

Toshiba Global Commerce Solutions
4690 OS



User's Guide

Version 6 Release 4

Toshiba Global Commerce Solutions
4690 OS



User's Guide

Version 6 Release 4

Note

Before using this information and the product it supports, be sure to read Safety Information- Read This First, Warranty Information, Uninterruptible Power Supply Information and the information under Appendix D, "Notices," on page 543.

September 2013

This edition applies to Version 6 Release 4 of the licensed program Toshiba 4690 OS (program number 5639-P70) and to all subsequent releases and modifications until otherwise indicated in new editions.

If you send information to Toshiba Global Commerce Solutions (Toshiba), you grant Toshiba a nonexclusive right to use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

© Copyright Toshiba Global Commerce Solutions, Inc. 2013.

© Copyright IBM Corporation 2010, 2012

Contents

Figures	xxiii
Tables	xxv
Safety	xxvii
About this guide	xxix
Who should use this guide	xxix
Terminal models	xxix
Where to find more information	xxx
4690 V6 library	xxx
Notice statements	xxx

Part 1. Understanding the 4690 Operating System 1

Chapter 1. Getting started.	3
System panel paths chart	3
What is the 4690 OS?	5
Operating modes	6
4690 Network File System	7
4690 Virtual File System	8
Embedded Shared File System	8
Operating system configurations	8
Diskette, CD, and DVD drives and media used with your system	9
CD-ROM support for the 4690 Operating System	10
DVD support for the 4690 Operating System	10
Optical diskettes	12
Handling diskettes, CDs, and DVDs	12
Write-protecting your diskettes	12
Formatting your diskettes	13
Keeping backup copies of diskettes	13
About your hard disk drive.	13
Precautions to use with your hard disk drive	14
Disabling the write verify function for your hard disk drive	14
Identifying drives	14
Consoles	15
Operating modes on your system	15
User modes	16
Application modes.	16
What are windows?	16
Security on your system	17
Your operator identification	18
Your password	18
Specially authorized operators	18
Console ID security	19
NFS security.	20
NetBIOS security	20
Enhanced security	21
Secure Shell.	30
IP Security	30
4690 secure logon through LDAP server	30
Command line logging	31
Data security in dump data	33

Securely Deleting Files	33
Using menus	33
Understanding panel information	34
Message line	35
Understanding messages	35
Function keys line	35
Status line	35
Using special keys	36
Using keys to control panel information	36
Using function keys	40
Using system request keys	42
Using system keys Ctrl Alt +	43
Using system reset keys	44
Confirm Ctrl-Alt-Delete (IPL key sequence)	44
Using keys in Command Mode	45
Using the Supplemental Option	45
CD-ROM	45
Memory Key	45
Hard Drive	45
 Chapter 2. Configuring your store system	 47
Using your planning worksheets	47
Saving or reconfiguring the default configuration	48
Signing on	48
Using auto sign on	49
Using auto sign off	50
Selecting the Configuration option	50
Configuring legacy terminals (4683 and 4693)	52
Terminal device groups	52
Keyboard layouts	52
Terminal load definition	52
Alphanumeric display character set	53
Model 1 and Model 2 printer character set	53
Model 3 and Model 4 printer character set	53
Terminal screen saver	53
Network file system data	53
Font/logo utility	54
Configuring your TCxWave 6140 Series, SurePOS 4694, 350, and 700 Series terminals	58
Terminal device characteristics	58
Terminal load definitions	59
Keyboard layouts	59
Alphanumeric display character set	59
Terminal screen saver	59
Network file system data	59
Java applications	61
Preload bundles	65
Font/logo utility	71
Configuring your store controllers	71
Non-DDA system	72
DDA system	72
Configuring your system	75
Activating your configuration	75
Activating your terminal configuration	76
Activating your controller configuration	77
Activating your system configuration	78

Setting a locally administered LAN address	79
Changing the LAA prefix	80
Disabling LAA	80
Chapter 3. Store controller operations	81
Using the SYSTEM MAIN MENU	81
Running your store's primary application	81
Running your store's secondary application	82
Using File Utilities	82
Using System Installation and Update Aids	84
Collecting and reporting problem analysis data	86
Using command mode	91
Using store control functions	91
Power management	93
Using terminal functions	96
Using controller functions	99
Using TCC functions	100
Using LAN functions	101
Using Communications Functions	105
Chapter 4. Point-of-Sale terminal operations	109
Quick reference table for system functions	109
Using a keyboard with Java applications	110
Using a keyboard and video display on a controller/terminal	110
Using a touch screen on a terminal	111
Touch Screen Firmware Update	112
Dual touch devices	112
Unintentional touch rejection	112
Tone Only Keyboard Interface	113
Requesting system functions	113
Using uninterruptible power supply	116
Terminal screen saver	117
Displaying messages	117
Displaying wait messages	118
Displaying offline messages.	118
Displaying system messages	118
Clearing the message display	119
Setting the date and time on an offline terminal	119
Displaying the date format	119
Setting the date on an offline terminal	119
Setting the time on an offline terminal	120
Changing the volume of the keyboard tone	120
Displaying the terminal number	120
Enabling the storage retention function	120
Disabling the storage retention function	121
Displaying the available memory	121
Dumping terminal storage	121
Swapping terminal video screens.	121
Switching to the enhanced mode graphical extensions screen	122
Using the Set Terminal Characteristics program	122
Requesting the STC program from the terminal keyboard.	122
Entering terminal numbers	122
Displaying terminal configuration data	126
Changing a terminal number	126
Resetting a terminal number to zero	128
Changing a terminal configuration	128

Formatting the terminal hard disk drive	130
Exiting an STC function	131
Other system functions	132
Chapter 5. Working in Command Mode	133
Selecting Command Mode	133
Exiting from Command Mode	133
Commands and command lines	133
Typing commands into the command line.	133
Processing commands	134
Logging command line activity.	134
Editing keys	135
Inserting characters.	136
Entering several commands on a command line	136
Organizing your information.	137
Files and file names on FAT systems	137
Directories and subdirectories on a FAT system	138
File systems supporting long file names	140
Directories and subdirectories on a VFS system	141
Store controller roles	141
Using global file name characters	142
The * character	142
The ? character	142
The ^ character	143
Protecting your files.	143
Redirecting input and output	143
Methods for redirecting input and output	143
Redirecting output to a file or device	144
Adding output to the end of a file or device	144
Getting input from a file or device	144
Piping commands	145
Filtering data	145
Chapter 6. Using system commands	147
Formatting diskettes	147
Formatting your hard disk drive	148
Making copies of diskettes	148
Copying diskettes on a one-drive system	148
Copying diskettes on a two-drive system	149
Displaying what is in a file	149
Listing the files on your diskette	149
Listing files on a specific drive	150
Listing files on another store controller.	150
Renaming a file	150
Erasing a file	150
Chapter 7. Command summary.	153
Types of commands	153
Using commands on the LAN	153
Quick reference tables	154
Command format	156
Batch commands	157
Using parameters in batch files	158
Chaining batch files.	160
Nesting batch files	161
Commands for batch files	161

ADXSDEL command	163
Purpose	163
Formats	163
Description	163
Examples	163
ASSIGN command	165
Purpose	165
Formats	165
Type	165
Description	165
Examples	165
BACKUP command	167
Purpose	167
Formats	167
Type	167
Description	167
Examples	168
BATCH command	170
Purpose	170
Format	170
Description	170
Example	170
CHDIR command	171
Purpose	171
Formats	171
Type	171
Abbreviation	171
Description	171
Examples	171
CHKDSK (check disk) command	173
Purpose	173
Formats	173
Type	173
Description	173
Examples	174
CLS command	178
Purpose	178
Format	178
Type	178
Description	178
Example	178
COMMAND command	179
Purpose	179
Formats	179
Type	179
Description	179
COMP command	181
Purpose	181
Formats	181
Type	181
Description	181
Examples	182
COPY command	184
Purpose	184
Formats	184
Type	184

Description	184
Examples	185
Combining files	187
Appending files	188
Using device names with COPY	189
DEFINE command	190
Purpose	190
Formats	190
Type	190
Description	190
Examples	191
DIR (directory) command.	193
Purpose	193
Formats	193
Type	193
Description	193
Examples	194
DISKCOMP command.	198
Purpose	198
Formats	198
Type	198
Description	198
Examples	198
DISKCOPY command	200
Purpose	200
Formats	200
Type	200
Description	200
Examples	200
ECHO Command	203
Purpose	203
Formats	203
Type	203
Description	203
Examples	203
ELSE Command	205
Purpose	205
Format	205
Type	205
Description	205
Example	205
ERASE Command	206
Purpose	206
Formats	206
Type	206
Abbreviation	206
Description	206
Examples	206
EXIT Command	208
Purpose	208
Format	208
Type	208
Description	208
Example	208
FCOPY Command	209
Purpose	209

Formats	209
Type	209
Description	209
Examples	209
FIND Command	210
Purpose	210
Formats	210
Type	210
Description	210
Examples	210
FOR Command	213
Purpose	213
Format	213
Type	213
Description	213
Examples	213
FORMAT Command	214
Purpose	214
Formats	214
Type	214
Description	214
Examples	215
FSET Command	218
Purpose	218
Formats	218
Type	218
Description	218
Examples	219
GOTO Command	221
Purpose	221
Format	221
Type	221
Description	221
Example	221
IF Command	223
Purpose	223
Formats	223
Type	223
Description	223
Examples	223
LIST Command	226
Purpose	226
Format	226
Type	226
Description	226
Example	226
MKDIR Command	227
Purpose	227
Formats	227
Type	227
Abbreviation	227
Description	227
Examples	227
MORE Command	229
Purpose	229
Formats	229

Type	229
Description	229
Examples	229
ORDER Command	231
Purpose	231
Formats	231
Type	231
Description	231
Examples	231
PATH Command	232
Purpose	232
Formats	232
Type	232
Description	232
Examples	232
PAUSE Command	234
Purpose	234
Format	234
Type	234
Description	234
Examples	234
PRINT Command	235
Purpose	235
Formats	235
Type	236
Description	236
PROMPT Command	237
Purpose	237
Formats	237
Type	237
Description	237
Examples	237
QUERY Command	239
Purpose	239
Format	239
Description	239
Examples	239
REM Command	240
Purpose	240
Formats	240
Type	240
Description	240
Examples	240
RENAME Command	241
Purpose	241
Format 1.	241
Format 2.	241
Type	241
Abbreviation	241
Description	241
Examples	242
RESTORE Command	243
Purpose	243
Formats	243
Type	243
Description	243

Examples	244
RMDIR Command	247
Purpose	247
Formats	247
Type	247
Abbreviation	247
Description	247
Examples	247
SECURITY Command	248
Purpose	248
Formats	248
Type	248
Description	248
Example	248
SHIFT Command	249
Purpose	249
Format	249
Type	249
Description	249
SORT Command	251
Purpose	251
Formats	251
Type	251
Description	251
Examples	251
TREE Command	255
Purpose	255
Formats	255
Type	255
Description	255
Examples	255
TYPE Command	257
Purpose	257
Format	257
Type	257
Description	257
Examples	257
UNLOCKP Command	258
Purpose	258
Formats	258
Type	258
Description	258
Examples	258
VER Command	259
Purpose	259
Format	259
Type	259
Description	259
Example	259
VERIFY Command	260
Purpose	260
Formats	260
Type	260
Description	260
Examples	260
VOL Command	261

Purpose	261
Formats	261
Type	261
Description	261
Examples	261
Creating an image file of a diskette	261
Burning an ISO image on a CD or memory key	262
Purpose	262
Formats	262
Description	262
Examples	263
Chapter 8. Creating and editing files	265
Using DR EDIX	265
Starting DR EDIX	265
Entering DR EDIX commands	266
Entering text	266
Moving the cursor	266
Editing text	267
Saving your work	268
Understanding DR EDIX backup files	268
Editing another file	269
Exiting DR EDIX	269
Understanding the panel	269
Working with windows and buffers	272
Using block commands	273
Searching for and translating text.	275
Going to a line by number	277
Merging files	277
Changing the file name	278
Summary of DR EDIX commands	278
DR EDIX error messages	280
Using XE Editor	282
Starting XE Editor	282
Entering text	283
Moving the cursor	283
Editing text	284
Saving your work	284
Exiting XE Editor.	285
Editing multiple files	285
Understanding the panel	285
Using block commands	287
Searching for and translating text.	288
Going to a line by number	288
Importing files	288
Changing the file name	289
Command line logging and the XE editor	289
Chapter 9. Using Java with the operating system	291
Understanding Java with the operating system.	291
Configuring your system	293
Configuring terminal functions for Java.	294
Configuring controller functions for Java	294
NFS backup for Java 2 terminal applications	294
Installing and running a Java application	296
Running Java 2 JVM applications from the command line.	297

Running Java 6 JVM applications from the command line.	297
Running Java applications from a terminal	297
Configuring multiple graphical Java applications	298
Configuring multi-app	298
Property file	298
Configuring multi-app on a terminal	299
Running multi-app on a controller	299
Configuring classpath	299
Window Manager Dialog	299
Using graphics with the operating system.	301
Using multiple video displays	302
Supported video cards	302
Supported video dongles.	303
Configuring dual display	303
Configuring the default display.	303
Secondary video display function limitations.	303
Accessing multiple displays from a single application	303
Chapter 10. Enhanced terminal preload and Java configuration	305
Overview	305
Java application setup.	306
Preload bundles	309
Terminal load definitions	309
Migrating legacy setups to 4690 OS Version 6 Release 2 or later	310
RAM disk preloading	311
MultiApp	311
Setting fields on the Java Application Setup Screen	311
JavaTOF.	312
Controller/terminal setup	313
Terminal Configuration Activation	313
Terminal Preload Rebuild Utility	314
Terminal loading	315
File naming conventions	318
Frequently asked questions (FAQs)	321
Chapter 11. Understanding USB	323
Multiple Device Support	323
USB Memory Keys	324
Configuring your system	324
Using USB with the operating system	324
USB devices with 4690 OS	324
Chapter 12. Enabling MBrowser	327
Launching MBrowser	327
System flags	328
Customizing MBrowser	330
Launching MBrowser from a terminal	332
Program Exit Code	335
Operational Environment.	336
Chapter 13. The PDF document viewer.	337
Launching adxpdf	337
Command line syntax	338
Launching adxpdf from a terminal	339
Program Exit Code	339
Operational Environment.	340

Chapter 14. 4690 Directory Services	341
Directory Services requirements	341
Configuring Directory Services	341
Initializing the directory	341
Verifying installation	342
ADXLDIFF.DAT	342
The ADXLDIFF.DAT Header	342
ADXLDIFF.DAT Directory Entries	343
Root Entries	344
The Initial Administrator	345
4690 User Authorization Roles and Groups	346
Password Policy Settings	347
Support for Long User IDs	348
Logging on to using Directory Services IDs	349
Updates to the Enhanced Security API	349
Additions to the Enhanced Security API	350
IsMemberOf	350
GetGroups	351
Replication	352
Connecting to a 4690 OS Directory Server	353
Transport Layer Security	354
Additional information	355

Part 2. 4690 Operating System Utilities 357

Chapter 15. Using the Display/Alter File Utility	359
Accessing the Display/Alter File Utility	359
Logging ADXCSJOL Activity	359
Using File Mode	359
Selecting a Disk Drive	359
Changing the Current Directory	359
Selecting a File	360
Displaying and Altering the File	360
Scanning a File	360
Using Physical Sector Mode	361
Selecting a Disk Drive	361
Selecting a Sector	361
Displaying and Modifying a Sector	361
Scanning a Sector	362
Using the Critical Sector Repair Function	363
 Chapter 16. Using the software maintenance utilities	 365
Apply Software Maintenance utility	365
ASM verification of maintenance	367
ASM history and product-level files	368
LAN (MCF) system considerations	371
Optional ASM indicators	372
Combining a delta with maintenance	373
Remote application of combined/compressed maintenance updates	373
RMA and Director method	374
RCP method	376
Build software maintenance control file utility	379
Building a software library	381
Installing a user's product	381
Applying and maintaining user modifications	382
User checksums	383

Transferring modules to other subdirectories	383
Chapter 17. Using the streaming tape drive utility	385
Understanding the streaming tape drive utility	385
Accessing the streaming tape drive utility.	386
Using the streaming tape drive utility from the host	386
Using the streaming tape drive utility from an application program	386
Status file definition.	386
File data accessing considerations	387
Tape cartridge identification and contents.	387
Using tape utility functions	388
Backing up files to tape	388
Restoring files from tape	388
Listing tape contents	388
Erasing tape	389
Adjusting tape tension.	389
Creating a selection list file	389
LAN (MCF Network) system considerations	389
Using the streaming tape drive	390
Backing up entire hard disk drives	390
Backing up selected files.	390
Backing up files that applications are updating	391
Chapter 18. Using the optical drive utility.	393
Introduction to the optical drive utility	393
Comparison with streaming tape drive utility.	394
Advantages.	394
Disadvantages	394
Formatting the optical disk	394
Accessing the optical drive utility	395
Using the optical drive utility from the host	395
Using the optical drive utility from an application program	396
Logical drive name	396
Using the optical drive from command mode	396
Status file definition.	397
File data accessing considerations	397
Disk identification and contents	398
Optical drive utility functions	398
Backing up files to optical disk.	398
Restore files from optical disk	399
List optical disk contents	399
Erase files from optical disk.	400
Format optical disk	400
Selection list file creation.	400
LAN system considerations	400
Understanding file distribution and DISTFILE	400
Supplemental Option using the CD-ROM considerations	401
Recommended usage for optical drive	401
Entire hard disk drives and CD/DVD media	401
Selected files	402
Backing up files that applications are updating	403
Write recovery.	403
Alternative drives	404
RCP command interface	404
Cancelling a backup that is in progress	404

Chapter 19. Using the File Compress/Decompress Utility	405
Accessing the File Compress/Decompress Utility through Panels	405
Using the File Compress/Decompress Utility from Host Processor	406
Using the File Compress/Decompress Utility with Diskettes	406
LAN System Considerations	407
LAN (MCF Network) Attributes and the File Compress/Decompress Utility	407
File Compress/Decompress Utility and Hard Disk Drive Space	408
Error Logging	408
Chapter 20. Using the disk rebuild utility	409
Introduction to the LAN disk rebuild utility	409
V6 LAN disk rebuild considerations	410
Restoring data on a hard disk drive	411
Formatting a hard disk drive	411
Locating controller IDs for NetBIOS-based systems	412
Determining IP configuration information	412
Using valid paths	413
Loading the disk rebuild utility	414
Entering the LAN type	414
Selecting a transfer method	415
Entering the IP information	415
Entering the LAN (MCF Network) node ID	416
Recovering data	416
Checking the audit trail	416
Nested subdirectories	417
Chapter 21. Using Terminal RAM Disk Preload Utilities	419
Creating List Files	419
Running the Image Build Utility	420
Running ADXRTCCL	421
Completing the Terminal RAM Disk Load	421
Using a .BAT File	421
Chapter 22. System Services Process Utility	423
Using the System Services Process Utility	423
Return codes	424
Chapter 23. Controller ASM preload	425
Overview	425
Command Processing	426
Available commands	427
Trigger – Control command execution	427
Extract – Extract an archive file	428
Messages and error logging	430
Chapter 24. Directory Backup and Restore Utility (ADXZUDIR)	433
Overview	433
General Syntax	434
File Exclusion	434
Automatic Mode	436
Creating a backup	437
Restoring a backup	438
Testing a backup	438
Listing the contents of a backup	438
Manual Mode	439
Creating a ZIP archive	439

	Extracting a ZIP archive	439
	Testing a ZIP archive	439
	Listing the contents of a ZIP archive	439
	Using a response file	440
	Creating an Image of a 4690 Store Controller	440
	Restoring an image of a 4690 Controller	441

Part 3. Multiple Controller Feature Network. 443

Chapter 25. Using the Multiple Controller Feature	445
Advantages of the Multiple Controller Feature	445
Concepts and Elements of the Multiple Controller Feature	445
Distributed Files	446
File Distribution Attributes	446
File Type Attribute	446
File Mode Attribute	447
Per Update	448
At Close	448
Nested Subdirectories	448
Store Controller Roles	448
Master Store Controller	449
Alternate Master Store Controller.	449
Subordinate Store Controller	449
File Server	449
Alternate File Server	449
Combining Store Controller Roles	449
Nodes.	450
Node IDs	450
Node Names	451
Logical File Names	451
Logical Names Table	451
Distribution Exception Logs	452
How the Exception Logs are Used	452
Exception log tracking	452
Exception Log Reconciliation	453
 Chapter 26. Multiple Controller Feature (MCF) Networks	 455
TCC Support and Background Applications	455
Defining File Types and Modes	455
Mirrored File - Distribute At Close	455
Mirrored File - Distribute Per Update	456
Compound File - Distribute At Close	456
Compound File - Distribute Per Update	456
Connecting to a Host	457
Host Support Requirements.	457
Two-Controller Configuration	457
Multiple Controller Configuration	458
Selecting Store Controller Roles	458
 Chapter 27. Using MCF Networks	 461
Displaying the Status of Store Controllers	461
Activating/Deactivating the Master Store Controller	461
Activating/Deactivating the File Server Controller	462
Switching to Alternate Store Controller Roles	463

Chapter 28. Using the Distributed File Utility	465
Accessing the Distributed File Utility	465
Distribution Exception Logs	465
Processing the Master and File Server Exception Logs.	466
Displaying the Exception Log	466
Modifying the Exception Log	467
When to Display and Modify the Distribution Exception Logs	467
Log Error on Clearing of Exception Log	467
Displaying and Modifying a File's Distribution Type	467
Displaying a File's Distribution Attributes	468
Changing a File's Distribution Attributes	468
Changing a Local File to a Distributed File	469
Changing a Distributed File to a Local File	469
Distributing a File	469
Using the Distributed File Utility in Command Mode	469
 Chapter 29. Connecting OS/2 or DOS to the MCF Network	473
Connecting OS/2 to the MCF Network	473
Operating System Restrictions and Required Software	473
Using the Universal Naming Convention	474
Network Access Instructions	474
System Commands.	477
Using the Application Program Interface	477
Connecting DOS to the 4690 MCF Network	478
DOS and LAN Program Requirements.	478
The PC LAN Program Naming Convention	478
Establishing network access instructions	479
System commands	481
Using the Application Program Interface	482
Alternative procedures for OS/2 and DOS	482
Creating 4690 logical names for LAN (MCF network) resources	482
Using a single 4690 controller with OS/2 or DOS	483
 Chapter 30. MCF network problem analysis and recovery	485
Recovering the master store controller hard disk drive	485
Step 1. Install the software on the replacement hard disk drive	485
Step 2. Check for inactive changes waiting to be activated	485
Step 3. Activate the acting master's configuration.	486
Step 4. Stop all applications	486
Step 5. IPL the master store controller.	486
Step 6. Erase all file server exception log entries for the master store controller.	487
Step 7. IPL all the other store controllers	487
Step 8. Distribute subdirectories	487
Step 9. Reassign acting master and file server.	488
Reconnecting a store controller to the MCF Network system.	488
When the exception log or spool file fills up	488
Exception log	488
Spool file	489
When the file server hard disk drive is full	489
Freeing space.	489
When a store controller is listed as a non-MCF Network	489
Node list entry file	490
Copying distributed files to and from diskettes	491
Recovering when distributed files have been changed to local	491

Chapter 31. Store controller backup	493
How store controller backup works	493
LAN	493
Store Controller Backup Operations	493
Allowing Store Controller Backup	493
Preventing Store Controller Backup	494
Recovering from Store Controller Backup	494
 Appendix A. Loading Diagnostics for POS Systems over the Network	495
Configuring Remote Diagnostics on 4690	495
Understanding DHCP Alternate Boot Behavior	496
 Appendix B. Systems Network Architecture 3270 Emulation	499
3270 Emulator description	499
3270 Emulation on the store controller	499
3270 Emulation on the Point-of-Sale terminal	500
Equipment supported by the 3270 Emulator	500
Keyboard templates	501
Modifying your system configuration	501
Changing your configuration using menus	502
Starting 3270 Emulator Sessions in the Controller	502
Stopping the 3270 Emulator in the Controller	503
Using the 3270 Emulator with Switched Lines	504
Configuring a Printer Emulation at the Store Controller	504
Using 3270 Emulator Keyboard Functions at the Store Controller	504
Graphic Keystrokes	504
Unsupported Functions	505
3270 Emulator Key Functions	505
Using 3270 Emulator Printing Functions at the Store Controller	511
Local Printing	511
Operator-Initiated Local Printing	511
Host Printing	511
Operator Guidance	511
Operator Guidance Messages	512
Using the Application Programming Interface	514
Operator Guidance from the Application Programming Interface	515
3270 Emulation on the Terminal	516
Configuration	516
Using 3270 Emulation on the Terminal	517
Using Printing Functions at the Terminal	518
API	519
Starting and Stopping the Terminal Emulator	519
 Appendix C. Using the Supplemental Option on 4690	521
Using the Supplementals	521
Using the Supplemental Option on Enhanced 4690 Controllers	521
Using the Supplemental Option through the CD-ROM	521
Booting Enhanced Supplementals over the network	522
Securing the Supplementals	523
Functions on the Supplementals	523
Using Store Control Functions	523
Using the Menu Panels	524
Commands on the Supplementals	528
Command Format	529
ADXCSU1L Command	531
Purpose	531

Formats	531
Type	531
Description	531
Example	531
ADXFSM3Z Command	533
Purpose	533
Formats	533
Type	533
Description	533
Examples	534
ADXFSM6Z Command	535
Purpose	535
Formats	535
Type	535
Description	535
Examples	535
ADXNSX0L Command	536
Purpose	536
Formats	536
Type	536
Description	536
Example	536
DVRLOAD Command	537
Purpose	537
Formats	537
Type	537
Description	537
Example	538
DVRUNIT Command	539
Purpose	539
Formats	539
Type	539
Description	539
Example	539
DVRUNLK Command	540
Purpose	540
Formats	540
Type	540
Description	540
Example	540
SYS Command	541
Purpose	541
Formats	541
Type	541
Description	541
Examples	541
Disk Rebuild Utility	541
Appendix D. Notices	543
Telecommunication regulatory statement	545
Electronic Emission Notices.	545
Federal Communications Commission (FCC) Statement	545
Industry Canada Class A Emission Compliance statement	545
Avis de conformité à la réglementation d'Industrie Canada	545
Australia and New Zealand Class A Statement.	545

European Union Electromagnetic Compatibility (EMC) Directive	
Conformance Statement	545
Germany Class A Statement	546
Japan Voluntary Control Council for Interference Class A statement	547
Japan Electronics and Information Technology Industries Association (JEITA)	
statement	547
Korean communications statement	548
Russian Electromagnetic Interference (EMI) Class A statement.	548
People's Republic of China Class A electronic emission Statement	548
Taiwan Class A compliance statement	548
European Community (EC) Mark of Conformity Statement	549
Electrostatic Discharge (ESD)	549
Japanese Electrical Appliance and Material Safety Law statement	549
Japanese power line harmonics compliance statement.	549
Cable ferrite requirement.	550
Product recycling and disposal	550
Battery return program	551
For Taiwan:	551
For the European Union:	551
For California:	552
Flat panel displays	552
Monitors and workstations	553
Trademarks.	554
Glossary	555
Index	575

Figures

1. Paths of 4690 OS panels from SYSTEM MAIN MENU	4
2. Write-protecting 3½-in. diskettes and optical diskettes	13
3. Main Java Applications screen	61
4. Application screen	62
5. New Java Application screen	63
6. Rename Java Application screen	64
7. Preload Bundles screen	65
8. New Preload Bundle screen.	66
9. Rename Preload Bundle screen	67
10. Edit Preload Bundle screen	68
11. Add Bundle Entry screen	69
12. Edit Bundle Entry screen	70
13. Controller performance	87
14. Point-of-Sale terminals	109
15. System function and status indicator keys on Point-of-Sale keyboards	115
16. Directory Tree	139
17. Using parameters	159
18. Chained batch files	160
19. Nested Batch Files.	161
20. Example of application configuration file	332
21. Sample application load definition configuration file	334
22. Root Entries	344
23. LDAP Sync Replication	353
24. Types of Files Stored on Each Store Controller	450

Tables

1.	Using windows in a multiple-user mode	17
2.	Personal computer keyboard key functions	36
3.	Auxiliary Console ASCII Keyboard key functions	38
4.	Java interface key functions	40
5.	Function keys used in single- and multiple-application Mode	41
6.	System request key descriptions	42
7.	Enhanced Options Menu	43
8.	Auto sign-on file contents, ADXC SOAU.DAT	49
9.	Model 4A printer default font files	56
10.	4610 printer models TI5, TF7, TM7, 1NR, 2CR, or 2NR default font files	56
11.	Model 4A printer default font files	57
12.	Performance Utilization Guidelines	90
13.	Editing key functions	135
14.	Working with files	154
15.	Working with directories	154
16.	Working with disks and diskettes	154
17.	Protecting your data	154
18.	Modifying your user environment	155
19.	Other commands	155
20.	Using driver commands	155
21.	Using batch file commands	155
22.	Using commands on the LAN (MCF Network).	156
23.	BACKUP parameters	168
24.	Information generated by CHKDSK	176
25.	COMMAND parameters	179
26.	COPY parameters	185
27.	Error message help levels	190
28.	DIR parameters	194
29.	FIND Parameters	210
30.	FORMAT parameters	214
31.	PRINT options	236
32.	QUERY Command Options	239
33.	RESTORE Parameters	243
34.	Sort Parameters	251
35.	DR EDIX cursor control keys	266
36.	DR EDIX status table	271
37.	DR EDIX window commands	272
38.	DR EDIX commands and error messages	278
39.	XE Editor cursor control keys	283
40.	Text control keys	286
41.	FAQs about JavaTOF enhancements in 4690 OS Version 4	321
42.	MBrowser Launcher Exit Codes	335
43.	Adxpdf Launcher Exit Codes	339
44.	Additional information.	355
45.	Format of history and product-level files	371
46.	Cursor Movement Keys	505
47.	Field Cursor Movement Keys	506
48.	Program Attention Keys	507
49.	Delete Keys	508
50.	Print Control Keys	508
51.	Miscellaneous Keys	509
52.	Emulator-Specific Keys	510

Safety

Before installing this product, read Safety Information- Read This First.

قبل تركيب هذا المنتج، يجب قراءة الملاحظات الأمنية

Antes de instalar este produto, leia as Informações de Segurança.

在安装本产品之前，请仔细阅读 **Safety Information** (安全信息)。

安裝本產品之前，請先閱讀「安全資訊」。

Prije instalacije ovog produkta obavezno pročitajte Sigurnosne Upute.

Před instalací tohoto produktu si přečtěte příručku bezpečnostních instrukcí.

Læs sikkerhedsforskrifterne, før du installerer dette produkt.

Lees voordat u dit product installeert eerst de veiligheidsvoorschriften.

Ennen kuin asennat tämän tuotteen, lue turvaohjeet kohdasta Safety Information.

Avant d'installer ce produit, lisez les consignes de sécurité.

Vor der Installation dieses Produkts die Sicherheitshinweise lesen.

Πριν εγκαταστήσετε το προϊόν αυτό, διαβάστε τις πληροφορίες ασφαλείας (safety information).

לפני שתתקינו מוצר זה, קראו את הוראות הבטיחות.

A termék telepítése előtt olvassa el a Biztonsági előírásokat!

Prima di installare questo prodotto, leggere le Informazioni sulla Sicurezza.

製品の設置の前に、安全情報をお読みください。

본 제품을 설치하기 전에 안전 정보를 읽으십시오.

Пред да се инсталира овој продукт, прочитајте информацијата за безбедност.

Les sikkerhetsinformasjonen (Safety Information) før du installerer dette produktet.

Przed zainstalowaniem tego produktu, należy zapoznać się z książką "Informacje dotyczące bezpieczeństwa" (Safety Information).

Antes de instalar este produto, leia as Informações sobre Segurança.

Перед установкой продукта прочтите инструкции по технике безопасности.

Pred inštaláciou tohto zariadenia si pečítajte Bezpečnostné predpisy.

Pred namestitvijo tega proizvoda preberite Varnostne informacije.

Antes de instalar este producto, lea la información de seguridad.

Läs säkerhetsinformationen innan du installerar den här produkten.

About this guide

This guide assists you in using the 4690 OS Version 6 Release 4 (hereafter referred to as the operating system). It contains procedures for using the functions of the operating system under normal operating conditions. It assumes you have set up your store controller according to the instructions in the guide for your specific store controller. It also assumes you have set up your point-of-sale terminals according to the installation guides.

Who should use this guide

This guide is written for store personnel who use the operating system and its utilities. It is designed to assist you in using the controllers, terminals, and utilities.

Terminal models

The 4693-xx1, 4694, SurePOS 300 Series Model 350, SurePOS™ 700 and the TCxWave 6140 Series terminals are called *Mod1* terminals. Although all are called *Mod1* terminals, each terminal model supports some features that other models do not support. Additionally, the SurePOS 300/700 Series and TCxWave 6140 Series terminals provide USB capabilities.

The 4693-xx2 terminals are called *Mod2* terminals. These terminals attach to a *Mod1* terminal and depend upon that *Mod1* terminal for control and communication with the store controller.

Note: 4683 terminals are not supported on 4690 OS V6R3. References to 4683 terminals only apply to previous versions of the OS.

The controller/terminal (for example, a 4693-5x1 controller/terminal) combines the function of the store controller and point-of-sale terminal in a single product. The terminal portion of a controller/terminal is considered to be a *Mod1* terminal.

Note: Refer to the support Knowledgebase for information about which 4694 and SurePOS systems are supported as combination controller/terminals. 4693 systems are only supported as a controller in a non-Java environment or as an alternate in a Java™ environment.

4690 V6R3 introduced support for the SurePOS 300 Series Model 350 terminal (4810-350). This is a *Mod1* terminal. The 4810-350 has the following characteristics:

- It can only be used as a terminal in Enhanced Mode
- It cannot be used as a store controller
- The RS232 Sureport card is not supported by 4690 OS
- Aside from a single port for the cash drawer, there are no RS485 ports available.

4690 V6R4 introduced support for the TCxWave Series terminal (6140-100). The 6140-100 is a *Mod1* terminal with the following characteristics:

- It can only be used as a terminal in Enhanced Mode
- It cannot be used as a store controller
- No RS485 ports available, including no cash drawer port

- Cash drawer support is provided, for a single cash drawer, through USB-attached cash drawers
- RS232 support is provided through RS232-to-Serial dongles
- The Power Button functions as the only available Dump Button
- Secondary video support through a video dongle

Where to find more information

Current versions of the following Toshiba Global Commerce Solutions documentation are available on the Toshiba support site.

1. On the right side of the web page under popular links, select **Publications**.
2. Click on the publication related to your product.

4690 V6 library

Note: References to related 4690 publications in this guide are references to the publications in the 4690 Version 6 library. For example, the *4690 OS Version 6: User's Guide* is referred to as the *4690 OS: User's Guide*
4690 OS: Planning, Installation, and Configuration Guide, G362-0541
4690 OS: User's Guide, G362-0542
4690 OS: Messages Guide, G362-0543
4690 OS: Communications Programming Reference, G362-0544
4690 OS: Programming Guide, G362-0545
4690 OS: Master Index, G362-0546
4680 BASIC: Language Reference, SC30-3356

Notice statements

Notices in this guide are defined as follows:

Notes	These notices provide important tips, guidance, or advice.
Important	These notices provide information or advice that might help you avoid inconvenient or problem situations.
Attention	These notices indicate potential damage to programs, devices, or data. An attention notice is placed just before the instruction or situation in which damage could occur.
CAUTION	These statements indicate situations that can be potentially hazardous to you. A caution statement is placed just before the description of a potentially hazardous procedure step or situation.
DANGER	These statements indicate situations that can be potentially lethal or extremely hazardous to you. A danger statement is placed just before the description of a potentially lethal or extremely hazardous procedure step or situation.

Part 1. Understanding the 4690 Operating System

Chapter 1. Getting started

This chapter gives you important information about your system and how to operate it. The chapter describes the information you see on your screen and how to use or interpret that information. You can also find information on disks, diskettes, CD-ROMs, and DVDs, operating modes, passwords, NetBIOS security, operator authorization and identification, menu panels, and function keys.

Read this chapter before starting your system.

System panel paths chart

Figure 1 on page 4 is a quick-reference diagram of panel titles and options. The figure represents the operating system menus that appear when you make a selection on the SYSTEM MAIN MENU panel. Each line represents the particular path (or options selected) you take to get to the panel at the end of the line.

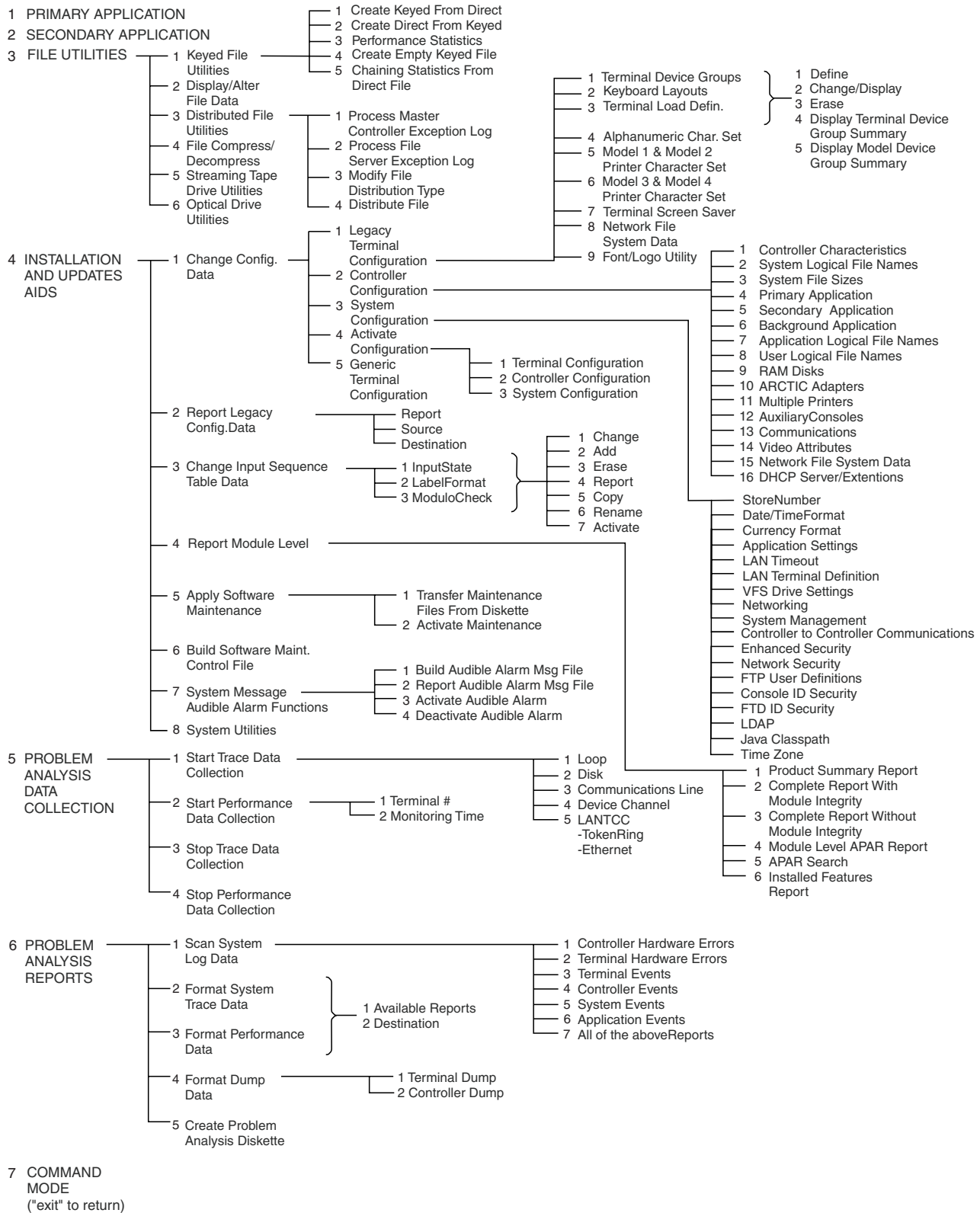


Figure 1. Paths of 4690 OS panels from SYSTEM MAIN MENU

What is the 4690 OS?

The 4690 Operating System (OS) is a group of programs that enable you to control the organization and use of data on hard disk drives, tapes, diskettes, optical diskettes, CDs, and DVDs. The operating system enables you to use applications (programs) that run your store operations, use devices such as printers, configure your system, or create files and directories.

The operating system is a flexible system that can be operated to suit your store's special needs. It is a multitasking, multiuser system that supports data redundancy and backup functions. For example, this system gives you the option of running one application or several applications at the same time. The system enables you to run some applications using menus and run some applications by entering commands.

Your operating system is *menu-driven*. That is, you choose the system function you want to use from a menu displaying several options. These options represent tasks that you can perform in your store. By choosing from the options that are displayed, you can run your store programs, create keyed files, monitor system performance, and get information about your terminals.

Your system also includes a *command-driven* option. This option lets you perform certain functions not available as menu options. For example, by typing in commands, you can create memos or other correspondence, or you can format or copy a diskette.

The operating system supports a special feature called the 4690 Multiple Controller Feature (MCF). This optional feature enables one store controller to be a central point of control in the system and communicate with other store controllers using a Data Distribution Application (DDA).

X.25 communications is available for use with the operating system. This function enables the store controller to attach to a X.25 network by a X.25 Interface Co-Processor/2 adapter. You can use up to four virtual circuits for each X.25 Interface Co-Processor/2 adapter. The operating system supports the attachment to X.25 networks operating according to the 1984 or 1988 CCITT recommendations.

Some of the functions of the operating system are used by your store controller; other functions are used by the point-of-sale terminal. Some POS terminal models can be used as both a controller and a terminal. Refer to the support Knowledgebase for current information about the capabilities of specific POS hardware.

The SurePOS 700 Series Models 7x4 and 7x5 can only be used with 4690 OS in Enhanced Mode.

The SurePOS 300 Series Model 350 can only be used with 4690 OS as an Enhanced Mode terminal.

The TCxWave 6140 Series terminal can only be used with 4690 OS as an Enhanced Mode terminal.

The operating system provides Java language-based capability at both the server and the client. This, plus the extension of TCP/IP capability to the client, enables Java applications at both the server and client to run concurrently with existing CBASIC or C applications. You can make functional enhancements to existing applications by developing a new Java application that communicates with an

existing application. See the *4690 OS: Programming Guide* for information on designing applications with the Java™ programming language.

The operating system offers a terminal offline function (TOF), which is a Java-language based program. With TOF, you can load into memory all Java classes that are required by an application so that if the terminal loses its connection with the controller, the application can continue to operate. See the *4690 OS: Programming Guide* for more information about using the terminal offline function.

The operating system has a graphics capability designed to work with the Java support. The graphics capability is accessible only through Java. Existing applications can use the graphical support by using Java extensions to provide a graphical user interface.

Because Java programs often have file names greater than eight characters in length, the operating system provides support for file names greater than eight characters. 4690 OS V2 and later versions use a 4690 Virtual File System (VFS), which is configured through system configuration to allow this support. See “4690 Virtual File System” on page 8 for more information. See the *4690 OS: Planning, Installation, and Configuration Guide* for worksheets for configuring VFS drive settings. See the *4690 OS: Programming Guide* for information on using long file name support.

4690 OS V2 and later versions also allow the configuration of a Network File System (NFS). Using NFS settings, you can configure mount point group and user ID information to mount to drives on a remote system. See “4690 Network File System” on page 7 for information on the Network File System within the 4690 OS environment. The operating system provides support for systems using SurePOS 700 Series Models 7x1, 7x2 and 7x3. Also, the operating system provides support for systems using the SurePOS 700 Series Models 7x4 and 7x5, but only in Enhanced mode. These systems provide a universal serial bus (USB) I/O attachment interface for point-of-sale I/O devices. The operating system supports the USB family of point-of-sale I/O devices, which include keyboards, displays, and cash drawers. It also supports other I/O devices that are designed to and conform to USB point-of-sale device interface specifications. The SurePOS 300 Series Model 350 can only be used with 4690 OS as an Enhanced Mode terminal. The TCxWave 6140 Series terminal can only be used with 4690 OS as an Enhanced Mode terminal.

The operating system supports and allows you to attach several double-byte character set (DBCS) I/O devices including the APA display, the 4689 printer, the Keyboard-V, the Keyboard-VI, the PLU POS keyboard, and the 50-key POS keyboard with JUCC MSR. For information on programming these devices, see the *4690 OS: Programming Guide*.

Operating modes

4690 OS V6 introduces a new infrastructure laying the way for the future while continuing to provide the expected benefits of the OS. As a result, 4690 OS V6 provides two operating modes, Classic and Enhanced.

Classic Mode

Classic Mode continues to use the OS infrastructure on which previous releases were based. This mode is intended to allow 4690 OS V6 to support the functions and much of the same hardware that have been supported in previous 4690 versions. Some new hardware and new functions available in 4690 OS V6 are not

supported in Classic Mode. Other functions, such as the new Remote Management Agent (RMA) software distribution support, is available in both modes.

Some 4690 OS V6 functions are available only in Classic Mode. These include, but are not limited to:

- Tape backup devices
- Optical drives (O:)
- Display/Alter - physical mode
- Disable Write Verify function

Enhanced Mode

Enhanced mode introduces a new hardware interface layer below Classic 4690 OS. The current 4690 OS user interfaces and programming APIs continue to be available allowing current applications to run in Enhanced Mode. This mode supports some additional System x[®] servers not supported by Classic Mode, including selected blade server models. Enhanced Mode is also required in order to exploit other new functions such as USB flash memory drives.

Systems running supported versions of 4690 OS that meet the minimum resource requirements of 4690 OS V6 may be migrated to 4690 OS V6 and continue to run in Classic Mode. The 4690 OS V6 installation CD is only available as an Enhanced Mode installation. 4690 OS V6 provides support for some System x servers only in Enhanced Mode. See Appendix L of the Planning, Installation, and Configuration Guide, "Classic To Enhanced Conversion Utilities," for information on converting controllers from Classic Mode to Enhanced Mode.

Some new functions in 4690 OS V6 are available only in Enhanced Mode. These include, but are not limited to:

- Running Supplemental system from hard disk
- USB memory key support
- Creating Supplementals on CD or USB memory key
- Java version 1.6 support
- F: drive for Java 1.6
- RMA Master Agent on 4690 OS controller
- Enhanced Options menu

Enhanced Mode terminals

Enhanced Mode support is available for recent models of SurePOS 300/700 Series and TCxWave 6140 Series terminals. Some of the functions available only in Enhanced Mode include:

- Power[®] management Deep Sleep mode (for capable hardware)
- Wake On LAN
- Java 6

4690 Network File System

The operating system supports the use of a Network File System (NFS). An NFS allows you to mount remote file systems across homogeneous and heterogeneous systems. The 4690 NFS provides an NFS client and an NFS server. The NFS server can export local directories for use by remote NFS clients. The NFS configuration in the 4690 OS V2 and later versions environment is performed through controller and terminal configuration.

The 4690 NFS client uses resources available on a remote site. A 4690 terminal can have only an NFS client, not an NFS server. A 4690 controller can be an NFS client and an NFS server simultaneously. Within the 4690 OS V2 and later versions environment, NFS supports only Data Distribution Architecture (DDA)-type files, but not 4690 keyed files.

Within the 4690 OS V2 and later versions environment, NFS is configured during terminal and controller configuration. NFS allows:

- You to mount to drives on another controller
- Other computers to mount to your system's C or D drive, or both
- Other computers to mount to your system's F drive (supported only in Enhanced Mode)
- Other computers to mount to your system's M or N drives, or both

For more information on NFS servers and clients, see the "NFS Server (ADXHSINL.386)" section, in the "Using TCP/IP in The Operating System" chapter of the *4690 OS: Communications Programming Reference*.

4690 Virtual File System

The operating system provides support for file names greater than eight characters in length through the use of a 4690 Virtual File System (VFS). The VFS drive setting must be enabled through system configuration. When you enable VFS drive settings, the operating system creates logical drives M: and N:. The drive determines where the VFS directory will be located. However, the information is actually stored on drives C: and D:. Drive M: information is stored on drive C:, and drive N: information is stored on drive D:.

Once you have enabled VFS, you can use drives M: and N: to provide long file name support locally, or configure NFS to access long file name support from a remote system.

Note: The operating system supplies a 32-bit program that implements 4690 VFS. VFS within environments using 4690 OS V2 and later versions should not be confused with an external VFS architecture.

Embedded Shared File System

The Embedded Shared File System (F:) is a file space shared with 4690's embedded Linux technologies. It can be read and written directly using the F: device specifier by 4690 users and user programs.

One embedded function that accesses this space is Java 6. Direct file accesses from Java 6 default to this file space. See "File systems supporting long file names" on page 140 for information on file name support in this file space.

Note: Files in the F: file space are not copied by the LAN Disk Rebuild Utility, and they can not be made distributed files. Any need for backup or redundancy of data on F: is the responsibility of the user. Files which require automatic redundancy (distribution) or LAN Disk Rebuild Utility capability should be placed on other 4690 file spaces that support these functions.

Operating system configurations

You can have different store system configurations in your store depending on the number of store controllers and additional features:

- A system with one store controller

If your store has one store controller and at least one terminal, you have a *single controller system*. The store controller controls the terminal.

- A system with two or more store controllers that can provide store controller backup

If your store has at least two store controllers, each store controller can control terminals on its own Terminal-Controller Communications (TCC) Network. However, a store controller is not required to have a network of terminals. If one of the store controllers fails, the other one not responsible for a network can automatically take control of the failing store controller's network.

When you configure your system, you select which store controller will provide backup for another. This kind of system is called a *store controller backup system*. It ensures that the terminals on a network can still be used if their own store controller cannot be used.

You can have a store controller backup system without the network described in the following paragraphs. If you do not have the network described in the following paragraphs, you must have some important store controller services that enable your store controllers to synchronize and reconcile data throughout the system through user-supplied programs.

- A system with two or more store controllers, the Multiple Controller Feature (MCF), and an Ethernet network

Your systems' functions depend on the type of store controller you have. With an Ethernet network, the operating system supports both LU 6.2 peer-to-peer communications and a DDA.

In the LAN with a DDA system (MCF Network), the store controllers are in a local environment (they are in the same store), they work together on the same electronic network, and they send files on the network to each other.

In this LAN system, one of the store controllers is a central point of control. It controls the prime (or main) versions of certain files. You configure your system from this store controller. It is called a *master store controller*.

A second store controller must be responsible for taking over for the master store controller if the master store controller fails. It is designated as the backup store controller and must be able to act as the master store controller. This backup store controller is called the *alternate master store controller*.

This kind of system can have other special store controller types besides the master store controller and alternate master store controller. One of the store controllers must be assigned to control the process of updating user application files. This store controller is called a *file server*, and it also has an alternate. A store controller can perform more than one role at the same time, so the master store controller can also act as the file server. Toshiba recommends that the roles of master store controller and file server be assigned to the same store controller.

To learn more about store controller roles, see the section on Store Controller Configuration in the *4690 OS: Planning, Installation, and Configuration Guide*.

Diskette, CD, and DVD drives and media used with your system

Your system can have the following types of media, CD-ROM or DVD drives:

- 3½-in. drive (stores up to 2.88 MB of data on a 3½-in. diskette). Note that this requires a 4-MB diskette.

- 3½-in. drive (stores up to 1.4 MB of data on a 3½-in. diskette). Note that this requires a 2-MB diskette.
- 3½-in. drive (stores up to 720 KB of data on a 3½-in. diskette)
- 3½-in. optical drive (stores up to 128 MB, 230 MB, 540 MB, or 640 MB of data). Note that this requires an optical diskette and a 3½-in. optical drive.
- CD-ROM drive (IDE-compliant)
- Multi-player, multi-burner, DVD RAM drives
- DVD-RW, DVD+RW, DVD RAM, CD-R, and CD-RW media

Choosing the correct diskette for operation in your diskette drive is necessary.

CD-ROM support for the 4690 Operating System

Use of wildcards is supported when accessing a CD-ROM on 4690 OS. For example the command `dir p:*. *` works on a CD-ROM. User prompts and errors are displayed in a manner consistent with like function on floppy diskettes or hard drives. The following functions are supported:

- CD [ISO9660] Full (8.3 filename) read with wildcards:
 - Commands:
 - '**Dir**' fully displaying the CD contents
 - '**CD**' able to change directories
 - '**Copy**' able to copy from the CD to other targets
 - '**Type**' able to display file contents for textual files
 - '**Eject Disc**' capability

Note: The 'Eject Disc' function is only supported in Enhanced Mode with the Enhanced Options menu.

DVD support for the 4690 Operating System

The 4690 Operating System supports several hardware devices, one of which is the DVD multi-burner hardware device. This supported hardware device is able to read and write several optical media (CD-RW, DVD-RW, DVD+RW and DVD RAM) that use the UDF (Universal Disk Format).¹

With the 4690 Operating System supporting the UDF file system, the user can use the regular commands supported by other file systems, such as `dir`, and `copy`. Due to the many versions of the UDF file system, the 4690 Operating System only supports optical media written and formatted by the 4690.

Even though regular commands and operations work using UDF, these operations reflect some characteristics particular to the media and the file system. Therefore, the same drive assignment (**p:**) will be shared as with the previous CD unit support. The 4690 operating system automatically selects the file system support (UDF versus ISO 9660) based upon the optical media contents and format.

The current support for the ISO 9660 file system remains unchanged. Line commands like `dir`, and `copy`, and others, are unsupported with this file system. USB DVD and USB DVD Multi-burner units are supported by the UDF file system in Enhanced Mode only.

1. The Optical Storage Technology Association (OSTA) developed and maintains this format implementation of the ISO/IEC 13346 (also known as ECMA-167).

Important

CD media formatted with UDF is different (in terms of file system support) than CD media formatted with the ISO 9660 file system: UDF supports line commands, the latter does not. Also, UDF CD media are not bootable whereas you can use ISO 9660 media to boot and install the 4690 operating system.

Using optical media formatted with the UDF file system

An optical media formatted with UDF file system can be used as any other file system with the following characteristics:

- The operating system provides the Optical Drive utility (see Chapter 18, “Using the optical drive utility,” on page 393) to allow you to transfer data between the store controller's hard disk drives and CD/DVD drives with write-capable CD/DVD media.
- The 4690 UDF media only support long names that contain A through Z and .. The maximum path length that can be used from the command line (including the drive designation) is approximately 106 characters.
- Whenever writing to the optical media, the tray of the optical unit is locked until the command `unlockp` is issued. This condition occurs because the optical media save data and directory structures in the cache memory. Also, some commands issue temporary files that lock the open tray. If the media is locked inside the optical unit and the media tray will not eject, then issue the `unlockp` command before trying to open the tray.
- The UDF line commands, `format` and `chkdsk`, are enhanced for optical media. `chkdsk` is not supported for optical media in Enhanced mode. If the media is locked inside the optical unit and the media tray will not eject, then use `unlockp` to unlock the open tray after a write has being attempted.
- The first time you use an optical media, always use the *long* format option (`format p: -L`). After this step is completed, you can issue a short format command. (Depending upon the optical media, a regular long format could take up to 60 minutes to format and a short format is much shorter in duration.) Additionally, always make sure that the long format option precedes a short format. This requirement is essential with media formatted with a non-UDF file system. For example, attempting a short format over a CD-RW with an ISO9660 file system will fail.

Notes:

1. Media formatted on 4690 Enhanced must not be written with 4690 Classic, and media formatted on 4690 Classic must not be written with 4690 Enhanced. Media formatted *and written* on 4690 Enhanced, however, may be read with 4690 Classic, and media formatted *and written* on 4690 Classic may be read with 4690 Enhanced.
 2. A DVD+RW formatted with the long option causes the led light to blink a couple of times and then the led light is off during the remaining formatting process.
- Any interruption during the write operation (formatting, running `chkdsk p: -F`) can damage the media state. Always perform a `chkdsk p: -F` after such occurrences and, if necessary, preserve the data in another location.
 - For detailed information on the UDF file commands, `chkdsk`, `format`, `unlockp`, see Chapter 7, “Command summary,” on page 153. The 4690 operating system supports only one UDF optical session per media.

- Redirecting (<, <<, >, >>) and piping (|) commands are troublesome when issued from the optical media p: drive. Toshiba recommends that you issue these commands from the hard drive unit. For example, use `c:\>chkdsk p: -F -V >> out.txt`, instead of `p:\>chkdsk -F -V >> out.txt`,
- Avoid leaving a UDF media inside of the optical unit when powering off the controller. For the system to recognize the media and file system after this happens, you must open and close the unit.
- Lower-quality UDF media might be unusable or marginally usable. The ability to read or write marginal media may also be inconsistent across different UDF-capable drives. In some cases, for example, a disc might not successfully complete a format operation on one drive, but might be successful on another.

Optical diskettes

The operating system provides the Optical Drive utility (see Chapter 18, “Using the optical drive utility,” on page 393) to let you transfer data between the store controller’s hard disk drives and the 3½-in. Optical Disk Drive’s diskettes.

You cannot perform an initial program load (IPL) from the optical drive because of hardware and software restrictions. Therefore, Supplementals and the Installation Diskettes should not be put on an optical diskette.

The optical drive does not interfere with other devices.

The optical disk uses a standard file allocation table (FAT) and cannot be partitioned.

Note: Not supported in Enhanced mode.

Handling diskettes, CDs, and DVDs

Some of your important information might be stored on diskettes, CDs, or DVDs, so you should learn to treat them with special care. For more details see the guide to operations for your specific store controller.

Write-protecting your diskettes

You can read data from a diskette or write data to a diskette. If you want to write data to a diskette, you need to know if it is write-protected. This is important to know because you can accidentally write over information that you might want to keep if your diskette is not write-protected.

Optical diskettes

If the square hole in the optical diskette is blocked, it is not write-protected.

To avoid writing over the information on an optical diskette, you can write-protect it by sliding the plastic tab in the corner of the diskette so that the square hole is open.

If you decide that later you want to write information on the optical diskette, slide the tab so that it blocks the hole.

See Figure 2 on page 13 for a diagram of how to write-protect your optical diskette.

3½-in. diskettes

If the square hole in the 3½-in. diskette is blocked, it is not write-protected.

To avoid writing over the information on a 3½-in. diskette, you can write-protect it by sliding the plastic tab in the corner of the diskette so that the square hole is open.

If you decide that later you want to write information on the 3½-in. diskette, slide the tab so that it blocks the hole.

See Figure 2 on page 13 for a diagram of how to write-protect your 3½-in. diskette.

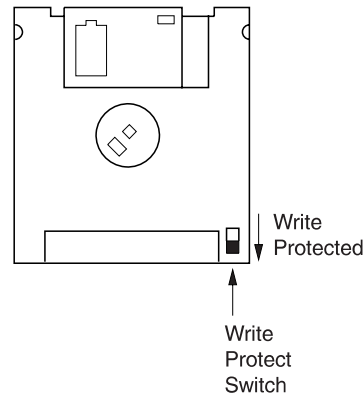


Figure 2. Write-protecting 3½-in. diskettes and optical diskettes

Formatting your diskettes

If you have purchased unformatted diskettes, you must format each diskette before you use it.

For information on formatting your diskettes, see “FORMAT Command” on page 214.

Keeping backup copies of diskettes

Your diskettes contain data that is important to your store’s operation and the operation of your system. To protect this data, *back up* (copy) your important diskettes. Once you create backup copies, use them and store the original diskettes. If your backup copies are lost or damaged, you can use your originals to make other backup copies.

“DISKCOPY command” on page 200 contains the step-by-step procedure for copying diskettes using the DISKCOPY command.

Note: You cannot use DISKCOPY with optical diskettes.

About your hard disk drive

Like your diskettes, your hard disk drive must be prepared before you can use it for the first time. If you try to use the hard disk drive before preparing it, you get a message:

INVALID DRIVE SPECIFICATIONS

For information on preparing your hard disk drive, see CPREP and DPREP in Chapter 20, “Using the disk rebuild utility,” on page 409. The preparation of your hard disk drive is performed during installation. For information on preparing your hard disk drive, see the *4690 OS: Planning, Installation, and Configuration Guide*.

If you want to upgrade your hard disk drive or replace a functioning hard disk drive, first IPL the machine with the Supplemental Option using the CD-ROM. Power off the machine when the Supplemental System Main Menu panel appears. Then remove the existing hard disk drive and install the new hard disk drive. If you did not CPREP or DPREP the new hard disk drive before installing it, IPL the machine using the Supplemental Option using the CD-ROM and then, perform CPREP or DPREP.

If you want to replace a hard disk drive that is not functioning, remove the hard disk drive and install a new one. IPL the machine with the Supplemental Option using the CD-ROM. Run CPREP or DPREP to format the new hard disk drive. If CPREP or DPREP was run on the new hard disk drive prior to installing it in the machine that had the non-functioning hard disk drive, you must run CPREP or DPREP again after the hard disk drive is installed.

Attention: Failure to follow the steps that are indicated can result in a corrupt format on the hard disk drive.

Precautions to use with your hard disk drive

- Keep magnets away from your hard disk drive.
- Avoid dropping, shaking, or jarring your hard disk drive.

Disabling the write verify function for your hard disk drive

For hard disk drives that have the Self-Monitoring, Analysis and Reporting Technology (SMART) capabilities, you can disable the write verify function and improve system performance. The write verify function causes a read of the data just written following any writing of data to the hard disk drive to force a cyclic redundancy check (CRC) of the data. If the data is corrupted, the operating system can rewrite the data to the hard disk drive.

This function was needed at one time as a reliability check for writing to hard disk drives; however, the reliability of hard disk drives has improved to the point where this function is optional.

If the file ADX_SDT1:ADXNOWRV.DAT file exists on a hard disk drive that has SMART capabilities, the write verify function will be disabled. If the file ADX_SDT1:ADXNOWRV.DAT file exists on a hard disk drive with no SMART capability, the write verify function will continue to be used to help protect the hard disk data. It is the responsibility of the user to create the file on the hard disk drive. The file does not need to contain any information. 4690 OS Enhanced Mode does not use this file.

Identifying drives

The system identifies each diskette, hard drive or USB memory key by one of the letters A to F. (F: is a special drive on Enhanced controllers.) An optical drive is assigned drive O. The CD/DVD drive is assigned drive P. The drives are not physically labeled but are assigned these values or letters by your system.

Note: F: drive is not distributed or copied on LDRU.

The system also enables you to create pseudo drives in memory. These are in addition to the diskette and hard disk drives. They are called *random access*

memory (RAM) disks, and you create them when you configure your system. For information about system configuration, see the 4690 OS: Planning, Installation, and Configuration Guide.

RAM disks enable you to use random access memory in the store controller or terminal as if it were a diskette or a hard disk drive. You must assign letters for the RAM disks as if they were a diskette or a hard disk drive. Use these letters for configuring RAM disks at the store controller: T, U, V, W, and Q (Enhanced RAM Disk). Use the letters X, Y, or Q (Enhanced RAM Disk) for RAM disk configuration in the terminal.

Using 4690 Virtual File System (VFS) drive settings, you can create two virtual drives to enable long file name support. During system configuration, you can enable long file name support on drive D. Long file name support is automatically enabled on drive C. Virtual drive M is always created for the C drive. Virtual drive N is created if you enable long file name support on drive D.

The operating system also allows you to create up to six logical drives to be used as mount points for NFS configuration. Drives G, H, I, J, K, and L can be defined as mount point data for a mount group during terminal and controller NFS configuration. For more information on terminal and controller NFS configuration, see “Network file system data” on page 53.

Consoles

Console is the term used to mean a keyboard and a display that are used together. The keyboard and display of the store controller together are called the *main console*. More than one console can be attached to a store controller. The store controller can have up to eight optional *auxiliary consoles*. Auxiliary consoles are additional keyboards and displays of specific model types that attach to the store controller.

There are a few differences between auxiliary and main consoles. The 316x auxiliary consoles have one fewer usable lines on their screens, which results in differences in displaying information.

You can perform most of the same functions at auxiliary consoles and main consoles.

The operating system includes an AUXILIARY CONSOLE CONTROL panel that enables you to look at and control auxiliary console applications from any console. See “Using system request keys” on page 42 for details on accessing this panel. See Table 5 on page 41 for a description of the specific functions available with the AUXILIARY CONSOLE CONTROL panel.

The 316x family of auxiliary consoles uses an ASCII keyboard that is different from the main console keyboard. The 315x family of auxiliary consoles can use either an ASCII keyboard or a PS/2® keyboard.

Operating modes on your system

Your operating system lets you define the operating modes you need for your system and for your operators. Specifying the modes of operation lets you control the security of your system, and helps you tailor your system to meet your specific personnel and resource needs.

You must specify a *user mode* for your system and an *application mode* for your operators. The user mode defines how many users can have active applications running on the system at one time. The application mode specifies the number of applications each operator is authorized to run at one time.

User modes

For the main or auxiliary consoles, the system can operate in either *single-user mode* or *multiple-user mode*. You specify the user mode you want when you configure your system. See the *4690 OS: Planning, Installation, and Configuration Guide* for information about system configuration.

In single-user mode, each console can have only one operator signed on to the store controller at a time. No other operator can have access to the store controller using that console as long as another operator is still signed on. Multiple operators can be signed on to the store controller in single-user mode, but they must each be signed on using a different console.

In multiple-user mode, more than one operator can be signed on to the store controller using the same console. An operator can disconnect from the store controller, which allows another operator to sign on using the same console. This operator can use the console to sign on to the store controller even though the first operator might have active applications running. Only one operator can have actual physical control of each console's screen and keyboard at any one time.

Application modes

Operators on the system can work in either *single-application mode* or *multiple-application mode*.

An application mode must be specified for each operator on the system. You specify the application mode for each operator in the *System Authorization File*. See the *4690 OS: Programming Guide* for information on the System Authorization File.

An operator authorized for single-application mode can run only one application at a time. Each application must complete running before another one can be started.

Multiple-application mode is the most powerful way to use your store controller. In this mode, an authorized operator can run several applications at the same time. There is no waiting for one application to complete before starting another. Each application that you start runs in its own *window*.

What are windows?

One of the most important features of the operating system is its capability to run more than one application at a time. Windows are only available in multiple-application mode. You can, for example, run a report, run a trace, and edit a file all at the same time using a single store controller. These applications operate as if each had its own keyboard and screen, even though only one application physically has control at any given time. Each application runs in its own *window*.

A window operates like the channels on a television. You have only one screen, but you can switch from window to window without ending the applications you are currently running. You can run a report on one window, switch to another window and write a memo, or switch to another window and check the status of one of your terminals.

Table 1 on page 17 lists how operators can use windows on one console in a multiple-user mode.

Table 1. Using windows in a multiple-user mode

Window	Operator	Application
1	A	Running a report
2	B	Running a report
3	C	Running a report
4	B	Running a store program
5	C	Running a store program
6	A	Editing a file

You open a window when you start an application. This process varies with the application you are starting. For example, when you select one of the options Primary Application, Secondary Application, or Command Mode from the SYSTEM MAIN MENU, the window opens immediately. If you select any of the other options on the SYSTEM MAIN MENU, the window does not open immediately. Instead, another menu appears; you select an option from that menu, and then the window opens.

Each application that you start runs in a window that belongs to you until you end the application or until it is complete. The window is closed when the application is complete or you end the application. Each operator can have up to eight active windows, or there can be a total of eight active windows at the same time.

Once a window is opened, you can change or close it using the WINDOW CONTROL panel. See “Using system request keys” on page 42 for details on how to access this panel and “Using function keys” on page 40 for information on how to use this panel.

Security on your system

Your system offers a safeguard against unauthorized access by requiring that all operators have a unique identification, a password, and authorization to perform certain functions on the system. Your system verifies this information before it allows an operator access to files. See the *4690 OS: Programming Guide* for information about how to set up System Authorization Files.

In addition to the standard system security, you can enable Enhanced Security. Enhanced Security provides more stringent password requirements and additional authorization requirements. See “Enhanced security” on page 21 for more information.

Notes:

1. The options included in this product can help your company address the PCI DSS requirements.
2. The customer is responsible for evaluation, selection, and implementation of security features, administrative procedures and appropriate controls.
3. PCI DSS is Payment Card Industry Data Security Standards.

With Version 6 Release 3, Directory Services has been added to 4690 OS controllers in order to support enterprise managed user IDs. The 4690 logon process has been enhanced so that it can access an LDAP directory for user ID

information, including password and authorization attribute data. See Chapter 14, “4690 Directory Services,” on page 341 for more information on Directory Services.

Your operator identification

The *operator identification* is your unique identification on the system. This ID can be a name that is specified by store procedures or by an authorized person at your store.

The ID can contain up to nine valid characters. Valid characters are any letters (A - Z), any numbers (0 - 9), or one of these special characters:

() { }

Directory Services IDs can be up to 72 characters long and use the ascii characters 0x21 through 0x7E.

When you type in your ID, you see it displayed on the screen.

Your password

Your system also requires you to enter a *password*. This password authorizes you to sign on and use your system.

Your password can be up to eight valid characters. Valid characters are the same as for your ID. For security, your password should have at least five characters. When Enhanced Security is enabled, a minimum password length is configured, ranging from 2 to 8 characters. Directory Services passwords can be up to 72 characters long and use the ascii characters 0x21 through 0x7E.

Unlike your ID, your password does not appear on the screen. You should select a password that someone else cannot easily guess, but one that is fairly easy for you to remember. Your password should be secret and should be shared only with authorized persons who need to know it.

If you enter an incorrect password when signing on, the sign-on panel reappears and prompts you to enter your operator ID and password again.

An authorized person at your store will tell you how often you must change your password and how to do it. When Enhanced Security is enabled, passwords can be configured to expire after a set number of days. You can change your expired password during sign-on.

Prompt to change expired password from SSH

When you attempt to sign on to Secure Shell (SSH) and your password has expired, SSH prompts you to change the password. Your SSH client software must support this function for it to operate.

Specially authorized operators

Some functions on your system can be used only by operators having special authorization. These functions include stopping another operator's window, checking background applications, or requesting system control functions. The authorization to use these functions is designated in the System Authorization File.

Each operator on the system has an *authorization record* in the System Authorization File. This file defines the functions that a specific operator can use. If you try to use a system function that is not authorized in your authorization record,

you are denied access and a message tells you that you are not authorized to use the selected function. For more information about the System Authorization File, see the *4690 OS: Programming Guide*.

When Enhanced Security is enabled, an enhanced authorization file is used. This file contains all settings in the original System Authorization File plus additional enhancements. For more information, see “Enhanced security” on page 21.

The following sections discuss the types of special operators on the system and the functions they are authorized to perform.

Privileged window operator

A *privileged window operator* is authorized to stop windows belonging to other operators using function keys. See Table 5 on page 41 for more details.

Background control operator

Background applications are programs that do not require a screen and keyboard and can start automatically when your system is powered on. Background applications do not run in a window. An example of a background application is a credit processing program.

A *background control operator* is authorized to use the BACKGROUND APPLICATION CONTROL panel to start or stop an application that is listed on that screen.

System control operator

The *system control operator* can use the options on the SYSTEM CONTROL FUNCTIONS menu panel. These options include setting system date and time, enabling or disabling the terminal storage retention function, and setting the system message display level.

Authorizations for menu options

Not all operators are authorized to use all of the options on the menu. Your system requires that each operator be authorized for menu options. Some operators might be authorized to make certain selections from a menu that other operators might not be authorized to use.

Console ID security

System configuration in 4690 OS V4 or higher versions allows the user to enable console ID lockout. If enabled, the user selects the number of invalid attempts before lockout, and the duration of the lockout. The number of invalid attempts must be between 3 and 10. The duration must be from 5 minutes to 60 minutes, or until midnight, meaning that the ID will be locked out until midnight.

The ID lockout status is not persistent. The status is kept in memory but is not written to disk, which means that a controller IPL will reset all IDs to be active. The intent of the lockout function is to thwart attempts at password guessing, so the time lost to IPL the controller to reset an ID is, in itself, a deterrent.

Unlock ID

The UNLOCKID utility can be used to unlock a user ID that has been locked through console ID lockout. To use this utility, type UNLOCKID xxxx where xxxx is a 1 – 9 character ID to unlock. The additional parameters passed are ignored

UNLOCKID will write a return code to stdout.

1 - Missing ID

- 2 - ID is too long
- 0 - Good return code

Note: Return code **0** does not mean the ID was removed from the list. It means **if that ID was on the list, it was removed.**

NFS security

4690 OS V2 and later versions provide authentication using a personal computer, NFS daemon (PCNFSD). The user name and password must be defined in the PCNFSD.DAT file in the ADX_SDT1 subdirectory. On the 4690 NFS Server, start PCNFSD.286 if you want to provide the authentication. Most PC-based clients will attempt to use the PCNFSD authentication.

If you are using the 4690 NFS Client to access a system that uses user ID and group IDs, access control is based on these IDs.

See the *4690 OS: Communications Programming Reference* for more information on PCNFSD authentication.

NetBIOS security

NetBIOS security can be enabled in the 4690 LAN stack as an optional feature. It applies to the entire controller and cannot be segmented to individual drives on the controller or to specific types of access to individual drives. NetBIOS security is supported only when an NT 4.0 workstation is used to access the 4690 controller.

Note: NetBIOS security is not supported on systems configured with Controller-to-Controller Communication over Internet Protocol (CCC/IP). Refer to the *4690 OS: Planning, Installation, and Configuration Guide* for more information about CCC/IP.

- The first method is NetBIOS security is not enabled. This is the default.
- The second method is to enable NetBIOS security to require a user ID and password when accessing resources. The 4690 DOS LAN Server code determines if security is enabled by checking for the existence of a file named ADX_SDT1:ADXLNSEF.DAT.
 - If this file does not exist, NetBIOS security is not enabled (default mode).
 - If this file does exist and the first line contains an ASCII value of 0, NetBIOS security is disabled and no access is allowed.
 - If this file does exist and the first line contains an ASCII value of 1, NetBIOS security is enabled and access is allowed using a user ID and password.
 - Any ASCII value other than 0 or 1 in the first line is read as a 0 value resulting in NetBIOS security being disabled and no access being allowed.

Creating the NetBIOS security file

After the first line in the ADX_SDT1:ADXLNSEF.DAT file, the following lines must contain an account. An account consists of one user ID and one password separated by a space on a single line.

To create the NetBIOS security file (ADX_SDT1:ADXLNSEF.DAT), from the controller in command mode, type: ADXLNSEF *uuuu pppp* where:

uuuu is the alphanumeric user ID (using any four (maximum) alphanumeric characters from 0 to 9 and from a to z)

pppp is the alphanumeric password (using any four (maximum) alphanumeric characters from 0 to 9 and from a to z)

Each user ID and each password can consist of one alphanumeric character to a maximum of four alphanumeric characters. Valid numeric characters are from 0 to 9 and valid alphabetic characters are from a to z and can be uppercase or lowercase. The user ID and password can contain all numeric, all alphabetic (uppercase and lowercase), or any mixture of these alphanumeric characters. The password is encrypted prior to being written to the ADXLNSEF.DAT file.

Note: Using 0000 (zero) for **both** the user ID and password is not allowed.

Additional functions include deleting an account, changing an account, listing all accounts, flagging any changes made to accounts, and changing the NetBIOS security mode.

- To delete an account, edit the ADX_SDT1:ADXLNSEF.DAT file using an editor (such as XE or DREDIX). Press the control/backspace key to delete the account in the file.
- To change an account, edit the ADX_SDT1:ADXLNSEF.DAT file using an editor (such as XE or DREDIX).
- To list or view all accounts or to flag changes made to accounts, use an editor (such as XE or DREDIX).
- To change NetBIOS security, edit the ADX_SDT1:ADXLNSEF.DAT file and change the first line to either 0 or 1, as applicable.

When using a Windows® operating system, do NET USE or Map Network Drive as follows:

- Type: NET USE ?: \\ADXLXyyN\z-uuuu-pppp where:
 - ? is the drive letter you want NT to assign to the controller connection.
 - yy is the 4690 node name
 - z is the drive on the 4690 controller to be accessed (do not attempt to access E:, F:, M: or P: drives on 4690 OS).
 - uuuu is the alphanumeric user ID (using any four (maximum) alphanumeric characters from 0 to 9 and from a to z).
 - pppp is the alphanumeric password (using any four (maximum) alphanumeric characters from 0 to 9 and from a to z).

Enhanced security

Enabling Enhanced Security	22
Disabling Enhanced Security	22
Enhanced Security password rules	22
Configuring Enhanced Security	23

Enhanced Security is an optional function that gives the 4690 OS a higher level of security. It is designed to help 4690 OS users meet the PCI DSS password requirements.

In the initial version of Enhanced Security, users either had total security access, or no access. Enhanced Security has been updated so that users can be given access to some features of Enhanced Security, and denied access to other features. The Enhanced Security updates allow broader access to Enhanced Security features without compromising store security.

Enhanced Security uses a new and encrypted authorization file, ADXEPW0F.DAT, which holds a maximum of 2,048 IDs and models. The original legacy file, ADXC0U0F.DAT, is still available. User IDs added through Enhanced Security are added to the new file and not the old file. An Enhanced Security API provides

application programs access to Enhanced Security features. Refer to the *4690 OS: Programming Guide* for more information about the API.

Enabling Enhanced Security

Enhanced Security is enabled in system configuration. From the System Configuration screen, select **Enhanced Security**. Select the box labeled **Enable enhanced security**.

When you enable Enhanced Security, you are required to enter a master ID and password. This information is used to create an initial master operator record in the Enhanced Security operator authorization file. This master ID has all authorization flags enabled, allowing it to perform any task on the 4690 OS. The master ID cannot be removed without disabling Enhanced Security. The password for the master ID is initially set as expired, and must be changed the first time it is used to logon. Save your changes, exit the System Configuration screen, and activate system configuration. When activation completes successfully, you must IPL all controllers to enable Enhanced Security.

When you enable Enhanced Security, you are also given the option to migrate operator IDs from the existing authorization file, ADXC SouF.DAT. To do this, select the box labeled **Convert legacy authorization file**. The passwords for all migrated IDs are set as expired. If an existing operator ID is the same as the master ID, it will be ignored during migration.

Notes:

1. If LDAP support is enabled, you cannot enable Enhanced Security.
2. If Auto Sign-on has been implemented, it is disabled when you activate Enhanced Security.
3. There must be an active acting master controller to change an expired password.

Disabling Enhanced Security

Enhanced Security is disabled in system configuration. From the System Configuration screen, select **Enhanced Security**. Deselect the box labeled **Enable enhanced security**. Save your change, exit the System Configuration screen, and then activate System Configuration. After activation completes successfully, you must IPL all controllers to disable Enhanced Security. When Enhanced Security is disabled, the enhanced operator ID file is no longer available. The 4690 OS reverts to its original logon process using the original legacy operator authorization file, ADXC SouF.DAT.

Important: Do not erase the ADXC SouF.DAT file and then disable Enhanced Security. You will not be able to sign on to your system.

Enhanced Security password rules

If your system has Enhanced Security enabled, there are several new rules for passwords used to log onto the system console:

- Passwords must contain both alphabetic and numeric characters.
- Passwords can not be reused within four cycles.
- Passwords can not include repeated characters of more than two. For example, a password cannot contain "AAA".
- Passwords can not include sequences of more than two. For example, a password cannot contain "456" or "abc".
- Passwords can not include the ID of the user.

Additional rules can be established when you configure Enhanced Security:

- The minimum password length can be configured to range from two to eight characters.
- Passwords can be configured to expire after a certain number of days.
- Passwords can be required to contain the following:
 - A lowercase letter
 - An uppercase letter
 - A number
 - A special character
- Passwords can be required to have four characters change from the previous password.

Note that there are certain situations where a rule is not applied. For example, users with the proper authorization can reset the password of any user. In this case the requirement to have four characters changed from the previous password is not applied. The user changing the password is not required to know the old password and the system does not save password information in an unencrypted format.

Configuring Enhanced Security

After Enhanced Security is enabled, it can be customized by authorized users through the Enhanced Security application. To start the application, select **Installation and Update Aids** from the main menu, then select **System Utilities**. This displays the Home Page. From the home page, select **Enhanced Security** from the links on the left side of the panel. The Enhanced Security application allows you to:

- Modify password settings
- Work with operator authorization records
- Work with Authorization models

The Enhanced Security application is not part of 4690 configuration. It has no concept of active and inactive settings. Changes take affect as soon as they are saved. Access to Enhanced Security information is granted in sixty minute intervals. When you use the Enhanced Security application, you must save any changes within sixty minutes or the session will time out and any changes will be lost.

Migration issues: When you install a level of 4690 OS that contains the Enhanced Security updates, Enhanced Security will check to see if the previous version was active. If it was, the following will occur:

- The Master ID authorization level will be initialized to 99.
- The authorization level for all other records will be set to zero.
- Records with Enhanced Security authorization will also be authorized to modify Password Settings.

Password settings: The Password Settings attribute, has been added to user authorization records. It controls user access to the Enhanced Security Password Settings screen. Users not authorized to run Enhanced Security cannot be authorized to use the Password Settings screen. Users with Enhanced Security authorization can optionally be granted access to the Password Settings Screen.

The following settings are configurable on the Password Settings page:

- Minimum Password Length – The minimum password length can range from two to eight characters.

- **Number of Days Until Passwords Expire** – Passwords can be configured to never expire. You can also choose to expire passwords in 1, 7, 10, 14, 30, 45, 60, 90, 120, or 180 days.
- **Number of Days to Display a Password Expiration Warning** – This sets the number of days that users will receive warnings that their password is about to expire. The warning appears when the user logs on. The warning can range from one to fifteen days.
- **Password Content Option** – Enabling this option requires new passwords to contain an upper case character, a lower case character, a number, and a special character. Existing passwords are not invalidated.
- **Four Character Change Option** – Enabling this option requires new passwords to contain at least four characters that were not in the previous password.

Note: The Password Content Option and the Four Character Change Option are not available unless the Minimum Password Length is set to four or more.

Authorization Manager: The Authorization Manager is used to maintain operator authorization records. These records contain the user ID, password, and authorization level for every user on your system. The Authorization Manager allows you to:

- Create new authorization records
- Delete authorization records
- Edit the authorization settings of records
- Rename the user ID of authorization records
- Set the user password of authorization records

Creating new authorization records: To create a new authorization record, select **New** on the Authorization Manager's main screen. You then enter the new user ID, and password. You also select a model from which to copy the authorization settings. If you have not created any models, you can use the default model. The default model has no authorization settings enabled. You are also given the option to set the password as expired. This will force the user to change their password the first time they sign on.

Deleting authorization records: To delete an authorization record, select the record to highlight it, and then select **Delete**. You will be asked for confirmation. If you delete your own authorization record, you will no longer be authorized to perform Enhanced Security operations.

Renaming authorization records: You can rename a user ID by selecting that ID and then selecting **Rename**. You are then prompted to enter the new user ID.

Setting passwords: You can change a user's password by selecting the user ID and then selecting **Set Password**. You are then prompted to enter the new password. You have the option of setting the new password as expired. This will force the users to change their password the first time they sign on.

Authorization levels: An authorization level has been added to user authorization records. The authorization level will range from 0 to 99, with 0 being the lowest level of authorization. The Master ID is created with an authorization level of 99. An authorization level affects a user's ability to perform Enhanced Security operations. The following operations are affected:

- **New** – New records are always based on a model or an existing record. When users create a new authorization record, the record used as a model must have

an authorization level less than or equal to their own. For example, a user with an authorization level of 40 could not create a new record by copying a record that has an authorization level of 60. The new record inherits the authorization level of the model, not the current user. The authorization attributes of the new record are the intersection of the user authorization attributes and the model authorization attributes (see “Editing authorization attributes”). When copying from the system default, the new record will have an authorization level of zero, and all authorization attributes are disabled.

- **Edit** – Users may only edit records that have an authorization level less than or equal to their own authorization level. For records they are allowed to edit, they can modify the authorization level, setting it to a value from 0 up to its own authorization level.
- **Delete** – Users may only delete records that have an authorization level less than or equal to their own authorization level.
- **Rename** – Users may only rename records that have an authorization level less than or equal to their own authorization level.
- **Set Password** – Users may only change the password of records that have an authorization level less than or equal to their own authorization level.

The new Enhanced Security features allow users to do very specific tasks without giving them full Enhanced Security access. For example, store managers can be given Enhanced Security authorization without the ability to modify the general password settings. By using levels, they can be granted permission to create and edit authorization records for cashiers, but denied access to other authorization levels. Users will not be able to increase their authorization, or create new authorization records with greater authorization than they currently have. The Enhanced Security Updates allow much more flexibility in the setup and maintenance of authorization records.

Editing authorization attributes: Initially, users with authority to work with authorization records had access to every available authorization attribute. Now, users can only modify the authorization attributes that they have authorization for. Other attributes will remain unchanged.

For example, Bob has Enhanced Security authorization, allowing him to create, edit, and delete authorization records. Bob does not have Apply Software Maintenance authorization. If Bob edits Sue’s authorization record, he cannot give her Apply Software Maintenance authorization if she does not have it. If she has Apply Software Maintenance authorization, Bob cannot remove that authorization. If Bob creates a new record, that record will not have Apply Software Maintenance authorization.

When a user creates a new authorization record it is based on a model. The authorization attributes enabled for the new record are the attributes that both the user and the model have enabled, in other words the intersection of the user attributes and the model attributes.

Editing authorization records: Editing an authorization record allows you to specify which operating system functions an operator is allowed to perform. The following authorization settings can be enabled or disabled for each operator:

- **System Modes**
 - **Multiple Application Mode** – Allows an operator to run more than one application at a time, using windows. System keys “w”, “s”, “p”, and “n” are enabled by this selection.

- Privileged Window Control – Allows an operator to stop an application window that is owned by another operator.
- Background Application Control – Allows an operator to stop, start, and check the status of background applications. System key “b” is enabled by this selection.
- System Control Functions – Enables the options listed under Store Control Functions, so that this operator can be authorized for one or more store control functions. System key “c” is enabled by this selection.
- Auxiliary Console Control – Allows an operator to check the status of auxiliary consoles from the main console. If the operator is also a Privileged Window Control operator, he can stop applications on an auxiliary console. System key “a” is enabled by this selection.
- Multiple Sign-on Mode – Allows an operator to sign on to more than one console concurrently.
- Store cControl Functions
 - Terminal Functions – Enables the options listed under Terminal Functions, so that this operator can be authorized to perform one or more of the functions.
 - Controller functions – Enables the options listed under Controller Functions, so that this operator can be authorized to perform one or more of the functions.
 - TCC Functions – Enables the options listed under TCC Functions.
 - Multiple Controller Functions – Enables the options listed under Multiple Controller Functions.
 - System Functions – Enables the options listed under System Functions.
 - Communication Functions – Enables the options listed under Communication Functions.
- Terminal Functions
 - Display Terminal Status – Shows the operator various information about the state of the terminal. This information contains such things as the state of storage retention, date, time, etc.

Note: This function only applies to terminal that are online at the time of the request. Terminals that are in standby, suspended, or powered off do not respond.

 - Enable Terminal Storage Retention – Terminal storage retention should be enabled for normal operations. It allows the terminal to quickly resume normal operation after a short power failure. Enabling storage retention also allows for programmable power commands for Advanced Power Management (APM) and Advanced Configuration and Power Interface (ACPI) to be executed.
 - Disable Terminal Storage Retention – Terminal storage retention should be disabled before the terminal power is disconnected. This function also forces all programmable power commands for APM and ACPI to power off the terminal.
 - Start Terminal Application – Allows you to start an application in a terminal.
 - Stop Terminal Application – Allows you to stop an application currently running in a terminal.
 - Java Application Functions – Displays the list of JAVA application functions available in the terminal.
 - Load Terminal Storage – Allows you to reload a terminal. This is normally done after software maintenance has been applied. On Enhanced Mode

terminals, this will only reload the 4690 OS code not the underlying hardware interface layer. To reload the hardware interface layer, use the Load Terminal Configuration Data option below.

- Dump Terminal Storage – Causes the contents of the terminal's storage to be placed in a file on the controller. Then the terminal storage is reloaded.
- Load Terminal Configuration Data – Causes one or more terminals to reload the configuration data for their previously loaded terminal number. On Enhanced Mode terminals, this will load both the underlying hardware interface layer and the 4690 OS code.
- Controller Functions
 - Display Controller Status – Shows the status of this store controller. If this controller is part of a LAN system, it also lists the logical connections to other controllers on the LAN.
 - Enable Controller RAM Disk – Allows the configured RAM disks to be allocated at the next IPL.
 - Disable Controller RAM Disk – Prevents the configured RAM disks from being allocated at the next IPL.
 - Load Controller Storage – Loads the memory of this (or a specified) controller's storage.
 - Dump Controller Storage – Makes a copy of the store controller's memory. A full or a partial dump can be chosen. A full dump will cause the store controller to reload.
- TCC Functions
 - Allow Store Controller Backup – Prepares this (or a specified) store controller to take control of another store controller's terminals, if needed.
 - Prevent Store Controller Backup – Prevents this (or a specified) store controller from taking control of another store controller's terminals. If the store controller is currently providing backup, it will release control of the other store controller's terminals.
 - Resume Store TCC Control – Returns control of the terminals to this store controller.
- Multiple Controller Functions
 - Activate Master Controller – For a LAN system, this function makes this controller assume the role of the Master Store Controller. A store controller can be configured as Master, but it does not act as Master until it is activated.
 - Activate File Server Controller – For a LAN system, this function makes this store controller assume the role of the File Server.
 - Deactivate Master Controller – For a LAN system, this function removes this store controller from the role of the Master Store Controller.
 - Deactivate File Server Controller – For a LAN system, this function removes this store controller from the role of the File Server.
- System Functions
 - Set System Message Level – The message level is a number from 1 to 5 that determines what messages are displayed on the SYSTEM MESSAGE display screen. When set to 1, only the most important messages will be displayed. When set to 5, all messages will be displayed.
 - Set System Date and Time – The date and time entered through this selection will be used throughout the system.
- Communication Functions
 - Enable Link – Allows you to enable a SNA link.

- Disable Link (No Force) – Allows you to disable a link when all applications using the link have completed normally.
- Disable Link (Force) – Allows you to disable a link immediately. Applications using the link might end abnormally.
- Display Link – Allows you to display the link status.
- Enable LAN for LU 6.2 Communications – Allows you to enable the LAN for use by LU 6.2 applications.
- Disable LAN for LU 6.2 Communications – Allows you to disable the LAN immediately for use by LU 6.2 applications.
- Display LAN Status for LU 6.2 Communications – Allows you to display the status of the LAN for use by LU 6.2 applications.
- SSH Secure Remote Logon – Allows you to logon to the system remotely using SSH.
- SSH Secure FTP Logon – Allows you to logon to the system using SSH Secure FTP.
- System Main Menu
 - File Utilities – Enables the options listed under File Utilities, so that this operator can be authorized to perform one or more of the functions.
 - Installation and Update Aids – Enables the options listed under Installation and Update Aids, so that this operator can be authorized to perform one or more of the functions.
 - Problem Analysis Data Collection – Enables the options listed under Problem Analysis Data Collection, so that this operator can be authorized to perform one or more of the functions.
 - Problem Analysis Reports – Enables the options listed under Problem Analysis Reports, so that this operator can be authorized to perform one or more of the functions.
 - Command Mode – Allows the operator to open a command prompt window, which allows them to type commands and start user programs.
 - Group and User number – Identifies the operator while running in Command Mode. These two number are used to determine file ownership and access rights in Command Mode.
 - Group 2 User 1 has owner access rights to application files. Initially owner rights are (READ, WRITE, DELETE).
 - Group 2 User not 1 has group rights for application files. Initially group rights are (READ ONLY).
 - Groups 3 through 254 with any user can not access application files.
- File Utilities
 - Keyed File Utilities – Allows you to examine statistics on current keyed files, and create new ones.
 - Display/Alter File Data – Allows authorized users to display and edit files in a binary format.
 - Distributed File Utilities – Allows you to work with distributed files, and the file distribution exception logs.
 - File Compress/Decompress – Allows you to compress and decompress one or more files.
 - Streaming Tape Drive Utilities – Allows you to backup and restore files using a streaming tape drive.
 - Optical Drive Utility – Allows you to backup and restore files using an optical disk.

- Installation and Update Aids
 - Change Configuration Data – Allows you to make configuration changes to your store system.
 - Report Legacy Configuration Data – Allows you to show configuration information in a report.
 - Change Input Sequence Table Data – Allows you to define tables that tell the operating system how to react to possible inputs.
 - Report Module Level – Allows you to obtain a report of the module levels of each software product that is installed on the controller.
 - Apply Software Maintenance – Allows you to apply and manage changes to the 4690 Operating System, and other supported application programs.
 - Build Software Maintenance Control File – Allows you to build a control file to use with the Apply Software Maintenance utility.
 - System Message Audible Alarm Functions – Allows you to manage a list of messages that cause the controller's alarm to sound when they are logged.
 - System Utilities
- System Configuration
 - System Settings – Allows you to make configuration changes to the System Settings section of System Configuration.
 - Java Configuration – Allows you to make configuration changes to the Java Configuration section of System Configuration.
 - System Security – Allows you to make configuration changes to the System Security section of System Configuration.
 - FTP User Definitions – Allows you to make changes to FTP user definitions. Note that this authorization only applies if you have encoded your FTP user definition file.
- Problem Analysis Data Collection
 - Start Trace Data Collection – Allows you to start logging various disk and communication events.
 - Start Performance Data Collection – Allows you to start logging performance information on a terminal or the controller.
 - Stop Trace Data Collection – Allows you to stop the logging of various disk and communication events.
 - Stop Performance Data Collection – Allows you to stop the logging of performance information.
- Problem Analysis Reports
 - Scan System Log Data – Allows you to examine the controller's system log.
 - Format System Trace Data – Allows you to report on previously collected system trace information.
 - Format Performance Data – Allows you to report on previously collected performance data.
 - Format Dump Data – Allows you to view formatted output of a terminal or controller dump.
 - Create Problem Analysis Diskette – Allows you to copy problem analysis data to diskette.
- User Defined Attributes – Eight user defined attributes are available. User programs can query the state of these attributes for any reason they wish.

Creating authorization model records: Authorization models are authorization records without a password. The user ID serves as the “name” of the model.

Authorization models are used when creating authorization records. They are very useful when there is a need to create many records with the same authorization level. Authorization Model records are created and maintained just like authorization records, except there is no ability to set or change passwords.

Secure Shell

Secure Shell (SSH) is a standards-based application and an associated set of networking protocols. It provides secure, encrypted communications over an untrusted network (for example, the Internet) between servers and clients. Refer to the *4690 OS: Communications Programming Reference* for information about SSH.

IP Security

IP Security (IPsec) enables secure communications over untrusted networks by securing data traffic at the IP layer. This allows individual users or organizations to secure traffic for all applications without having to make any modifications to the applications. Therefore, the transmission of any data (such as, file transfer or application-specific company data) can be made secure. Refer to the *4690 OS: Communications Programming Reference* for information about IPsec.

4690 secure logon through LDAP server

4690 V4 or higher supports secure logon at the store controllers by implementing a central LDAP server. Both 4690 TCP/IP and Java 2 must be installed on your system.

Note: This function is not the same as Directory Services, which uses OpenLDAP to establish a directory on 4690 controllers.

Note: This discussion assumes that the user is familiar with basic LDAP and SSL concepts.

Without LDAP enabled, user logons are directed to the file, ADX_IDT1:ADXCSOUF.DAT, for user IDs, user passwords and user authorization information. When LDAP is enabled, logons are directed to the LDAP server as an LDAP query.

Note: If Enhanced Security is enabled, you cannot enable LDAP.

- If the user ID is located within the configured search domain of the server's Directory Information Tree (DIT), the user password and authorization information is returned to the 4690 client and this information is used for the active logon session.
- If the server is unreachable, then a local logon is made available. The logon is directed to the ADX_IDT1:ADXCSOUL.DAT file, which must be created by the user prior to accessing the local logon option. A utility, ADXULDIF.286, which creates this file for the user, is shipped with 4690 optionals.

It is the responsibility of the enterprise administrator to limit those who can log on through the local logon option.

LDAP server requirements

There are several requirements on the server side in order for secure logon to function properly.

- For each entry in the server's DIT that represents a 4690 user, these attribute:value pairs must be present with each pair on a separate line:
 - uid:4690 user ID

- uidNumber:encrypted 4690 user password in character format
- destinationIndicator:14-byte 4690 user authorization in character format

This is an example of a possible string for user ID 1:

```
uid:1
uidNumber:373438383439383520
destinationIndicator:01030000000000000000000000000000
```

- The user must edit the LDAP configuration file of the server to include the schemas that contain definitions for the attribute:value pairs.
- The server must implement Secure Socket Layer (SSL) to communicate with the 4690 clients.

ADXULDIF utility

Adxuldif.286 is shipped as part of the install/asm package and is located under the 4690OPT directory. The utility aids customers in producing user data from ADX_IDT1:ADXC SouF.DAT, the existing 4690 file. This data can then be ported to the LDAP server to create entries within the DIT.

The utility has the following usages:

adxuldif -a

Creates LDIF type entries for all users found in the ADXC SouF.DAT file.
Writes output to the ADX_IDT1:ADXUSERS.DAT file.

adxuldif -l <userID...>

Creates LDIF type entries for the user IDs listed on the command line if the IDs are found in the ADXC SouF.DAT file. Writes output to the ADX_IDT1:ADXUSERS.DAT file.

adxuldif -s <userID>

Creates the ADX_IDT1:ADXC SouL.DAT file with the user's information found in the ADXC SouF.DAT file.

The ADXC SouL.DAT file is needed to be able to use local logon if the LDAP server is unavailable.

Command line logging

The command line logging function allows the logging of commands at the POS controller. The function logs the command, date, time, and operator ID to a special log file. Other function includes:

- Logging of file access from DREDIX and the XE editor
- Logging of file writes by ADXC SJOL
- Logging of SFTP session commands

Master log file ADXC LLOG.DAT

ADXC LLOG.DAT is the master log file. The directory, ADX_CLOG will hold all local, log files. Each command- line session contains its own log file, however this file is temporary and will be concatenated onto the master log file and deleted when the controller IPLs.

Timestamp format: The individual log files consist of the following:

UUUUUUUU.Cnn

UUUUUUUU is the one to eight character user ID and nn is 00-99, the next sequential value available in the directory. As a customer, you control the size of the master log file.

Header line

OpID PID NodeID Store#

Command line data

[Timestamp] {command line data}

Example:

```
OpID=99999999 PID=003E NodeID=CC Store#=0001
07:35:56.46 Thu 05-19-2005 ADX_UPGM:AUTOEXEC.BAT
07:35:56.51 Thu 05-19-2005 ECHO OFF
07:35:56.55 Thu 05-19-2005 PROMPT $p$g$ev
07:35:56.59 Thu 05-19-2005 ECHO This message displayed by ADX_PGM:AUTOEXEC.BAT
07:35:56.62 Thu 05-19-2005 ADX_UPGM:AUTOEXEC.BAT
07:35:59.79 Thu 05-19-2005 cd adx_clog
07:36:01.13 Thu 05-19-2005 dir
07:36:12.94 Thu 05-19-2005 cd adx_spgm
07:36:29.27 Thu 05-19-2005 cd ..
07:36:34.46 Thu 05-19-2005 cd adx_spgm
07:36:35.89 Thu 05-19-2005 dir
07:36:39.65 Thu 05-19-2005 exit
```

ADXCSJ0L logging

ADXCSJ0L creates a log file in the ADX_CLOG directory each time it is invoked and if the ADXCMDLG logical name is defined. Changes to data by ADXCSJ0L are normally within a sector. The 4690 OS writes an entire sector (or one half as is displayed) when the WRITE function key is pressed and ASCII is displayed.

The log files consist of:

UUUUUUUU.Jnn

UUUUUUUU is the one to eight character user ID; nn is 00-99, the next sequential value available in the directory.

Operator ID, Filename, Sector #, and data

The ID of the operator, filename, sector number and any additional data.

Header line

The header line consists of [Timestamp] OPID NodeID [Text message]

Changed sector data

The changed sector data follows the header line.

Example:

```
17:35:02 Tue 08-16-2005 OpID=1 NodeID=R0 CREATE ADXCSJ0L LOGGING FILE

Filename=BB.ERR Sector#=0000

4C4F4144 494E4720 44454641 554C5420 LOADING DEFAULT
50724F50 45525449 45532046 494C450D PROPERTIES FILE.
0A636F6D 2E69626D 2E4F5334 3639302E .com.ibm.OS4690.
466C6578 6F734578 63657074 696F6E3A FlexosException:
20464C45 584F5345 58434550 54494F4E FLEXOSEXCEPTION
20287263 3D307838 30393031 35323229 (rc=0x80901522)
0D0A0961 7420636F 6D2E6962 6D2E4F53 ...at com.ibm.OS
34363930 2E4E6174 69766534 3639302E 4690.Native4690.
72657475 726E4576 656E7473 284E6174 returnEvents(Nat
69766534 3639302E 6A617661 3A353433 ive4690.java:543
290D0A09 61742063 6F6D2E69 626D2E6A )...at com.ibm.j
```


706F732E	73657276	69636573	2E4F5334	pos.services.OS4
3639302E	5072696E	74657243	6F6D706C	690.PrinterCompl
6574696F	6E546872	6561642E	72756E28	etionThread.run(
5072696E	74657243	6F6D706C	6574696F	PrinterCompleti
6E546872	6561642E	6A617661	3A313134	nThread.java:114

DREDIX and XE logging

The 4690 operating system logs the read/write access from the editors, but not each change. The DREDIX editor creates a BAK file that can be used to compare the changes made to the files.

The format of the individual log files for the DREDIX and XE editors are as follows:

UUUUUUUU.Xnn

UUUUUUUU is the one to eight character user ID; nn is 00-99, the next sequential value available in the directory.

SFTP logging

The SFTP log files will be in c:\adx_clog as *USERNAME.F##*, where *USERNAME* is the ID that the SFTP server used to establish the 4690 OS SFTP session. Also, for a 4690 OS SFTP client, the user ID is the user ID used to start the SFTP client. All actions performed on files are logged.

System messages

System messages W990 and W991 indicate that a file was read or written by ADXCSJOL.

Data security in dump data

Payment card information in dumps can be secured via System Configuration. When this setting is enabled, all potential payment card numbers contained in controller and terminal memory dumps and Linux process core files are obscured.

From the System Configuration screen, select Data Security. Select / deselect the box labeled **Protect payment card information in dump data**. Save your change, exit the System Configuration screen, and then activate System Configuration. After activation completes successfully, you must IPL all controllers and terminals to enable / disable Data Security.

Payment card protection is supported for enhanced controller dumps, enhanced controller/terminal dumps, enhanced/classic terminal dumps and enhanced core dumps.

Securely Deleting Files

With a 4690 operating system, you can also use the ADXSDEL command to erase files. The ADXSDEL command offers a higher level of security than just using the ERASE command alone. The ADXSDEL.286 command overwrites at least three times a targeted file before removing it from its directory entry; therefore, deleting sensitive information more securely. See “ADXSDEL command” on page 163 for a description of the ADXSDEL command.

Using menus

Your system uses menus to help you perform your tasks. These menus appear at your main or auxiliary console and show a list of options that represent tasks you can perform.

When you sign on, the SYSTEM MAIN MENU appears:

```
CS01S000          SYSTEM MAIN MENU

Select one of the following:

    1 Primary Application
    2 Secondary Application
    3 File Utilities
    4 Installation and Update Aids
    5 Problem Analysis Data Collection
    6 Problem Analysis Reports
    7 Command Mode

Type your selection number, then press ENTER.  _

Message Line
F1Help F2      F3      F4      F5      F6      F7      F8      F9Signoff
Status Line
```

Notes:

1. The status line is not shown on the 316x auxiliary console screen.
2. On your panel, the actual titles of the applications appear, instead of the words “Primary Application” and “Secondary Application” that are shown here. If no applications are configured, lines 1 and 2 are blank after the selection number.

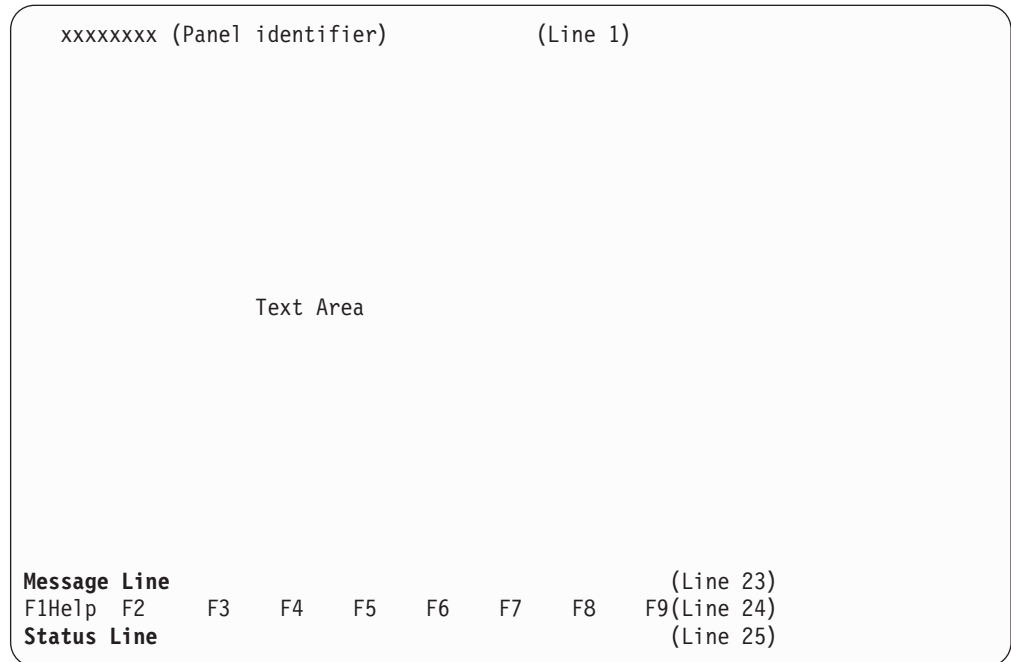
This menu contains a list of options representing groups of tasks you can perform on the system. You are prompted to select one of the options from the menu.

If you select Primary Application (1) or Command Mode (7), the applications begin running immediately. If you select any other option, another panel appears and you are prompted to select an option from a second menu. Selecting an option from this second menu causes another panel to appear. This time you are prompted to either select an option or enter data. Each panel tells you whether to enter data or select another option.

Some menus can show a *default value* when you are prompted to enter a value or select an option. This default value means that your system selects the value shown on your screen if you do not specify another value.

Understanding panel information

The menu screen is 25 lines long. The last three lines provide status and message information.



Note: The status line is not shown on the 316x auxiliary console screen.

Message line

The *message line*, line 23, displays messages from the application you are running. It is highlighted or red when a message appears.

Understanding messages

Several different types of messages appear at your store controller display. Some messages give you information or prompt you to enter data, some indicate a problem requiring action to be taken, and some contain identifiers while others do not. For information on message types and on messages that are related to problems, see the *4690 OS: Messages Guide*.

Function keys line

The *function keys line*, line 24, shows the function keys that are active on a panel. Function keys let you move from panel to panel, display help information, sign off from the system, and perform other system functions. Active keys are indicated by a highlighted or blue function that is displayed immediately to the right of the function key. Function keys with no function that is displayed are not active for that panel.

The function keys displayed on your panel match the function keys that are found on your keyboard. See “Using function keys” on page 40 for information on how to use these function keys.

Status line

The *status line*, line 25, gives you status information about your system. This line displays the time of day, gives information about active windows (in multiple-application mode only), or lets you know that a system message can be displayed. A system message gives you information about the events taking place on your system.

Status information appears as follows:

time=HH:MM Current Window=x Number of Windows=y SYSTEM MESSAGE AVAILABLE

Where:

- HH:MM** Shows the time in hours and minutes.
- x** Tells which window is currently active (can have values 0 through 8). It is set to 0 when no window is active.
- y** Indicates the number of windows that are owned by the current operator (can have values 0 through 8). It is set to 0 when no window is owned by the current operator.

Window status information appears only in multiple-application mode. The status line does not appear on the 316x auxiliary consoles.

SYSTEM MESSAGE AVAILABLE tells you that a system message has been written to the SYSTEM MESSAGE DISPLAY panel since the last time you displayed it. See “Using system request keys” on page 42 for information on how to access the SYSTEM MESSAGE DISPLAY panel, and see “Using function keys” on page 40 for information on how to use the panel. The system message indicator is cleared when you display this panel.

Using special keys

Your console’s keyboard has keys that you can use to do certain functions. This section describes the functions of those keys.

Using keys to control panel information

Some of the keys on your keyboard allow you to control the information appearing on your screen. Table 2 on page 36 shows the key combinations that are needed on a PS/2 or PC keyboard.

Note: Table 3 on page 38 shows the key combinations that are needed at an auxiliary console ASCII keyboard.

PS/2 and PC keyboard key functions

See the guide to operations of your specific store controller for the location of the special keys on your PC or PS/2 keyboard.

Table 2. Personal computer keyboard key functions

Function	Key	Description
PAGE UP	PgUp, Page Up	Displays the preceding page of information on your panel when there are multiple pages to be displayed.
PAGE DOWN	PgDn, Page Down	Displays the next page of information on your screen. If you are entering data in a series of pages, this key saves your data from page to page until you process the data by pressing the Enter key.
BACKSPACE	← Backspace	Moves the cursor to the preceding character or space on the line and deletes that character.

Table 2. Personal computer keyboard key functions (continued)

Function	Key	Description
PRINT SCREEN	PrtSc, Print Screen (on PS/2 Enhanced Keyboard)	Prints all the information that is currently on your screen.
ESCAPE	Esc	Ends a function. Any data entered since the last time you pressed Enter is lost.
HOME	Home	Returns your cursor to the first entry or selection field of your panel. Note: On some panels, pressing Shift and then Tab gives you the same result as the Home key. Using this key combination can also cause the cursor to move to the preceding field or entry.
END	End	Moves your cursor to the last entry or selection field on your panel.
ENTER	Enter <↓	Completes the entry of data on a panel and processes the data you entered. The data entered is accepted and processed if there is no error. If you are selecting the default on a panel, pressing this key causes the default selection to be processed.
CURSOR DOWN	↓	Moves the cursor down to the next line.
CURSOR UP	↑	Moves the cursor up to the previous line.
CURSOR RIGHT	→	Moves the cursor one character to the right.
CURSOR LEFT	←	Moves the cursor one character to the left.
SYSTEM REQUEST	Alt+SysRq	Displays the SYSTEM KEYS panel.
FUNCTION KEYS	F1 to F10	The functions are defined at the bottom of each panel.
BREAK	Ctrl+Break	Manually ends some functions.
TAB	→	Moves the cursor to the next field or position.
BACKTAB	Shift+ ←	Moves the cursor to the preceding field or position.
INSERT	Ins, Insert	Enables characters to be inserted at the cursor.
DELETE	Del, Delete	Causes the character at the cursor to be deleted. The remaining characters shift left.

Table 2. Personal computer keyboard key functions (continued)

Function	Key	Description
REFRESH (For auxiliary consoles only)	Ctrl+5	<p>Reinitializes the auxiliary console if a disconnect is forced while you are signed on. (For information on disconnect, see Table 5 on page 41.)</p> <p>For auxiliary consoles attached to the controller by RS-422 (EIA 422) lines, you must press Ctrl+5 after the auxiliary console has been switched off and switched on again, or if the connection to the controller has been broken.</p> <p>For auxiliary consoles attached by RS-232 (EIA 232) lines, the operating system automatically performs the REFRESH function when the auxiliary console is turned on or reconnected to the system.</p>

Auxiliary console ASCII keyboard key functions

The keyboard of the auxiliary console ASCII keyboard requires some key combinations that are different from the PS/2 keyboard. Table 3 on page 38 contains the key combinations for this keyboard. See the *3161/3164 Guide to Operations* or the *3151 Guide to Operations* for the location of the special keys on your auxiliary console.

Note: The 315x family of auxiliary consoles can support either a PS/2 or ASCII keyboard. Table 3 on page 38 lists the key combinations for the ASCII keyboard. Table 2 on page 36 describes the PC or PS/2 keyboard key combinations.

Table 3. Auxiliary Console ASCII Keyboard key functions

Function	Key	Description
PAGE UP	Send Line	Displays the preceding page of information on your panel when there are multiple pages to be displayed.
PAGE DOWN	PA3 (Ctrl+ Keypad 3)	Displays the next page of information on your panel when there are multiple pages to be displayed. If you are entering data in a series of pages, this key saves your data from page to page until you process the data by pressing Enter.
BACKSPACE	<—Backspace	Moves the cursor to the preceding character or space on the line, deleting that character.
PRINT SCREEN	SEND	Lets you print all the information that is currently on your screen, including the information on the message and status lines.
ESCAPE	Esc	Ends a function. Any data entered since the last time you pressed Enter is lost.

Table 3. Auxiliary Console ASCII Keyboard key functions (continued)

Function	Key	Description
HOME	Home	Returns your cursor to the first entry or selection field of your panel. Note: On some panels, pressing Shift , and then Tab gives you the same result as the Home key. Press and hold Shift and Tab together.
END	PA1 (Ctrl+ Keypad 1)	Moves your cursor to the last entry or selection field on your panel.
ENTER	Return <↵	Completes the entry of data on a panel and processes the data you enter. The information you entered is accepted and processed if there is no error. If you are selecting the default on a panel, pressing this key causes the default selection to be processed.
CURSOR DOWN	↓	Moves the cursor down to the next line.
CURSOR UP	↑	Moves the cursor up to the previous line.
CURSOR RIGHT	→	Moves the cursor one character to the right.
CURSOR LEFT	←	Moves the cursor one character to the left.
SYSTEM REQUEST	Jump	Displays the SYSTEM KEYS panel.
FUNCTION KEYS	F1 to F12	The functions are defined at the bottom of each panel.
BREAK	Ctrl+C	Manually ends some functions.
TAB	Tab →	Moves the cursor to the next field or position.
BACKTAB	ShiftTab →	Moves the cursor to the preceding field or position.
INSERT	Insert	Allows characters to be inserted at the cursor.
DELETE	Del	Causes the character at the cursor to be deleted. The remaining characters shift left.
REFRESH	Ctrl+5	Reinitializes the auxiliary console if a disconnect is forced while you are signed on. (For information on disconnect, see Table 5 on page 41.) For auxiliary consoles attached to the controller by RS-422 (EIA 422) lines, you must press Ctrl+5 after the auxiliary console has been switched off and switched on again, or if the connection to the controller has been broken. For auxiliary consoles attached by RS-232 lines, the operating system automatically performs the REFRESH function when the auxiliary console is turned on or reconnected to the system.

Java interface key functions

For systems using a Java interface and which do not have a mouse attached, the following keyboard controls are available.

Note: If an entry field within the Java configuration panel is highlighted, pressing any entry key (such as Backspace) or any alpha or numeric key on the keyboard will delete the highlighted text.

Table 4. Java interface key functions

Function	Key	Description
NAVIGATE FORWARD	Tab →	Move focus to the next field or item.
NAVIGATE FORWARD	Ctrl + Tab	Navigates forward out of a data entry field.
NAVIGATE BACKWARD	Shift+Tab ←	Move focus to the previous field or item.
SELECT	Spacebar	Selects a radio button, activates a Java button, selects an item from a combination box, or selects the function that currently has focus.
CURSOR RIGHT	→	Moves the cursor one character to the right or expands a menu item.
CURSOR LEFT	←	Moves the cursor one character to the left or compacts a menu item.
CURSOR DOWN	↓	Moves the cursor down to the next line.
CURSOR UP	↑	Moves the cursor up to the previous line.
HELP	F1 or Shift+F1	Displays the help text panel for that topic. Note: To close the help panel, press the spacebar when the Close button has focus.
ACTIVATE TITLE BAR	Alt	Activates or deactivates the title bar menu.
SYSTEM REQUEST	Alt+Print Screen	Displays the System Request keys.
PRINT SCREEN	Print Screen	Sends an image of the desktop to the print spooler.
OPEN WINDOW LIST	Ctrl+Esc	Opens the window control list.
TITLE BAR	F10+↓	Displays the title bar pulldown menu.

Using function keys

Each panel displays the function keys that are active on that panel on the function keys line (line 24). The key and the function it performs on the panel both appear. See “Function keys line” on page 35 for a description of this line. The function of some keys can vary according to the panel on which they appear. For example, F9 has the Disconnect function on one panel and the Sign-off function on another.

The available function keys vary according to the application mode of the operator. Table 5 on page 41 summarizes function keys available in single-application and multiple-application mode with the exception of SWAP and STOP. These two functions are available only in multiple-application mode.

Table 5. Function keys used in single- and multiple-application Mode

Function	Function key	Description
Display HELP panel	HELP or Help	A HELP panel for the menu option you are using appears. When you want to exit HELP, press HELP again.
Page back 10 pages	Pg-10	Moves back 10 pages in a multiple-panel report, or to the top of the report if there are fewer than 10 pages.
Page down 10 pages	Pg+10	Moves forward 10 pages in a multiple-panel report, or to the bottom of the report if there are fewer than 10 pages.
Return to the previous option panel without saving your changes	QUIT or Quit	The last panel from which you selected an option appears. Any changes you made since the last time you pressed Enter are lost.
Remove inactive background applications	Clear	Removes all background applications from the BACKGROUND APPLICATION CONTROL panel that are not active and are not permanent. Permanent background applications are those that exist when the system is powered on.
Display the newest messages	Newest	The SYSTEM MESSAGE DISPLAY panel shows the most recent messages written to the message file.
Display the oldest messages	Oldest	The SYSTEM MESSAGE DISPLAY panel displays the oldest messages.
Disconnect from the system without affecting your applications	Disconnect	Displays the sign-on panel; you must sign on again to gain access to the store controller. Your applications continue to run until they finish or need keyboard input.
Sign off from the system	Sign-off	Displays the sign-on panel and ends all applications running. A message warns you that all your active applications will be ended if you sign off.
Swap or switch to another window	Swap	<p>Passes control to the window you specified on the WINDOW CONTROL menu. The panel for the specified window appears, and you can enter data to the application running on the window.</p> <p>You cannot swap to a window owned by another operator.</p>
Stop a specific application running in a window	Stop	Causes the application running in the window you specified to stop. The application is ended and the window becomes available for another operator to use. You can stop an application owned by another operator, if you have privileged window control.
Start a background application running	Start	Causes the application you specified on the BACKGROUND APPLICATION CONTROL menu to run.

Table 5. Function keys used in single- and multiple-application Mode (continued)

Function	Function key	Description
Refresh the panel	Refresh	Causes the AUXILIARY CONSOLE CONTROL menu to be updated.
View auxiliary console WINDOW CONTROL SCREEN	View	Accesses a copy of the WINDOW CONTROL SCREEN of the auxiliary console you specified on the AUXILIARY CONSOLE CONTROL menu.
Stop auxiliary console applications and sign off operator	StopAll	Causes all applications running on the auxiliary console you specified on the AUXILIARY CONSOLE CONTROL menu to stop and then signs off the auxiliary console operator.

Using system request keys

System requests are entered by using the SYSTEM REQUEST (SysRq) key to access the SYSTEM KEYS menu. See “PS/2 and PC keyboard key functions” on page 36 for a description of the key for the console you are using. On this menu, the system request functions are denoted by a letter key followed by a description of the key’s function. You can access the SYSTEM KEYS menu from any other menu. The system request functions that are available to you vary according to the application mode you are using.

Table 6. System request key descriptions

Function	Key	Description
View/control applications on auxiliary console	A or a	Available only on the main console; this function enables you to view and control the applications on auxiliary consoles.
Display BACKGROUND APPLICATION CONTROL menu	B or b	Displays the BACKGROUND APPLICATION CONTROL menu. You must be authorized as a background control operator to use this menu.
Display STORE CONTROL FUNCTIONS menu	C or c	Displays the STORE CONTROL FUNCTIONS menu. (See Chapter 3, “Store controller operations,” on page 81 for information on store controller functions.) You must be authorized to access this menu.
Display SYSTEM MESSAGE DISPLAY panel	M or m	Displays the SYSTEM MESSAGE DISPLAY panel. This panel lets you see the messages in the system message file. The most recent messages appear. You can view all of the messages in the file using the function keys described on the panel. If a new message is received while you are looking at this panel, the new message does not appear until you press one of the function keys.
Move to next window	N or n	Moves you to the next window.
Move to preceding window	P or p	Moves you to the preceding window.
Display SYSTEM MAIN MENU	S or s	Displays the SYSTEM MAIN MENU.

Table 6. System request key descriptions (continued)

Function	Key	Description
Switch to terminal	T or t	Switch to terminal (controller/terminal only). When operating a controller/terminal with a shared display and keyboard, you can switch control of the keyboard and video display from the store controller mode to the terminal mode by pressing the t key. To switch back to the store controller mode, press the SysRq key to access the SYSTEM KEYS menu. If you are not signed on to the store controller you can still use the SysRq key from the signon panel to access the SYSTEM KEYS menu to switch to the terminal mode.
Display WINDOW CONTROL panel	W or w	Displays the WINDOW CONTROL panel showing all currently active windows. From this panel, you can swap windows or stop an application that is running on a window.

Press **Quit** to return to the panel from which you accessed the SYSTEM KEYS menu.

Using system keys Ctrl Alt +

When running in Enhanced Mode, 4690 OS has an Enhanced Options Menu that allows you to start special processes. To display this menu, press and hold down **Ctrl Alt +** on your controller keyboard. The Enhanced Options Menu allows you to perform the functions described in Table 7.

Table 7. Enhanced Options Menu

Function	Key	Description
Return to 4690	Esc	Removes the Enhanced Option Menu.
Cold restart	1	Initiates a full system restart. Property is vxcon.menu.coldboot.
Start 4690 supplemental system from hard drive	2	Boots the 4690 OS supplemental system from an image on the hard drive. Property is vxcon.menu.hdsupp.
Generate supplemental OS CD	3	Generates a bootable CD-R or CD-RW of the 4690 OS supplemental system. Property is vxcon.menu.suppcd.
Generate supplemental memory key	4	Generates a bootable memory key of the 4690 OS supplemental system. Property is vxcon.menu.suppmk.
Dump Controller Storage	5	Generates a dump. Property is vxcon.menu.dumpstorage.
Eject Disc	6	Ejects the CD/DVD media. Property is vxcon.menu.eject.

Menu options can be disabled by creating or editing the file C:/ADX_IDT1/VX_CTLPR.DAT. If the file is empty, all the options will be available. Setting one or

more of the properties above to zero will disable the option. For example, `vxcon.menu.coldboot=0` will disable the cold restart function. The property is enabled if it is set to a non-zero value, or if it is missing from the file.

Note: Changes to the file do not take effect until the next controller IPL.

Some keyboards do not have a + key on their numeric keypad. In this case, you can change the key sequence that invokes the Enhanced Options Menu by adding the `vxcon.key.popup` property to the `C:/ADX_IDT1/VX_CTLPR.DAT` file. For example:

```
vxcon.key.popup = ctrl alt e
```

That example changes the key sequence to Ctrl+Alt+E.

Note: This new key sequence does not work if you IPL from the supplementals.

The ANPOS keyboard does not have a + key on the numeric keypad. Instead, use the key immediately to the right of the 9 key on the numeric keypad.

Note: Due to a restriction in the operating system, do not define the *n* or the *x* key to invoke the Enhanced Options Menu.

Using system reset keys

If you need to restart your system, use the system reset keys. (Restarting your system is sometimes referred to as an *initial program load [IPL]*.) The system reset keys are a combination of Ctrl+Alt+Delete on your PC keyboard. Press and hold down the **Ctrl**, **Alt**, and **Delete** keys together until the screen clears, and then release.

For Enhanced 4690 controllers and terminals, Ctrl+Alt+Delete causes 4690 to reload (not a re-IPL of system).

Note: IPLing your system causes any applications currently running to stop. No combination of keys is available on the auxiliary console ASCII keyboard to IPL your system.

Confirm Ctrl-Alt-Delete (IPL key sequence)

This is an optional selection that, when enabled, prompts the user to confirm that the controller should IPL. When enabled, and a user is logged onto the system, the System Keys (CSO1S000HG) panel is displayed. This is enabled in System Options of System Configuration.

1. If you press the **Ctrl-Alt-Delete** key sequence the message will be displayed on the message line:
Do you really want to load controller storage? (1 = Yes)
2. If you press **1** the controller will IPL. Any other key will either perform the action from the System Keys menu that is associated with that key, or put up an error message indicating an invalid function was requested. If the logon screen is displayed, the message will appear on that screen, but will indicate that F7 will confirm the IPL.
3. If the system has not yet booted to the logon screen, **Ctrl-Alt-Delete** will IPL without confirmation.

Using keys in Command Mode

Your system also uses keys for editing command lines when you are working in the command-driven option, Command Mode (option 7 on the SYSTEM MAIN MENU panel). See “Editing keys” on page 135 for details on these keys.

Using the Supplemental Option

Supplementals arrive with your system. This section provide information on how to use the Supplemental Option from various media.

Note: You must use Classic Supplementals if you use a Classic mode controller, and you must use the Enhanced Supplementals with an Enhanced mode controller.

CD-ROM

To generate a CD with 4690 Supplementals, select item **3** on the Enhanced Options Menu described in Table 7 on page 43.

You can use the Supplemental Option using the CD-ROM for certain system functions and for correcting certain problems.

Memory Key

To generate a bootable memory key with 4690 Supplementals, select item **4** on the Enhanced Options Menu described in Table 7 on page 43. The controller hardware must support booting from memory keys, and the BIOS of the controller must have the boot order set to boot from USB key, USB FDD, or USB hard drive depending on the model of the controller.

Note: When the system is booted from the Supplementals memory key, read and write operations on the physical CD-ROM drive are not supported.

Hard Drive

To run 4690 Supplementals from the hard drive, select item **2** on the Enhanced Option Menu described in Table 7 on page 43.

Notes:

1. When the system is running Supplementals from the hard drive, the hard drive formatting utilities (**CPREP** and **DPREP**) are not supported because the hard drive is in use.
2. When the system is booted from the Supplementals hard drive, read and write operations on the physical CD-ROM drive are not supported.

For information on using the Supplementals from the CD-ROM, memory key, and hard drive, see Appendix C, “Using the Supplemental Option on 4690,” on page 521.

Chapter 2. Configuring your store system

This chapter is a guide to configuring your store system. It takes you step-by-step through the process of configuring your terminals and store controllers with store-specific characteristics. It also shows you how to activate your configuration once you have configured your system.

Before performing the steps in this chapter, make sure you have completed the configuration worksheets in the *4690 OS: Communications Programming Reference*. This guide refers to these worksheets.

You can configure your terminals, store controllers, or store system in any order. This guide follows the order shown in the menus on the panels.

Configuration screens that use Java graphics are available also in a text version. The text version is used when Java graphics are not available. To force the use of text mode, add a logical name, ADXUIMD, and set the value to TEXT. The logical name can be set to GUI also. You cannot force graphics mode if Java graphics are not supported in your environment. Some environments where Java graphics are not supported are:

- When the controller is not configured for Java graphics
- In remote operation sessions
- In telnet sessions

You can activate any portion of your configuration in any order you want. However, it might be important for you to consider the time of activation with respect to store operations. For example, your store cannot be operating when you activate changes made to system file size or system logical names.

The changes you make when configuring your system do not become effective until you activate your changes and re-IPL your system. The default configuration on your system is still in effect until you go through the process of activating the new configuration and re-IPLing your system.

Note: Because of memory requirements and file usage conflicts, no other applications should be run at the same time you are performing a configuration function (for example, changing, activating, printing, displaying, or reporting configuration data).

Using your planning worksheets

You should have your planning worksheets available when you are ready to start configuration. These worksheets are located in the appendixes in the *4690 OS: Planning, Installation, and Configuration Guide*.

The worksheets are designed to help simplify the configuration process for you. Some of the worksheets contain data about your terminals and store controllers. Other worksheets help you plan or organize your configuration data.

Saving or reconfiguring the default configuration

If you want to keep a specific default when configuring the system, do not change it. If you have changed your default configuration and decide you want to keep it as it was, do not activate the new configuration. The current configuration is in effect until you activate the new configuration.

Signing on

When your system starts up, a sign-on panel appears. To sign on:

1. Type in your operator ID. If you are using the default ID supplied with your system, type:
99999999
If you are using another ID, type that ID. If you need help setting up an operator ID, see “Your operator identification” on page 18.
If you are using an enterprise managed ID provided through Directory Services, type that ID followed by the character '@'. The '@' character is optional if your ID is longer than nine characters.
2. Press **Enter**. Your system prompts you to enter your password:
Type your Password____
When complete, press **Enter**.
3. Type your password. If you are using the default password supplied with your system, type:
99999999
If you are using a password you created, type that password. If you need help creating a password, see “Your password” on page 18.
4. Press **Enter**.

If you enter an incorrect password, the sign-on panel appears again and prompts you to re-enter your operator ID and password.

If Enhanced Security is enabled and your password has expired, the system prompts you to enter a new password and to re-enter it for verification.

If Enhanced Security is enabled and your password is about to expire, the system displays the number of days until expiration. The system prompts you to either press **F2** if you want to change your password or to press **Enter** if you want to complete sign-on without changing your password.

If your enterprise managed password provided through Directory Services expires, or is about to expire, follow the procedures set up by your enterprise for updating passwords. Once your password has expired, you will not be able to sign on until an updated password is sent from the enterprise to 4690 Directory Services.

After you enter your operator ID and password, the SYSTEM MAIN MENU appears:

CS01S000

SYSTEM MAIN MENU

Select one of the following:

1

Primary Application

2

Secondary Application

3

File Utilities

4

Installation and Update Aids

5

Problem Analysis Data Collection

6

Problem Analysis Reports

7

Command Mode

Type your selection number, then press Enter. _

F1Help

F2

F3

F4

F5

F6

F7

F8

F9Signoff

Using auto sign on

The auto sign-on feature allows you to bypass the 4690 OS log-on screen and proceed directly into an application, such as GSA/SA. Traditionally, 4690 OS applications need an operator ID and password to be keyed in manually.

Note: Even if Ctrl + Alt + Delete confirm is configured, it is bypassed during auto login. When you press **Ctrl + Alt + Delete** at the appearance of the Sign On screen, the system is auto logged in to the initial screen of the primary application.

Due to operation authorization, a file named ADXCISOAU.DAT is available for auto sign-on support. The ADXCISOAU.DAT file contains the following fields:

- Operator ID
- Password

If the ADXCISOAU.DAT file does not exist when OCF begins, you will not be automatically signed on to 4690 OS. If the ADXCISOAU.DAT file does exist, the operator ID and password contained in this file are used to automatically log you on to the 4690 OS. If the password is not provided in this file or if the operator ID or password are incorrect, you are prompted to input the operator ID and password manually. After log on, if the primary application from the SYSTEM MAIN MENU was defined in the controller configuration, it begins automatically too. If the primary application was not defined in the controller configuration, the SYSTEM MAIN MENU is displayed after log on. Additionally, the auto log-on implementation is designed to prevent recursive auto log on, if you press F9 (LogOff). Otherwise, the SYSTEM MAIN MENU is displayed.

To implement auto sign-on, create the ADXCISOAU.DAT file in ADX_SPGM with a valid operator ID and password. The file restrictions are as follows:

- ADXCISOAU.DAT must be located in ADX_SPGM.
- This file must contain the operator ID and password for auto sign-on as shown in Table 8:

Table 8. Auto sign-on file contents, ADXCISOAU.DAT

Offsets	Length	Description
0	9	operator ID (if the length is less than 9, fill in with blanks)
9	1	blank
10	8	password (if the length is less than 8, fill in with blanks)
18	1	blank

Using auto sign off

The Auto Sign Off function allows you to signoff or disconnect a 4690 system console user who has been inactive (no system keyboard or mouse input) for a specifiable length of time. This function is available as a set of 4690 application services in 16-bit and 32-bit C, Java, and CBASIC for a 4690 background application. For 4690 Enhanced systems, there are additional 4690 application services available to monitor, signoff, and disconnect system and auxiliary consoles.

Refer to the *4690 OS: Programming Guide* for more information about the Auto Sign Off APIs.

Selecting the Configuration option

The 4690 OS V3R3 or later introduces the Generic Terminal Configuration option, which supports configuring 4694 and SurePOS terminals. Using Generic Terminal Configuration, the 4690 OS detects the I/O devices that are attached and auto-configures the 4694 or SurePOS terminals. Generic Terminal Configuration is similar to how SurePOS has been configuring USB devices. It supports RS-485 devices on a 4694 terminal, and RS-485 devices that are supported on SurePOS models.

The 4690 OS V3R3 or later introduces the Generic Terminal Configuration option, which supports configuring 4694, SurePOS and TCxWave 6140 Series terminals. Using Generic Terminal Configuration, the 4690 OS detects the I/O devices that are attached and auto-configures the 4694, SurePOS or TCxWave 6140 Series terminals.

Legacy Terminal Configuration refers to how 4683 and 4693 terminals have always been configured in the past.

The RS-485 device IDs are used to identify devices that are attached to a 4694 or SurePOS terminal using Generic Terminal Configuration. The RS-485 devices are detected in numerical order from the lowest device ID to the highest device ID.

The 4690 OS provides a USB interface to support Toshiba USB I/O devices that are attached to a 4694 (with a USB port) or SurePOS terminals using Generic Terminal Configuration. The order of enumeration of USB devices is dependent on the ports into which they are plugged. The Generic Terminal Configuration allows combining of Toshiba USB I/O and RS-485 devices on a terminal. RS-485 devices have priority over USB I/O devices when both types of devices are being used. For example, if an RS-485 keyboard and a USB keyboard are attached, only the RS-485 keyboard is supported.

The 4690 OS provides a USB interface to support Toshiba USB I/O devices that are attached to a 4694 (with a USB port), SurePOS or TCxWave 6140 Series terminals using Generic Terminal Configuration.

The priority and detection order of devices that are attached to the terminal controls the assignment of the ANDISPLAY, ANDISPLAY2 and ANDISPLAY3 designations to the POS display devices. The first display processed is the ANDISPLAY, and so on. This priority and detection of device order can be changed through the Generic Terminal Configuration Device Characteristics settings, if needed. The Device Characteristics setting also controls the assignment of the SYSTEM Display.

Generic Terminal Configuration provides automatic driver loading for devices detected at boot time. Devices attached to the system unit after boot time might fail to load properly. USB hot plug is not supported as a rule. Only USB media devices like floppy drives or memory keys are hot plug capable. Power off the system unit when connecting or disconnecting any other hardware.

Note: Prior versions of 4690 permitted limited removal and reinsertion of identical hardware devices, but this capability is not consistent with the base unit hardware support statement, and therefore it is no longer tested or supported.

For additional information regarding Generic Terminal Configuration and Legacy Terminal Configuration, see the *4690 OS: Planning, Installation, and Configuration Guide*.

1. From the SYSTEM MAIN MENU, select the Installation and Update Aids option. To select this option, type **4** and press **Enter**.

The INSTALLATION AND UPDATE AIDS menu appears:

```
CS01S004      INSTALLATION AND UPDATE AIDS

Select one of the following:

    1 Change Configuration Data
    2 Report Legacy Configuration Data
    3 Change Input Sequence Table Data
    4 Report Module Level
    5 Apply Software Maintenance
    6 Build Software Maintenance Control File
    7 System Message Audible Alarm Functions
    8 System Utilities

Type your selection number, then press Enter. _

F1Help F2  F3Quit  F4    F5    F6    F7    F8    F9
```

2. From this menu, select the Change Configuration Data option. To select this option, type **1** and press **Enter**.

The CONFIGURATION menu appears:

```
CSCMS001      CONFIGURATION

Select one of the following:

    1. Legacy Terminal Configuration
    2. Controller Configuration
    3. System Configuration
    4. Activate Configuration
    5. Generic Terminal Configuration

Type your selection number.      _

When complete, press ENTER.
To exit configuration, press F3.

F1HELP F2  F3QUIT  F4    F5    F6    F7    F8    F9    F10
```

3. Follow the menus that appear on your store controller display and select options or enter data as prompted.

Press **Enter** to save your data. When you are finished entering your data, press **F3** to quit and return to the last option panel. Be sure you have pressed **Enter** before pressing **F3**. QUIT (F3) lets you return to a previous option panel, but does not save data you entered.

If you want to change your data after pressing Enter, return to the appropriate panel and enter the correct data. Press **Enter** again and return to the panel you came from by pressing **F3**, or, when available, **PgDn** (page down). Pressing **PgDn** takes you to the next panel. Press **PgUp** (page up) to return to the previous panel. A message on the panel indicate when the PgDn or PgUp keys can be used.

When you want to return to the CONFIGURATION menu, use the Esc key. Press **Esc** to scroll back through the option panels you have filled out until you reach the SYSTEM MAIN MENU panel. Press **Esc** each time a panel appears to move back along the path to the main menu.

Configuring legacy terminals (4683 and 4693)

Select option **1** (Terminal Configuration) from the CONFIGURATION menu. The TERMINAL CONFIGURATION menu appears. From this menu, you can select options to define terminal device groups, terminal load definitions, keyboard layouts, printer character sets, alphanumeric display character sets, terminal screen saver data, NFS data, or use the font/logo utility. Use the data from your worksheets in the *4690 OS: Planning, Installation, and Configuration Guide* to configure your terminals.

Note: 4683 terminals are not supported on 4690 OS V6R3. References to 4683 terminals only apply to previous versions of the OS.

Terminal device groups

A terminal device group is a group of common devices that attach to your terminal. This option lets you define, change, display, or erase the configuration data for the terminal device groups, display a terminal device group summary, or display a model device group summary.

Device groups also contain data for Java application execution.

Keyboard layouts

This option lets you define, change, display, or erase your keyboard layouts, display a list of defined keyboard layouts, or display a list of model keyboard layouts.

Terminal load definition

The terminal load definition contains the name of the device group, terminal mode, keyboard layout, command tail, and application program to be loaded into the terminal. This option lets you define, change, display, or erase a terminal load, or display a list of model terminal load definitions.

If you are defining or changing a terminal load configuration, a menu appears that asks you for specific information about that terminal (such as what keyboard layout it has).

The terminal load definition contains selections for enabling TCP/IP, selecting TCP/IP address method, and Java features. If any of these features are selected, you must also select the feature definition.

Alphanumeric display character set

This option lets you change the default character set for an alphanumeric display.

Model 1 and Model 2 printer character set

This option lets you change the default character set for a model 1 or model 2 terminal printer.

Model 3 and Model 4 printer character set

This option lets you change the default character set for a model 3 or model 4 terminal printer.

Terminal screen saver

This option lets you change the terminal screen saver data.

Note: This function is provided for VGA-attached video displays. It is not provided for 4683-421 terminals.

Network file system data

Configuring terminal NFS mount point data allows you to assign a terminal or group of terminals to a remote host. By assigning terminals to a mount point group and defining mount points within that group, you can define different resources to different terminals.

Note: Before performing any Java-based configuration utilities (such as NFS configuration), you must enable Java graphics in the controller configuration. Then, you must activate the change and IPL your system before Java graphics are enabled.

To access **Terminal NFS** configuration:

1. Select option **4** (Installation and Update Aids) from the SYSTEM MAIN MENU.
2. Select option **1** (Change Configuration Data).
3. Select option **1** (Terminal Configuration).
4. Select **Network File System Data**.

The panel for Terminal NFS Configuration is displayed.

Within the 4690 OS V2 or later versions, you can define a total of eight mount groups. Every terminal can be assigned to one of these mount point groups. By default, each terminal is initially assigned to mount group 8. You can define mount point data for this mount group to change the data for all terminals. Or, you can override the default by assigning various terminals to different mount groups and then defining different mount point data.

By default, no mount point data is defined for any of the mount groups. Once you define mount point data for a mount group, this data is retained. For example, if you define mount point data for mount group 1 and then want to add a new terminal to mount group 1 at a later time, you do not have to redefine the fields for mount group 1, you only need to add the new terminal to the group.

The following fields can be assigned during terminal NFS configuration.

NFS Mount Group

This indicates the mount group to which the terminal or group of terminals is assigned.

Drive Use this field to select a logical drive to which you can assign NFS properties. You can assign up to six drives per mount group.

NFS Mount Group Description

Use this field to indicate a short description for the NFS mount group. The default is NFS Mount Group Default.

NFS Server's IP Address

Use this field to enter the TCP/IP address for the system you want to access. The default is 0.0.0.0.

User ID

If you are using the 4690 NFS Client to access a system that uses a user ID, enter that user ID in this field. On some systems a user ID is automatically created when a user name is requested and created. Access control is based on this ID. The remote system allows access only to directories and subdirectories to which this user ID is authorized. Additionally, file permissions are retained; if the user is only allowed read access on the remote system, then the user ID on the 4690 NFS client is only allowed read access.

Group ID

If you are using the 4690 NFS Client to access a system that uses a group ID, enter that group ID in this field. On some systems a group ID is automatically created when a user name is requested and created. Access control is based on this ID. The remote system allows access only to directories and subdirectories to which this group ID is authorized. Additionally, file permissions are retained; if the group is only allowed read access on the remote system, then the group ID on the 4690 NFS client is only allowed read access.

Remote Resource Name

This field indicates the location or drive and directory you want to access on the remote system. For example, C:\TEMP to access the TEMP subdirectory on the remote resource's C: drive.

Font/logo utility

The Font/Logo Utility converts a specified bitmap file into a logo file and downloads a font file or logo file to a DBCS-enabled printer. The utility has a menu-driven interface; however, you can perform the same functions from the command line. To select the font/logo utility:

1. Select option **4** (Installation and Update Aids) from the SYSTEM MAIN MENU.
2. Select option **1** (Change Configuration Data).
3. Select option **1** (Terminal Configuration).
4. Select option **9** (Font/Logo Utility).

The main panel in the utility lists three functions:

- Logo conversion
- Font download
- Logo download

Logo conversion

The logo conversion function creates a logo file from a user-defined bitmap file. The bitmap file can be monochrome, 16-color, 256-color, or 24-bit color.

Perform the following steps to convert a bitmap file into a logo file from the LOGO CONVERT panel:

1. Type the name of the converted bitmap file in the bitmap name field.
2. Type the name of the logo file in the logo name field.
3. Select whether you want to override an existing file with the same name.
4. Press **Enter** to start the conversion. A message displays at the bottom of the panel to indicate the result of the conversion.

h

Perform the following steps to convert a bitmap file to a logo file from the command line:

1. Switch to Command Mode by selecting option 7 from the SYSTEM MAIN MENU.
2. Switch to the ADX_SPGM directory.
3. Type **ADXFONTL C** *bitmap-file logo-file*, and press **Enter**. A message displays indicating the result of the conversion.

Notes:

1. If the bitmap file is located in the ADX_SPGM directory, the file name can be used. If the bitmap file is not located in the ADX_SPGM directory, the complete directory path and file name must be used.
2. Check the ADX_SDT1:ADXFNTLF.DAT file for detailed information about the logo conversion.

Font download

Using the font download function, a font file can be downloaded to one of the next options:

- Model 4A printer
- 4610 model TI5 printer
- 4610 model TF7 printer
- 4610 model TM7 printer
- 4610 model 1NR printer
- 4610 model 2CR printer
- 4610 model 2NR printer
- USB APA display

Printer font downloads are available in DBCS locales, while the display font download is only available in Japan and Korea. Chinese character sets are supported on these printers, although Chinese fonts are not downloaded to the 4610 printer models TI5, 2CR, or 2NR.

Multiple font downloads are supported with the 4610-TI5 printer. You can download multiple font files to the same printer at one time. To download multiple fonts, create a configuration file containing the font file name list to download. The operating system provides a default configuration file, ADX_SPGM:ADXPFD CF.DAT.

Perform the following steps to download a font to a printer or USB APA display. From the FONT DOWNLOAD panel:

1. Type the name of the font or the configuration file name to be downloaded in the font file name field.
2. Type the terminal number or range of terminals to which the font is to be downloaded in the terminal number field.

3. Select the device to which you want to download the font.
4. Press **Enter** to begin the download. Messages indicating the process and the result of the download appear on the system display or terminal.

Perform the following steps to download a font to a printer from the command line:

1. Switch to the ADX_SPGM directory.
2. Type the following command and press **Enter**:

ADXFONTL F *device-type terminal-number font-file*, where *device-type* can be from 1 to 5:

1. Model 4A printer
2. 4610 printer models TI5, TF7, TM7, 1NR, 2CR, or 2NR
3. USB APA display (ANDISPLAY)
4. USB APA display (ANDISPLAY2)
5. USB APA display (ANDISPLAY3)

Notes:

1. Before downloading large font files, disable terminal storage retention in the TERMINAL FUNCTIONS panel.
2. For detailed information about font download, see the log file, ADX_SDT1:ADXFNTLF.DAT.
3. Font file limitations are:
 - For the Model 4A printer, 256 single-byte characters can be stored, while a total of 12300 locations are available for double-byte characters.
 - For the 4610 printer models TI5, TF7, TM7, 1NR, 2CR, or 2NR, up to 13501 double-byte characters for both a thermal station and the impact station can be downloaded.
 - If a font file contains more than the maximum characters, the portion that exceeds the maximum will not be downloaded.
4. The operating system provides the following default font files for you to download to the printer:

Table 9. Model 4A printer default font files

Language	File name	Font size (height x width)
Japanese	ADXF9MJF.DAT	9x10 (SB), 9x16 (DB)
Korean	ADXF9MKF.DAT	9x10 (SB), 9x16 (DB)
Simplified Chinese	ADXFAMPF.DAT	9x10 (SB), 16x10 (DB)
Traditional Chinese	ADXFAMTF.DAT	9x10 (SB), 16x10 (DB)

Table 10. 4610 printer models TI5, TF7, TM7, 1NR, 2CR, or 2NR default font files

Language	Station	File name	Font size (height x width)	Character set
Japanese	Thermal	ADXF4SJF.DAT	24x12	Single Byte
Japanese	Thermal	ADXF4DJF.DAT	24x24	Double Byte
Japanese	Impact	ADXF6SJF.DAT	16x16	Single Byte
Japanese	Impact	ADXF6DJF.DAT	16x16	Double Byte
Korean	Thermal	ADXF4SKF.DAT	24x12	Single Byte
Korean	Thermal	ADXF4DKF.DAT	24x24	Double Byte
Korean	Impact	ADXF9SKF.DAT	9x8	Single Byte

Table 10. 4610 printer models TI5, TF7, TM7, 1NR, 2CR, or 2NR default font files (continued)

Language	Station	File name	Font size (height x width)	Character set
Korean	Impact	ADXF9DKF.DAT	9x16	Double Byte
Simplified Chinese	Thermal	ADXF4SPF.DAT	24x12	Single Byte
Simplified Chinese	Thermal	ADXF4DPF.DAT	24x24	Double Byte
Simplified Chinese	Impact	ADXF6SPF.DAT	16x16	Single Byte
Simplified Chinese	Impact	ADXF6DPF.DAT	16x16	Double Byte
Traditional Chinese	Thermal	ADXF4STF.DAT	24x12	Single Byte
Traditional Chinese	Thermal	ADXF4DTF.DAT	24x24	Double Byte
Traditional Chinese	Impact	ADXF6STF.DAT	16x16	Single Byte
Traditional Chinese	Impact	ADXF6DTF.DAT	16x16	Double Byte

- The operating system provides some default font files that can be downloaded to the USB APA display. These font files are obtained from the Point-of-Sale Subsystem (POSS) implementation. The font styles for the USB APA display are different from the font styles for the SIO APA display.

Table 11. Model 4A printer default font files

Language	File name	Font size (height x width)
Japanese	ADXA6DJF.DAT	16x8 (SB), 16x16 (DB)
Korean	ADXA6DKF.DAT	16x8 (SB), 16x16 (DB)

- If the font file is located in the ADX_SPGM directory, the file name can be used. If the font file is not located in the ADX_SPGM directory, the complete directory path and file name must be used.

Logo download

You can download logos to a 4689-3G1 printer or the thermal station of a TI5, 2CR, or 2NR printer by using the logo download function.

Logos downloaded to the 4689-3G1 printer must be 53 bytes (fixed) in width and 144 pixels (fixed) in height. Logos downloaded to the 4610-TI5 printer can be 1–72 bytes in width and 1–244 bytes in height.

To download a logo to a printer, perform the following steps from the LOGO DOWNLOAD panel:

- Type the name of the logo file to be downloaded in the logo file name field.
- In the terminal number field, type the terminal number or range of terminals to which the logo file is to be downloaded.
- Select the kind of device to which the logo is to be download.
- Press **Enter** to begin the download. Messages indicating the process and the result of the download appear on the system display or terminal.

Perform the following steps to download a font to a printer from the command line:

- Switch to the ADX_SPGM directory.
- Type the following command and press **Enter**:

ADXFONTL L *device-type terminal-number logo-file*, where *device-type* can be 1 or 2:

1. 4689-3G1 printer
2. 4610-TI5 printer

Notes:

1. For detailed information about the logo download, see the ADX_SDT1:ADXFNTLF.DAT file.
2. For the 4610-TI5 printer, only the thermal logo is supported.
3. If the logo file is located in the ADX_SPGM directory, the file name can be used. If the logo file is not located in the ADX_SPGM directory, the complete directory path and file name must be used.

| **Configuring your TCxWave 6140 Series, SurePOS 4694, 350, and 700 Series terminals**

From the CONFIGURATION menu, select option 5 (Generic Terminal Configuration).

The screenshot shows a terminal window with the title 'CSCMS001 CONFIGURATION'. The prompt 'Select one of the following:' is displayed. A list of five options is shown: 1. Legacy Terminal Configuration, 2. Controller Configuration, 3. System Terminal Configuration, 4. Activate Terminal Configuration, and 5. Generic Terminal Configuration. Option 5 is highlighted with a blue bar. Below the list, the prompt 'Type your selection number.' is followed by a blue cursor. Below that, the prompt 'When complete, press ENTER.' is displayed. At the bottom of the terminal window, a row of function keys is visible: F1HELP, F2, F3QUIT, F4, F5, F6, F7, F8, F9, and F10.

Note: All other SurePOS 4694, SurePOS 300/700 Series and TCxWave 6140 Series terminal configuration information (keyboard layouts, alphanumeric display character sets, terminal screen saver, and Network File System (NFS) data) is defined through the Generic Terminal Configuration panels and should be defined before activating configuration.

The 4690 Operating System supplies default configuration files for both terminal device characteristics and terminal load definitions. Because the operating system can use these defaults, you do not have to create new configurations in order for the terminals to operate. You can choose to use the defaults supplied or define new terminal device characteristics or load definitions.

Use the data from your worksheets in the *4690 OS: Planning, Installation, and Configuration Guide* to configure your terminals.

Terminal device characteristics

The terminal device characteristics option allows you to define, change, display, or erase the devices attached to the SurePOS 4694, 300 Series, 700 Series and TCxWave 6140 Series terminal. With this option, you can create default terminal

device characteristics for an entire store. Changes to the terminal device characteristics could then be made at the store level rather than at each terminal within the store.

Terminal load definitions

A terminal load definition contains the name of the terminal device characteristics, the keyboard layout, command tail, and application program to be loaded into the SurePOS 4694, 300 Series, 700 Series and TCxWave 6140 Series terminal. This option lets you define, change, display, or reset a terminal load.

The terminal load definition also contains selections for selecting terminal mode, enabling TCP/IP, selecting TCP/IP address method, and Java features. If any of these features are selected, you must also select the feature definition. Additionally, if you have a Java or primary application defined, you must use one of the four keyboards defined (for example, the 50-key keyboard). However, you can change the keyboard layout name.

Keyboard layouts

This option lets you define, change, display, or erase your keyboard layouts, display a list of defined keyboard layouts, or display a list of model keyboard layouts.

Alphanumeric display character set

This option lets you change the default character set for an alphanumeric display.

Terminal screen saver

This option lets you change the terminal screen saver data.

Note: This function is provided for VGA-attached video displays.

Network file system data

Configuring terminal NFS mount point data allows you to assign a terminal or group of terminals to a remote host. By assigning terminals to a mount-point group and defining mount points within that group, you can define different resources to different terminals.

Note: Before performing any Java-based configuration utilities (such as NFS configuration), you must enable Java graphics in the controller configuration. Then, you must activate the change and IPL your system before Java graphics are enabled.

To access **Terminal NFS** configuration:

1. From the SYSTEM MAIN MENU, select option **4** (Installation and Update Aids).
2. Select option **1** (Change Configuration Data).
3. Select option **1** (Terminal Configuration).
4. Select **Network File System Data**.

The panel for Terminal NFS Configuration is displayed.

Within the 4690 OS V2 or later versions, you can define a total of eight mount groups. Every terminal can be assigned to one of these mount-point groups. By default, each terminal is initially assigned to mount group 8. You can define mount-point data for this mount group to change the data for all terminals. Or, you

can override the default by assigning various terminals to different mount groups and then defining different mount-point data.

By default, no mount point-data is defined for any of the mount groups. After you define mount-point data for a mount group, this data is retained. For example, if you define mount-point data for mount group 1 and then want to add a new terminal to mount group 1 at a later time, you only need to add the new terminal to the group. You do not have to redefine the fields for mount group 1.

The following fields can be assigned during terminal NFS configuration:

NFS Mount Group

This indicates the mount group to which the terminal or group of terminals is assigned.

Drive Use this field to select a logical drive to which you can assign NFS properties. You can assign up to six drives per mount group.

NFS Mount Group Description

Use this field to indicate a short description for the NFS mount group. The default is NFS Mount Group Default.

NFS Server's IP Address

Use this field to enter the TCP/IP address for the system that you want to access. The default is 0.0.0.0.

User ID

If you are using the 4690 NFS Client to access a system that uses a user ID, enter that user ID in this field. On some systems, a user ID is automatically created when a user name is requested and created. Access control is based on this ID. The remote system allows access only to directories and subdirectories to which this user ID is authorized. Additionally, file permissions are retained; if the user is only allowed read access on the remote system, then the user ID on the 4690 NFS client is only allowed read access.

Group ID

If you are using the 4690 NFS Client to access a system that uses a group ID, enter that group ID in this field. On some systems, a group ID is automatically created when a user name is requested and created. Access control is based on this ID. The remote system allows access only to directories and subdirectories to which this group ID is authorized. Additionally, file permissions are retained; if the group is only allowed read access on the remote system, then the group ID on the 4690 NFS client is only allowed read access.

Remote Resource Name

This field indicates the location or drive and directory that you want to access on the remote system. For example, C:\TEMP to access the TEMP subdirectory on the remote resource's C: drive.

Java applications

The configuration of (multiple Java applications) MultiApp and Java TOF can be accessed through the Generic Terminal Configuration (GTC). The main Java Applications screen shows you a list of the Java applications.

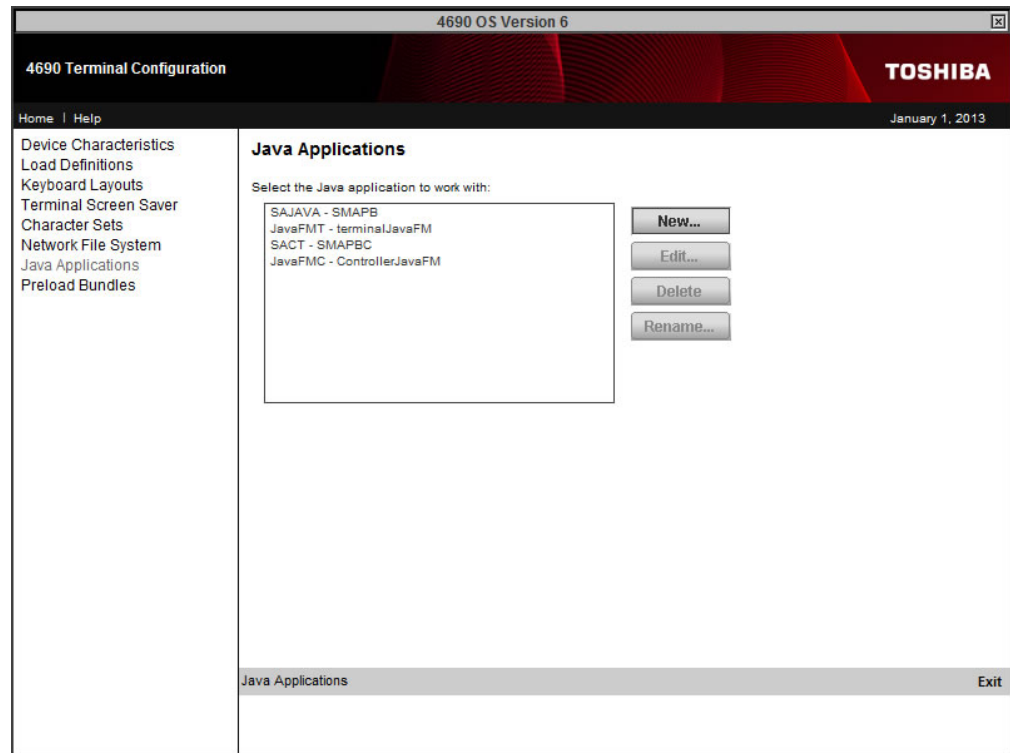


Figure 3. Main Java Applications screen

From the main Java Applications screen you can:

- Create a new Java application by selecting **New**.
- Select an application from the list, then select **Edit** to modify it.
- Select an application from the list, then select **Delete** to delete it.
- Select an application from the list, then select **Rename** to rename it.

The Application screen is used to configure the Java application and its parameters to configure classpath, other JVM options, classname, or enter a list of arguments.

4690 OS Version 6

4690 Terminal Configuration

TOSHIBA

Home | Terminal Configuration | Help

January 1, 2013

Application

Advanced

TOF Resource Creation

TOF Dependency Checking

Include List

Exclude List

JAR List

Application

Enter the Java application information.

Classpath:

r:\java\lib\OS4690.zip;r:\c:\java\jpos4690.zip;r

Other JVM options:

Classname:

IBMDefault

Application arguments:

Editing the Java application SAJAVA.

Save | Exit

Figure 4. Application screen

To create a new Java Application record, you must assign it a name. Optionally, you can add a description. You can also select an existing entry as a model for the new record.

4690 OS Version 6

4690 Terminal Configuration

TOSHIBA

January 1, 2013

Home | Help

Device Characteristics
Load Definitions
Keyboard Layouts
Terminal Screen Saver
Character Sets
Network File System
Java Applications
Preload Bundles

New Java Application

Enter the name and description of the new Java application:

Name

Description

Copy default values from:

SAJAVA - SMAPB
JavaFMT - terminalJavaFM
SACT - SMAPBC
JavaFMC - ControllerJavaFM

Submit

Cancel

Java Applications

Exit

Figure 5. New Java Application screen

The Rename Java Application screen enables you to change the name and description of Java application records.

The screenshot shows the '4690 OS Version 6' window with a 'TOSHIBA' logo. The main title is '4690 Terminal Configuration'. A sidebar on the left lists navigation options: Home, Help, Device Characteristics, Load Definitions, Keyboard Layouts, Terminal Screen Saver, Character Sets, Network File System, Java Applications, and Preload Bundles. The main content area is titled 'Rename Java Application' and contains the instruction 'Change the name and description of this Java application:'. Below this are two text input fields: 'Name' with the value 'JavaFMT' and 'Description' with the value 'terminalJavaFM'. At the bottom of the form are 'Submit' and 'Cancel' buttons. A footer bar at the bottom of the main area contains 'Java Applications' on the left and 'Exit' on the right.

Figure 6. Rename Java Application screen

The Advanced Application screen contains settings used for running multiple Java applications when the terminal is offline.

The TOF Resource Creation screen enables you to extract resources from existing zip or jar files to include in the JavaTOF zip file.

The TOF Dependency Checking screen determines the Java classes used by a given application.

Preload bundles

A preload bundle is a list of files to zip together and load onto a drive at terminal boot time. It provides an enhancement over moving files individually. The preload bundle screens are used to define the bundles. When and where to load them is specified in the terminal load definition configuration.

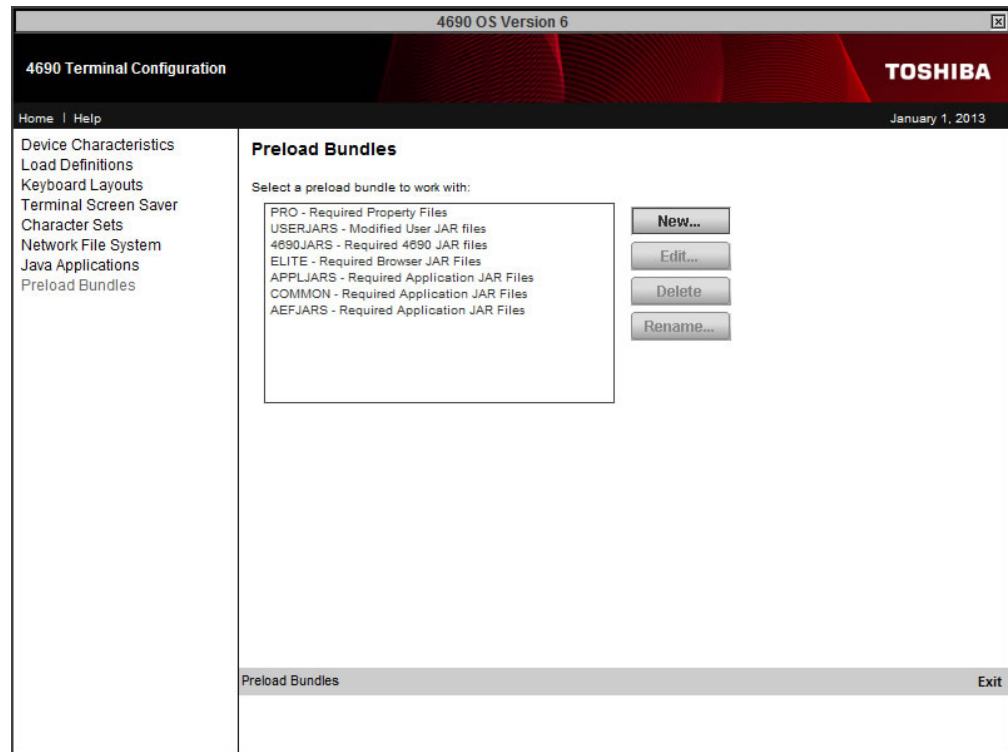


Figure 7. Preload Bundles screen

From the main Preload Bundles screen you can:

- Create a new preload bundle by selecting **New**.
- Select a preload bundle from the list, then select **Edit** to modify it.
- Select a preload bundle from the list, then select **Delete** to delete it.
- Select a preload bundle from the list, then select **Rename** to rename it.

Select **New** to specify the name, description, and default values for a new preload bundle.

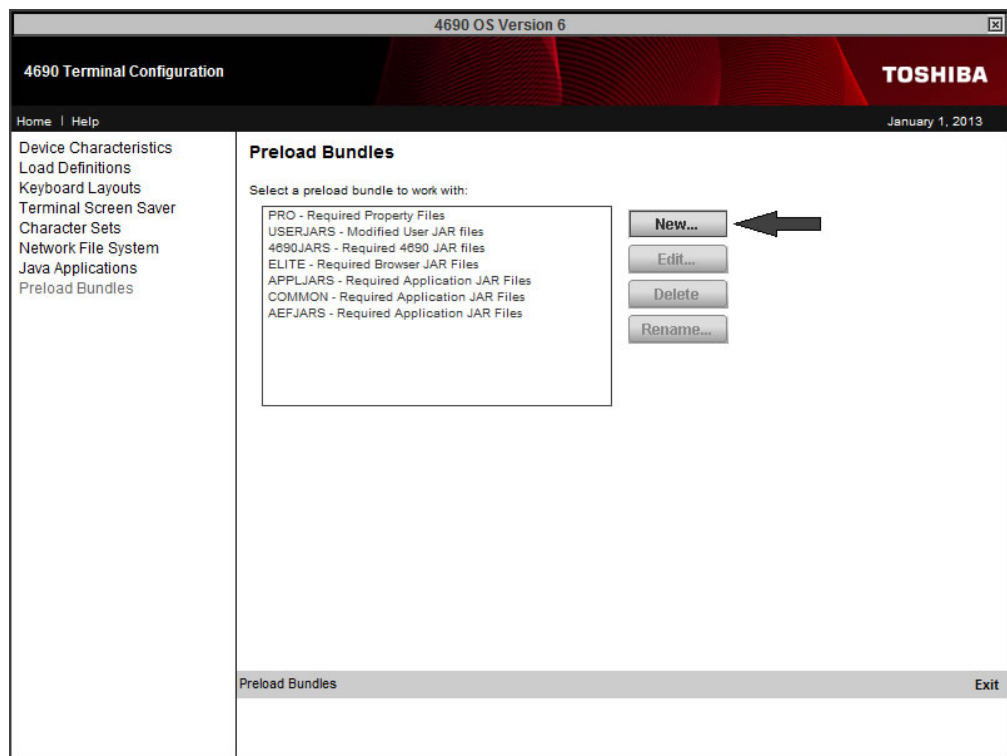


Figure 8. New Preload Bundle screen

After you have defined one or more preload bundles, you can select a preload bundle and edit, delete, or rename it.

The Rename screen enables you to change the name and description of a preload bundle.

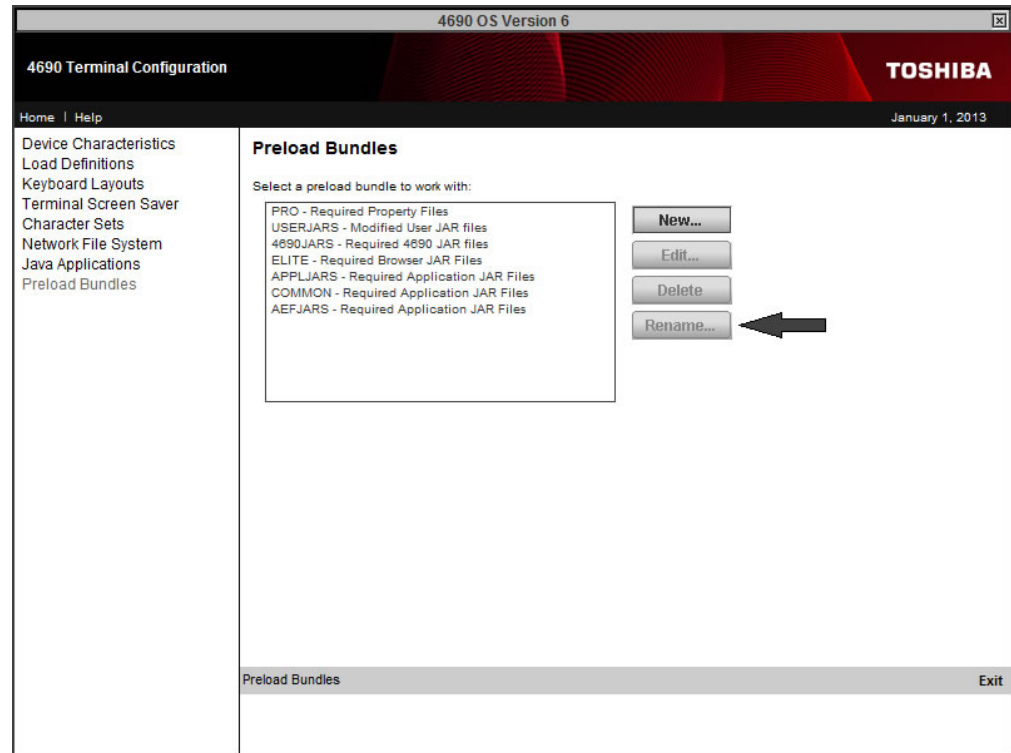


Figure 9. Rename Preload Bundle screen

Editing a preload bundle allows you to add bundle entries, one at a time. For each entry, you specify the file name and attributes of the entry. You can optionally select check boxes that specify that the file is optional, the file will be searched for recursively, and the entry will be placed in the root directory of the target drive.

The Edit Preload Bundle screen enables you to add a file to the preload bundle, change a file name or its attributes, or remove a file from the preload bundle by selecting **Delete**.

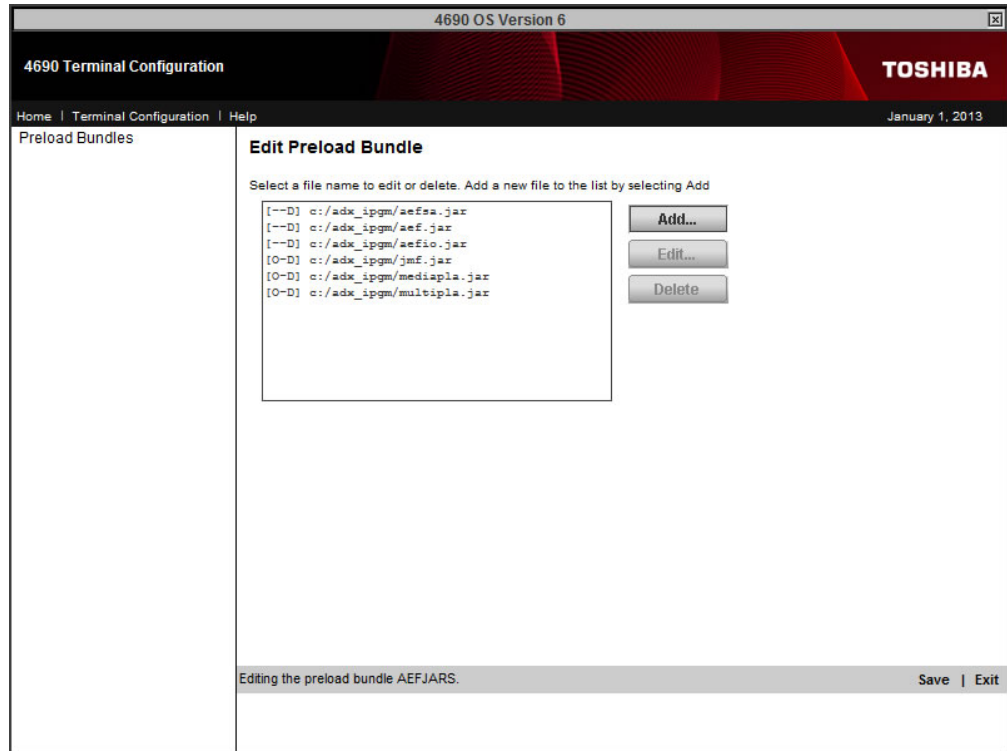


Figure 10. Edit Preload Bundle screen

The Add Bundle Entry screen enables you to add an entry to the preload bundle. A preload bundle entry consists of a file name and three optional attributes: Optional, Recursive Search, and Root Directory.

The screenshot shows a web-based interface for '4690 OS Version 6'. The top navigation bar includes '4690 Terminal Configuration', 'Home | Terminal Configuration | Help', and the date 'January 1, 2013'. The Toshiba logo is in the top right. The main content area is titled 'Preload Bundles' and contains an 'Add Bundle Entry' form. The form has a text input field for the file name, followed by three checkboxes: 'File is optional', 'Recursively search for the file', and 'Place in root directory of target drive'. Below these are 'Submit' and 'Cancel' buttons. A status bar at the bottom indicates 'Editing the preload bundle AEFJARS.' with an 'Exit' button.

4690 OS Version 6	
4690 Terminal Configuration	TOSHIBA
Home Terminal Configuration Help	January 1, 2013
Preload Bundles	Add Bundle Entry
	Enter the file name and attributes for this new entry:
	<input type="text"/>
	<input type="checkbox"/> File is optional
	<input type="checkbox"/> Recursively search for the file
	<input type="checkbox"/> Place in root directory of target drive
	<input type="button" value="Submit"/> <input type="button" value="Cancel"/>
	Editing the preload bundle AEFJARS. <input type="button" value="Exit"/>

Figure 11. Add Bundle Entry screen

The Edit Bundle Entry screen enables you to modify the file name and modify any of its associated attributes.

The screenshot displays the '4690 OS Version 6' terminal configuration window. The title bar includes '4690 OS Version 6' and a close button. The main header features '4690 Terminal Configuration' on the left and the 'TOSHIBA' logo on the right. Below the header, a navigation bar contains 'Home | Terminal Configuration | Help' and the date 'January 1, 2013'. The interface is divided into two main sections. The left section, titled 'Preload Bundles', is currently empty. The right section, titled 'Edit Bundle Entry', contains the following elements: a prompt 'Update the file name and attributes for this entry:' followed by a text input field containing 'c:/edx_ipgm/multipla.jar'; three checkboxes labeled 'File is optional' (checked), 'Recursively search for the file' (checked), and 'Place in root directory of target drive' (unchecked); and two buttons, 'Submit' and 'Cancel'. At the bottom of the right section, a status bar shows 'Editing the preload bundle AEFJARS.' and an 'Exit' button.

Figure 12. Edit Bundle Entry screen

Font/logo utility

This option either converts a specified bitmap file into a logo file or downloads a font file or logo file to a DBCS-enabled printer. Create the bitmap with a painting tool, and then use this utility to convert the bitmap into a logo file. The download logo file function downloads a font file or logo file to a specific terminal, or a Model 4A or 4610-T15 printer. The downloaded file resides on the printer until it is redefined. For information on running the font/logo utility, see “Font/logo utility” on page 54.

Configuring your store controllers

When configuring your store controller, the questions that appear depend upon whether MCF is enabled. Select option **2** (Controller Configuration) from the CONFIGURATION menu.

Note: Before performing any Java-based configuration utilities (such as NFS), you must enable Java graphics in controller configuration. Then, you must activate the change and IPL your system before Java graphics are enabled.

If MCF is enabled, the following questions and panels appear.

Are you configuring a Store System that uses the
Toshiba Multiple Controller Feature (LAN) to support
the Data Distribution Application? ___ Y=Yes, N=No

If your system does support DDA, see “DDA system” on page 72. If your system does not support DDA, see “Non-DDA system” on page 72.

The following panel appears:

CSCMS005

LAN CONFIGURATION

This LAN media type is set. The current LAN media type is shown. To change, enter the new value:

11 = Token Ring
2 = Ethernet

This TCC media type is set. The current TCC media type is shown. To change, enter the new value:

21 = Token Ring
2 = Ethernet

When complete, press ENTER.

F1 F2 F3QUIT F4 F5 F6 F7 F8 F9 F10

Note: Token Ring is no longer supported as of 4690 OS Version 6 Release 4 however the Configuration menus on the system have not been changed to remove it.

If you do not have MCF enabled, the following question appears on the panel.

Are you configuring a store system that uses
SNA communication on a LAN? ___ (Y=Yes, N=No)

Non-DDA system

If you have a non-DDA system but you have enabled MCF, type **N** and press **Enter** to reply **No** to the Data Distribution Application question. The system then asks if you use your LAN (MCF Network) for LU 6.2 communications on a LAN. When you answer this question, the LAN CONFIGURATION panel appears asking you to define your LAN and TCC media types. Once you have defined your LAN and TCC media types, a menu appears listing items you can configure for your store controller. The system asks you to type an **X** beside the items that you want to configure for this store controller. After you have selected these items, press **Enter**. The system processes the items in order on the menu.

Note: You have LU 6.2 communication capability regardless of how you answer the first question. The second question refers only to LU 6.2 communications on a LAN.

A menu appears for each item you choose. See the worksheets in the *4690 OS: Planning, Installation, and Configuration Guide* for assistance with configuring the items on this menu. The worksheets explain how to configure every menu item.

DDA system

Note: Configuration must be performed on the master store controller.

If your system does support DDA, type **Y** and press **Enter** to reply Yes to the Data Distribution Application question. If you reply Yes, you automatically have LU 6.2 communications capability. The LAN CONFIGURATION panel appears, prompting you to define your LAN and TCC media types. Once you have defined your LAN and TCC media types, the system displays the CONTROLLER CONFIGURATION panel.

CONTROLLER CONFIGURATION

Enter Store Controller IDs:

CONTROLLER TYPE	ID
-----	--
MASTER	
ALTERNATE MASTER	
FILE SERVER	
ALTERNATE FILE SERVER	

When complete, press ENTER.

F1HELPF2F3QUITF4F5F6F7F8F9F10

You must enter a two-character ID for each store controller listed on the panel. These IDs must correspond to the IDs you entered during installation. See “Node IDs” on page 450 for an explanation about these IDs.

There are four input areas on this panel. This does not mean that you must have four store controllers. One store controller can perform two roles. Each title represents a role. See “Store controller roles” on page 141 for more information on these roles.

As you fill in IDs for the master and alternate master controller, default IDs appear for the file server and the alternate file server. You can change these IDs.

After completing this menu, press **Enter**. The system displays the next CONTROLLER CONFIGURATION menu. This panel contains the IDs of your master controller, file server, and their alternates. It also contains a list of blanks. These blanks are for subordinate store controllers.

CONTROLLER CONFIGURATION

- To configure a controller, position the cursor on the appropriate ID and press the ENTER key.

- To add a new subordinate Store Controller, enter its ID.

ID	CONTROLLER TYPE
--	-----
	Alt Master & Alt File Server
	Master & File Server

To return to the Configuration Main Menu, press ESC.

F1HELP F2 F3QUIT F4 F5 F6 F7 F8 F9ERASE F10

Assign an ID to each subordinate store controller in your store system. As you fill in each ID, the system places “Subordinate” in the CONTROLLER TYPE column. Each store controller ID must be different. See the *4690 OS: Planning, Installation, and Configuration Guide* for information on ID requirements.

Note: You can add store controllers to your configuration at this point. They all do not have to be store controllers that you have defined on previous panels or at installation. But, the added store controllers must be store controllers that you defined when enabling features.

Configuring specific items

After entering the IDs of your subordinate store controllers, you must configure specific items for each one, and for the master and alternate master controllers. Position the cursor over a controller ID and press **Enter**.

A menu containing a list of items appears with the ID of the store controller to be configured at the top of the panel. The system asks you to type an **X** beside the items that you want to configure for this store controller. After you have chosen the items, press **Enter**. The system processes the items in order on the menu.

You can choose to process each item separately. If so, mark only one X on this menu each time the system returns to it.

You will notice that default files have already been created for each of your store controllers. Each time you complete an individual menu item, press **Enter** to return to this menu.

See the *4690 OS: Planning, Installation, and Configuration Guide* for assistance with configuring the items on this menu. The worksheets explain how to configure every menu item.

Your controller configuration files are now defined. Proceed with the next steps of configuration.

Configuring controller NFS

Configuring NFS mount point data on a controller is similar to terminal NFS configuration except that you only have one mount group per controller.

Note: Before performing any Java-based configuration utilities (such as NFS configuration), you must enable Java graphics in controller configuration. Then, you must activate the change and IPL your system before Java graphics are enabled.

To access **Controller NFS** configuration:

1. Select option **4** (Installation and Update Aids) from the SYSTEM MAIN MENU.
2. Select option **1** (Change Configuration Data).
3. Select option **2** (Controller Configuration).

Network File System Data is an option on the Controller Characteristics panel.

Within the 4690 OS V2 or later versions, you can define one mount point group per controller. No mount points are initially defined in the group. The following fields can be assigned during controller NFS configuration:

Drive Use this field to select a logical drive to which you can assign NFS properties. You can assign up to six drives per mount group.

NFS Mount Group Description

Use this field to indicate a short description for the NFS mount group. The default is NFS Mount Group Default.

NFS Server's IP Address

Use this field to enter the TCP/IP address for the system you want to access. The default is 0.0.0.0.

User ID

If you are using the 4690 NFS Client to access a system that uses a user ID, enter that user ID in this field. On some systems a user ID is automatically created when a user name is requested and created. Access control is based on this ID. The remote system will allow access only to directories and subdirectories to which this user ID is authorized. Additionally, file permissions are retained; if the user is only allowed read access on the remote system then the user ID on the 4690 NFS client is only allowed read access.

Group ID

If you are using the 4690 NFS Client to access a system that uses a group ID, enter that group ID in this field. On some systems a group ID is automatically created when a user name is requested and created. Access control is based on this ID. The remote system will allow access only to directories and subdirectories to which this group ID is authorized.

Additionally, file permissions are retained; if the group is only allowed read access on the remote system then the group ID on the 4690 NFS client is only allowed read access.

Remote Resource Name

This field indicates the location or drive and directory you want to access on the remote system. For example, C:\TEMP to access the TEMP subdirectory on the remote resource's C: drive.

Configuring your system

Select option **3** (System Configuration) from the CONFIGURATION menu. The SYSTEM CONFIGURATION menu appears, where you can:

- Specify a store number
- Specify the date and time formats for your store
- Specify the currency formats for your store
- Define application settings (define an application name prefix and enable or disable the application file timeout function)
- Enable or disable the LAN timeout function
- Add, edit, and delete LAN terminal definitions
- Enable or disable the VFS Name Server for long file-name support
- Specify networking information
- Specify system management information
- Specify controller to controller communication information
- Specify system security settings
- Define Java configuration data
- Specify system options information

Refer to the *4690 OS: Planning, Installation, and Configuration Guide* for assistance with configuring your system.

Activating your configuration

This option activates the changes made during configuration. The changes that you entered are not active until you activate your configuration.

Select option **4** (Activate Configuration) from the CONFIGURATION menu. The ACTIVATE CONFIGURATION menu appears.

ACTIVATE CONFIGURATION

Select the changes that you wish to activate.

1

Terminal Configuration

2

Controller Configuration

3

System Configuration

Type your selection number, then press Enter.

F1HELP

F2

F3QUIT

F4

F5

F6

F7

F8

F9

F10

As you select each option from this menu, you are actually activating that portion of the configuration. When you press **Enter**, a message tells you if your changes were activated.

Activating your terminal configuration

Select option **1** (Terminal Configuration) from the ACTIVATE CONFIGURATION menu. After you press **Enter**, a message appears telling you whether your changes were accepted.

If your changes are accepted, a message appears telling you that your changes have been activated.

If you have changed anything that affects a terminal, you can make your changes effective in the terminal from either the terminal or the store controller.

- From the terminal:
Run the Set Terminal Characteristics program. See “Using the Set Terminal Characteristics program” on page 122 for information on how to run the Set Terminal Characteristics program.
- From the controller:
Use the Load Terminal Configuration Data option on the TERMINAL FUNCTIONS menu. See “Load terminal configuration data” on page 98 for more information on how to load the configuration data from the controller.

Note: If you have changed anything that affects the terminal component of a controller/terminal or its attached Mod2 terminal, you must IPL the controller/terminal.

If your changes are not accepted, a message appears telling you so and identifies the problem. Correct the data and try to activate the configuration again.

Activating your controller configuration

Select the **Controller Configuration** option from the **ACTIVATE CONFIGURATION** menu. After you press **Enter**, the system returns to the **CONFIGURATION** menu where a message appears telling you whether your changes were accepted.

If your changes are accepted, a message appears telling you that your changes have been activated. The changes become effective when you IPL your store controller.

To IPL your store controller, press and hold **Ctrl+Alt+Del** at the same time, and then release.

Note: To IPL a remote store controller on a multiple-controller system:

1. At the **SYSTEM KEYS** menu, select option **c** (Access the Store Control Functions Screen).

SYSTEM KEYS

System Keys are used to request special control functions. Their action is independent of the current application(s). They have no direct effect on the current application, even though its screen will no longer be displayed.

Type one of the following letters or a function key.

- | | |
|---|--|
| m | Access the System Message Display Screen. |
| c | Access the Store Control Functions Screen. |
| b | Access the Background Application Control screen. |
| s | Start new application. (displays the System Main Menu) |
| w | Access the Window Control screen. |
| n | pass control to the next higher numbered window owned by this operator (Next). |
| p | pass control to the next lower numbered window owned by this user (Preceding). |
| a | Access the Auxiliary Console Control Screen. |
| t | Switch to Terminal Mode. |
| j | Switch to Terminal Java Console. |
| X | Switch to enhanced mode graphical extensions screen. |

F1HELP F2 F3QUIT F4 F5 F6 F7 F8 F9Disconnect

2. At the **STORE CONTROL FUNCTIONS** menu, select option **2** (Controller Functions).

STORE CONTROL FUNCTIONS

Select one of the following:

- 1 Terminal Functions
- 2 Controller Functions
- 3 TCC Functions
- 4 Multiple Controller Functions
- 5 System Functions
- 6 Communications Functions

Type your selection number, then press Enter.

F1HELP F2 F3QUIT F4 F5 F6 F7 F8 F9 F10

3. At the CONTROLLER FUNCTIONS panel, select option **4** (Load Controller Storage).

CONTROLLER FUNCTIONS

Select one of the following:

- 1 Display Controller Status
- 2 Enable Controller RAM Disk
- 3 Disable Controller RAM Disk
- 4 Load Controller Storage
- 5 Dump Controller Storage

Type your selection number, then press Enter.

F1HELP F2 F3QUIT F4 F5 F6 F7 F8 F9 F10

If your changes are not accepted, a message appears telling you so and identifies the problem. Correct the data, and then try to activate the configuration again.

Activating your system configuration

Select the **System Configuration** option from the ACTIVATE CONFIGURATION menu. After you press **Enter**, the system immediately returns to the ACTIVATE CONFIGURATION menu where a message tells you whether your changes were accepted.

If your changes are accepted, a message tells you that your changes have been activated. The changes become effective when you IPL both your store controller and your terminals.

If your changes were not accepted, a message tells you so and identifies the problem. Correct the data, and then try to activate the configuration again.

To IPL your store controller, press and hold **Ctrl+Alt+Del**, at the same time, and then release.

IPL your terminal as follows:

- To IPL your terminal at the terminal:
 1. Deactivate storage retention
 - a. At the SYSTEM KEYS menu, select option **c** (Access the Store Control Functions) panel.
 - b. At the STORE CONTROL FUNCTIONS menu, select option **1** (Terminal Functions).
 - c. At the TERMINAL FUNCTIONS menu, select option **3** (Disable Terminal Storage Retention).
 2. Power Off the terminal.
 3. After a few moments, power on the terminal again. The terminal reloads using the new system configuration information.
- To IPL your terminal at the store controller:
 1. At the SYSTEM KEYS panel, select option **c** (Access the Store Control Functions Screen).
 2. At the STORE CONTROL FUNCTIONS menu, select option **1** (Terminal Functions).
 3. At the TERMINAL FUNCTIONS menu, select option **7** (Load Terminal Storage).

Your store system has now been activated and IPLed. You can begin store operations.

If your changes were not accepted, a message appears telling you so and identifies the problem. Correct the data, and then try to activate the configuration again.

Note: The terminal component of a controller/terminal is IPLed when the controller/terminal is IPLed.

Setting a locally administered LAN address

The operating system enables you to set a locally administered address (LAA) to create a unique address for each POS terminal in your LAN system. The operating system creates the LAA using terminal unique data (the terminal number and store number) along with a user-specified LAA prefix. The exact format of the LAA is determined by:

- Terminal number - 3 digits
- Store number - 4 digits
- User-specified LAA prefix - 5 digits

Note: If you are configuring the LAA on a system that already has TCC over IP configured, there will be two addresses in the DHCPINFO table for each terminal. One address is the terminal's default LAN address and the other

address is the user-created LAA. These addresses might cause the system to run out of available IP addresses. If this happens, clear the unused addresses using DHCPINFO or increase the number of available IP addresses.

To create and use the LAA in a point-of-sale terminal:

1. Select **7** (Command Mode) from the SYSTEM MAIN MENU to get to a C: prompt.
2. Create a configuration file called ADXTSAWF.DAT in the ADX_SDT1 subdirectory. This file will contain the user-specified LAA prefix, and must contain the following LAA trigger string (including quotes) where *n* must be a hexadecimal digit from 0 (zero) to F. For example, the valid range for the LAA prefix is 40000 to 4000F.

"LAA 4000n"

Once this file is created, the file distribution attribute for ADXTSAWF.DAT should be set to Compound - Distribute on Close, which will force distribution of the file to all store controllers on the LAN system.

Note: ADXTSAWF.DAT should exist on all store controllers within the LAN system.

3. Run Set Terminal Characteristics (STC), option 2, 2, 2, 2. You should run STC on each terminal in which you want to have the LAA enabled. After STC runs and the terminal has loaded its configuration, the terminal will IPL. Once the terminal IPLs, the LAA for the terminal will be set.

Changing the LAA prefix

To change the LAA prefix, modify the configuration file ADXTSAWF.DAT and ensure that the changes are distributed to all store controllers on the LAN system. Run STC, option 2, 2, 2, 2, on each terminal to force the change.

Disabling LAA

To disable the LAA in a point-of-sale terminal, delete or rename the LAA configuration file ADXTSAWF.DAT in the ADX_SDT1 subdirectory. Make sure the file is deleted or renamed on all store controllers in the LAN system. Run STC, option 2, 2, 2, 2, on each terminal to force the change.

Chapter 3. Store controller operations

This chapter is a guide to store controller operations. It gives you information on running your store's applications, using system installation and update aids, and using store controller functions.

Using the SYSTEM MAIN MENU

When you sign on to a store controller or auxiliary console, the SYSTEM MAIN MENU appears:

```
CS01S000          SYSTEM MAIN MENU

Select one of the following:

1  Primary Application
2  Secondary Application
3  File Utilities
4  Installation and Update Aids
5  Problem Analysis Data Collection
6  Problem Analysis Reports
7  Command Mode

Type your selection number, then press Enter. _

F1Help F2  F3  F4  F5  F6  F7  F8  F9SIGNOFF
Time=xx:xx Current Windows=x Number of Windows=x
```

Note: The status line is not shown on the 316x family of auxiliary console screens, but you can configure the 315x auxiliary consoles to show the status line. See the *4690 OS: Planning, Installation, and Configuration Guide* for information on these auxiliary consoles.

The options displayed on the menu represent tasks you can perform in your store operations. Using options from the menu, you can perform tasks such as running your store's programs, updating your system, building keyed files, or formatting diskettes. The options you choose depend on your store's requirements.

To select an option from the menu, type the number of the option, and press **Enter**. The following sections describe the tasks you can do using each option on the menu.

Running your store's primary application

When you want to run your store's primary application, select option **1** on the SYSTEM MAIN MENU.

Note: On your panel, the actual title of your program will be displayed instead of the words "Primary Application" that are shown here. Primary applications are defined for each store controller. Each of your store controllers can have the same or different primary applications depending on your requirements.

When you press **Enter**, your primary application begins. The next panel you see depends on your application.

For more information on primary and secondary applications, see the *4690 OS: Programming Guide*.

Running your store's secondary application

When you want to run your store's secondary application, select option **2** on the SYSTEM MAIN MENU.

Your store can specify up to 14 secondary applications.

Note: On your panel, this option will display a title your store specifies instead of the words "Secondary Application" that are shown here. Secondary applications are defined for each store controller. Each of your store controllers can have the same or different secondary applications depending on your requirements.

A panel appears showing the titles of up to 14 applications that are specified by your store. The following panel is a sample only; your panel displays the actual number and title of your secondary applications.

SECONDARY APPLICATION

Select one of the following:

- 1 Application A
- 2 Application B
- 3 Application C
- 4 Application D
- 5 Application E
- 6 Application F
- 7 Application G
- 8 Application H
- 9 Application I
- 10 Application J
- 11 Application K
- 12 Application L
- 13 Application M
- 14 Application N

Type your selection number, then press Enter

F1 F2 F3QUIT F4 F5 F6 F7 F8 F9 F10

To run one of the applications that are shown on this panel, type the number of the application, and then press **Enter**.

Using File Utilities

To select the File Utilities option from the SYSTEM MAIN MENU, type **3** and press **Enter**.

The FILE UTILITIES menu appears:

CS01S003

FILE UTILITIES

Select one of the following:

1

Keyed File Utilities

2

Display/Alter File Data

3

Distributed File Utilities

4

File Compress/Decompress

5

Streaming Tape Drive Utilities

6

Optical Drive Utility

Type your selection number, then press Enter. ____

F1

F2

F3

F4

F5

F6

F7

F8

F9Signoff

Type the number of the utility you want to access, and press **Enter**.

Keyed File Utilities

The operating system provides a Keyed File Utility to allow you to create and manage keyed files. To start this utility, select the **Keyed File Utilities** option on the FILE UTILITIES menu. See the *4690 OS: Programming Guide* for more information on the Keyed File Utilities.

Display/Alter File Data

The operating system provides a Display/Alter File Data utility for displaying and altering files. To start this utility, select the **Display/Alter File Data** option from the FILE UTILITIES menu. You must be authorized to use this panel. When you select this option, the system prompts you to enter a security code on the next panel.

See Chapter 15, “Using the Display/Alter File Utility,” on page 359 for more information on this utility.

Distributed File Utilities

The operating system provides Distributed File Utilities for processing distributed files. To start this utility, select the **Distributed File Utilities** option from the FILE UTILITIES menu.

For more information on the Distributed File Utilities, see “Distributed Files” on page 446.

File Compress/Decompress

The operating system provides a File Compress/Decompress Utility to compress and decompress files on your store system. To start this utility, select the **File Compress/Decompress** option on the FILE UTILITIES menu. For more information on the File Compress/Decompress utility, see Chapter 19, “Using the File Compress/Decompress Utility,” on page 405.

Streaming Tape Drive Utilities

If you have an optional streaming tape drive attached to your store controller, the operating system provides a set of streaming tape utilities to enable you to use this device. To start this function, select the **Streaming Tape Drive Utilities** function from the FILE UTILITIES menu. For more information on the Streaming Tape Drive Utilities, see Chapter 17, “Using the streaming tape drive utility,” on page 385.

Optical Drive Utility

If you have an optional optical drive, the operating system provides an Optical Drive Utility to enable you to use this device. To start this utility, select the **Optical Drive Utility** function on the FILE UTILITIES menu. For more information on the Optical Drive Utility, see Chapter 18, “Using the optical drive utility,” on page 393.

Using System Installation and Update Aids

To select the Installation and Update Aids option from the SYSTEM MAIN MENU, type **4** and press **Enter**.

The INSTALLATION AND UPDATE AIDS menu appears:

CS01S004 INSTALLATION AND UPDATE AIDS

Select one of the following:

1

Change Configuration Data

2

Report Legacy Configuration Data

3

Change Input Sequence Table Data

4

Report Module Level

5

Apply Software Maintenance

6

Build Software Maintenance Control File

7

System Message Audible Alarm Functions

8

System Utilities

Type your selection number, then press Enter. ____

F1Help F2 F3Quit F4 F5 F6 F7 F8 F9

Type the number of the option you want, and press **Enter**.

Change Configuration Data

The Change Configuration Data option lets you reconfigure your store controller, terminal, or system, or lets you activate your configuration. See Chapter 2, “Configuring your store system,” on page 47 for detailed information on using this menu.

The Change Configuration Data option also lets you report Generic Terminal Configuration. See “Report Legacy Configuration Data” for more information.

Report Legacy Configuration Data

The Report Legacy Configuration Data option lets you display, file, or print the data you defined during legacy configuration. If file destination is specified, the report is filed in ADX_SDT1:ADXCSDTF.DAT.

Because of memory requirements and file usage conflicts, no other applications should be run at the same time you are performing a configuration function (for example, changing, activating, printing, or displaying configuration data).

Note: Reporting Generic Configuration Data is accessed through the Generic Terminal Configuration option. You can display, file, or print the data you defined during Generic Terminal Configuration.

To report Legacy Configuration Data for Generic Terminal Configuration, follow these steps:

1. Select **Installation and Update Aids** (option 4) from the SYSTEM MAIN MENU.

2. Select **Change Configuration Data** (option 1).
3. Select **Generic Terminal Configuration** (option 5).
4. After the 4690 Configuration screen is displayed, select **Home**.
5. Select **Configuration Reports**.
6. Select whether to report active or inactive configurations.
7. Specify the destination of the report output. If file destination is specified, the report is filed in the ADX_SDT1:ADXCSDTF.DAT file.
8. Optionally select Report All Terminal Configuration, which is the same as selecting all records on the Device Characteristics, Load Definitions, Java Applications, and Preload Bundles pages.
9. Optionally select Report System Configuration, which includes the system configuration settings in the report.
10. Select **Create Report**.

Before performing any Java-based configuration utilities, you must enable Java graphics in the controller configuration. Then, you must activate the change and IPL your system before Java graphics are enabled.

Change Input Sequence Table Data

The input sequence table defines the input data for terminal devices, such as the keyboard and the scanner. The Change Input Sequence Table Data option on the menu lets you build an input sequence table.

Note: Because of memory requirements, no other applications can run at the same time you are building an input sequence table.

For information on building an input sequence table, see the *4690 OS: Programming Guide*.

Report Module Level

This option reports module-level information for all software products on your system. If file destination is specified, the report is filed in ADX_SDT1:ADXCSSDF.DAT.

Apply Software Maintenance

Use the Apply Software Maintenance option to apply changes or updates to the operating system, the 4680 or 4690 application program products, and your application programs. These changes must be on a set of diskettes with a product control file on the first diskette or on a CD-ROM. The product control file describes which product the changes are intended for and the module level of each module to be updated.

For more information about this utility, see “Apply Software Maintenance utility” on page 365.

Build Software Maintenance Control File

The Build Software Maintenance Control File lets you create or edit a product control file. This file is required if you choose to use the Apply Software Maintenance utility to maintain your own products or modifications to 4680 or 4690 application program products.

For more information about this utility, see “Build software maintenance control file utility” on page 379.

System Message Audible Alarm Functions

When you select the **System Message Audible Alarm Function** option, the SYSTEM MESSAGE AUDIBLE ALARM FUNCTION panel appears:

```

CS1S001      SYSTEM MESSAGE AUDIBLE ALARM FUNCTION

Select one of the following:

1  Build audible alarm message file
2  Report audible alarm message file
3  Activate audible alarm
4  Deactivate audible alarm

Type your selection number.          1

Enter the requested information, then press Enter.

F1HELP F2  F3QUIT F4  F5  F6  F7  F8  F9  F10

```

Build Audible Alarm Message File: This utility lets you build a message file that causes the operating system to sound an audible alarm at the store controller when certain system messages are waiting in the system log. This function is useful for alerting the operator at the store controller to situations that require immediate action.

The file contains default entries that you can keep and add to, or delete. You can customize the functions of the audible alarm file to meet the needs of your store. For detailed information on building this file, see the *4690 OS: Programming Guide*.

Report Audible Alarm Message File: A status report, indicating the status of the audible alarm, can be displayed, printed, or written to a file. The report contains the following information:

- Store controller ID.
- If the alarm is activated or deactivated.
- Length of time the alarm is set to sound.
- Message numbers in the alarm file. These are the numbers of the messages that prompt the alarm to sound.

Activate and Deactivate Audible Alarm: These options let you make specifications about the audible alarm. You can activate the alarm for a specific time period, deactivate the alarm, or switch off a ringing alarm. You need to switch off a ringing alarm (option 4) only if you specified that the alarm is to sound until you switch it off using this option.

If the store controller is one of multiple controllers using DDA, you must also identify the store controller by specifying the store controller ID.

Collecting and reporting problem analysis data

The Problem Analysis Data Collection and Problem Analysis Reports options on the SYSTEM MAIN MENU let you collect data for resolving problems, and print, display, or write the data to a file. These options should be used only if there are problems with your system. See the *4690 OS: Messages Guide* for more information. See “Measuring system performance” on page 86 for detailed information about system performance.

Measuring system performance

The most consistent factor that adversely affects system performance is overuse of one or more resources. The key resources in the operating system are the hard

disk drive, the store controller processing unit, and the terminal device channel. The operating system provides a performance-gauging facility that is called the *Performance Data Collection* (also known as the *Performance Monitor*) and a Performance Report Utility. The Performance Monitor measures the use of these resources in the operating system. You start the Performance Monitor using system panels. The system panels of the Performance Report Utility control the device specification for the output.

In 4690 OS V6, the results shown by the Performance Monitor may be different depending on the mode of the system being measured. This is due to the new infrastructure that V6 Enhanced Mode uses to interface to the hardware.

The 4690 Performance Report Utility generates two types of output or report styles: graph summary and raw utilization. The Raw Utilization Data is designed primarily for file output that can be pulled back to a host for analysis. This is useful when you want to collect data for up to 24 hours.

The graphic report helps define sharp increases and decreases in component use. The system generates this graphic output for intervals of 1/60th of the total Performance Monitor runtime when the monitoring time is set to 60 minutes or less. These intervals represent the *granularity* of the graph. This granularity (or period of time that is represented by one bar) can be computed by a formula.

For example, if you want to know the granularity for a 30-minute runtime, you would use the following formula to show the interval time in seconds:

$$(30 \text{ min}) \times (60 \text{ sec per min}) / (60 \text{ intervals}) = 30 \text{ seconds/interval}$$

Figure 13 shows a bar graph output from the Performance Report Utility for controller processor utilization:

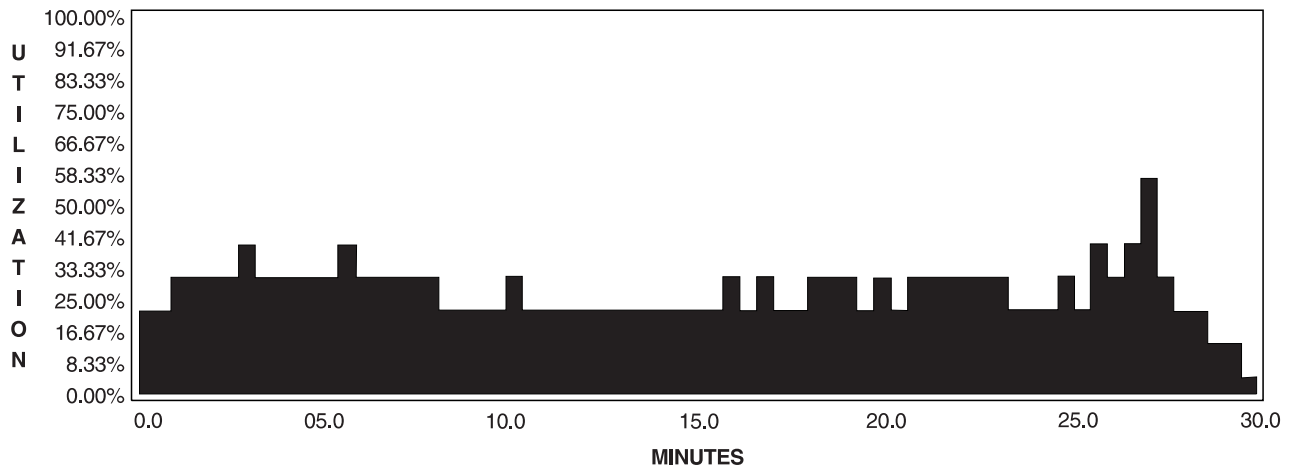


Figure 13. Controller performance

See the *4690 OS: Messages Guide* for detailed instructions on running the Performance Monitor and the Performance Report Utility.

The 4690 Performance Monitor allows you to collect data for as little as 10 minutes or as long as 24 hours on the CPU and hard disk drives. The Device Channel utilization from a specific terminal is limited to a maximum of one hour of data collection.

When the monitoring is longer than one hour, the Report Utility allows graphing of any range of time between the start and stop collection time. The data collection interval for runs set for one hour or more will be 60 seconds. For runs that are set for 24 hours, data for 1,440 intervals of 60 seconds are collected per device.

A 24-hour graph will show 60 bars. Each bar represents 24 minutes or 24 of the combined intervals that are collected. If you requested that only four hours be graphed from the 24-hour data collection, each of the 60 bars will represent four minutes. If the 4 hours requested for the report began with the 10th hour (or 600th minute), the time scale on the bottom of the graph summary will show minute "600" for the 1st bar with six more incremental counts up to minute "839" for the last bar. (240 minutes in four hours.) This is like "zooming in" on a subset of the total 24 hour graph to get a finer granularity.

For monitoring times of one hour or less, the graph option shows all of that time.

You can print or display the Raw Utilization Data, or send it to a file just as the Graph Summary. The raw data gives the device utilization for every interval collected. For example, if the data is collected for one hour or longer, each interval represents 60 seconds, and gives the interval count along with the device utilization for that interval.

The data is given as six intervals per line (or per record). The following is an example of two lines of the data for the Hard File Performance along with its possible header records.

```

Store 201                SYSTEM PERFORMANCE REPORTS
                        Controller CC                      11/25/93
                        HARD FILE 1 DISK PERFORMANCE

Date of monitoring:    11/25/93
Time of monitoring:   From 09:21:15      To 09:33:15

        The total data collection time was    12.00 minutes.

0001   9.5  0002  15.3  0003  16.5  0004  15.3  0005  17.8  0006  24.0
0007  22.2  0008  17.4  0009   6.2  0010   7.6  0011  11.8  0012  16.0
```

The 8th interval, or 8th minute in this example, 0008 was 17.4% utilized on this device. The device statistics that summarize the graph output are the only part of graph that is missing in the headers of the raw data.

Print or display the Raw Data to see the exact format. Leading records will contain the data lines that look similar to the graph report. These records contain the report name, date, time, and so on. The text will vary with different countries.

A statistical analysis application on a host could collect this data for many stores for selected 24-hour periods, and check for trends or monitor for sustained periods of high usage.

Example of system performance data collection

If you run the Performance Data Collection for 24 hours, the Performance Report Utility could create any or all of the following types of output for one device of all devices.

- A graph of all 24 hours compressed into 60 bars; each bar represents 24 minutes.
- A graph of 2 hours between 12 noon and 2:00 p.m. (12:00 to 14:00). The graph is 60 bars; each bar represents 2 minutes in granularity.

- A graph of 30 minutes from 9:00 a.m. until 9:30 a.m. The graph is 30 bars; each bar represents 60 seconds granularity. (The minimum granularity for monitoring lasting greater than one hour is one minute.)
- Print the raw utilizations for 5 p.m. until 8 p.m. (17:00 to 20:00).
This produces each minute's utilization for the 180 minutes from 5 p.m. until 8 p.m. The number in front of the utilization is the minute relative to the first minute the data collection was started. If the collection started at 3 p.m. the numbered utilizations will start at number 0121 and go to 0300. The utilizations are expressed to a tenth of a percent (14.2 is fourteen and two tenths percent). This is a much finer utilization than that displayed on the graph, which is in multiples of 8.33% utilization.
- Create a file of raw data for all 24 hours.
Each device would create 1 440 utilizations, written as six of these per line (or record). That is 240 records plus the header records, which number about 10 per device.

Measuring performance at the store controller

Two situations exist where there is a particular benefit in running the Performance Monitor:

- Run the performance monitor when you feel that your response times are greater than you expect them to be. The results of measuring the CPU and hard disk drive utilizations can help you pinpoint which component is over utilized. Generally, a utilization of more than 50% is considered high.
The Performance Report Utility enables you to determine which system components are probably involved in system degradation. This report, coupled with knowing which applications were active, and approximately what the traffic load from the terminals was at the time the measurements were taken, can help you determine the cause of the performance problems.
- Invoke the performance monitor when the system is running satisfactorily, but you want to know if you can safely add more load to the system (for example, to add more terminals or run an additional background program). If all the components measured have a utilization factor of considerably less than 50%, you can add more load to your system. To verify that you can add more load, you should contact your Toshiba representative to have your performance requirements needs analyzed.

Two controller measurements are made:

- Controller processing utilization
- Hard disk drive utilization

(The operating system does not measure host link utilization, or Ethernet utilization for Ethernet attached terminals. The Ethernet utilization would be very low for point-of-sale traffic and there are tools available to measure both of these networking resources if that is necessary.)

Do not expect any of these controller measurements to be 100% utilized, or store controller utilization to ever be zero. There are timer functions that are always running.

System utilizations

Resource limitations are guidelines that say, "If the system is kept within these boundaries, the system will maintain its performance stability." These guideline utilizations are shown in Table 12 on page 90.

When one of these limits is exceeded, it might represent performance degradation in the system. If there is evidence of poor performance, then one of these resources might be causing the bottleneck.

These system utilizations are guidelines for performance stability in a multi-tasking environment. They represent utilizations sustained over a period of time, not just spikes on the graph representing a few seconds or a minute.

Table 12. Performance Utilization Guidelines

Device utilization	Percentage
CPU utilization	60 %
Hard disk drive utilization	50 %
Host utilization	50 %
Device channel utilization	40 %

Note: These utilizations do not represent the full capacity of the system.

The guidelines shown in Table 12 on page 90 allow you to run other applications simultaneously without impacting the sales throughput and response time.

The idea is that there are system resources left to support other tasks. But, these other tasks are not without their limitations.

Terminal applications have a higher priority to controller resources than applications executing in the store controller. Background applications in the store controller slow down before noticeable impact appears in the terminal.

Consideration

A batch application (like applying 10,000 file updates to a file that takes x amount of time) will theoretically take 100% of the system. For example, it might be 75% on the hard disk drive and 25% in the CPU if this task were measured by itself. The 75% hard disk drive utilization is not bad. It will however, have some impact on applications that are running at the same, or lower, application priority.

Terminal requests to the store controller automatically have higher priority than any controller applications so these will not be impacted. A batch process like this should be assigned an application lower than “5,” (for example, priorities 6 through 9), which is the system default for all applications as well for applications that are started at the consoles. Then, this batch task would have very little impact on console tasks, which means impact to operator console response time would not be impacted by this application.

The guidelines refer to values for the interactive activity of the system where good system response times are important to you.

Processor utilization while printing on the 4690

While a printer attached to the 4690 is actually printing, the processor utilization reported from ‘Performance Data Collection’ registers between 70% and 90% utilization. Technically, this utilization report is correct. However, most of the cycles are lower in priority than all application priorities.

This utilization is generated from the operating system “polling” the printer to see if it is ready to receive the next buffer of data. If any other application is ready to execute, the processor is made available, and printer polling stops. The utilization

report would be more accurate if the processor utilization did not record these cycles, but it is difficult to discard them from the report.

The actual impact from the print spooler “printing” is actually very low because printers are very slow devices.

The same high utilization appears when the printer is “off” or not attached and there is work in the print queue for this controller.

The best rules to follow regarding printer activity are:

- Ignore processor utilization that is shown during periods of printer activity.
- Omit printer activity when trying to collect valid processor utilization measurements.

Be sure the print queue is empty as well. This queue can be queried by issuing the PRINT command from the command line.

Measuring performance at the terminal

The device channel utilization measures the traffic that a specified terminal receives on the link attaching most of its devices. These devices include:

- Keyboards
- Readers
- Scanners
- Displays
- Printers
- RS-232-attached devices

Typically, this utilization should be consistently under 10% unless the RS-232 port is transmitting or receiving large blocks of data.

Using command mode

This option is the command-driven option on your system. Unlike the other options on the SYSTEM MAIN MENU, this option does not use panels to help you perform a task. To use this option, you must know how to enter commands, and you must be familiar with command formats.

Chapter 5, “Working in Command Mode,” on page 133 through Chapter 8, “Creating and editing files,” on page 265 show you how to use command mode to help you perform tasks such as formatting or copying diskettes or creating and editing files.

Using store control functions

The STORE CONTROL FUNCTIONS menu provides functions that help you control certain activities of your point-of-sale terminals and store system. For example, there are store control functions for enabling and disabling the terminal storage retention function and setting system date and time. You must be authorized as a System Control Operator to use store control functions. See “Specially authorized operators” on page 18 for more information on this authorization.

To display the STORE CONTROL FUNCTIONS menu, select option **c** (Access the Store Control Functions Screen) from the SYSTEM KEYS menu.

SYSTEM KEYS

System Keys are used to request special control functions. Their action is independent of the current application(s). They have no direct effect on the current application, even though its screen will no longer be displayed.

Type one of the following letters or a function key.

m	Access the System Message Display Screen.
c	Access the Store Control Functions Screen.
b	Access the Background Application Control screen.
s	Start new application. (displays the System Main Menu)
w	Access the Window Control screen.
n	pass control to the next higher numbered window owned by this operator (Next).
p	pass control to the next lower numbered window owned by this user (Preceding).
a	Access the Auxiliary Console Control Screen.
t	Switch to Terminal Mode.
j	Switch to Terminal Java Console.
X	Switch to enhanced mode graphical extensions screen.

F1HELP F2 F3QUIT F4 F5 F6 F7 F8 F9Disconnect

The STORE CONTROL FUNCTIONS menu appears:

CS02S002 STORE CONTROL FUNCTIONS

Select one of the following:

1. Terminal Functions
2. Controller Functions
3. TCC Functions
4. LAN Functions
5. System Functions
6. Communications Functions

Type your selection number, then press Enter.

F1Help F2 F3Quit F4 F5 F6 F7 F8 F9 F10

Each option on this menu displays an additional list of functions.

Type the number of the option you want, and press **Enter**. The panel for that option appears. It contains the list of functions specific to the option. Type the number of the function you want, and press **Enter**. Each time a function you have selected is complete, the system returns to the options panel from which you selected that function. For example, if you select **Display Terminal Status**, when the function is complete, the system returns to the TERMINAL FUNCTIONS menu.

Power management

Advanced Power Management (APM) is an industrial power management standard. 469x and 4800-7x1, 7x2 and 7x3 terminals, when equipped with APM hardware and enabled, can implement a subset of APM BIOS (V1.2). These terminals can be placed into a low power, standby state by program control, pressing the power switch, or by loss of AC power. In the standby state, the terminal memory is retained, the CPU is not processing, and some terminal devices are powered down, but the OS, application, and data are maintained. In addition, when the terminal is in the APM Standby or Suspend state, it is not communicating with the controller and will be off-line (the indicating light is off).

Under program control, the terminal can be placed into standby using the programmable power functions of ADXSERVE. See *Designing Applications with 4680 BASIC* in the 4690 OS: Programming Guide for more information on ADXSERVE functions.

To enable the terminals to use standby mode, go to the Terminal System Function screen at the store controller and select the menu item **Enable Storage Retention**. At that point you select the amount of time to wait for inactivity before entering standby mode. The default is 30 minutes, which means that if there is no activity at a terminal for 30 minutes, it will enter standby mode. Entering all zeros provides the terminal storage retention function without an inactivity timeout. Applications can provide an inactivity time out setting. Storage Retention can also be enabled using the ADXSERVE function. See *Designing Applications with 4680 BASIC* in the 4690 OS: Programming Guide for more information on ADXSERVE functions.

To return the terminal to its fully active state, the power switch can be pressed or an external interrupt from an I/O device can be done, such as a key press on the keyboard, mouse movement, or touching the touch display. In the case of AC power loss, a backup battery must be installed to maintain the system, and AC power must be restored before the terminal can be returned to its full power state.

APM is still used by 469x and 4800-7xx terminals, as well as 4800-7xx controllers, running 4690 OS in Classic mode.

Advanced Configuration Power Interface (ACPI) is the successor to APM and is a widely accepted power management specification that allows operating systems to control the power distribution to peripherals such as displays, printers, and hard drives. ACPI provides several power states with varying levels of power savings. Support was added to 4690 OS for ACPI S3 (suspend to RAM or deep sleep) and S5 (soft power off) power management for certain terminal types.

ACPI power state S5, also called soft power off, allows the system to go into a "soft" power off mode, with minimal devices still drawing power from the system. These devices may be used to generate a wake event, such as Wake on LAN. S5 is supported on TCxWave 6140 Series Model 100, SurePOS 300 Series Model 350 and SurePOS 700 Series Models 7x1, 7x2, 7x3, 7x4 and 7x5. S5 Wake on LAN is supported for Enhanced Mode only, but Wake on Timer is supported for Classic and Enhanced mode.

ACPI power state S3, also called Deep Sleep or Suspend to RAM, allows the system to recognize a sleep request, suspends the system to RAM, notifies the software, and powers down the appropriate hardware. The CPU is shut down and all I/O devices not connected to USB ports enabled for S3 wake-up are powered down, but memory is retained in RAM (which retains power). When the system

recognizes a wake event, it restarts the hardware, restores the system state, and notifies the software. This allows for quick restarts without requiring reloading. The terminal Deep Sleep solution does not allow applications to continue during sleep mode. S3 is supported on TCxWave 6140 Series Model 100, SurePOS 300 Series Model 350 and SurePOS 700 Series Models 7x3, 7x4, and 7x5. S3 is only supported on those SurePOS 700 Series terminals with Deep Sleep hardware support and only for Enhanced mode. S3 is not supported on controller/terminals or controllers.

Entering a Power Management state

There are several methods for Enhanced terminals to enter Deep Sleep state:

- Pressing the power button (when properly configured)
- Programmatically (via an ADXSERVE call)
- Based on an inactivity timer

On all Enhanced terminals which support Deep Sleep, except TCxWave 6140 Series terminals, you can configure the function that is performed when the power button is pressed. The choices are Shutdown, Suspend (Deep Sleep), or Dump.

Under program control, the terminal can be placed into Deep Sleep using the programmable power functions of ADXSERVE. See Chapter 17, "Designing Applications with 4680 BASIC" in the 4690 OS: Programming Guide for more information on these ADXSERVE functions.

Note: Terminal Storage Retention must be enabled to allow the application put the terminal into Deep Sleep mode. If storage retention is disabled, the power down ADXSERVE call causes the system to initiate a soft power down instead of entering Deep Sleep mode. To allow application compatibility, ADXSERVE calls when using ACPI in Enhanced terminals is analogous to making the same calls on APM Classic terminals. For example, with storage retention enabled, the ADXSERVE call to power down the terminal causes a properly configured ACPI Enhanced terminal to enter Deep Sleep mode, whereas that call causes an APM terminal to enter standby mode.

You can use the OCF system function menu to enable Terminal Storage Retention for all terminals on the system and to set an inactivity timeout value. 4690 OS V6R2 uses this inactivity timer event to enter the Deep Sleep state (for Enhanced mode terminals that support this function) or to enter the standby state (for Classic mode terminals).

Note: Going to sleep via the Power Button is not supported with the TCxWave 6140 Series because the Power Button always functions as a dump button with this Series, regardless of the configuration.

Waking from a Power Management State

There are several methods to wake an Enhanced mode terminal from either Deep Sleep or soft power off state:

- Power button
- Wake on LAN (WOL)
- Wake on timer (or alarm)

When in Deep Sleep or soft power off states, the hardware power button may be used to wake the system. The 4690 OS V6R3 Enhanced mode supports Wake on LAN from Deep Sleep and soft power off states for TCxWave 6140 Series, SurePOS 700 Series Models 7x3, 7x4, 7x5, and the SurePOS 300 Series Model

350. Wake on LAN (WOL) is also supported from soft power off state for SurePOS 7x1/7x2 systems. WOL support must be enabled in the BIOS setup (the default is off).

When the ADXSERVE power off call is used by the application, an optional parameter may be specified to set a wake-up time.

Note: The wake up time used is the one set with the ADXSERVE call, not the one set in BIOS.

Deep Sleep mode also has the following wake methods:

- PS/2 mouse or keyboard
- Standard USB mouse or keyboard
- Port A USB
- Wake on LAN

With the hardware properly enabled (BIOS setting and/or jumper) a touch of a key on a PS/2 attached system keyboard or a movement of a PS/2 attached mouse will wake from Deep Sleep mode. A standard USB keyboard or mouse attached to one of the standard (non-powered) USB ports can be used to wake the system. For information related to jumper settings, refer to the Power Management Configuration "POS PowerManagement_1.4a_2009.pdf" Knowledgebase document which describes power management options for Point of Sale hardware.

A new SurePort card has been developed on which port A is designated to support wake from Deep Sleep when a USB POS keyboard is attached. The other hardware power management requirements must also be in place, and a jumper might have to be moved.

Although the operating system does not directly provide an application to send Wake on LAN commands to terminals, Enhanced mode terminals can wake-up using this method if the function is enabled in BIOS. The RMA software solution includes the ability to send wake-up commands to terminals. Customers or business partners who wish to provide alternate or additional wake-up software may refer to the 4690 OS Communications Programming Reference chapter "Using Communications and Systems Management" for details regarding a file that 4690 OS provides to map terminal numbers to MAC addresses.

Note: RS-485 devices cannot be used to wake ACPI systems.

Using terminal functions

The Terminal Functions option provides functions that help you control the activities of your store terminals. To display the TERMINAL FUNCTIONS menu, type **1**, and press **Enter**. The TERMINAL FUNCTIONS menu appears:

CS03S058

TERMINAL FUNCTIONS

Select one of the following:

1. Display Terminal Status
2. Enable Terminal Storage Retention
3. Disable Terminal Storage Retention
4. Start Terminal Application
5. Stop Terminal Application
6. Java Application Functions
7. Load Terminal Storage
8. Dump Terminal Storage
9. Load Terminal Configuration Data

Type your selection number, then press Enter. 1

F1Help F2 F3Quit F4 F5 F6 F7 F8 F9 F10

Display terminal status

This function gives you information about a specific point-of-sale terminal. When you select this option, you are prompted to specify a terminal number. When you enter a terminal number, the DISPLAY TERMINAL STATUS menu appears, displaying the following information:

- The date and time showing on the terminal
- Whether the terminal storage retention function is enabled or disabled
- Whether trace or performance monitoring is taking place
- What application (if any) is running on the terminal
- The partner terminal (if any)

Note: Terminals that are in Deep Sleep (S3) or APM standby do not provide status.

Enable terminal storage retention

The storage retention function on your point-of-sale terminal enables the use of a battery backup if your hardware supports it. These machines will switch to battery backup when AC power is lost. Storage retention enables various power modes depending on the machine type and the function performed.

Note: The storage retention function is not available on a controller/terminal. This backup function might or might not be used in your store.

When you enable the storage retention function from the store controller, you enable the function for all terminals on that TCC Network. If you want to enable the function on specific terminals, you must enable the function from the specific terminal. See “Enabling the storage retention function” on page 120 for more information on enabling this function on a single terminal.

If you have multiple controllers using the DDA, you can enable storage retention from the master store controller only. When you enable storage retention from the master store controller, the system enables all of the terminals on that master store controller’s TCC Network. To enable storage retention on terminals on a TCC Network for a store controller that is not the master store controller, you must enable storage retention from the master store controller.

Note: If you enable the terminal storage retention, you are prompted to use or change an inactive time out period. This setting will allow terminals to enter a low power state, depending on their hardware support and configuration, after the entered time has been exceeded for terminal inactivity. You can enter a zero time to disable to inactive timer.

Disable terminal storage retention

When you disable the storage retention function for terminals from the store controller, you disable the function for all terminals on that TCC Network.

Note: The storage retention function is not available on a controller/terminal. If you want to disable this function on specific terminals, you must disable the function from the specific terminal. See “Disabling the storage retention function” on page 121 for more information on disabling this function on a single terminal.

If you have multiple controllers using the DDA, disable storage retention from the master store controller only. When you disable storage retention from the master store controller, the system disables all of the terminals on that store controller’s TCC Network. To disable storage retention for terminals on a TCC Network for a storage controller that is not the master store controller, you must disable storage retention from the master store controller.

Start terminal application

This option lets you run an application on a specific terminal. When you select this option, the START TERMINAL APPLICATION menu appears and prompts you to specify a terminal number and application name.

Type in the terminal number and application program name, and press **Enter**.

Stop terminal applications

Sometimes you might need to stop an application that is currently running on a terminal. When you select the **Stop Terminal Application** option, you are prompted to enter the terminal number. This function stops all applications on that terminal that is including any 3270 emulation. You can stop applications on only one terminal at a time.

Note: Once you have stopped terminal applications, you must start the primary application before starting any secondary applications.

Use Java application functions

This option displays the list of Java functions available in the terminal. When you select this option, the JAVA APPLICATION FUNCTIONS menu appears. From the JAVA APPLICATION FUNCTIONS menu you can display the current Java application status, start a Java application, or stop a Java application.

Displaying Java application status: When you select this option, the DISPLAY JAVA STATUS panel appears. Use this panel to display the current Java application status. This panel indicates:

- Whether the Java application is active or inactive
- How the Java application was initially started
- Class and parameters passed to the Java Virtual Machine

When you select this option, you are prompted to enter the terminal number. Type in the terminal number and press **Enter**.

Starting Java applications: When you select this option, the START JAVA APPLICATION menu appears and prompts you to specify a terminal number and class and parameters. This panel also shows you the Java Virtual Machine, which cannot be modified.

Stopping Java applications: When you choose this option, you are prompted to enter the terminal number. This function stops the Java application on that terminal.

Type in the terminal number and press **Enter**.

Load terminal storage

This option can be used after applying software maintenance to load the terminal with the maintenance changes.

Note: This option does not work on the terminal portion of a controller/terminal. Terminal storage on a controller/terminal is loaded when the controller/terminal is IPLed.

When you select this option, you are prompted to specify a terminal number. Type the terminal number, and press Enter. If you want to load terminal storage for all terminals, enter an asterisk (*). (You can only use an asterisk (*) to reload all terminals from the acting Master Controller.) On Enhanced Mode terminals, this function will only reload the 4690 OS code, not the underlying hardware interface layer. To reload the hardware interface layer, use the Load Terminal Configuration Data option below.

Note: If you have multiple controllers using DDA and you choose this option from the master store controller, the system loads all of the terminals on the TCC Network.

Dump terminal storage

Use this option to collect problem analysis data. This option dumps the contents of the specified terminal's storage to a file. The terminal's storage is reloaded.

Note: This option does not work on the terminal portion of a controller/terminal. The terminal storage of a controller/terminal is dumped when the controller/terminal is dumped. See "Dump controller storage" on page 100 for information on dumping a controller/terminal.

When you select this option, you are prompted to specify a terminal number. Type the terminal number, and press **Enter**. The system allows you to dump the storage on one terminal only.

For more information on terminal dumps, see the *4690 OS: Messages Guide*.

Load terminal configuration data

Use this option to load the active terminal configuration for one or all terminals. If you want to load the active terminal configuration for all terminals, enter an asterisk (*). See "Configuring legacy terminals (4683 and 4693)" on page 52 and "Activating your terminal configuration" on page 76 for how to change and activate a terminal configuration.

Note: If you have multiple controllers using DDA and you choose this option from the master store controller, the system loads the active terminal configuration for all terminals on the TCC Network.

When you select this option, you are prompted to either specify a terminal number or an asterisk (*) to load the active terminal configuration for all terminals. (You can only use an asterisk (*) to reload all terminals from the acting Master Controller.) Type a terminal number or an asterisk (*), then press **Enter**. Any application running at the specified terminal or terminals will be ended.

If you have a Mod2 terminal attached to a Mod1 terminal, you do not need to load terminal configuration data on the Mod2 terminal if the Mod2 is powered on and configured. If the Mod2 terminal is powered on and configured when you load terminal configuration data on the Mod1 terminal, the terminal configuration data will be loaded onto the Mod2 terminal. On Enhanced Mode terminals, this function will reload both the underlying hardware interface layer and the 4690 OS code.

Note: This option does not work on the terminal portion of a controller/terminal or its attached Mod2 terminal. The active terminal configuration for a controller/terminal and its attached Mod2 terminal is made effective when the controller/terminal is IPLed.

Using controller functions

The Controller Functions option provides functions that help you control certain activities of your store controllers.

To select this option, type **2** from the STORE CONTROL FUNCTION menu, and press **Enter**. The CONTROLLER FUNCTIONS menu appears:

CS03S046

CONTROLLER FUNCTIONS

Select one of the following:

1. Display Controller Status

2. Enable Controller RAM Disk

3. Disable Controller RAM Disk

4. Load Controller Storage

5. Dump Controller Storage

Type your selection number, then press Enter. 1

F1Help F2 F3Quit F4 F5 F6 F7 F8 F9 F10

These options show you the store control functions that you can perform on a store controller.

Display controller status

When you select this option, the system displays the status of the store controller you are using. If this store controller is also part of a multiple controller system using DDA, the system also lists the connections to other store controllers.

Enable controller RAM disk

Use this option to activate a RAM disk on this store controller. The next time this store controller IPLs, the RAM disks that are defined are activated.

Disable controller RAM disk

Use this option to deactivate a RAM disk on this store controller. The next time this store controller IPLs, the defined RAM disks are not activated.

Note: If a store controller cannot complete the IPL, or if the system configuration cannot be loaded because a RAM disk is too large, you must disable the RAM disk. In this case, use the Supplemental Option using the CD-ROM to disable the RAM disk.

Load controller storage

You usually select this option only after reconfiguring a store controller. When you select this option, the system loads the memory of this store controller (or another, that you specify by ID).

Notes:

1. If you have multiple controllers using DDA and you choose this option from the master store controller, the system allows individual controllers or all the controllers on the TCC Network to be IPLed. If you are using a store controller that is not the master store controller, the system loads only the store controller from which the request was entered.
2. This option loads both the terminal and controller portions of a controller/terminal.
3. For enhanced controllers this option will reload 4690 OS on the controller, not IPL the controller.

Dump controller storage

Use this option to collect problem analysis data. Selecting this option dumps the contents of the store controller to a file. This file is located at the store controller that is dumped.

Note: This option dumps the contents of both the terminal and controller portions of a controller/terminal.

For more information on store controller dumps, see the *4690 OS: Messages Guide*.

Using TCC functions

The TCC Functions option provides functions that help you control certain activities of your TCC Network.

To select this option, type **3** from the STORE CONTROL FUNCTIONS menu, and press **Enter**. The TCC FUNCTIONS menu appears:

CS03S054

TCC FUNCTIONS

Select one of the following:

1. Allow Store Controller Backup

2. Prevent Store Controller Backup

3. Resume Store TCC Control

Type your selection number, then press Enter. 1

F1Help F2 F3Quit F4 F5 F6 F7 F8 F9 F10

Allow store controller backup

When you choose this option, the system asks you for the ID of the store controller that is going to back up the network.

Note: If you have multiple controllers using DDA and you choose this option from the master store controller, the system enables store controller backup for

another store controller (which you can specify) or all store controllers on the TCC Network. If you are using a store controller that is not the master store controller, the system enables backup for this store controller only.

See Chapter 31, “Store controller backup,” on page 493 for a step-by-step explanation of the store controller backup and control resumption process.

Prevent store controller backup

When you choose this option, the system requests the ID of the store controller providing backup. This option prevents the store controller from providing backup.

If the store controller is actively providing store controller backup when you select this option, it stops providing it immediately.

Note: If you have multiple controllers using DDA and you choose this option from the master store controller, the system prevents store controller backup for another store controller (which you can specify) or all store controllers on the TCC Network. If you are using a store controller that is not the master store controller, the system prevents backup for this store controller only.

See Chapter 31, “Store controller backup,” on page 493 for a step-by-step explanation of the store controller backup and control resumption process.

Resume store TCC control

Use this option to return control of the network to the primary store controller.

Note: The backup store controller that assumed control from the original one is still enabled for store controller backup. The Resume Store TCC Control option does not change that status.

Before performing the resume function, you must IPL (power off and then power on) the disabled store controller. Be sure you do this before executing the resume function.

See Chapter 31, “Store controller backup,” on page 493 for a step-by-step explanation of the store controller backup and control resumption process.

Using LAN functions

The LAN Functions option provides functions that help you control certain activities of your LAN (MCF Network) system using DDA.

To select this option, type **4** from the STORE CONTROL FUNCTIONS menu, and press **Enter**. The MULTIPLE CONTROLLER FUNCTIONS menu appears.

CS03S064

MULTIPLE CONTROLLER FUNCTIONS

Select one of the following:

1. Activate Master Controller

2. Activate File Server Controller

3. Deactivate Master Controller

4. Deactivate File Server Controller

Type your selection number, then press Enter. 1

F1Help F2 F3Quit F4 F5 F6 F7 F8 F9 F10

Note: These options have an effect only on a multiple controller system using DDA.

Activate master controller

When you choose this option, this store controller becomes the central point of control for all other store controllers in the system and keeps the prime version of distributed files.

Use this option when:

- The alternate master store controller is needed to take over control for the master store controller.
- You return control of the system to the master store controller. You must be working at the master store controller when you do this.

Note: The VFS server only supports distribution of long file names when it is running on the active master.

Activate file server controller

When you select this option, the system assigns the file server store controller to keep the prime versions of your non-system-mirrored files. For an explanation of mirrored files, see “Mirrored File” on page 447.

The same rules apply for activating a store controller as a file server controller as for a master store controller. Also, the activating and deactivating must be done in the same order as with a master store controller and an alternate master store controller. The order is as follows:

1. Go to the alternate file server controller; activate it as the file server controller.
2. Once the original file server controller is back in operation, deactivate the alternate file server controller. Do this by selecting the **Deactivate File Server Controller** option discussed in the next section.
3. Return to the configured file server controller and select **Activate File Server Controller**.

After you have completed these steps and the store controllers are each functioning according to their original assignments, continue with the following step.

4. At the configured file server controller, restart necessary background applications using the keys described in Table 5 on page 41.

Note: If you have one store controller performing two roles, you must activate each role using the appropriate options.

These steps assume that you have a system with two store controllers. One acts as the master store controller and file server, the other as the alternate for both roles. If the first one goes down, you must select the **Activate Master Store Controller** option and the **Activate File Server** option on the other one. This is also true when giving control back to the first one.

Deactivate master controller

When you choose this option, the system removes the store controller as the central point of control for the system.

Deactivate file server controller

When you choose this option, this store controller is no longer assigned to keep the prime versions of your non-system-mirrored files.

Examples of MCF functions

This section contains examples that show how you might use the MCF Functions to deactivate and activate your master store controller and your file server during maintenance.

Example 1: You have a multiple controller system using DDA with two store controllers functioning as the master store controller and file server and their alternates: one store controller functions as the master and the file server; the other functions as the alternate master and the alternate file server. In this example, the first store controller will be called M, and the second AM.

Store controller M develops problems with its hard disk drive. The hard disk drive must be replaced. Go to store controller M and select **Deactivate Master Controller**. Then, at store controller AM, select **Activate Master Controller**.

Store controller AM (the alternate master store controller) now functions as the master store controller. The configured (now deactivated) master store controller still functions as the file server controller.

Note: If the acting master cannot perform the role, the multiple controller system using DDA will be temporarily without a master store controller until the alternate master store controller is activated as the acting master store controller. In the time between the master store controller going down and the alternate store controller being brought up, no file updates are sent on the LAN (MCF Network).

Example 2: Use the same configuration initially described in “Example 1” on page 103: one store controller functioning as the master store controller and file server, the other performing the alternate roles. As in Example 1, you have activated the alternate master store controller to function as the master store controller while hardware maintenance is performed on the configured master store controller.

When maintenance is complete, the master store controller is ready to resume its role as master store controller. First, you must deactivate the alternate master store controller so that it will no longer perform the master store controller roles. To do this, go to the alternate master store controller. Using the MULTIPLE CONTROLLER FUNCTIONS menu, deactivate it as the master store controller.

Then, working at the configured master store controller, select **Activate Master Controller**. These two selections return each store controller to its originally configured role.

Note: When the configured master store controller comes back up after maintenance, it automatically assumes the role of alternate master store controller. It continues to do so until you restore the role assignments to their original store controllers.

Using System Functions

The System Functions option provides general functions for your system.

To select this option, type **5** from the STORE CONTROL FUNCTIONS menu, and press **Enter**. The SYSTEM FUNCTIONS menu appears:

CS03S059

SYSTEM FUNCTIONS

Select one of the following:

1. Set System Date and Time

2. Set System Message Level

Type your selection number, then press Enter. 1

F1Help

F2

F3Quit

F4

F5

F6

F7

F8

F9

F10

Set system date and time

The date and time do not have to be set daily. This function is used to set the date and time for the first time or to reset time, for example, when changing to daylight savings time.

You must perform this function on the acting master store controller.

When you select this function, the SET SYSTEM DATE AND TIME menu appears. You are prompted to enter the date and time in the format shown on the menu. Enter the date and time as requested, and press **Enter**.

Set system message level

This option lets you specify the messages you want to display on the SYSTEM MESSAGE LEVEL panel based on severity code. A *severity code* is assigned by the system to every error or event taking place on your system. Using the Set Message Display option, you can specify which errors or events you want displayed on the SYSTEM MESSAGE DISPLAY panel.

See “Using system request keys” on page 42 for information on how to access the SYSTEM MESSAGE DISPLAY panel, and see “Using function keys” on page 40 for information on how to use the panel.

Severity level	Type of error/event
1	Affects the entire system or multiple terminals (for example, a hard disk drive failure error)
2	Affects a single physical unit such as a terminal or an input/output device (for example, a terminal program error or an input/output device failure)
3	Affects a programming function (for example, the program supporting terminal requests might not be operating correctly)
4	Indicates a statistical error/event has been detected (for example, an error in a communications line or in a device channel)
5	Denotes normal expected occurrences (for example, events occurring as a result of operator action)

Severity codes range from 1 (the most severe error) to 5 (informational type of error). The following table explains the severity code levels:

Severity Level	Type of Error/Event
1	Affects the entire system or multiple terminals (for example, a hard disk drive failure error)

- 2 Affects a single physical unit such as a terminal or an input/output device (for example, a terminal program error or an input/output device failure)
- 3 Affects a programming function (for example, the program supporting terminal requests might not be operating properly)
- 4 Indicates a statistical error/event has been detected (for example, an error in a communications line or in a device channel)
- 5 Denotes normal expected occurrences (for example, events occurring as a result of operator action)

When you select the **Set Message Display** option, you are prompted to enter a message level from **1** to **5**.

Note: For normal store operations, a message level of 3 is recommended and is the default. This message level setting ensures that the operator receives all the messages needed for operator recovery. Type the number of the message level you want, and press **Enter**. Only messages indicating errors or events of the severity code you specify and those that are more severe are displayed on the SYSTEM MESSAGE DISPLAY panel.

For information on the format of system messages displayed on the SYSTEM MESSAGE DISPLAY panel, see the *4690 OS: Messages Guide*.

Using Communications Functions

The Communications Functions option provides functions that let you enable and disable communication links. You must enable and disable links at each store controller.

To select this option, type **6** from the STORE CONTROL FUNCTIONS menu, and press **Enter**. The COMMUNICATIONS FUNCTIONS menu appears:

CS03S070
COMMUNICATIONS FUNCTIONS

Select one of the following:

- 1. Enable Link
- 2. Disable Link (no force)
- 3. Disable Link (force)
- 4. Display Link
- 5. Enable LAN for LU 6.2 Communications
- 6. Disable LAN for LU 6.2 Communications
- 7. Display LAN Status for LU 6.2 Communications

Type your selection number, then press Enter. 1

F1Help
F2
F3Quit
F4
F5
F6
F7
F8
F9
F10

Enable link

When you select this option, the system prompts you for the name of the communication link defined during configuration. Entering the name of the communication link initializes the link for communication. The first time a link is enabled, initialization takes from 2 to 10 minutes.

The Enable Link option verifies that the specified link is configured and that the communications adapter defined for it by configuration exists. Be aware of the following constraints when enabling a link:

- For all communications except X.25, only one link can be enabled for each communications port.
- For X.25 communications, you can have up to four links enabled for each X.25 communications port.
- When a communication link is enabled, you must disable the link to remove the communication support.

If the communication line is *leased*, the connection is made when the communication link is enabled. If the communication line is *switched*, connection is not made until the user dials.

When SNA communications is loading and initializing, it uses less priority than all other background applications you have running, so as not to interfere with them. You can improve the loading time of SNA communications (at the expense of background programs) by defining a User Logical File Name parameter of ADXHPRIO and setting its value to a number from 1 through 9. This number represents the background priority to be used when SNA is loading and initializing.

Notes:

1. Even though command complete messages indicate otherwise, for X.25 communication on a switched virtual circuit (SVC)-IN, link connection is made when the link receives a call. Also, when a local communication link is enabled, the connection is made.
2. A communications link can also be enabled by configuring and starting a background application called ADXHSNLL.286. You can use this background application to automatically enable an SNA link at IPL time.

ADXHSNLL expects to receive the following parameters, separated by commas and used without any embedded spaces:

- *Linkname* is the name of the SNA link you want to enable.
- *Retry* and *interval* are used when ADXHSNLL is unsuccessful in its first attempt to open the link. That is, whenever ADXHSNLL tries to open the link and receives a return code from the SNA driver that implies a temporary outage, ADXHSNLL waits the number of seconds specified by *interval*, and tries again to open the link, up to the number of times specified by *retry*. The range of acceptable values for *retry* is 0 through 9999 times. If you specify a number outside this range, 2880 is used. If you are specifying a peer link, use 0. The range of acceptable values for *interval* is 0 through 999 seconds. If you specify a number outside this range, 60 is used.

Disable link (no force)

When you choose this option, you must enter the name of the active communication link you want to disable.

Disable without force allows all active LU applications using the communication link to complete, but no new LU application can start using the link.

When the last LU application has completed using the link, connection is broken and the communications adapter can now be used to support another communication link.

Disable link (force)

When you choose this option, you must enter the name of the active communication link you want to disable.

Disable with force places the communication link in a state such that every attempt to use it results in an error. This error should cause most applications to end, but user-supplied applications might have to be cancelled.

When the last LU application has completed using the link, the connection is broken and the communications adapter can now be used to support another communication link.

Note: This option is not available for a local communication link.

A communications link can also be disabled by configuring and starting a background application called ADXHSNSL.286. This could be used to automatically disable an SNA link when switching master controllers. ADXHSNSL expects to receive one parameter, the name of the SNA link you want to disable.

Defining the ADXHSNSL program must be done when two controllers (one master and one backup) have identical host link definitions and are using ADXHSNLL. You must define ADXHSNSL using the Background Applications menus under Controller Configuration. ADXHSNSL must be defined for both the master and alternate master controller. To define ADXHSNSL, use the following settings:

Background application	Configuration option
ADXHSNLL	Start when master Stop when not master
ADXHSNSL	Start when not master Stop when master

Display link

This option displays the state of a specified communication link.

Enable LAN for LU 6.2 communications

Enabling the LAN verifies that the LAN is configured for communications by LU 6.2 applications and that the Ethernet network defined by configuration exists.

The Enable LAN for LU 6.2 option prepares for communication between a single store controller and all other store controllers on the LAN.

If configuration defines the LAN links as *not auto-activated*, connections are not made until an LU 6.2 application attempts communication across the LAN. If configuration defines the LAN links as *auto-activated*, connections have been made when the enable completes.

Disable LAN for LU 6.2 communications

Disabling the LAN ends communication across the Ethernet network. The communication links are placed in a state such that every attempt to use them by an LU 6.2 application results in an error. This error should cause most applications to end, but user-supplied applications might have to be canceled for the communication link to be used again.

When the last LU 6.2 application has completed using the link, the connections are broken.

Display LAN status for LU 6.2 communications

This option displays the state of a specified communication link.

Chapter 4. Point-of-Sale terminal operations

This chapter shows how to request system functions at your point-of-sale terminal.

Figure 14 on page 109 shows the different Point-of-Sale terminals and the devices that are associated with the terminals.

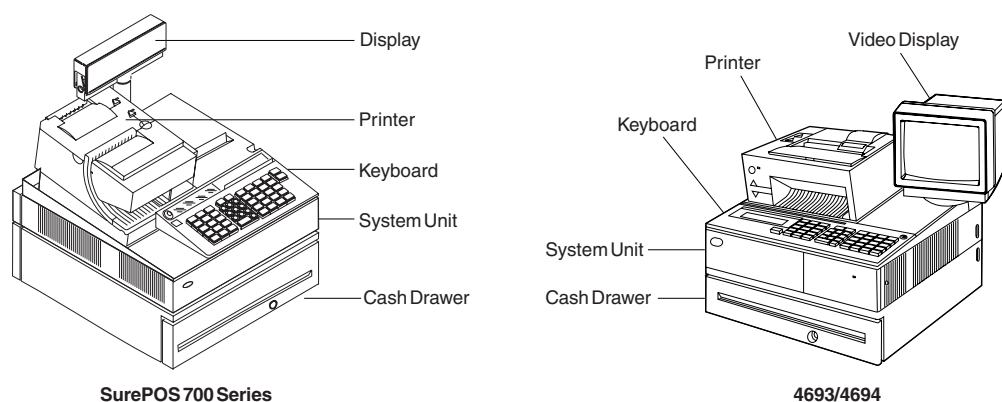


Figure 14. Point-of-Sale terminals

System functions can be used to display system messages, change or display a terminal number or configuration, change the volume of the keyboard tone, or set the time and date on an offline terminal. Some of these functions are used under special operating conditions rather than in daily operations.

Quick reference table for system functions

The following table provides a quick reference for system functions.

System function	Key code*	Page
Display Wait Message	1	118
Display Offline Message	2	118
Display System Message	3	118
Change Keyboard Tone Volume**	6	120
Display Terminal Number	7	120
Display Date Format at Terminal	5,0	119
Set Date at the Terminal	5,1	119
Set Time at the Terminal	5,2	120
Switch to enhanced mode graphical extension screen	5,8	122
Swap Terminal Video Screens	5	121
Change Keyboard Tone Volume**	6	120
Display Terminal Number	7	120

System function	Key code*	Page
Load Set Terminal Characteristics (STC) Program	7,1	122
Enter Terminal Number	1xxx	126
Display Current® Terminal Configuration	1xxx	122
Change Terminal Number	1xxx	126
Reset Terminal Number to Zero	1000	128
Change Terminal Configuration	2,2,2,2	128
Format Terminal Hard Disk Drive	2,2,2,2	130
Exit an STC Function	3,3,3,3	131
Enable Storage Retention	7,2	120
Disable Storage Retention	7,3	121
Display Memory Available	9,6	121
Dump Terminal Storage	9,8,9,8	121
* The complete key sequence is S1 , <i>key code</i> , S2 .		
** Some keyboards do not support this function.		

Using a keyboard with Java applications

The operating system enables you to use a keyboard that is attached to the PS/2 keyboard port of the terminal. This enables keyboard functions to be accessed from Java applications. A PS/2 keyboard or an Alphanumeric Point-of-Sale (ANPOS) keyboard can be attached to the keyboard port. Configuration for the keyboard port is provided in the terminal device group configuration. See the *4690 OS: Planning, Installation, and Configuration Guide* for details on terminal device group configuration.

Access to the keyboard at the terminal is limited to a Java application. When a PS/2 keyboard is configured, a POS keyboard must also be configured. An ANPOS keyboard can be shared so that both PS/2 keyboard functions and POS functions are provided depending on the active console (either terminal or Java console).

On SurePOS 300/700 Series systems or TCxWave 6140 Series systems, if any keyboard is identified during system IPL, the keyboard is used as the point-of-sale keyboard. The ANPOS keyboard attached to the PS/2 port is used for the controller or for the Java application. If a keyboard is not identified during system IPL, an ANPOS keyboard is identified, and terminal video is configured, the ANPOS is used as a shared keyboard. If no ANPOS keyboard is attached to the PS/2 port on the SurePOS 300/700 Series or TCxWave 6140 Series, the operating system uses the first ANPOS keyboard found as the system keyboard. If no other keyboard is found, the ANPOS keyboard is used as the system keyboard.

If Java is enabled, you have access to a Java console using the terminal video. To switch between the terminal console and the Java console, press **Alt+SysRq**. To display the Java console, press **Alt+SysRq+J**. To display the terminal console, press **Alt+SysRq+T**.

Using a keyboard and video display on a controller/terminal

The controller/terminal can be configured so that the controller portion and the terminal portion share the use of a keyboard and video display, share the video display only, or do not share a keyboard and the video display. When the controller/terminal is sharing a keyboard, all keystrokes are directed to the current operational environment. If a magnetic stripe reader (MSR) is attached to the

shared keyboard, data sent from the reader is always directed to the terminal side even if the current operational environment is the controller.

When the controller/terminal is sharing the video display, all display output from running the controller or terminal applications is accepted regardless of the current operational environment. The last display output for each side is displayed when that side becomes the current operational environment.

Terminal and controller applications that run on the controller/terminal continue to run regardless of the current operational environment.

Using a touch screen on a terminal

The terminal can be configured so that you can have a touch screen as the only keyboard and display device or include a keyboard and other displays according to current operating system restrictions. When the terminal is configured to use a touch screen, a keyboard is not required; the system functions that are provided by the keyboard are provided by the touch screen pseudo keyboard driver.

To configure a touch screen for a non-SurePOS 300/700 Series system, indicate that the terminal will use a video display. Then indicate the following for the video:

- In a controller/terminal terminal device group, the video must be shared. Indicate that the terminal will share the video and keyboard or that the terminal will share the video only. On controller/terminals, touch is supported only for terminal applications.
- The video display will be a touch screen display.
- Indicate whether an MSR and/or a keypad will be attached to the touch screen.

To configure a touch screen for a SurePOS 300/700 Series or TCxWave 6140 Series terminal, indicate that the terminal will use a video display. To do this, use the Terminal Device Characteristics in Generic Terminal Configuration. Additional configuration options are not needed. Touch displays on Enhanced Mode terminals must be USB-attached.

Touch Screen Firmware Update

4690 OS supports automatic firmware update for TCxWave 6140 Series system unit's built-in touch screen and certain models of 4820 monitors. To enable this function, the user needs to edit or create (if it doesn't exist) the terminal property file `vx_trmpr.dat` with the following info:

```
video.firmware.flash.disabled=false  
touch.firmware.flash.disabled=false
```

The user must then run `adxrtcl` and `adxnszal` on all controllers and reload the terminal.

Dual touch devices

Dual touch requires that both touch devices are USB attached 4820 touch screens.

Unintentional touch rejection

In some customer environments, the touch display of an infrared touch panel might detect unintended touches to the screen due to such things as loose clothing or insects. To avoid these unintended touch events, add the logical name, `ADXUTRSL`, to define an unintentional touch rejection (UTR) sensitivity level.

Add `ADXUTRSL` to `\adx_idt1\adxtrmuf.dat` to define the sensitivity level for the touch displays of all terminals attached to a controller. Add `ADXUTRSL` to `\adx_idt1\adxctluf.dat` to define the sensitivity level for the touch displays of the terminal side of a controller/terminal.

Unintentional touch rejection is available on SurePoint™ Display Models 4820-x1x and 4820-xLx. The factory default unintentional touch rejection sensitivity level for the touch device is disabled. The level defined for `ADXUTRSL` changes the setting that exists in the touch device.

Define `ADXUTRSL` to one of the following to adjust the unintentional touch rejection sensitivity level:

ADXUTRSL=0

Disabled - The UTR function is disabled. This setting allows for normal infrared touch functionality.

ADXUTRSL=1

Light – Light touches are required to register a touch; hovering just off the screen or touches by a sleeve, for example, would not register a touch.

ADXUTRSL=2

Medium Light – Medium light, slightly harder, touches are required to register a touch.

ADXUTRSL=3

Medium Hard – Medium hard touches are required to register a touch.

ADXUTRSL=4

Hard – Hard touches are required to register a touch; This setting requires a firm impact to register a touch.

ADXUTRSL=-

Leave unintentional touch rejection sensitivity level as is. This allows touch displays that have been set to differing sensitivity levels to retain those levels.

If ADXUTRSL has not been defined, 4690 will set the Unintentional touch rejection sensitivity level to disabled (same as ADXUTRSL=0). If an invalid value is defined for ADXUTRSL, 4690 will leave the sensitivity level as is (same as ADXUTRSL=-).

Tone Only Keyboard Interface

The Tone Only Keyboard Interface, TOKI, enables a touch display that does not have an integrated keypad to provide a POS tone function.

Add ADXTOKI to \adx_idt1:adxtrmuf.dat to enable/disable the TOKI function for the touch displays of all terminals attached to a controller. Add ADXTOKI to \adx_idt1:adxctluf.dat to enable/disable the TOKI function for the touch displays of the terminal side of a controller/terminal.

TOKI is available on SurePoint Display Models 4820-x1x and 4820-xLx. The setting defined for ADXTOKI changes the setting that exists in the touch device.

Define ADXTOKI to one of the following:

ADXTOKI=0

Disabled (default)

ADXTOKI=1

Enabled

ADXTOKI= -

Leave TOKI setting as is. This allows touch displays that have been set to different TOKI settings to retain those levels.

Defining ADXTOKI is required to enable TOKI. If ADXTOKI is not defined, then 4690 will override the current setting and disable TOKI (same as 0 value).

If an invalid option is specified, then TOKI is left as is (same as "-" value)

A change in TOKI setting forces an IPL in order for the touch device to provide a POS tone function.

TOKI should be enabled only on systems where it will not cause conflicts with other devices that have the capability to provide the POS tone function. 4690 supports only one POS keyboard. If TOKI is enabled and a POS keyboard or a touch display with an integrated keypad is also present, results are unpredictable as to which device will be selected as the POS keyboard and provide the tone function.

Requesting system functions

System functions are requested at the terminal by using the system function keys on the terminal keyboard. Each keyboard model has system function keys. Figure 15 on page 115 gives the location of system function keys on all keyboards.

System function keys should be selected and operated the same way on the 50-Key Modifiable and Matrix keyboards. To request a system function, use a *keying sequence*. Each function has a specific keying sequence that is keyed in as follows:

50-Key Modifiable or Matrix keyboard:

1. Press **S1**. This key precedes all system function requests. It tells the system that the information that follows is system request information.

2. Using the numeric keypad, enter the numeric code for the specific system function you want to use.
3. Press **S2**. This key enters your system request information and ends the system function.

Alphanumeric keyboard:

The Alphanumeric keyboard's system function keys are the two upper-left keys on the keyboard (see Figure 15 on page 115). The tops of the keys are labeled **F1** and **F2**; the front sides of the keys are labeled **S1** and **S2**. Press and hold the **Ctrl** key and then press either key to enter system function requests on the Alphanumeric keyboard.

1. Press **Ctrl** and **F1** together. The **Ctrl** key must be used with the function keys each time a system request is made.
2. Enter the numeric code for the specific system function you want to use.
3. Press **Ctrl** and **F2** together. This key sequence enters your system request information and ends the system function.

ANPOS keyboard:

The ANPOS keyboard has raised lettering to the right of the system function keys (see Figure 15 on page 115). The upper key is **S1**, and the key below it is **S2**. Press and hold the **Ctrl** key and then press either key to enter system function requests on the ANPOS keyboard.

1. Press **Ctrl** and **S1** together. The **Ctrl** key must be used with the function keys each time a system request is made.
2. Enter the numeric code for the specific system function you want to use.
3. Press **Ctrl** and **S2** together. This key sequence enters your system request information and ends the system function.

4693 Point-of-Sale (POS) keyboard (with or without an MSR), a **USB 50-Key keyboard** (with or without an MSR), a **4693 Alphanumeric Point-of-Sale (4693 ANPOS) keyboard** with an MSR, or a **4693 Modifiable Layout or USB 133-Key keyboard** with an MSR, **USB ANPOS keyboard** (with or without MSR), an LCD/Video (SurePoint™ Solution) keypad, a **50-Key keyboard with JUCC MSR**, **Modular ANPOS keyboard**, a **Modular 67-Key keyboard**, or a **Modular 67-Key LCD keyboard**:

1. Press **Ctrl** and **S1** together. The **Ctrl** key must be used with the function keys each time a system request is made.
2. Type the numeric code for the specific system function you want to use.
3. Press **Ctrl** and **S2** together. This key sequence enters your system request information and ends the system function.

Keyboard-V (POS) keyboard (with JUCC MSR), **Keyboard-VI (POS) keyboard** (with JUCC MSR), and the **50-key (POS) Keyboard** (with JUCC MSR):

1. Press **Ctrl** and **S1** together. The **Ctrl** key must be used with the function keys each time a system request is made.
2. Type the numeric code for the specific system function you want to use and press **Enter**.
3. Press **Ctrl** and **S2** together. This key sequence enters your system request information and ends the system function.

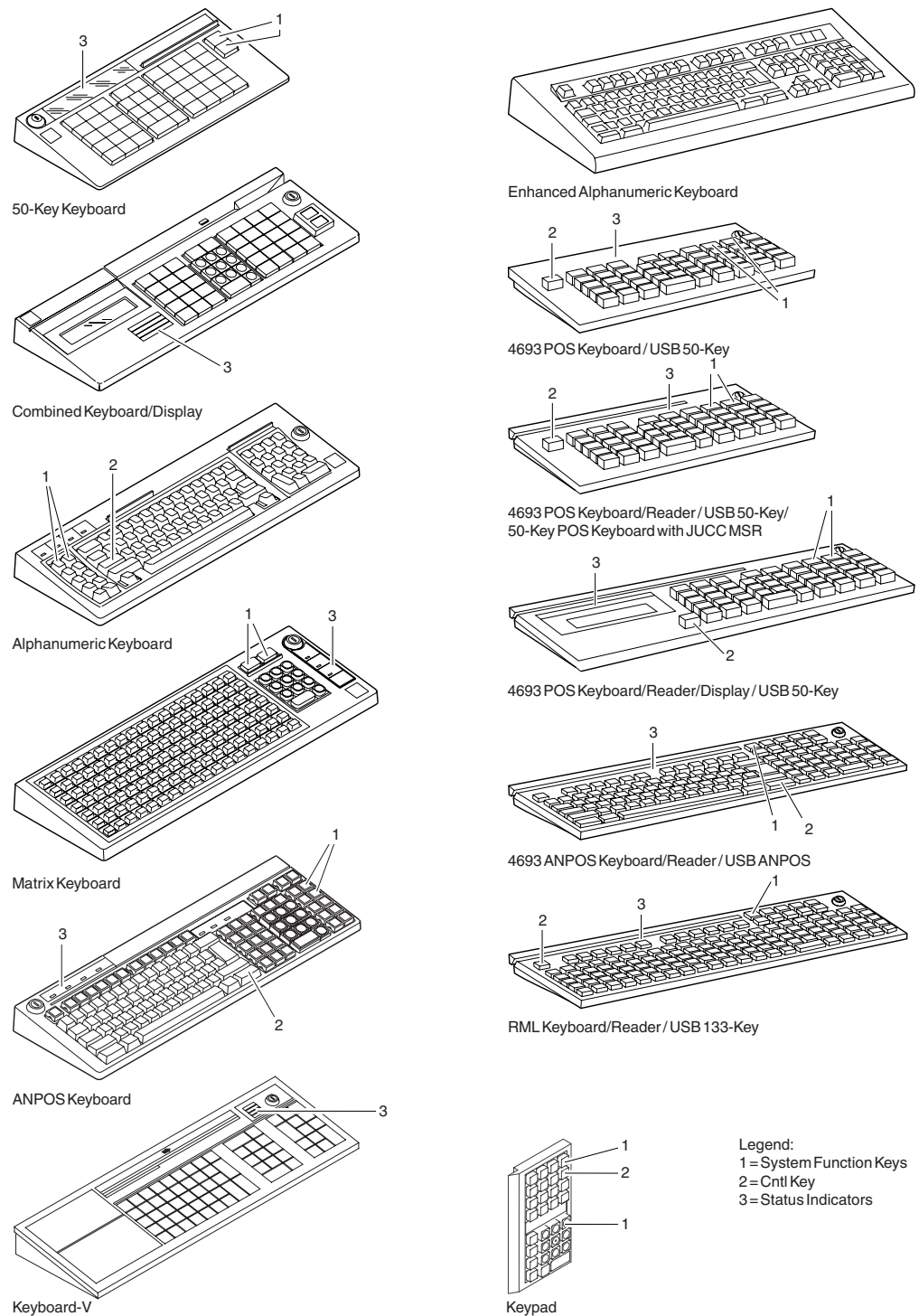


Figure 15. System function and status indicator keys on Point-of-Sale keyboards

Touch Screen pseudo keyboard:

When the terminal is configured to use a touch screen, a keyboard is not required; the system functions that would be provided by the keyboard are provided by the touch screen pseudo keyboard driver. The touch screen pseudo keyboard on a USB display can be started by touching the lower left corner of the display three

consecutive times, then touching the lower-right corner of the display one time. After this touch sequence, the pseudo keyboard appears.

The touch screen pseudo keyboard on an RS-485 display can be started by powering off the display and, within three seconds, powering on the display and touching the lower-left corner of the display one time.

The touch screen support pseudo keyboard driver provides a system function mode that uses the touch screen to display a system function mode screen. The system function mode screen contains a touch-driven keypad. The keypad contains the S1 key, S2 key, and the numeric keys 0 through 9. To enter normal operation:

1. Press **S1**. This key precedes all system function requests.
2. Using the numeric keypad, enter the numeric code for the specific system function you want to use.
3. Press **S2**. This key enters your system request information and ends the system function.

If the display demonstrates a location error (that is, the mouse pointer location does not match the point of the display being touched), the display might need to be calibrated.

In Classic Mode, the calibration utility that is valid for USB touch screens only, improves the accuracy of the display and is invoked using a system function (S1, 55, S2). The terminal must have Java loaded on it for the calibration utility to work correctly. Once the calibration utility starts, follow the instructions to calibrate the display.

In Enhanced Mode in order to calibrate the touch screen, the display cable of the touch screen must be plugged into the primary video port, and the USB touch cable must be plugged into a powered USB port. This configuration is required regardless of whether the touch screen being calibrated will be used as the primary or the secondary display. Invoke the (S1, 55, S2) system function to calibrate the display as the primary touch screen and invoke the (S1, 56, S2) system function to calibrate the display as the secondary touch screen. Once the calibration utility starts, follow the instructions to calibrate the display. When the calibration utility exits, the system will be rebooted. If a touch screen is later moved to a different system, the calibration utility will need to be run again for that touch screen on the new system. If a touch screen is moved from the primary to the secondary port, or from the secondary to the primary port on the same system, the calibration utility will also need to be run again for that touch screen.

Notes:

1. In order to calibrate RS-485 touch screens, a utility diskette is needed.
2. USB infrared touch screen displays do not require calibration. This includes such models as the 4820-2xx or 4820-5xx.

Using uninterruptible power supply

Your terminal can be equipped with an uninterruptible power supply (UPS), which enables you to continue using your terminal during brief power outages. If your terminal has an UPS that is configured for control by the operating system, messages are displayed when power outages occur and when power is restored. These messages are used to determine whether you should continue to use the terminal or shut it down in an orderly manner.

The length of time that your terminal can operate on the battery power of the UPS is typically fifteen minutes. (See the specifications of your UPS to determine the exact time for your UPS.) However, your terminal can be configured to turn off the UPS after a certain interval when the power remains off. Your UPS is turned off if its battery runs low. Because the time the UPS can run the terminal during a power failure is limited, you should bring terminal operations to an orderly stop if a power failure lasts more than several minutes.

Note: External UPS devices are not supported on SurePOS 700 Series terminals with the Battery Backup feature installed. The battery backup must be removed before an external UPS can work with the SurePOS 700 Series system.

Terminal screen saver

The terminal screen saver avoids burn in of terminal video displays by taking over the video screen when a period of video inactivity has elapsed. Video inactivity is defined as no updates (writes) to the video display.

The terminal screen saver saves the video display by clearing the screen then writing the terminal screen saver message at an incremental location every 1/2 second. The terminal screen saver message continues to wrap around the video screen until the terminal screen saver is stopped.

Any input from the keyboard or touch display stops the terminal screen saver, restores the video display, and restarts the video inactivity period. This input is not passed along to the application. Any write to the video display also stops the terminal screen saver.

The terminal screen saver message, the attribute for the message, and the length of the video inactivity period are configured by using the Terminal Screen Saver Data option from the TERMINAL CONFIGURATION menu. The terminal screen saver screen uses the video display format configured for your terminal. To enable or disable the terminal screen saver for individual terminals, use the Terminal Device Group for your terminal. The terminal screen saver is enabled by default. It saves the video display with the message "TOSHIBA" in white letters on a blue background (white letters on grey background on monochrome video displays) after a video inactivity period of 15 minutes.

Notes:

1. This function is provided for VGA-attached video displays. It is not provided for 4683-421 terminals.
2. The terminal screen saver is ineffective when used with any application that constantly updates the screen (for example, the Supermarket Application Full Screen [Scrolling Receipt] feature, which constantly updates the status portion of the screen with the time and date).

Displaying messages

Each point-of-sale keyboard has indicators that provide status information about the system. When one of these indicator lights comes on, your system is sending a message to your terminal. The message sent to your terminal can be a Wait, Offline, or System message.

Figure 15 on page 115 shows the status indicators on the point-of-sale keyboards.

When more than one display is configured for the Point-of-Sale Terminal, all system messages are shown on the display that is configured as the system display.

Displaying wait messages

When the Wait indicator light comes on, the application running on the terminal is waiting for some action to end (for example, waiting for a program to load). The keyboard is locked, and only system function requests are accepted. A message related to the wait condition can be displayed.

If you display the message when the Wait light is on, the message indicates the reason for the current wait condition.

If you display the message when the Wait light is off, the message indicates the reason for the last wait condition.

To display a Wait message, type **S1, 1**, and **S2**.

The Wait message appears at the terminal system display. For example:

```
W321 PROCESSING...  
WAIT FOR PROMPT
```

Displaying offline messages

When the Offline indicator light on your keyboard comes on, normal system communications have been interrupted, and your terminal is *offline* (not communicating with the store controller). Use a system function required to display a message related to the offline condition. Terminal offline does not apply to the terminal portion of a controller/terminal.

If you display the message when the Offline light is on, the message indicates the reason for the current offline condition.

If you display the message when the Offline light is off, the message indicates the reason for the last offline condition.

To display an Offline message, type **S1, 2**, and **S2**.

The Offline message appears at terminal display. For example:

```
W004 CONTROLLER DOES  
NOT RESPOND
```

Displaying system messages

Your terminal receives numerous messages from the system, prompting messages, information messages, or status messages. Your system keeps the last five messages in the order of the most current to the least current. The least current message is deleted if the most current message increases the number beyond five. These messages can be displayed whether the Message Pending light is on or off.

When the Message Pending light is on

When the Message Pending light comes on, there is a message waiting that has not been seen. The Message Pending light stays on until the message is displayed. If you display the message when the Message Pending light is on, the most current message is displayed. If the Message Pending light stays on after the message has been viewed, another message is waiting to be displayed. The new message can be displayed by repeating the key sequence.

When the Message Pending light is off

If the Message Pending light is off when you display a message, the message shown is the next message after the last one viewed. You can continue displaying messages by repeating the key sequence. When the end of the messages is reached, the following message appears:

```
W204 NO SYSTEM  
MESSAGE AVAILABLE
```

If you continue displaying messages after the W204 message is displayed, the next message shown is the most current message.

To display a system message, type **S1**, **3**, and **S2**.

The system message appears at your terminal display. For example:

```
W008 PROGRAM IS  
LOADING...
```

Clearing the message display

To clear a message from the display, type **S1** and **S2**.

The message no longer appears.

Setting the date and time on an offline terminal

The date and time are usually set on an offline terminal under the following conditions:

- When the terminal previously lost communications with the store controller (terminal is offline), and you are returning to normal communications (bringing terminal back online)
- When communications cannot be established with the store controller

Terminal offline does not apply to the terminal portion of a controller/terminal.

Displaying the date format

When you reset the date, use the date format that is specified for your store during configuration. If an invalid date format is used, a message is received at the terminal system display. To ensure that you are using the valid date format for your store, type **S1**, **5**, **0**, and **S2**.

The correct date format displays in message W009 in one of the following forms:

- **DATE FORMAT:MMDDYY**
- **DATE FORMAT:DDMMYY**

Where:

MM Is the month.
DD Is the day.
YY Is the year (last two digits).

When you set the date, enter the date in the date format that is displayed for your store.

Setting the date on an offline terminal

1. Display the date format that is specified for your store during configuration (type **S1**, **5**, **0**, and **S2**).

2. To set the date, type **S1, 5, 1**, the date, and **S2**.

The date is set to the date you entered. If an invalid date format is used, a message appears at your terminal system display.

Setting the time on an offline terminal

Enter the time in the following format:

HHMM

Where:

HH Is the hour (00 through 23)

MM Is the minute (00 through 59)

To set the time, type **S1, 5, 2**, the time, and **S2**.

The time is set to the time you entered. If an incorrect time format is used, a message appears at your terminal system display.

Changing the volume of the keyboard tone

To change the volume of the tone on your terminal keyboard, type **S1, 6**, and **S2**. A high tone is changed to low, and a low tone is changed to high. Your terminal sounds the old tone volume once, and then sounds the new tone volume twice.

On the 4693 and USB keyboards (POS keyboard, ANPOS keyboard, and Modifiable Layout keyboard), press **Ctrl+5** (on the numeric keypad) to change the tone volume. Your terminal sounds the new tone volume once.

Note: On keyboards that have an integrated display, the tone volume cannot be changed.

Displaying the terminal number

To display your terminal number, type **S1, 7**, and **S2**.

The terminal number appears as follows:

```
W012 TERMINAL  
NUMBER = xxx
```

Where:

xxx Is a 3-digit terminal number from 001 to 999.

Enabling the storage retention function

Terminals have a storage retention function. Storage retention is not available for the terminal portion of a controller/terminal and is only available on certain 4694 models. This function can be enabled or disabled on an individual terminal (as described in this section) or on all terminals at once. See “Enable terminal storage retention” on page 96 for information on enabling all terminals at the same time.

To enable the storage retention function on a single terminal, type **S1, 7, 2**, and **S2**.

The storage retention function is enabled. To verify that the storage retention function is enabled, display the DISPLAY TERMINAL STATUS menu for your terminal (see “Display terminal status” on page 96 for information on how to display this menu).

Note: Enabling storage retention for a single terminal is only effective until the terminal is IPLed. The store setting are picked up upon completion of the IPL.

Disabling the storage retention function

This system function enables you to disable the storage retention function on a single 4683 or 4693 Mod1 terminal. Storage retention is not available on the terminal portion of a controller/terminal. See “Disable terminal storage retention” on page 97 for information on disabling all terminals at the same time.

To disable the storage retention function on a single terminal, type **S1, 7, 3, and S2**. The storage retention function is disabled.

To verify that the storage retention function is disabled, display the DISPLAY TERMINAL STATUS panel for your terminal (see “Display terminal status” on page 96 for information on how to display this panel).

Note: Disabling storage retention for a single terminal is only effective until the terminal is IPLed. The store setting are picked up upon completion of the IPL.

Displaying the available memory

To display the total amount of available memory in the terminal and the size of the largest contiguous block of free memory, type **S1, 9, 6, and S2**. For terminals with free or contiguous memory greater than 99 MB, the available memory is displayed in kilobytes (KB).

Dumping terminal storage

This system function allows you to dump terminal storage to a file on the controller for use in problem analysis. To dump terminal storage to a file, perform the following steps:

1. Type **S1, 9, 8, 9, 8, S2**.
2. Press **Enter**. U008 is displayed while the contents of terminal storage is dumped to the controller. Terminal storage is automatically reloaded when the dump process is completed.

Swapping terminal video screens

If a terminal has Java enabled, this function allows you to toggle between displaying the terminal Java video screen and displaying the terminal application/system-display video screen.

To swap, or toggle, to the screen which is not currently displayed, type **S1 5 S2**.

Switching to the enhanced mode graphical extensions screen

To switch to the enhanced mode graphical extensions screen, type **S1 5 8 S2**. This sequence has the same effect as typing **Alt+SysRq+X** on an ANPOS or standard keyboard. This function is not available on Classic systems.

Using the Set Terminal Characteristics program

This section explains how to use the Set Terminal Characteristics (STC) program. The Set Terminal Characteristics (STC) program enables you to perform the following functions:

- Enter a terminal number
- Display terminal configuration data
- Change a terminal number
- Change a terminal configuration
- Reset a terminal number to zero
- Format a terminal hard disk drive
- Exit STC function

Note: The STC program cannot be used on the terminal portion of a controller/terminal.

When there is more than one display configured for a terminal, the STC program displays information at your system display. See the *4690 OS: Communications Programming Reference* for a description of how to run the STC program using the SYSTEM CONTROL menu or how to start it remotely from the host. You can also use a key sequence at your terminal.

Note: Configuration changes can not be made to a Mod2 terminal unless the configuration changes are made to the Mod1 terminal first.

Requesting the STC program from the terminal keyboard

To request the STC program, type **S1, 7, 1, and S2**.

A message tells you that the program is loading. When the program is loaded, you are prompted for one of the following:

- If your terminal has a terminal number assigned, you are prompted to enter a request. This message appears:

```
Z002  ENTER REQUEST,  
KEY S2
```

Note: To end the function at this point, type **3, 3, 3, 3, S2**. The STC function ends and all configured applications for your terminal are reloaded.

The following sections explain how to use STC to perform each of the functions.

Entering terminal numbers

The STC program is automatically selected by the terminal's power-on sequence for a terminal that does not currently have a terminal number (for example, a newly installed terminal). The terminal number is used to uniquely identify each terminal. It is also used to locate the terminal's configuration information.

Notes:

1. If the terminal number that is entered already exists or has been created in the active Legacy Terminal Configuration, the terminal uses the Legacy

Configuration path. The terminal might display a series of messages about I/O devices not configured but attached, and configured devices not attached. Press **S2** to proceed with the terminal reload.

2. If the terminal number that is entered already exists or has been created in the active Generic Terminal Configuration, the terminal uses the Generic Terminal Configuration path. The terminal displays the Configuration Complete message for a few moments and then proceeds with the terminal reload.
3. If the terminal number that is entered does not exist in the active Legacy Terminal Configuration or the active Generic Terminal Configuration, the terminal uses the store default Generic Terminal Configuration. The terminal displays the Configuration Complete message for a few moments and then proceeds with the terminal reload.

Choosing the default terminal system display

When a Mod1 terminal is loaded with more than one display attached and without an assigned terminal number, the system chooses one of the displays as the default system.

The default system display will be the first display that is detected. For terminals configured using Generic Terminal Configuration, the system display will be the first of the following which is attached:

- RS-485 POS display with the lowest device ID. (Device IDs are described in the *4690 Operating System Messages Guide*, Appendix C. Reference Information.)
- USB display plugged into the port which is enumerated before any other USB port with a display plugged into it.
- Video display

Using generic terminal configuration

Each terminal (4694, SurePOS 300/700 Series or TCxWave 6140 Series) must have a terminal number.

Note: For this procedure, use the numeric keypad section of your keyboard unless instructed otherwise.

To enter a terminal number:

1. Power On any devices attached to the terminal that have their own power switches, and then power On the terminal. Your terminal begins a power-on sequence.
2. Observe your terminal display while your terminal goes through its power-on sequence. As your terminal goes through this sequence, a series of messages appears on your terminal display. See the *4690 OS: Messages Guide*.
At the end of the power-on sequence, the STC program is automatically loaded. After the program loads, your terminal displays:

Z001

Notes:

- a. To end the function at this point, type **3, 3, 3, 3, S2**. Your display goes blank and the STC program ends.
- b. If a terminal number is zeroed, a default terminal number (12285) is still used for TCC communications during the configuration process.
3. Enter the appropriate terminal number (1xxx, for example) and press **S2**.

Where:

- 1 Is a keypad identifier.

xxx Is a 3-digit terminal address from 001 to 999.

Note: Write down the number you have selected for this terminal. This number is needed to perform the steps that follow later.

If a mistake is made when entering the number, press **S1** to clear the display, and then reenter the correct number.

If a terminal number is entered that is not valid, an error message appears, and you are prompted to reenter the number.

When a valid terminal number is entered, this message appears:

Z012 CONFIGURATION COMPLETE

The terminal begins its power-on sequence.

Using legacy terminal configuration

Each terminal (4683, 4693, or 4694 Mod1) must have a terminal number.

Note: For this procedure, use the numeric keypad section of your keyboard unless instructed otherwise.

To enter a terminal number:

1. Power On any devices attached to the terminal that have their own power switches, and then power On the terminal. The terminal begins a power-on sequence.
2. Observe the terminal display while the terminal goes through its power-on sequence. As the terminal goes through this sequence, a series of messages appears on the terminal display. See the *4690 OS: Messages Guide* for information about these messages.

At the end of the power-on sequence, the STC program is automatically loaded. After the program loads, the terminal displays:

Z001

Notes:

- a. To end the function at this point, type **3, 3, 3, 3, S2**. The display goes blank, and the STC program ends.
 - b. If a terminal number is zeroed, a default terminal number (12285) is still used for TCC communications during the configuration process.
3. Enter the appropriate terminal number (1xxx, for example) and press **S2**.

Where:

1 Is a keypad identifier.

xxx Is a 3-digit terminal address from 001 to 999.

Note: Write down the number chosen for this terminal. This number is needed to perform steps that follow later.

If a mistake is made when entering the number, press **S1** to clear the display, and then reenter the correct number.

If an invalid terminal number is entered, an error message appears, and you are prompted to reenter the number.

When a valid terminal number is entered, this message appears:

Z010 TO VIEW CONFIG
KEY S2, ELSE KEY S1

4. If you do not want to view the configuration data, press **S1** to bypass the data and continue with the next step.

If you want to view the configuration data, press **S2**. A series of messages appears on your display giving you data about your terminal devices. (For information about these messages, see the *4690 OS: Messages Guide*.) Press **S2** each time you want to see the next message in the series. At the end of the series, this message appears:

```
Z012  CONFIGURATION COMPLETE  
KEY S2
```

5. If your configuration is correct, press **S2**. Your terminal now performs one of the following procedures:
 - If initial diagnostics have never been run on your terminal or nonvolatile random access memory (NVRAM) has been cleared, the Diagnostic Monitor program is automatically loaded and runs a series of tests on each device that is configured for and is attached to the terminal. When the program completes, the initial application specified for your terminal during configuration is automatically loaded.
 - If initial diagnostics have already been run on the terminal, the initial application specified for your terminal during configuration is automatically loaded.

On the Mod2 terminal

Note: You cannot enter a terminal number for the Mod2 until you have entered a terminal number for the attached 4683 or 4693 Mod1 terminal and it has completed the STC sequence.

Each terminal must have a terminal number. To enter a terminal number:

Note: For this procedure, use the numeric keypad section of your keyboard unless otherwise instructed.

1. Power On the Mod2 after you have successfully completed the procedure for entering a terminal number on the Mod1 terminal. The Mod2 terminal prompts you for a terminal number.
2. Enter the appropriate terminal number (1xxx, for example) and press **S2**.

Where:

1 Is a keypad identifier.

xxx Is a 3-digit terminal address from 001 to 999.

Note: Write down the number chosen for this terminal. This number is needed to perform the steps that follow later.

If a mistake is made when entering the number, press **S1** to clear the display, and then reenter the correct number.

If an invalid terminal number is entered, an error message appears, and you are prompted to reenter the number.

When a valid terminal number is entered, this message appears:

```
Z010  TO VIEW CONFIG  
KEY S2, ELSE KEY S1
```

3. If you do not want to view the configuration data, press **S1** to bypass the data and continue with the next step.

If you want to view the configuration data, press **S2**. A series of messages appears on your display giving you data about your terminal devices. (For information about these messages, see the *4690 OS: Messages Guide*.) Press **S2** each time you want to see the next message in the series. At the end of the series, this message appears:

Z012 CONFIGURATION COMPLETE
KEY S2

4. If your configuration is correct, press **S2**. Your terminal now performs one of the following procedures:
 - If initial diagnostics have never been run on your terminal or NVRAM has been cleared, the Diagnostic Monitor program is automatically loaded and runs a series of tests on each device that is configured for and attached to your terminal. When the program completes, the initial application specified for your terminal during configuration is automatically loaded.
 - If initial diagnostics have already been run on the terminal, the initial application specified for your terminal during configuration is automatically loaded.

Displaying terminal configuration data

Note: This STC function is not available on SurePOS 300/700 Series or TCxWave 6140 Series systems.

The STC program lets you display data about your terminal configuration.

1. Request the STC program. See “Requesting the STC program from the terminal keyboard” on page 122.
2. Enter your current terminal number in this format:
1xxx
Where:
1 Is the keypad identifier.
xxx Is a 3-digit terminal number from 001 to 999.
3. Press **S2**. This message appears:
Z010 TO VIEW CONFIG
KEY S2 ELSE S1
4. To view the configuration data, press **S2**. A series of messages appears on your screen giving you data about the devices that are configured for and are attached to the terminal. (For information about these messages, see the *4690 OS: Messages Guide*.) Press **S2** each time you want to see the next message in the series. At the end of the series, a message tells you that configuration is complete. Press **S2** to exit the STC program and reload the initial application that is specified for your terminal.
5. If you do not want to view the configuration data, press **S1**. The data is bypassed, and a message tells you that configuration is complete. Press **S2** to exit the STC program and reload the initial application specified for your terminal.

Changing a terminal number

Use this procedure to change a terminal number. Terminal numbers might need to be changed when terminals are moved to different locations in a store.

At the SurePOS 300/700 Series and TCxWave 6140 Series

1. Request the STC program. See “Requesting the STC program from the terminal keyboard” on page 122.
2. Enter the new number to be assigned to your terminal in this format:
1xxx
Where:
1 Is the keypad identifier.

xxx Is a 3-digit terminal number from 001 to 999.

3. Press **S2**. This message appears:

Z012 CONFIGURATION COMPLETE

The terminal begins its power-on sequence.

At the 4683, 4693, or 4694 Mod1 terminal

1. Request the STC program. See "Requesting the STC program from the terminal keyboard" on page 122.
2. Enter the new number to be assigned to your terminal in this format:

1xxx

Where:

1 Is the keypad identifier.

xxx Is a 3-digit terminal number from 001 to 999.

3. Press **S2**. This message appears:

Z010 TO VIEW CONFIG
KEY S2 ELSE S1

4. To view the configuration data, press **S2**. A series of messages appears on your display giving you data about your terminal devices. (For information about these messages, see the *4690 OS: Messages Guide*.) Press **S2** each time you want to see the next message in the series. At the end of the series, a message tells you that configuration is complete. See Step 5. If you entered an incorrect terminal number, press **S1**, and return to Step 2.

If you do not want to view the configuration data, press **S1**. The configuration data is bypassed, and a message tells you that configuration is complete. Press **S2** and your terminal begins its power-on sequence.

5. Observe your terminal display while your terminal goes through the phases of the power-on sequence. During this sequence, a series of messages appears on your display. See the *4690 OS: Messages Guide* for information about these messages.

At the end of the power-on sequence, the new terminal number is assigned, and the initial application is loaded.

At the Mod2 terminal

Note: You cannot enter a terminal number for the Mod2 until you have entered a terminal number for the attached 4683 or 4693 Mod1 terminal, and it has completed the STC sequence.

1. Power On the Mod2 after you have successfully completed the procedure for entering a terminal number on the Mod1 terminal. The Mod2 terminal prompts you for a terminal number.
2. Enter the new number to be assigned to your terminal in this format:

1xxx

Where:

1 Is the keypad identifier.

xxx Is a 3-digit terminal number from 001 to 999.

3. Press **S2**. This message appears:

Z010 TO VIEW CONFIG
KEY S2 ELSE S1

4. To view the configuration data, press **S2**. A series of messages appears on your display giving you data about your terminal devices. (For information about these messages, see the *4690 OS: Messages Guide*.) Press **S2** each time you

want to see the next message in the series. At the end of the series, a message tells you that configuration is complete. See Step 5 on page 127. If you entered an incorrect terminal number, press **S1**, and reenter the terminal number.

If you do not want to view the configuration data, press **S1**. The configuration data is bypassed, and a message tells you that configuration is complete. Press **S2** and your initial application specified for your terminal during configuration is automatically loaded.

Resetting a terminal number to zero

Use this procedure on a terminal that you want to remove from the TCC Network.

1. Request the STC program. See “Requesting the STC program from the terminal keyboard” on page 122.
2. Enter the following terminal number:
1000
This resets your terminal number to 000, which is not a valid terminal number but is used as a temporary place holder.
3. Press **S2**.
4. If you are using this function on a SurePOS 300/700 Series or TCxWave 6140 Series system, this message appears:
Z012 CONFIGURATION COMPLETE
5. If you are using this function at a Mod1 terminal, the display goes blank, and the terminal begins its power-on sequence. During this sequence, a series of messages appears at your display. See the *4690 OS: Messages Guide* for information about these messages.
6. When the messages begin appearing on your display, power Off the terminal. The terminal number has now been reset to 000.

Note: At this time, your terminal has no valid terminal number assigned to it. You can reassign a terminal number when you reconnect the terminal to the TCC Network, or after you have performed hardware maintenance.

If you want to perform hardware maintenance, do so at this point. After you complete your hardware work, reenter the terminal number. See “Entering terminal numbers” on page 122 and enter the same number you wrote down when you performed that step.

7. If you are resetting a terminal number to zero at a Mod2 terminal, a message tells you to power Off the terminal after entering 1000. Power Off the Mod2 terminal. When this Mod2 terminal is powered on again, it prompts you for a new terminal number.

Note: If a terminal number is zeroed, a default terminal number (12285) is still used for TCC communications during the configuration process.

Changing a terminal configuration

Note: This STC function is not available on SurePOS 300/700 Series TCxWave 6140 Series systems.

This procedure is used to change the configuration of a terminal, but keep the same terminal number.

Changing a terminal configuration is a two-part procedure. One part of the procedure is performed at the store controller; the second part is performed at your terminal.

To change a terminal configuration, make the changes to the terminal load records or device group records. This procedure is done at the store controller (see “Configuring legacy terminals (4683 and 4693)” on page 52 for configuration information). After the changes are made to the configuration record, the changes must be loaded into the terminal. This procedure is done at the terminals that are affected by the configuration change, and is described in this section.

When adding a device to a Point-of-Sale Terminal, install the device before running the STC program; when removing a device from the terminal, disconnect the device before running the STC program.

At the Mod1 terminal

1. Request the STC program. See “Requesting the STC program from the terminal keyboard” on page 122.

2. Type **2, 2, 2, 2** and press **S2**.

This message appears:

```
Z010 TO VIEW CONFIG  
KEY S2 ELSE S1
```

3. To view the configuration data, press **S2**. A series of messages appears on your display giving you data about your terminal devices. (For information about these messages, see the *4690 OS: Messages Guide*.) Press **S2** each time you want to see the next message in the series. At the end of the series, a message tells you that configuration is complete. Press **S2**. Your terminal begins a power-on sequence.

Note: If you do not want to view the configuration data, press **S1**. The configuration data is bypassed, and a message tells you that configuration is complete.

4. Press **S2** and your terminal begins its power-on sequence.
5. Observe your terminal display while your terminal goes through the power-on sequence. During this sequence, a series of messages appears at your display. See the *4690 OS: Messages Guide* for more information about these messages.

At the end of the power-on sequence, the new configuration data is in effect, and all configured applications for the terminal are loaded.

At the Mod2 terminal

1. Request the STC program. See “Requesting the STC program from the terminal keyboard” on page 122.
2. Type **2, 2, 2, 2** and press **S2**.
3. A message tells you to power Off the Mod2 terminal and perform the reconfiguration procedure at the Mod1 terminal.

Note: If you change your mind at this point and decide to keep the current configuration, press **S1** and the initial application for your terminal is reloaded.

Power Off the Mod2 terminal. Go to the Mod1 terminal and perform the reconfiguration.

4. When the procedure is complete, power On the Mod2 terminal again. The STC program is loaded automatically on the Mod2, and you are prompted to enter a terminal number.
5. Enter the terminal number for the Mod2 terminal in this format:
 1xxx
 Where:
 1 Is the keypad identifier.
 xxx Is a 3-digit terminal number from 001 to 999.
6. To view the configuration data, press **S2**. A series of messages appears on your display giving you data about your terminal devices. (For information about these messages, see the *4690 OS: Messages Guide*.) Press **S2** each time you want to see the next message in the series. At the end of the series, a message tells you that configuration is complete. Press **S2** and your initial application specified for your terminal during configuration is automatically loaded.

Note: If you do not want to view the configuration data, press **S1**. The configuration data is bypassed, and a message tells you that configuration is complete.

Formatting the terminal hard disk drive

These procedures are used to format the hard disk drive in the terminal. Before formatting the hard disk drive, ensure that the application on the Mod2 terminal is cancelled or stopped.

On the SurePOS 300/700 Series and TCxWave 6140 Series

To format the hard disk drive:

1. Request the STC program. See "Requesting the STC program from the terminal keyboard" on page 122.
2. Type **2, 2, 2, 2** and press **S2**.

This message appears:

```
Z041 TO FORMAT DISK
KEY S1 ELSE KEY S2
```

Note: If you are entering a new terminal number or changing an existing terminal number and your terminal has a hard disk drive attached, the option to format the hard disk drive is automatically displayed.

3. Press **S1**. The formatting message appears as follows:

```
Z042 FORMATTING
```

When the formatting is complete, the format complete message appears. For example:

```
Z044 FORMAT COMPLETE
KEY S2
```

4. Press **S2** and your terminal begins a power-on sequence.
5. Observe your terminal display while your terminal goes through the power-on sequence. During this sequence, a series of messages appears at your display. See the *4690 OS: Messages Guide* for more information about these messages.

If an error occurs during the format of the hard disk drive, the following message displays:

```
Z043 FORMAT FAILED
KEY S2
```

Check your message log for the reason for the failure.

If a power failure occurs during STC, run STC again.

On 4683, 4693, or 4694 Mod1 terminals

To format the hard disk drive:

1. Request the STC program. See “Requesting the STC program from the terminal keyboard” on page 122.
2. Type **2, 2, 2, 2** and press **S2**.

This message appears:

```
Z010 TO VIEW CONFIG  
KEY S2 ELSE S1
```

3. To view the configuration data, press **S2**. A series of messages appears on your display giving you data about your terminal devices. (For information about these messages, see the *4690 OS: Messages Guide*. Press **S2** each time until you see the following message.

```
Z041 TO FORMAT DISK  
KEY S1 ELSE KEY S2
```

Note: If you are entering a new terminal number or changing an existing terminal number and your terminal has a hard disk drive attached, the option to format the hard disk drive is automatically displayed during the view configuration option.

4. Press **S1**. The formatting message appears as follows:

```
Z042 FORMATTING
```

When the formatting is complete, the format complete message appears. For example:

```
Z044 FORMAT COMPLETE  
KEY S2
```

5. Press **S2**. Continue pressing **S2** until a message informs you that configuration is complete.
6. Press **S2** and your terminal begins a power-on sequence.
7. Observe your terminal display while your terminal goes through the power-on sequence. During this sequence, a series of messages appears at your display. See the *4690 OS: Messages Guide* for more information about these messages.

If an error occurs during the format of the hard disk drive, the following message displays:

```
Z043 FORMAT FAILED  
KEY S2
```

Check your message log for the reason for the failure.

If a power failure occurs during STC, run STC again.

Exiting an STC function

If you decide to end an STC function, type **3, 3, 3, 3, S2**. The STC function ends and all configured applications for your terminal are reloaded.

Other system functions

Your system has functions for debugging terminal applications and for running tests on terminal devices. For information on using the debug function, see the *4690 OS: Programming Guide*. For information on tests for diagnosing problems, see the *4690 OS: Messages Guide*.

Chapter 5. Working in Command Mode

This chapter explains how to begin working in the command-driven option, Command Mode. It explains how to enter commands, change a command line, use editing keys, and organize your data into files and directories.

Selecting Command Mode

The Command Mode option lets you enter commands that help you create directories, copy files, create or edit files, or use applications that are designed to run on your personal computer.

Note: You must be authorized to use this option. For information on authorization, see your application manuals.

To select the Command Mode option, type **7** and press **Enter** from the SYSTEM MAIN MENU.

Note: Additional messages might be displayed before the prompt appears, or the prompt might differ from the example that is shown here.

The system prompt and cursor appear, indicating that the system is ready to accept a command. Your prompt looks like this:

```
C:>_
```

The alphabetic character shown in the prompt lets you know which drive you are using. In this case, the character C indicates that you are using the first hard disk drive.

Exiting from Command Mode

To exit from Command Mode and return to the SYSTEM MAIN MENU, type **Exit** at the system prompt and press **Enter**. The system returns to the SYSTEM MAIN MENU.

Commands and command lines

When you want to tell your system to perform a task in Command Mode, you type a *command*. A command is information that requests the system to perform an action.

Each command represents a function that your system can perform. For example, there are commands that let you create directories, display the information you have in a file, or rename a file. The command name often gives a clue about the function the command performs. For example, to rename a file, you use the RENAME command. See Chapter 7 for descriptions of the commands you can use on this system.

Typing commands into the command line

The information you type at the prompt is called the *command line*. You can type the information in either lowercase or uppercase. The command line generally contains this information:

```
C:>command filespec -parameter
```

where:

command

A command name (such as the DIR command), an application program (such as a word processing program), or a file that can be processed (such as a batch file). The batch file commands are described in “Batch commands” on page 157.

filespec

Tells your system where to find a file. (See “Finding files” on page 137, for information on how the system finds files.)

parameter

Additional information that you enter into the command line to specify the result you want. The parameter is denoted by a hyphen (-). Parameters vary according to the command you are using, and not all commands require parameters. See Chapter 7 for information on the parameters that are used with specific commands.

The following example shows a command line without a parameter. The example shows how to display all the files you have in a directory.

```
C:>DIR
```

The next example shows a command line with a parameter. The example shows how to display the files (DIR command) in a large directory and pause after each screen of entries (-P parameter).

```
C:>DIR -P
```

Processing commands

When you are ready to process the command you typed, press the **Enter** key.

Logging command line activity

If the logical name ADXCMDLG is defined and the command shell is started from the OCF menu, then 4690 OS will log when the command shell is started and stopped. The activity will be logged to a file, not to the system log. Each command line session will have its own temporary log file during the session. At the first IPL after the session ends, the temporary log file will be appended onto the master log file ADXCLOG.DAT and the temporary log file will be deleted. The directory ADX_CLOG will contain the log files, which are all local files.

The temporary log files for the command line have file names with the format of UUUUUUUU.Cnn, where UUUUUUUU is the one to eight character user ID and nn is the next sequential value (in the range 00-99) that is available in the directory. You are responsible for controlling the maximum size of the master log file.

Note: User IDs can contain some special characters that are not valid in file names. If the user ID would not be a valid file name, then the log file will not be created.

The adxclogl.386 utility can coalesce each of the individual log files in the ADX_CLOG directory into a single target file. This utility is available to run from a command line or as a background application when started from RCP (Remote Command Processor). The utility is invoked as follows:

```
adxclogl.386 <-f outputFile> <-l listFile>
```

Where:

- -f outputFile is the name (can include the full path name) of the output file that will contain the combined log files. It is optional and defaults to C:\ADX_CLOG\ADXCLLOG.DAT. If the file exists, the log data found in the existing session files will be appended. If it does not exist, it will be created.
- -l listFile is a local file name (can include the full path name) containing a list of the session log file extension characters to combine. This is an optional parameter and defaults to the file name ADX_SDT1:ADXCLOGF.DAT. If the specified file does not exist, the utility will default to combining files with extensions known at this file; C, J, X, and F for extensions Cnn, Jnn, Xnn, and Fnn. This option will allow the user to combine specific session log file types, if needed. The format of the file is a single character (Upper Case) per line such as:
 - C
 - J
 - X
 - F

Message W512 will be logged with information about the user that runs the utility and the exit return code. No messages will be sent to the screen.

Editing keys

After you type a command line, you might want to change the line or cancel it. Your system has several keys called *editing* keys that let you change a command line before pressing Enter or let you cancel a command line after you press Enter.

Some of these keys are a combination of Ctrl and a letter. Press and hold **Ctrl** while striking the letter key.

Table 13 summarizes the editing keys and their functions.

Table 13. Editing key functions

Key	Function
Left arrow	Moves the cursor one character to the left. No characters are changed.
Right arrow	Moves the cursor one character to the right. No characters are changed.
Up arrow, Down arrow	Retrieves the last command entered.
Backspace	Moves the cursor one character to the left and deletes the character at that position. Characters to the right of the cursor move one position to the left.
Delete	Erases the character at the cursor. Characters to the right of the cursor move one position to the left.
Ctrl+X	Deletes the command line and puts the cursor at the beginning of the line.
Ctrl+R	Displays the last command line. You can view, change, or rerun the command line.
Ctrl+S	Suspends a running command and its output. Command and output remain suspended until you use Ctrl+Q to restart or Ctrl+C to cancel.
Ctrl+Q	Restarts a suspended command and its output.

Table 13. Editing key functions (continued)

Key	Function
Ctrl+C	Stops a running command. Ctrl+C can also be used to stop a running program, but results are unpredictable.
Ctrl+V	Displays a list of the last 10 commands from which you can make a selection.
Ctrl+F	Displays the FIND CMD-LINE>. As you start to type characters, the system finds and displays the last command matching the characters you enter.
@#	Runs the command line specified with its history buffer number, #. Use this number when specifying a command line to run.

Inserting characters

When you are working in Command Mode, the system is normally in insert mode. While in insert mode, you need not use a special key to insert characters in the command line. To insert characters, move the cursor backwards and type the character or characters you want. The character or characters you type are inserted at the cursor. All characters to the right of the cursor are shifted to the right.

Entering several commands on a command line

The exclamation mark (!) lets you enter more than one command on a command line. This feature is called *chaining* commands together. When you place an exclamation mark between each command on the line and press **Enter**, the commands run sequentially as if they were entered on different command lines.

Note: You should be careful when using the exclamation mark (!) as part of other commands because the system can misinterpret the command. For example, REM Hi! This is a comment will make the system think the REM statement ends at the ! character and will try to execute a command with the name "This."

The following command line displays the current directory of drive A, renames a file, and displays the contents of another file. Notice that the system displays the subsequent command lines it runs.

```
A:>DIR ! RENAME CURRENT.EVT HISTORY.OLD ! TYPE MESSAGE.TXT
```

```
Volume in Drive f0: has no label
```

```
Directory of f0:
```

```
JANUARY          <DIR>          1-06-1985   9:34a
APRIL            <DIR>          3-30-1985  11:03a
ACCOUNTS PAY      1429          1-18-1985   1:42p
ACCOUNTS REC      13421         1-03-1985   8:46a
```

```
4 Files      324256 bytes free
```

```
A:>RENAME CURRENT.EVT HISTORY.OLD
```

```
A:>TYPE MESSAGE.TXT
```

```
To All Employees:
```

```
Please turn in your monthly status
reports by noon on Friday.
```


Thank you.

A:>_

Organizing your information

This section tells you how to organize your information on disk or diskette by grouping related information into files and directories.

Files and file names on FAT systems

A *file* is a collection of related information that is stored on diskette or disk. A file can contain information about your store operations, customers, sales, or any subject you select. For example, a file can be a mailing list, a list of prices, or a program.

You create a file or make changes to an existing file by using the text editor that is discussed in Chapter 8, "Creating and editing files," on page 265.

Finding files

When you are ready to work on a file, you must tell your system where to find it. To help the system find the file you want, your command line must contain a file specification or filespec. This filespec, which includes a *file name* and *file extension* (optional), tells the system exactly where the file is stored.

The filespec acts much like the address on a letter; it shows a path from a general location to a specific one.

For example, if you want to display the contents of the SHOWME file, which has a file extension of DAT and is found on your current drive, your command line would look like this:

```
C:>TYPE SHOWME.DAT
```

If you want to display the contents of the SHOWME file, and the file is not on your current drive, but on drive A, your command line would look like this:

```
C:>TYPE A:SHOWME.DAT
```

The filespec also can be a logical name. For example, if you want to display the contents of the C:\ADX_SDT1\ADXCShSF.DAT text file, which has a system logical name ADXCShSF, your command line would look like this:

```
C:>TYPE ADXCShSF
```

Note: Logical file names can take precedence over actual file names and directories. For example, if you have an actual file C:\\$YFILE, and a logical file name \$YFILE = C:\ADX_IDT4\MYFILE.DAT, the command TYPE \$YFILE will display the contents of MYFILE.DAT. In this example, if you have an actual subdirectory C:\MYITEMS\ and a logical name MYITEMS = C:\ADX_IDT1, the command CD MYITEMS will change the current directory to ADX_IDT1. See the *4690 OS: Messages Guide* for more information on logical names.

Naming your files

A file name is a label you give to your file that helps locate it. Each file name that you assign must be a valid one.

A valid file name:

- Can be up to eight characters long
- Can include any capitalized alphabetic character (A - Z)

Note: When naming a file, the system will change the characters A - Z to capital letters.

- Can include any of the numbers (0 - 9)
- Can use these special characters in the second, third, sixth, or seventh character position of a file name for compatibility with operating system names:

() { } @ \$ &

- Do not use ADX in the first three character positions. The prefix ADX is reserved for operating system files.
- Should not use the hyphen (-) character to avoid confusion with the command line parameters
- Can use @ in the fifth character position but only in an input sequence table file.

Examples:

VALID	INVALID
SLSDATA	SLSDATAFILE (too many characters)
STOR405	STOR#405 (special character in wrong position)
A(CONTL	
RE35FILE	

The file name can also include a file type. It serves as a “last name” for groups of related files. For example, you might want all your BASIC programs to have the BAS file extension. A file extension is optional, but it helps you to quickly identify the type of information that is found in a file. If you specify a file extension, you must include it in the filespec.

If you use a file extension, use a period between the file name and the file type. For example:

SALSDATA.BAS
INTEREST.DAT

A file extension can be one to three characters and can use the same letters, numbers, and special characters that are valid for file names.

Note: See the *4690 OS: Programming Guide* for a more detailed list of file naming rules and guidelines.

Directories and subdirectories on a FAT system

Just as a file is a collection of information, a *directory* is a collection of file names.

When you first begin working with a disk or diskette, a single directory is automatically created. This is called a *root directory*. This directory contains the names of files and other directories. The root directory has no name, but is indicated by the first slash or backslash in your filespec (see “Specifying paths between subdirectories” on page 139 for information on using the slash or backslash character to separate subdirectories).

For example, in the filespec \REPORTS\SALES\ANNUAL.DAT, the first backslash denotes the root directory.

If you are working in your root directory, you do not need to include the backslash for the root directory in your filespec.

When your root directory contains many files, you can create other directories, called *subdirectories*, to keep your data organized and easy to find. Create subdirectories using the MKDIR command discussed in Chapter 7, “Command summary,” on page 153. Use the same rules to name your directories that you use to name your files, except do not add file extensions to your subdirectory names. See Chapter 25, “Using the Multiple Controller Feature,” on page 445 for restrictions in using distributed files in nested subdirectories. See Chapter 20, “Using the disk rebuild utility,” on page 409 for restrictions on using the Disk Rebuild Utility with nested subdirectories.

See the *4690 OS: Programming Guide* for a description of the subdirectories that are defined and supported by the operating system.

After you create subdirectories, you can make any directory you select your default or current directory. Whenever you enter a file name without specifying a directory, the system searches your current directory first. Before you create other directories, your root directory is your current directory.

Finding files in directories

When you create subdirectories within your root directory, you set up a structure similar to a “tree.” Figure 16 shows how information is organized on your disk when you create subdirectories.

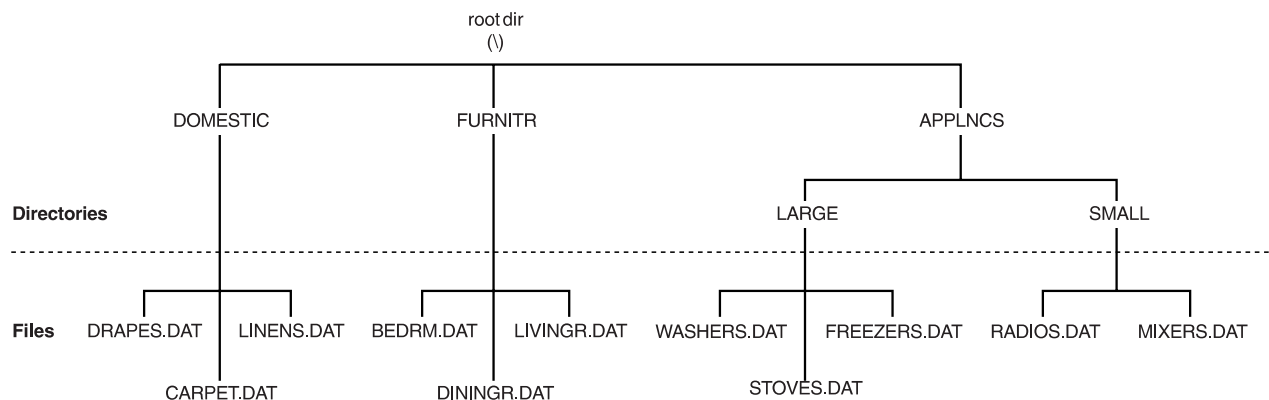


Figure 16. Directory Tree

Specifying paths between subdirectories

When you want to access a file, you must specify the *path* that leads to it. The path consists of directory names that are separated by a backslash character (\\) or a slash character (/).

Note: While your system accepts both the slash and the backslash character, it is probably less confusing to use one or the other. Examples in this guide use the backslash character.

For example, in Figure 16 on page 139, the path to file FREEZERS.DAT is:

```
C:>TYPE \\APPLNCS\\LARGE\\FREEZERS.DAT
```

If you are on one directory path and you want to access a file on a different path, you must specify the path leading from the root directory. For example, in Figure 16

on page 139, if you are in directory LARGE and you want to view file DRAPES.DAT in directory DOMESTIC, your path would look like this:

```
C:>TYPE \DOMESTIC\DRAPES.DAT
```

If you are specifying a path to a directory below your current directory, your path does not have to lead from the root directory. For example, if your path is from the root directory to directory APPLNCS, and you want to view file MIXERS.DAT, your path would look like this:

```
C:>TYPE APPLNCS\SMALL\MIXERS.DAT
```

Note: For additional information on subdirectories when using DDA, see Chapter 25, “Using the Multiple Controller Feature,” on page 445

Changing the current directory

If you want to change your current directory path, use the CHDIR command that is described in “CHDIR command” on page 171.

File systems supporting long file names

Some file systems in 4690 support file names greater than eight characters in length. The file name is a generic string, and the idea of file extension does not exist as an inherent part of the name. With these file systems, the maximum name length is 255 characters, and the maximum path length including directories is 260 characters. These file systems are VFS, NFS, Enhanced RAM Disk, Dynamic Mount (E:), Embedded Shared (F:), and (only in Enhanced Mode) CD/DVD (P:). Specified case is respected (file creation) on all these file systems. Specified case is enforced (file name matching) on all these file systems except (Enhanced) P:, where file name matching is case-independent as long as only one match exists.

Finding files

When you are ready to work on a file, you must tell your system where to find it. To help the system find the file you want, your command line must contain a file specification or filespec. This filespec, which includes a *file name* and *file extension* (optional), tells the system exactly where the file is stored.

The filespec acts much like the address on a letter; it shows a path from a general location to a specific one.

For example, if you want to display the contents of the STARTPAGE.HTML file that is located on your local M: drive, your command line would look like this:

```
M:>TYPE STARTPAGE.HTML
```

If you want to display the contents of the STARTPAGE.HTML file, and the file is not on your local M: drive, but on a remote system being accessed using NFS, your command line could look like this:

```
G:>TYPE STARTPAGE.HTML
```

The filespec also can be a logical name. For example, if you want to display the contents of the text file C:\ADX_SDT1\ADXCShSF.DAT, which has a system logical name ADXCShSF, your command line would look like this:

```
C:>TYPE ADXCShSF
```

Note: Logical file names can take precedence over actual file names and directories. For example, if you have an actual file C:\\$YFILE, and a logical file name \$YFILE = C:\ADX_IDT4\MYFILE.DAT, the command TYPE \$YFILE will display the contents of MYFILE.DAT. In this example, if you have an

actual subdirectory C:\MYITEMS\ and a logical name MYITEMS = C:\ADX_IDT1, the command CD MYITEMS will change the current directory to ADX_IDT1. See the *4690 OS: Programming Guide* for more information on logical names.

Naming your files

A file name is a label you give to your file that helps locate it. Each file name that you assign must be a valid one.

A valid file name:

- Can be up to 256 characters long
- Can include any capitalized alphabetic character (A - Z)
- Can include any of the numbers (0 - 9)
- Must not use ADX in the first three character positions
- Must not use :, /, \, *, ?, ", <, >, or | characters.
- Any character less than decimal 32 is not supported.
- Should not use either the decimal 127 or decimal 255 character.
- Should not use the hyphen (-) character to avoid confusion with the command line parameters
- A period (.) is treated just like any other character in the name. It does not indicate a file extension.

For example, the following are valid file names:

```
STARTPAGE.HTML  
HELLOWORLD.CLASS
```

Directories and subdirectories on a VFS system

Directories and subdirectories on a Virtual File System (VFS) work the same as on a FAT system with the following exceptions.

- The maximum path length, including directories and files, is 260 characters.
- The maximum directory depth is 60 levels including the root directory.

Store controller roles

If you are using a LAN system, you can also use files that are stored on other store controllers. In a LAN (MCF Network) system, each store controller has its own name. The name is formed by the system using the store controller ID you define at configuration. This name is called a *node name*. Each node name is different from all the others; the ID you defined becomes the unique part of the name. Node names help store controllers communicate with each other, keep files organized, and help you address a specific node on the system.

To specify that the file you want is on a different store controller, you must specify the node name as part of the filespec. For example, if you are connected to store controller CC and the file you want to copy is the PRICE.DAT file from the C: drive on store controller DD, type the following command:

```
C:>COPY ADXLXDDN::C:\PRODUCE\PRICE.DAT
```

Note: Specifying the node name as part of the filespec is not supported for the following drives: E:, F:, M:, P: and NFS drives. That is, ADXLXDDN::M: and ADXLXDDN::E: for example, are not supported in commands (where DD is the controller ID of the controller).

For more information on node names and store controllers, see “Node IDs” on page 450.

Using global file name characters

You can use three global file name characters for greater flexibility when you are working with commands:

- Asterisk (*)
- Question mark (?)
- Caret (^)

You can substitute a global file name character for other characters in file names and file extensions. When you type a command and use a global file name character in the file name or file extension, the system substitutes the characters that match the global file name character. The command you used is performed on all the files that match the global file name character in a specific directory.

The global file name character you select determines the number and position of the replaced characters. The following actions describe each global file name character.

The * character

When you use the asterisk character, the global file name character is replaced with any valid number of characters. If you use * in a file name, it is replaced with up to eight characters. If you use * in a file extension, it is replaced with up to three characters. For example, suppose that you have the following directory of files:

SALES.DAT	85JOBS.BAS
SALARY.DAT	TAXES.DAT
VOLUME.DAT	HOST.BAT
WEEKLY.BAS	SALSMEMO.BAS

If you want to copy all the files that have the *DAT* file extension to a diskette, type:

COPY *.DAT A:

These files are copied:

SALES.DAT
TAXES.DAT
VOLUME.DAT
SALARY.DAT

The ? character

This global file name character replaces only one character in the position that is shown. For example, suppose that you have the following list of files:

SALES1.DAT	SALESMON.DAT
SALES2.BAS	SALSREPT.DAT
SALES3.DAT	SALES.DAT

If you want to list all the files that begin with the six characters SALES and have a file extension of DAT, type this command:

DIR SALES?.DAT

Note: You can use multiple question marks to specify the exact number of characters for a file search.

These files are listed:

SALES1.DAT
SALES3.DAT

The ^ character

The ^ character tells the system to find all the files that do not match the global file name character. This character must be used as the first character in a file name.

Note: Use of the ^ character is not supported on drives supporting long file names.

For example, suppose that you have the following list of files:

SALSMEMO.BAS	VOLS84.DAT
SALSREPT.BAS	PRICES.DAT
YEAREND.DAT	SALPRICE.BAS

If you want to list all the files that do not begin with SAL, type:

```
DIR ^SAL*.*
```

These files are listed:

YEAREND.DAT
VOLS84.DAT
PRICES.DAT

Protecting your files

The operating system provides a way of protecting your files from unauthorized use. To ensure this protection, each user must have a *user ID* and a *group ID*. This ID is not the same as your operator ID, but it works basically the same way.

When you request a file or run a program, the system checks your user and group IDs to verify the type of authorization you have to use the file or program.

The operating system uses three kinds of authorizations that are called *classes*: *world*, *group*, and *owner*. Your store procedures define the file operations each class can perform.

Each class has a specific level of authorization that determines the type of functions you can perform on a file or program. For example, a user with owner class authorization can be authorized to read, write to, execute, or delete a file or program while a user with world class authorization can be authorized only to read the same file or program.

For more information on user and group IDs, see the *4690 OS: Programming Guide*.

Redirecting input and output

With the operating system, input is usually entered at the keyboard, and output is usually displayed on the screen. Input/output (I/O) redirection lets you use data from files or devices as input and lets you send output to a file, a device, or another command.

Methods for redirecting input and output

The following list shows four basic ways to redirect input and output. Each one consists of a command, followed by a redirection symbol, followed by a filespec or another command.

Method	Action
COMMAND > FILESPEC	Redirects the output from a command to a file or device
COMMAND >> FILESPEC	Adds the output from a command to the end of a file or device
COMMAND < FILESPEC	Uses a file or device as input to a command
COMMAND_1 COMMAND_2	Uses the output of COMMAND_1 as the input for COMMAND_2

If you add an asterisk to the redirection symbols >, >>, and |, the system redirects the standard error file of the command rather than the output of the command itself.

Method	Action
COMMAND >* FILESPEC	Redirects the command's standard error file to a file or device
COMMAND >>* FILESPEC	Adds the command's standard error file to the end of a file or device
COMMAND_1 * COMMAND_2	Uses the standard error file of COMMAND_1 as the input for COMMAND_2

You can combine two or more ways of redirecting I/O on the same command line.

You cannot use I/O redirection symbols in the filespec.

Redirecting output to a file or device

The > symbol directs the output from a command to a specified file or device. If the file does not exist, the operating system creates it. The following example sends the output from the DIR command to the PROGRAMS.LST file.

```
A:>DIR > PROGRAMS.LST
```

Adding output to the end of a file or device

The >> symbol takes the output from a command and adds it to the end of an existing file or device. The following example adds output from the DEFINE command to the end of the BACKLOG.DAT file.

```
DEFINE >> BACKLOG.DAT
```

Getting input from a file or device

The < symbol uses a file or device as the input to a command. In the following example, the MORE command displays information from the QUOTAS.LST file.

```
A:>MORE < QUOTAS.LST
```

This MORE command produces the following output:

```
SALES QUOTAS FOR FEBRUARY
-----
Barnes, B.      $ 5,000.00
Cummings, M.   $ 4,600.00
Fortt, M.      $ 6,000.00
```



```
Hinton, M.      $12,500.00
Norwood, E.    $ 3,000.00
```

```
-- More --
```

Piping commands

Using the output from one command as the input to another is called *piping*. When you use the `|` symbol between command names, the system uses the output from the first command as the input for the second command.

For example, assume that you have this directory of files on drive B:

```
A:>DIR B:
Volume in drive B: is PAPERWORK
Directory of B:

CLEANUP  BAT           25      6-06-1985   9:34a
MEMOS    25984          3-23-1985  11:03a
CATALOG  LST          17664    6-01-1985  10:13p
LETTERS  16256          4-16-1985   5:57p

    4 Files              90167 bytes free
A:>_
```

You can sort this directory alphabetically using the output from DIR as the input to the SORT filter. The sorted directory appears on the screen.

```
A:>DIR B: | SORT
    5 Files              90167 bytes free
Directory of B:
Volume in drive B: is PAPERWORK
%P00001  0           6-08-1985  10:18a
CATALOG  LST          17664    6-01-1985  10:13p
CLEANUP  BAT           25      6-06-1985   9:34a
LETTERS  16256          4-16-1985   5:57p
MEMOS    25984          3-23-1985  11:03a
A:>_
```

Notice that there is one new file in the sorted directory. This file is the temporary file used to pipe data between commands. The next time you use DIR, this file will have disappeared. When you pipe data between commands, do not run programs that erase or modify this file. Erasing this temporary file could cause your data to be incorrectly piped.

Note: The temporary file is not created if you pipe data between two or more commands, and all the commands are stored on disk rather than built into system.

NULL: device

The NULL: device can be used to discard messages that you are not interested in. You can use the following example to discard error messages.

```
COPY ADXNSSLG ADXNSSLG.BAK >* NULL:
```

Filtering data

Filters are commands that receive input, rearrange it, and send the resulting output to the screen, a file, device, or command.

Note: Filters alter the output that you see from your commands; they do not alter the contents of your files or affect the output of your commands. There are three filter commands: `FIND`, `MORE`, and `SORT`. Although you can use these filters with input from the keyboard, they are most often used to filter input from a file or command. These commands are discussed in detail in Chapter 7, “Command summary,” on page 153

Chapter 6. Using system commands

In this chapter, you learn how to use the commands that help you format your disks and diskettes, make copies of your diskettes, or display what is in a directory or a file. Other commands that help you manage your disks and diskettes are covered in more detail in Chapter 7, “Command summary,” on page 153.

Formatting diskettes

You must prepare diskettes and disks before you can use them the first time. The **FORMAT** command prepares disks and diskettes in a recording format that is acceptable for use on your system.

Attention: Any information that exists on your diskette is destroyed during formatting. If your diskette already has information on it, list its contents before formatting to make sure that you do not destroy any information you want to keep. List the contents of the diskette by using the **DIR** command that is described in “Listing the files on your diskette” on page 149

To format a diskette:

1. Make sure that you have the system prompt indicating that the system is ready to accept commands.
2. Type the **FORMAT** command followed by the letter of the drive you are using (the examples show drive A as the drive in use). Press **Enter**.

A:>FORMAT A:

Your system responds with the C: prompt.

3. Insert the diskette you want formatted into drive A. (If your system has two diskette drives, you can use A or B.)
4. Type **Y** (for Yes). The following message appears:

```
Formatting Drive A:
1,440K (18 sector, 80 track, double-sided, 3.5")
      HEAD 0      TRACK 0
```

When the diskette completes formatting, your system displays this message:

```
Formatted drive A:
2880 total sectors at 512 bytes/sector, 1 sectors/cluster.
1474560 bytes total disk space.
   512 bytes in boot area.
  9216 bytes in 2 FATs.
  7168 bytes in 224 root directory entries.
-----
1457664 bytes available on disk.
Format another floppy diskette (Y/N)?
```

The information on your panel tells you in bytes how much disk space and available space your diskette contains. The number of bytes that are shown on your panel will vary according to the number of tracks and the size of the sectors on the diskette. For more information on the **FORMAT** command, see “**FORMAT** Command” on page 214.

5. Remove the formatted diskette from the drive. If you want to format another diskette, type **Y**; if you do not, type **N**.

Formatting your hard disk drive

Like your diskettes, your hard disk drive must also be prepared before it can be used for the first time. To prepare your hard disk drive, see “Preparing Your Hard Disk Drive Using the Supplementals” on page 528.

Making copies of diskettes

Use the DISKCOPY command to copy all of the files from one diskette to another diskette. This command is for diskettes only. If you try to use it on a hard disk drive, you receive an error message.

This command prompts you to insert a source diskette and a destination diskette. The *source* diskette is the diskette you are copying from; the *destination* diskette is the diskette you are copying to.

If the destination diskette is not formatted like the source diskette, the DISKCOPY command formats it while it is copying the diskette.

Attention: Use the DISKCOPY command carefully. It writes over any information already on the destination diskette.

Copying diskettes on a one-drive system

1. Make sure that you have the system prompt indicating your system is ready to accept commands.
2. Type the DISKCOPY command:
A:>DISKCOPY A: A:
3. Press **Enter** and your system responds:
Insert source diskette into drive f0:
Press any key when you are ready
4. Insert the source diskette into the drive and press any key.
Your system responds:
Copying XX tracks, YY sectors per track, 2 side(s).
Insert destination diskette in drive A:
Press any key when you are ready
5. Remove the source diskette and insert the destination diskette. Press any key.
Your system responds:
Copying XX sectors per track, 2 sides
Depending on the size of your system memory, you might be prompted to insert the diskettes several times. When the diskette is copied, you are prompted:
Do you want to copy another diskette (Y/N)?
6. Type **Y** if you want to copy more diskettes; type **N** if you do not.

Note: You can make multiple copies of a diskette using the DISKCOPY command. The system prompts you to insert the source and destination diskettes. The system also prompts you to make more than one copy of the source diskette. See “DISKCOPY command” on page 200 for instructions on using the DISKCOPY command.

Copying diskettes on a two-drive system

1. Make sure that you have the system prompt that indicates that your system is ready to accept commands.
2. Type the DISKCOPY command, the letter of the drive that contains the source diskette, and the letter of the drive that contains the destination diskette:
`A:>DISKCOPY A: B:`
3. Press **Enter** and your system responds:
Insert source diskette into drive A:

Insert destination diskette into drive B:

Press any key when you are ready
4. Insert the diskettes as requested into drives A and B. Press any key.
Your system responds:
Copying XX tracks, YY sectors per track, 2 side(s).

Press any key when you are ready
When the diskette has been copied, you are prompted:
Do you want to copy another diskette (Y/N)?
5. Remove the diskettes. If you have other diskettes to copy, type **Y**; if you do not, type **N**.

Displaying what is in a file

When you want to look at the contents of a specified file, use the TYPE command. For example, to view the contents of the SALES.DAT file:

1. Insert the diskette that contains the file into drive A.
2. Enter this command:
`A:>TYPE SALES.DAT`
The contents of the SALES.DAT file appear on your panel:
The monthly sales meeting
will be held on October 28, 1987.

If you want to print the contents of the file, use the PRINT command that is described in Chapter 7

Listing the files on your diskette

Sometimes you might want to list all the files you have in your current directory or a directory you specify. The DIR command also gives you information about the files in your directories.

To list all the files in your current directory:

1. Insert the diskette containing the directory you want to display.
2. Type this command preceded by the letter for the diskette drive, and press **Enter**:
`:>DIR`

DIR displays information about the files in your current directory. It first displays diskette information such as volume identification and the name of the directory being listed. It then displays file information such as file names, file extensions, file

size in decimal bytes, and the date and time the file was created or last changed. It also tells you that the amount of free space left on your disk.

Listing files on a specific drive

If you want to see the files you have on a specific drive, type the DIR command followed by the letter of the drive you want:

```
A:>DIR B:
```

DIR lists the files in the current directory of drive B.

Listing files on another store controller

If you want to see the files you have on another store controller, type the node name followed by the letter of the drive you want:

```
C:>DIR ADXLXDZN::C:
```

In this example, *DZ* is the store controller ID assigned to the other store controller.

Any time you want to perform functions on store controllers other than the one you are using, specify the node name as the first part of the file specification.

Note: The NetBIOS protocol is not supported for the following drives: E:, F:, M:, P: and NFS drives. That is, *ADXLXDDN::M:* and *ADXLXDDN::E:* for example, are not supported in commands (where *DD* is the controller ID of the controller).

Renaming a file

Occasionally, you might want to change the name of a file or its file extension. The RENAME command lets you change a file without moving the file or changing any of the information in it.

To rename a file:

1. Insert the diskette containing the file you want renamed into drive A.
2. Type the RENAME command, the current filespec, and the new filespec.

```
A:>RENAME SALES.DAT REPORTS.DAT
      (old filespec) (newfilespec)
```

3. Press **Enter**.

When your filespec is renamed, the system prompt appears.

You can use the DIR command to check the new filespec.

Erasing a file

When you want to erase a file, use the ERASE command to delete it from your diskette.

Attention: Use the ERASE command carefully. Once you use this command on a file, the file is gone and cannot be recovered.

To erase a file on a diskette:

1. Insert the diskette containing the file to be erased into drive A.
2. Type the command, and then type the filespec of the file to be erased:

```
A:>ERASE SALES.DAT
```

3. Press **Enter**.

When your file is erased, the system prompt appears. You can use the DIR command to see that the file has been erased.

Chapter 7. Command summary

This chapter describes each command and gives examples that show command formats and parameters. The information in this chapter is applicable only in Command Mode.

The commands in this chapter describe how to do the following:

- Copy, view, and rename files
- Format your disk media
- Back up and restore files on your hard disk drive
- Create, delete, and search directories

Types of commands

Your system has two types of commands: internal and external. *Internal* commands are built into the system and can be used at any time. *External* commands are stored on disk and must be read from disk before they can be executed. For example, a file with an extension of .286 is considered an external command.

Some of the commands are available through the Supplemental Option using the CD-ROM that is shipped with your system. See Appendix C, “Using the Supplemental Option on 4690,” on page 521 for more information about the Supplemental Option using the CD-ROM.

Using commands on the LAN

You can use some of the commands that are described in this chapter on the LAN (MCF Network). Table 22 on page 156 lists the commands that you can use on a LAN. Also, the Type section of each command description tells you if you can use the command on a LAN.

To ensure the commands work correctly, the logical name must be defined with the complete path when issued from either the master controller or the alternate controller. Using a logical name within another logical name is not supported.

For example, when issuing a command from the alternate controller, use the complete path of the logical name as follows:

```
C:> define lname01 = ADXLXACN::C:\DIR01\FILE01.DAT
C:> print lname01
Printing - FILE01.DAT on PRN1:
C:> _
```

The commands you use on the LAN operate the same way they do with a non-LAN system. The difference is that on a LAN system, you can direct the command to operate at a specific store controller. The store controller can be different from the one you are using to issue the command. For example, if you are working at your master store controller and want to assign a drive on a subordinate store controller, you must specify the node name of that store controller and the command and drive parameters. The specification would look like this:

```
ASSIGN C:=ADXLXCBN::C:
```

This command reassigns drive C on the subordinate controller to drive C on the store controller CB. Notice that you must specify the full name of the node, ADXLXxyN, where xy is the ID you have assigned to that store controller.

Note: The NetBIOS protocol is not supported for the following drives: E:, F:, M:, P: and NFS drives. That is, ADXLXDDN::M: and ADXLXDDN::E: for example, are not supported in commands (where *DD* is the controller ID of the controller).

Quick reference tables

The following tables are a quick reference to the commands. They are organized according to the task you want to perform.

Table 14. Working with files

Task	Command
Compare two files	COMP
Copy a file	COPY
Erase a file	ERASE
Copy a file	FCOPY
Rename a file	RENAME
Display contents of a file	TYPE

Table 15. Working with directories

Task	Command
Change the current directory	CHDIR
Display files in a directory	DIR
Make a directory	MKDIR
Set up a search path	PATH
Remove an empty directory	RMDIR
Display subdirectories	TREE

Table 16. Working with disks and diskettes

Task	Command
Back up a disk	BACKUP
Display disk statistics	CHKDSK
Compare two diskettes	DISKCOMP
Copy a diskette	DISKCOPY
Prepare a diskette	FORMAT
Restore backed-up files to disk	RESTORE
Verify information written to disk	VERIFY
Display the volume label	VOL

Table 17. Protecting your data

Task	Command
Write a volume and enable or disable disk protection	DISKSET
Change the protection or attributes of a file	FSET
Display or change file access privileges	SECURITY

Table 18. Modifying your user environment

Task	Command
Reroute drive requests	ASSIGN
Create a new shell	COMMAND
Create a logical name	DEFINE
Exit a shell you have created	EXIT
Change the system prompt	PROMPT

Table 19. Other commands

Task	Command
Stop or clear a program or activity	BREAK
Search files or command output for text strings	FIND
Display internal operating system commands	LIST
Display a file or command output one panel at a time	MORE
Return an error level value	QUERY
Sort a file or command output	SORT
Transfer the 4680 loader program to drive C (see Appendix C, "Using the Supplemental Option on 4690," on page 521)	SYS
Display the version of the operating system	VER
Burn an ISO Image to a CD or to a memory key	BURNISO

Table 20. Using driver commands

Task	Command
Install a device for the first time (see Appendix C, "Using the Supplemental Option on 4690," on page 521)	DVRLOAD
Add another unit to an existing device (see Appendix C, "Using the Supplemental Option on 4690," on page 521)	DVRUNIT
Remove a driver previously installed with DVRLOAD (see Appendix C, "Using the Supplemental Option on 4690," on page 521)	DVRUNLK

Table 21. Using batch file commands

Task	Command
Tell the system to perform a nested batch operation	BATCH
Display or not display the commands in a running batch file	ECHO
Provide a conditional alternative for IF commands	ELSE
Repeat a command for different files or programs	FOR
Transfer control to a line in a batch file	GOTO
Test a condition to see if it is true or false	IF
Suspend batch processing	PAUSE
Put remarks in a batch file	REM
Use more than 10 parameters on a command line	SHIFT

Table 22. Using commands on the LAN (MCF Network)

Task	Command
Reroute drive requests	ASSIGN
Tell the system to perform a nested batch operation	BATCH
Change the current directory	CHDIR
Compare two files	COMP
Copy a file	COPY
Create a logical name	DEFINE
Display files in a directory	DIR
Erase a file	ERASE
Search files or command output for text strings	FIND
Change the protection or attributes of a file	FSET
Make a directory.	MKDIR
Display a file or command output one panel at a time	MORE
Set up a search path	PATH
Print a file	PRINT
Rename a file	RENAME
Remove an empty directory	RMDIR
Sort a file or command output	SORT
Display subdirectories	TREE
Display text	TYPE
Display the volume label	VOL

Command format

The commands in this chapter use the following command format notation:

[] Items shown inside the square brackets are optional. To include optional items, type only the information inside the brackets. Do not type the brackets.

CAPS Words shown in capital letters are called *keywords*. Keywords must be entered exactly as shown except you can enter the keywords in any combination of uppercase or lowercase letters. Your system automatically converts keywords to uppercase.

Italics Items shown in italics represent items that you substitute or supply. For example, *file name* in the command format means that you must substitute the name of your file in place of the word *file name*.

| A vertical bar means that you can enter one of the items that are separated by the bar, but not both. For example, the format ON|OFF indicates that you can enter either ON or OFF. Do not type the vertical bar.

... An ellipsis (...) indicates that you can repeat an item as many times as you want.

You should include all punctuation that is shown in the command format such as commas, slashes, backslashes, equal signs, and colons.

The commands in this chapter are listed in alphabetical order. Included with each command is this summary information:

- Purpose of command
- Format of command
- Type of command
 - Internal – Command processing is resident in the operating system and can be used at any time.
 - External – Stored on a disk and must be read from disk before they can be executed.
- How to use the command (description and example)

Unless otherwise noted, you cannot abbreviate command names.

Note: Many commands result in an immediate response from the system in the form of output to your display or printer (or other specified device). In some cases the output contains names of disk drives, and the names are different from the standard A, B, C, or D. The drive names you might see and their equivalents are as follows:

f0	=	drive A	h0	=	drive C
f1	=	drive B	h1	=	drive D

Table 13 on page 135 lists the command keys for the commands that are entered in Command Mode.

Batch commands

Batch files are files that contain one or more operating system commands. When you enter the name of the batch file on the command line, the system executes each command in the file from top to bottom.

You create a batch file by using the text editor (see Chapter 8, “Creating and editing files,” on page 265). All batch files have the file extension .BAT. When you run a batch file, do not type the file extension.

A batch file runs until all the commands within it have been run, or you cancel the batch file by pressing **Ctrl+C**. The operating system returns you to the directory from which you started your batch file as long as you do not use any commands in your batch file that change the current directory.

Your system lets you use I/O redirection with batch files. You can redirect I/O from the command line and from within a batch file. Redirection from within a batch file overrides redirection that is invoked from the command line. See Chapter 5, “Working in Command Mode,” on page 133 for information on redirecting I/O.

To run a batch file, enter one of the following formats on your command line:

```
batch_file_name drive: batch_file_name batch_file_name parameters
drive: batch_file_name parameters
```

A batch file can be run from any drive. Parameters are values the batch file uses for a specified application.

Suppose that you created a batch file that is called CLEANUP.BAT consists of the following commands:

```
CHDIR \  
DIR *.BAK  
ERASE *.BAK
```

If CLEANUP.BAT is in your current directory of drive A, you can run it by entering this command:

```
A:>CLEANUP
```

After you press **Enter**, the commands run in order:

1. CHDIR makes the root directory the current directory.
2. DIR displays all backup (BAK) files in the root directory.
3. ERASE deletes all backup (BAK) files from the root directory.

When processing is finished, you are in the root directory of drive A, and the system prompt appears:

```
A:>
```

If you are on drive C and want to run CLEANUP.BAT on drive A, enter this command line:

```
C:>A:CLEANUP
```

CLEANUP.BAT runs its commands against the files on drive A. When it is finished; it returns to drive C in the directory where you started.

If you want to run CLEANUP.BAT on drive C of store controller CD, enter this command line:

```
C:>ADXLXCDN::C:CLEANUP
```

AUTOEXEC.BAT is a batch file that runs whenever a Command Mode session is started. You can also create an alternative .BAT file based on the operator ID. If there is an alternative .BAT file for a specific operator ID, this file will run each time that operator starts Command Mode. For example, if there is a file 99999999.BAT in the ADX_UPGM subdirectory, this file will run each time operator 99999999 starts Command Mode.

If no .BAT file exists specific to the operator ID, then the AUTOEXEC.BAT file in the ADX_UPGM subdirectory will run if it exists. Otherwise, nothing is run.

Using parameters in batch files

Batch files are very useful; however, it is impractical to create a specific batch file for every situation you might encounter. The solution is to create general purpose batch files that can be used in many situations.

To create a general purpose batch file, replace specific information in the batch file, such as drive letters and file names, with placeholders. Then, when you run the batch file, supply the missing values on your command line. These command line values are called *parameters*. Up to ten different placeholders are allowed in your batch file. A placeholder consists of a percent sign followed by one of the digits 0 through 9.

Figure 17 on page 159 illustrates the use of parameters within a batch file.

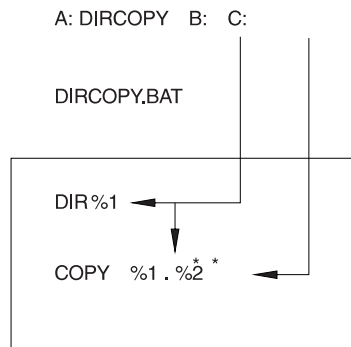


Figure 17. Using parameters

The system takes the first parameter on your command line and substitutes it for each %1 in your batch file, and substitutes the second parameter for each %2, and so on.

The following example shows CLEANUP.BAT with parameters in it:

```
CHDIR %1\
DIR %1*.BAK
ERASE %1*.BAK
```

If CLEANUP.BAT is on drive A, and you type: A:>CLEANUP C:

CLEANUP.BAT substitutes C: for each %1 it finds. Each command is run, displayed, and any output produced on the screen. This example runs CLEANUP.BAT with a parameter of C:.

```
B:>CLEANUP C:
```

```
B:>CHDIR C:\
```

```
B:>DIR C:*.BAK
```

```
Volume in drive C: is HARDDISK
Directory of C:
```

```
ONE      BAK      17    5-08-1985  14:04p
TWO      BAK      17    5-08-1985  14:04p
          2 Files      12345 bytes free
```

```
B:>ERASE C:*.BAK
```

```
B:>_
```

Notice that the ERASE command does not produce any output.

You can put several parameters in your batch file. For example, ERASEOLD.BAT contains these commands:

```
DEL %1
DIR %1
DEL %2
DIR %2
```

When you enter on the command line:

```
A:>ERASEOLD B:*.BAK C:*.BAK
```

The system substitutes B:*.BAK for %1 and C:*.BAK for %2.

Note: If the percent sign is used as part of a file name within a batch file, you must specify two percent signs. For example, to specify the TAX%.EXE file, you must enter it as TAX%%.EXE in the batch file.

You can put parameters in any order. You could invert the order of the commands in the ERASEOLD.BAT file, and the batch file still would work:

```
DEL %2  
DIR %2  
DEL %1
```

Chaining batch files

One of the commands that are included in a batch file can be the command to run another batch file. This feature lets you run two or more batch files consecutively by entering only one command line. To *chain* two batch files together, make the last command in the first batch file the command to run a second batch file. Figure 18 illustrates chaining batch files.

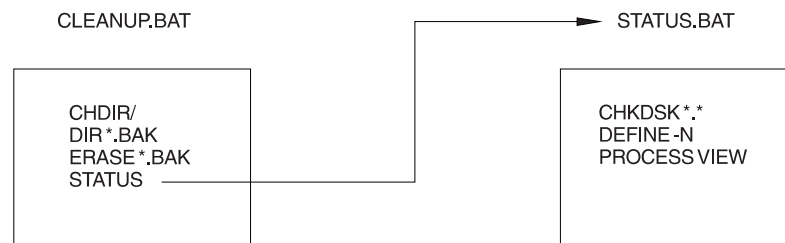


Figure 18. Chained batch files

To run both batch files, type CLEANUP on the command line and press **Enter**:

```
A:>CLEANUP
```

The system runs the commands in CLEANUP.BAT from top to bottom. When it reaches the command STATUS, control passes to STATUS.BAT and it runs this file as if you had entered STATUS on the command line. When the commands in STATUS.BAT finish, the system returns to the current directory.

If you place the command STATUS ahead of other commands in CLEANUP.BAT, control passes to STATUS.BAT. The remaining commands in CLEANUP.BAT are ignored.

Note: If in Figure 18 you include the command CLEANUP as the last command in STATUS.BAT, you create an infinite loop. CLEANUP.BAT runs its commands, control passes to STATUS.BAT which runs its commands, and then CLEANUP.BAT starts running again. The loop continues until you break out of it with **Ctrl-C**.

If you press **Ctrl-C** while running a batch file, you are prompted:

Terminate batch job (Y/N/A)?_

If you press **Y** (for Yes), the system ignores the remaining commands in the batch file. If you press **N** (for No), only the currently running command ends; batch processing continues with the next command in the batch file. If you press **A** (for All), the system stops the current batch file as well as any nested batch files.

Nesting batch files

Another way to run two or more batch files using one command line, is to use a *nested* batch file. A nested batch file contains the command `BATCH` followed on the same line by the name of another batch file. Unlike a chained batch file, when the second batch file is finished running, control returns to the first batch file.

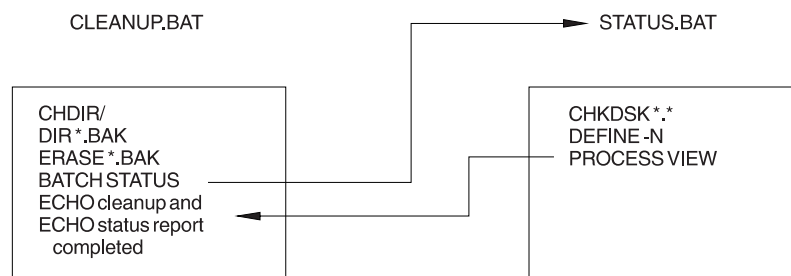


Figure 19. Nested Batch Files

In Figure 19 the system runs the commands in `CLEANUP.BAT` from top to bottom. When the command `BATCH STATUS` is reached, control is passed to `STATUS.BAT`. The system runs `STATUS.BAT` as if you had entered `STATUS` on the command line. When `STATUS.BAT` is finished, the system returns to `CLEANUP.BAT` and the commands continue to run there.

A batch file can contain more than one nested operation. The number of batch files you can nest is limited by the amount of memory available on your system. If you run out of memory while running a batch file, processing stops and an error message appears.

Commands for batch files

The following commands are especially appropriate for batch processing:

- `BATCH`
- `CLS`
- `ECHO`
- `FOR`
- `GOTO`
- `IF`
- `PAUSE`
- `REM`
- `SHIFT`

`BATCH`, `GOTO`, and `SHIFT` can only be used in a batch file. The other batch commands can also be entered on the command line, although they are more suited for batch use.

The following section provides an alphabetic list of commands available with the operating system.

ADXSDDEL command

Purpose

To clean the contents of or *securely delete* a file, and then delete it from the directory.

Formats

- (1) ADXSDEL filespec
- (2) ADXSDEL filespec -Q

Description

If you enter ADXSDEL by itself, you will receive an error message. Specify a file or a filespec in order to have a file or matching files cleaned and erased from the directory. Secure delete is only supported on the C: or D: drive. The following message will appear on any ADXSDEL command, even one that worked: 'This utility gives *secure deletion* of target files on C: or D: (including access across a network). For other target locations, *secure deletion* is not guaranteed, and it may simply give function equivalent to an ERASE.'

If you use the filespec *.* to clean and erase all the files in a directory or on a disk, the ADXSDEL command issues the following message to make sure that you want to clean and erase them: Are you sure (Y/N)? If you want to clean and erase all of the files on that particular directory, type **Y** and press **Enter**; otherwise, type **N** and press **Enter**. You cannot clean and erase the system file and the two special entries (. and ..) in each subdirectory.

Examples

Format 1

This format cleans and erases one or more files from the specified or current drive.

The next two examples show different ways of using format 1. The first example cleans and erases MYFILE.1 from the current directory of drive A:

```
C:>ADXSDDEL MYFILE.1
```

```
This utility gives secure deletion of target files on C: or D: (including access across a network).  
For other target locations, secure deletion is not guaranteed, and it may simply give a  
function equivalent to an ERASE.
```

```
C:>_
```

The next command cleans and erases all the files in the current directory of drive B:

```
D:>ADXSDDEL *.*
```

```
ADXSDDEL: Caution!
```

```
This utility gives secure deletion of target files on C: or D: (including access across a network).  
For other target locations, secure deletion is not guaranteed, and it may simply give a  
function equivalent to an ERASE.
```

```
Are you sure (Y/N)?
```

```
Y
```

```
D:>_
```

After you answer the prompt, ADXSDEL command carries out your instructions and returns you to the system prompt. If you answer **N**, the operating system cancels the ADXSDEL command and returns you to the system prompt. Your files are not cleaned or erased.

Format 2

This format uses the -Q parameter to clean and erase each of the files you specified. Press **Enter** after you type each response. The -Q parameter helps you avoid accidentally cleaning and erasing important files.

```
A:>ADXSDEL *.DAT -Q
FLPY0:LIST.DAT, delete (Y/N)?
N
FLPY0:MARKET.DAT, delete (Y/N)?
Y
FLPY0:REPORT.DAT, delete (Y/N)?
Y
A:>_
```

ASSIGN command

Purpose

To route requests from one drive to another drive.

Formats

Give a drive a new assignment:

(1) `ASSIGN original_drive=new_drive`

Reset a drive to its original assignment:

(2) `ASSIGN`

(3) `ASSIGN original_drive=`

(4) `ASSIGN original_drive=original_drive`

Type

Internal

Can be used on a LAN (MCF Network) or non-LAN system.

Description

In Formats 1, 3, and 4, you can assign more than one drive on a command line.

Some applications are designed to run only on drives A and B. By using `ASSIGN`, you can make applications use drives other than A and B, such as a hard disk drive or a RAM disk.

Changes to drive assignments affect only the program you are running in your current window. They do not affect other programs, commands you have already entered, or other users on the system.

Examples

Format 1

This format gives a drive a new assignment. Colons after the drive names are optional.

The following example routes requests for drives A and B to drive C:

```
A:>ASSIGN A=C B=C
```

```
A:>_
```

Format 2

This format resets all the drives to their original assignments:

```
A:>ASSIGN
```

```
A:>_
```

Format 3

This format resets a specific drive to its original assignment. Enter the original drive letter followed by an equal sign (=). The following example resets drive B:

```
A:>ASSIGN B=
```

```
A:>_
```

After the drive is reset, the command DIR B: displays the directory of drive B.

Format 4

This format also resets a specific drive to its original assignment. The following example resets drive A:

```
A:>ASSIGN A=A
```

```
A:>_
```

Multiple formats

You can use more than one format of the ASSIGN command on the same command line. In the example below, requests for drive A are routed to drive C, and drive B is reset to its original assignment.

```
A:>ASSIGN A=C B=
```

```
A:>_
```

LAN (MCF Network) Format

The following example uses format 4 of this command on a LAN (MCF Network) system. It assigns the original drive A to the new drive A on node DM.

```
A:>ASSIGN A=ADXLXDMN::A:
```

BACKUP command

Purpose

To back up one or more files from a source disk onto a backup disk.

Notes:

1. This command does not format the backup disk before backing up files. Use the FORMAT command to format the backup disk before executing the BACKUP command.
2. This command is not supported on the DVD-ROM drive.

Note:

Formats

Backing up the current directory:

- (1) BACKUP *source_drive: backup_drive:*
- (2) BACKUP *source_drive: backup_drive: -parameter*

Backing up one or more files:

- (3) BACKUP *source_drive: filespec backup_drive:*
- (4) BACKUP *source_drive: filespec backup_drive: -parameter*

Type

External

Intended for use with the Supplemental Option using the CD-ROM that is discussed in Appendix C, "Using the Supplemental Option on 4690," on page 521. Type can be used on a LAN (MCF Network) or on a non-LAN system.

Description

The BACKUP command backs up one or more files from a source hard disk drive or a RAM disk onto a backup diskette.

Backing up a file is not the same as copying it. Backup files are used to make copies of original files for safe keeping or to archive work. If files on the source drive become damaged, you can copy the backed up files onto the source drive with the RESTORE command.

Notes:

1. Backup files created with the BACKUP command on 4680 systems cannot be restored using the RESTORE command on the operating system. If you need to restore backup files from 4680 systems, copy the ADX_SPGM:RESTORE.286 file from the 4680 Operating System onto a diskette. Then, use this file to restore the files to the operating system.
2. This command cannot be used to back up the entire hard disk drive when the Store System has been IPLed from hard disk drive. If you want to back up a single file or a group of files, you can do so while online, but the files must not be in use when you back them up. If you get an access error or you want to back up the entire hard disk drive, you must IPL from the Supplemental Option using the CD-ROM to perform the backup.

Attention: Unless you specify the -A parameter, BACKUP erases existing files on the backup disk before it writes files from the source disk onto it.

Table 23 summarizes the parameters you can use with BACKUP. These parameters can be used singly or in any acceptable combination.

Table 23. BACKUP parameters

Parameter	Description
-A	Adds files to the backup disk without erasing files already on the disk.
-D:mm-dd-yy	Only backs up files modified on or after the specified date. The date must be specified in month, day, and year form. -D can be specified with the -A parameter.
-M	Only backs up files modified since the last backup.
-S	Backs up the current directory of the source disk and all the subdirectories attached to the current directory.

If you do not specify a directory path, files are backed up from the current directory.

If you do not specify a filespec, all the files in the directory are backed up.

After invoking BACKUP, you are prompted to insert a diskette. After BACKUP fills the diskette, it prompts you to insert another. You should label each diskette and record the date and diskette number.

BACKUP displays the name of each file as it backs it up.

Files are backed up to the root directory of the backup diskette.

This command can be used to back up files from the controller RAM disk; it will not back up files to the controller RAM disk.

Examples

Format 1

This format backs up the files from the current directory of the source disk onto the backup disk. In the following example, the source disk is in drive C, and the backup disk is in drive A:

```
A:>BACKUP C: A:
```

```
Insert the backup diskette 01 in drive f0:  
Warning! All files on diskette 01 will be erased.  
Press any key when you are ready ...
```

```
S
```

```
*** Backing up files to disk 01 ***
```

```
/ALPHA.TXT  
/CHAPTER.TXT  
/HOTPROG.BAS  
/INCOME.DAT
```

```
A:>_
```

Format 2

This format uses one or more of the backup parameters that are described in Table 23 on page 168. The following example uses the -A parameter to add files to the backup disk. Files already on the backup disk are not erased.


```
A:>BACKUP B: A: -A
```

```
Insert the backup diskette 01 in drive f0:  
Press any key when you are ready ...
```

```
S  
*** Backing up files to diskette 01 ***  
/CLIENTS.DAT  
/EXPENSES.DAT  
/SUPPLIES.LST
```

```
A:>_
```

Formats 3 and 4

Formats 3 and 4 are identical to formats 1 and 2 except that you specify one or more filespecs with your command. These formats enable you to back up selected files rather than the entire disk.

BATCH command

Purpose

To direct the operating system to perform a nested operation.

Format

BATCH *filespec*

Type

Internal

Type can be used on a LAN (MCF Network) or on a non-LAN system.

Description

Program control flows from the batch file that contains the BATCH command to the named *filespec*.

Filespec is the name of a batch file. If the batch file is not in the current directory, you must specify a directory path with *filespec*.

When the commands in the second *filespec* finish, control returns to the first batch file. Any additional commands in the first batch file are then run.

Example

This example shows a batch file on store controller CC that is executing a batch command on store controller CF.

```
C:> rem testing CC.....
```

```
C:> BATCH ADXLXCFN::TESTCF
```

CHDIR command

Purpose

To change the current directory. When you enter a file name without specifying which directory it is in, the operating system looks for the file in your current directory.

Formats

- (1) CHDIR
- (2) CHDIR *directory_path*

Type

Internal

Type can be used on a LAN (MCF Network) or on a non-LAN system, and it can be used with the store controller RAM disk.

Abbreviation

CD

Description

Changing the current directory implies that you have another directory on your disk to make current. If you try to change your current directory to a directory that does not exist, you receive an error message. For information on how to create a directory, see “MKDIR Command” on page 227.

If you do not specify a drive with the CHDIR command, the current drive is assumed.

Examples

Format 1

This format shows the current directory path of the current drive. In the following example, the current directory is also the root directory. Because of this, CHDIR does not show a directory path, only a drive letter.

```
A:>CHDIR
```

```
A:
```

```
A:>_
```

Format 2

You can use format 2 in several ways depending on which components of the directory path you specify. If you specify only a drive in format 2, it displays the current directory of that drive.

The next example includes a drive with the directory path. The command is entered from drive A and changes the current directory of drive B. The directory path (/) changes the current directory of drive B to its root directory.

```
A:>CHDIR B:/
```

```
A:>_
```

The next example of format 2 changes the current directory of drive A to the path root/BASEBALL/NATIONAL. The subdirectories BASEBALL and NATIONAL must already exist, or you receive an error message. If you are changing the directory path all the way back to the root directory, you must include a slash for the root directory.

```
A:>CHDIR BASEBALL/NATIONAL
```

```
A:/BASEBALL/NATIONAL/>_
```

The system prompt changes to show which directory and subdirectory you are in.

You can also use format 2 to attach additional directory levels to your current path, provided these directories already exist and are logically related to the current directory. The following example adds a subdirectory, GIANTS, to the current directory of drive A.

```
A:/BASEBALL/NATIONAL/>CHDIR GIANTS
```

```
A:/BASEBALL/NATIONAL/GIANTS/>_
```

LAN (MCF Network) format

On a LAN system you can access subdirectories on other store controllers.

```
A:>CHDIR ADXLCCN::C:/FOOTBALL
```

```
ADXLCCN::C:\FOOTBALL>_
```

This command changes your current subdirectory to FOOTBALL on drive C: of controller CC. Both the subdirectory FOOTBALL and the store controller CC must exist, or you receive an error message.

CHKDSK (check disk) command

Purpose

To display statistics about disks, files, and memory; and will analyze disks and correct errors.

Formats

- (1) CHKDSK
- (2) CHKDSK -F
- (3) CHKDSK -V
- (4) CHKDSK *filespec*
- (5) CHKDSK *filespec* -F -V
- (6) CHKDSK P: -F -V

Notes:

- 1. P: is not required, if done from the P: prompt:
P:> CHKDSK -F
- 2. CHKDSK P: is not supported in 4690 OS Enhanced mode.

Type

External

The -F option in formats 2 and 5 can be used on a store controller that has been IPLed from the hard-disk drive *only* when the specified drive is not the IPL hard-disk drive. If your controller IPLs from drive C:, you cannot use formats 2 or 5 with drive C:. To use the -F option on the IPL hard-disk drive, you must IPL from the Supplemental Option using the CD-ROM. All formats can be used with the store controller RAM disk. The format 6 is used on a CD or DVD; the -F option force check of a cleanly unmounted volume. The -V option is verbose mode.

Description

You should run CHKDSK occasionally for each disk to ensure the integrity of the file structures.

To prevent accidental changes to your disk, all Yes or No (Y/N) prompts from CHKDSK require you to press the **Enter** key after entering your response.

CHKDSK does not automatically correct the errors it finds in the directory or file allocation table. CHKDSK functions as if it was preparing to correct the disk (so you can analyze the possible results of the correction), but it does not actually write the corrections on the disk unless you specify the -F (fix) parameter.

You can use the CHKDSK command to check a disk on another store controller on the LAN (MCF Network) by using the LAN drive name (for example, ADXLXDDN::). However, if you use this option, the available memory information still refers to the machine on which CHKDSK was run.

Note: The output from the CHKDSK command is sent to STDERR. However, you can change the output to a file or to a printer by using the following commands.

To send the output to a file:

```
C:>CHKDSK >*FILE.DAT
```

To send the output to a printer:

```
C:>CHKDSK >*PRN:
```

See “Redirecting input and output” on page 143 for more information on redirecting output.

Examples

Format 1

This format displays a status report on the disk in the current drive. In the following example, CHKDSK analyzes the disk in drive C:

```
C:>CHKDSK
```

```
Volume T4690C in drive C:  created May 6, 1993  4:54pm

Disk:
    73383936 bytes total disk space.
        512 bytes in boot area.
    71680 bytes in 2 FATs.
    16384 bytes in 512 root-directory entries.
    1536 bytes in cluster round-off.
-----
    73293824 bytes total file space.
    12288 bytes in 3 hidden files.
    81920 bytes in 16 subdirectories.
    26628096 bytes in 590 user files.
-----
    46571520 bytes available on disk.

Memory:
    8388608 bytes total.
    1744896 bytes system.
    319488 bytes allocated.
-----
    6324224 bytes free.
```

Format 2

This format analyzes the directories and file allocation table on your disk and corrects any errors it finds. In Enhanced mode, additional information about memory status is displayed. This additional information includes the memory known to the 4690 OS kernel (label as OS4690 Memory). In addition, it includes memory used by the embedded Linux technologies (labelled as Enhanced memory). For example:

```
C:>CHKDSK - F
```

```
Volume VX4690_C32 in drive C: created Aug 31, 2009  8:07pm

Disk:
    114966394 KB total disk space.
        16 KB in boot area.
    28068 KB in 2 FATs.
    32 KB in 1024 root-directory entries.
    6 KB in cluster round-off.
-----
    114938304 KB total file space.
    10592 KB in 327 directories.
    4391840 KB in 5140 user files.
-----
    110535872 KB available on disk.

OS4690 Memory:
    1073741824 bytes total.
    22020096 bytes system.

Enhanced memory:
    1048576 KB RAM total.
    921144 KB RAM free.
```

90542080	bytes allocated.	2104504	KB swap total.
-----		2104420	KB swap free.
961179648	bytes free.	3025564	KB available.

If CHKDSK finds any lost allocation units (clusters) on the disk, it asks if you want to recover the lost data into files. If you answer yes, each chain of lost allocation units is recovered into its own file. The name of each file is FILE $nnnn$.CHK, where $nnnn$ is a sequential number that starts with 0000. The files are stored in the root directory of the current or specified drive. You can look at these files to see if they contain any useful information. If they do not, you can erase them.

Format 3

The -V (verify) parameter causes all path names on the specified or default drive to be displayed. If you specify a filespec, the number of noncontiguous blocks is reported. For example:

CHKDSK -V

Format 3 displays a list of all files (that includes any hidden ones) and their path names. It displays this in addition to the memory status report.

Format 4

Format 4 includes a filespec with CHKDSK. CHKDSK displays a report similar to the one in format 1, but it also includes the number of noncontiguous areas that are occupied by the specified file or files. CHKDSK accepts path names in file specifications.

In the following example, CHKDSK analyzes the diskette in drive A:

A:>**CHKDSK *.***

Disk in drive A: has no label.

Disk:

1474560	bytes total disk space.
512	bytes in boot area.
9216	bytes in 2 FATs.
7168	bytes in 224 root-directory entries.

1457664	bytes total file space.

1457664	bytes available on disk.

Memory:

8388608	bytes total.
1744896	bytes system.
270336	bytes allocated.

6373376	bytes free.

A:FORMAT.286

Contains 2 noncontiguous blocks.

Note: The Memory status display will be different in Enhanced mode, as described in “Format 2” on page 174.

Badly fragmented files (files with many noncontiguous areas) can slow down system performance when they are accessed; the disk drive head has to move around a lot to find parts of the file. To determine how fragmented your files are, use CHKDSK with a filespec of *.*. If your files are fragmented, you can correct this problem by copying them to another disk.

Format 5

Format 5 displays the number of noncontiguous areas that are occupied by the specified filespec and corrects any errors it finds in the directory or file allocation table. CHKDSK looks only for the filespec in the current directory.

CHKDSK filespec -F -V

The CHKDSK memory status report provides the following information:

Table 24. Information generated by CHKDSK

Term	Description
Total memory	Total memory available after the operating system kernel has been loaded into memory.
System memory	Memory allocated by the system
Allocated memory	Memory allocated by a process
Free memory	Memory not currently in use; the transient program area (TPA)

The sum of system memory, allocated memory, and free memory equals the total memory.

Format 6

In the following example, CHKDSK analyzes the Blocks and sizes, the Path names, the Connectivity, the Reference Counts and file Unique IDs, Free List and Logical Volume integrity of a CD or DVD.

The following messages will be displayed:

```
C:\>CHKDSK P: -F
```

```
    Initializing...
    Checking Sparing Table(s)...
    Checking Blocks and Sizes...
    Checking Pathnames...
    Checking Connectivity...
    Checking Reference Counts and Unique IDs...
    Checking Free List...
    Checking Logical Volume Integrity...
```

```
Directories: 1
Files:      1
Total blocks: 293600
Free blocks: 293554
```

If CHKDSK P: -F finds any error on the CD or DVD media; it will not ask if you want to correct the error, an automatic correction will be done.

Note: CHKDSK P: is not supported in 4690 OS Enhanced mode.

Format 7

In the following example, CHKDSK analyzes volume descriptors, logical volumes, volume descriptors sequences.

The following messages will be displayed:

```
    Initializing...
    LOCATING VOLUME RECOGNITION STRUCTURES
    LOCATING ANCHOR VOLUME DESCRIPTORS POINTERS RECORDS
    PROCESSING MAIN VOLUME DESCRIPTOR SEQUENCE
```


Comparing Volume Descriptor Sequences...
Comparing PVD's....
Comparing PD's....
Comparing USD's....
Comparing IUVD's...
Comparing LVD's...
GOVERNING LOGICAL VOLUME DESCRIPTOR:
IMPLEMENTATION USE VOLUME DESCRIPTOR:

Number of Files: 1
Number of Dirs: 1

Volume Format: OSTA UDF 2.01 Compliant

Volume unmounted cleanly, repair suppressed

CLS command

Purpose

To clear the panel.

Format

CLS

Type

Internal

Description

CLS clears the panel. Output returns to the panel when you run your next command, application, or batch file.

Example

C:>CLS

COMMAND command

Purpose

To change the way your user environment looks.

Formats

Invoke a copy of the primary shell:

- (1) COMMAND
- (2) COMMAND <string>
- (3) COMMAND -parameter <string>

Invoke a different shell:

- (4) COMMAND filespec -parameter

Type

External

Description

When you work with the operating system, the system prompt and command line are known as your user environment or *shell*. COMMAND lets you change how your user environment looks by displaying a different shell on the screen. The shell you see when you start the operating system is called the *primary* shell. A shell that you invoke with COMMAND is called the *secondary* shell.

COMMAND lets you:

- Invoke a secondary shell that is a copy of the primary shell (formats 1 through 3).
- Invoke a secondary shell that is different from the primary shell (format 4).

Table 25 summarizes the parameters that can be used with COMMAND:

Table 25. COMMAND parameters

Parameter	Description
-P	Makes the secondary shell permanent in memory. You cannot exit from this shell unless you log off or turn off your store controller.
-C <string>	Runs a command on the secondary shell, and then causes the secondary shell to terminate automatically when the command is finished. Substitute the command for <string> when you use this parameter.

Note: When you run a command on a shell that is different from your primary shell, the shell must be able to recognize the command. Some operating system commands are not compatible with other shells.

Examples

Format 1

Format 1 invokes a secondary shell that is a copy of your primary shell. The secondary shell has the same system prompt and recognizes the same commands as your primary shell. After you finish working with the secondary shell, use the EXIT command to return to the primary shell.

Format 2

Format 2 differs from format 1 only in that it accepts a command to run in the copy of the primary shell.

Format 3

The `-C <string>` parameter invokes a copy of the primary shell, runs a command in the new shell, and then automatically terminates the new shell. The following example invokes a secondary shell, runs the DIR command on it, and then terminates the secondary shell.

```
A:>COMMAND -C DIR
```

```
Volume in drive A: is $MANAGE
Directory of A:
```

EXPENSE	DAT	10368	3-04-1985	4:13p
INCOME	DAT	1613	6-27-1985	12:14p
BUDGET	DAT	31	2-15-1985	8:52a
LEVEL2	<DIR>		4-03-1985	9:10a
INVEST	BAS	2288	7-02-1985	3:25p

```
5 Files 141312 bytes free
```

```
A:>_
```

COMMAND allows you to precede the `-C` parameter with `-P`. However, this has the same effect as using `-C` by itself. You cannot have `-P` following `-C` in a COMMAND command line, because the `-C` parameter causes COMMAND to treat everything following the `-C` as input to another shell, not as input to COMMAND.

Format 4

Format 4 invokes the shell that is identified by the specified filespec. The file containing your shell must be an executable file. Executable files include files that have the extension 286. In the following example, the secondary shell is contained in the SHELL2.286 file, and it uses a percent sign as the system prompt. Because the secondary shell is not permanent in memory, EXIT returns you to the primary shell.

```
A:>COMMAND SHELL2.286
```

```
%EXIT
```

```
A:>_
```

Format 4 also lets you use one or both of the COMMAND parameters following the filespec designating the secondary shell. With the `-P` parameter you must log off or switch off your store controller to terminate the new shell. You cannot use EXIT.

COMP command

Purpose

To compare the contents of two files of equal size.

Formats

- (1) COMP
- (2) COMP *filespec_1*
- (3) COMP *filespec_1 filespec_2*

Type

External

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the store controller RAM disk.

Description

You can use COMP after copying a file to make sure that the copy is identical to the original.

If you want to compare two entire diskettes, use the DISKCOMP command, not COMP.

The COMP command lets you compare files as follows:

- On the same or different drives
- In the same or different directories
- With the same path name or file name (if the files are on different drives)

You can use global file name characters in the filespecs to compare two groups of files in which each pair of files is of equal size. COMP compares all the files matching *filespec_1* with the corresponding files in *filespec_2*.

COMP displays the path names and file names as each file is compared.

An error message appears if:

- COMP cannot find a file matching *filespec_1*.
- *filespec_1* and *filespec_2* are different sizes.
- A specified directory path is not valid.

If COMP finds mismatching information in the files, it displays the offset of the mismatching bytes in hexadecimal and decimal. After 10 unequal comparisons, COMP concludes that further comparing is unnecessary and displays:

10 Mismatches - ending compare

If the two files are identical, COMP displays:

Files compare ok

After the two files are compared, the next pair of files that match the two filespecs are compared. The process continues until no more files can be found that match *filespec_1*. COMP then displays:

Compare more files (Y)es?

—

At this point you can compare two more files or end the comparison. Type **Y** to compare two more files. You do not have to press **Enter**. You are immediately prompted for a new *filespec_1* and *filespec_2*.

Type **N** if you are finished comparing files. You do not have to press **Enter**. COMP returns you to the system prompt.

Examples

Format 1

Format 1 prompts you for *filespec_1* and *filespec_2* or a drive name. The following example compares GUESTS.LST on drive A with GUESTS.LST on drive B.

```
A:>COMP
```

```
Enter first filespec:  
GUESTS.LST
```

```
Enter second filespec or drive name:  
B:
```

```
A:GUESTS .LST and B:GUESTS .LST  
Files compare ok
```

```
Compare more files (Y)es?  
N
```

```
A:>_
```

Format 2

Format 2 specifies *filespec_1* only. COMP prompts you for *filespec_2* or a drive name. After you respond, COMP compares the files and displays the appropriate messages.

Format 3

Format 3 specifies both filespecs. If you only specify a drive name for *filespec_2*, COMP assumes that *filespec_2* has the same file name and file extension as *filespec_1* and looks for *filespec_2* in the default directory. The next example compares ANSI.SYS on drive A with ANSI.SYS on drive B.

```
A:>COMP ANSI.SYS B:
```

```
A:ANSI.SYS and B:ANSI.SYS  
Files compare ok
```

```
Compare more files (Y)es?  
N
```

```
A:>_
```

The following examples are all variations on format 3. The first example compares ORIGINAL.TXT on drive A with ARCHIVE.TXT on drive B.

```
A:>COMP A:ORIGINAL.TXT B:ARCHIVE.TXT
```

```
A:ORIGINAL.TXT and B:ARCHIVE.TXT  
Files compare ok
```

Compare more files (Y)es?
N

A:>_

The next example compares all the files in one directory with all the files in another directory. The files are listed as they are compared.

A:>**COMP A:/DIR1 A:/DIR2**

A:/DIR1/FILE1.DAT and A:/DIR2/FILE1.DAT
Files compare ok

A:/DIR1/FILE2.DAT and A:/DIR2/FILE2.DAT
Files compare ok

Compare more files (Y)es?
N

A:>_

The next example compares two unequal files, APPLES.TXT and ORANGES.TXT. COMP displays the following messages:

A:>**COMP APPLES.TXT ORANGES.TXT**

Compare error at offset 17H (decimal 23)
File 1 = 49H (decimal 73)
File 2 = 4FH (decimal 79)

Compare more files (Y)es?
N

A:>_

LAN (MCF Network) Format

This example uses format 3 of this command on a LAN system. It specifies a filespec on the current directory of store controller CC and a filespec on the current directory of the local store controller.

A:>**COMP ADXLXCCN::JWTST.BAT TESTJW.BAT**

ADXLXCCN::JWTST.BAT and C:TESTJW.BAT

Files are different sizes

COPY command

Purpose

To let you:

- Copy one or more files to another disk
- Copy files to the same disk provided you rename the copy or specify a different directory
- Combine files during the copying process
- Add one or more files to the end of another file
- Transfer data between any of the system devices

Note: When issuing a COPY command across a LAN (MCF Network) from a personal computer that is operating under DOS (that is, a non-4690 Store System LAN controller), use a fully qualified path name. The operating system does not support the use of the DOS NETSHARE command at the subdirectory level.

Using COPY on writable CD-RW or DVD media

When using COPY to put files on writable CD-RW or DVD media, note that the distribution attribute is *not* carried with the file to the optical media. Therefore, if you use the COPY command to create file backups, the distribution attribute is lost.

Formats

Copy a single file:

- (1) COPY *source*
- (2) COPY *source target*

Combine files:

- (3) COPY *source_1+source_2+...source_last target*

Add one or more files to the end of another file:

- (4) COPY *source_1+source_2+...source_last*

Type

External

COPY can be used on a LAN (MCF Network) or on a non-LAN system. It can also be used with the store controller RAM disk.

Description

Source is the filespec you want to copy from; *target* is the filespec you want to copy to. If you do not specify a drive with the source or target files, the current drive is assumed. If the target filespec is a directory path without a file name, files are copied into that directory without their names being changed.

You can use drive names, directory paths, and global file name characters with the source and target files. Note, however, that you can only use the asterisk (*) and the question mark (?) global file name characters with COPY. If you use global file name characters in the source filespec, the names of the files are listed as they are copied.

Files are usually copied in text (or ASCII) mode. To copy or combine binary files, use the -B parameter. You can combine ASCII and binary files by using the -A and

-B parameters on the same command line. When files are being copied, -B is the default for the source and destination. When files are being combined, -A is the default for the source and destination.

When you copy a file that is marked read-only, the copy does not have the read-only attribute.

Table 26 summarizes the parameters you can use with COPY.

Note: When you issue the COPY command, you must insert a space between the file specification and the parameter.

Table 26. COPY parameters

Parameter	Description
-A	Copies an ASCII (text) file. -A applies to the file preceding it and to all remaining files until you specify another -A or -B. When files are being combined, -A is the default for the source and target files.
-B	Copies a binary file. -B applies to the file preceding it and to all remaining files until you specify another -A or -B. When files are being copied, -B is the default for the source and target.
-S	Copies hidden and system files in addition to the specified file or files. You need to specify -S only once, and it can appear anywhere on the command line.
-V	Verifies that sectors are properly written on the target disk. Although -V slows down performance by forcing the operating system to verify your file twice, it is good practice to verify an important copy with -V. You need to specify -V only once, and it can appear anywhere on the command line.

Note: Suppose that you have a distributed file whose logical names define it as being on drive A of store controller CD. On store controller CM, the file is shown as being on a different drive. Do not use COPY to copy this file. Instead, use the Distributed File Utility, which is explained in Chapter 28, “Using the Distributed File Utility,” on page 465.

Examples

Format 1

Format 1 copies the source file to the current directory of the current drive. The target file is not renamed.

If you use format 1 on a single-drive system, COPY prompts you to insert the diskette that contains your source file, and then the diskette will contain your copy. Depending on the size of your store controller’s memory, you might be prompted to insert your diskettes more than once.

In the following example, format 1 copies the MYFILE.TXT file from drive B to the current drive (drive A). The file is not renamed.

```
A:>COPY B:MYFILE
```

```
1 file copied.
```

```
A:>_
```

The message 1 file copied. means that COPY created one target file.

When copying a file with a file name greater than eight characters in length and spaces within the file name, you must enclose the path and file name in quotes ("). In the following example, the file MY FILE WITH SPACES is copied from the M: drive.

```
COPY "M:\MY FILE WITH SPACES"
```

If the file is not enclosed in quotes, the copy process will fail.

Note: On a VFS drive, full wildcarding is not supported; only an asterisk (*) placed at the end of the file name is allowed. For example, the following two commands are allowed.

```
M:>copy myApp* \mydir\  
M:>copy * \mydir\
```

The following two commands are not allowed.

```
M:>copy *.* \mydir\  
M:>copy *.bat \mydir\
```

The following example uses format 1 with the -A parameter. SCORES.DAT is copied as an ASCII file from drive B to the current drive (drive A).

```
A:>COPY B:SCORES.DAT -A
```

```
1 file copied.
```

```
A:>_
```

You can also use global file name characters with format 1 to copy several files with one command line.

Format 2

Format 2 gives you the option of renaming your copy and specifying the drive it will be copied to. The next three examples show different uses of format 2.

The following example copies CURRENT.TXT to drive B and renames it ARCHIVE.TXT.

```
A:>COPY A:CURRENT.TXT B:ARCHIVE.TXT
```

```
1 file copied.
```

```
A:>_
```

Because the file was renamed, the target file could have been on the same drive as the source file.

The next example uses global file name characters to copy all the files on the current drive to drive B. COPY lists the files as it copies them. The files are not renamed. The -V parameter verifies the copies.

```
A:>COPY *.* B: -V
```

```
Copying -
```

```
COMMAND.COM  
STOCKS.DAT  
FORMAT.COM  
TEST.DAT  
CHAPTER.TXT
```

5 files copied.

A:>_

The previous example is very useful if the source files are fragmented. A file is fragmented if parts of the file are written on several areas of the disk and these areas do not touch. When a file is copied, these fragments are consolidated in the new file and the information is stored as one block (assuming that there is contiguous space available on the target disk). To determine if your files are fragmented, use the CHKDSK command that is described in “CHKDSK (check disk) command” on page 173.

The following variation on format 2 copies EXPENSES.DAT from the root directory of drive B to the subdirectory root/BUDGET, also on drive B. The copy is not renamed.

```
A:>COPY B:/EXPENSES.DAT B:/BUDGET
```

1 file copied.

A:>_

The preceding example assumes that directory BUDGET exists on drive B. If the directory did not exist, EXPENSES.DAT would have been copied into a file that is named BUDGET in the root directory of drive B.

LAN (MCF Network) Format

This example uses format 2 of this command on a LAN system. It copies JWTST.BAT from the root directory on the C drive of store controller CC to the default path on the local store controller.

```
A:>COPY ADXLCCN::C:\\JWTST.BAT JWTST.BAT
```

1 file copied

Combining files

Format 3

Format 3 adds a copy of one or more source files to the end of *source_1*, and then copies *source_1* to a target file. If the target file already exists, *source_1* is added to the end of it. The source files are combined only during the copy procedure. The content of each source file is the same as it was before copying.

COPY lists each source file as it combines them. The message 1 File combined. means that COPY created one target file. COPY lets you combine a maximum of 10 files.

COPY considers this process successful if it finds at least one, but not necessarily all, of the specified source files. If COPY cannot find any source files, it displays an error message.

The following example adds a copy of XYZ.DAT to the end of ABC.DAT on the current drive. The result is copied to drive B. The target file is named OLD.DAT.

```
A:>COPY ABC.DAT+XYZ.DAT B:OLD.DAT
```

Combining -

```
ABC.DAT
XYZ.DAT
```

1 file combined.

A:>_

You can also combine files from different drives and use the -A and -B parameters with different files on the same command line.

Note: This function should not be confused with the File Compress/Decompress Utility.

Appending files

Format 4

Format 4 adds one or more source files to the end of *source_1*, but does not copy *source_1* to a target file. Format 4 is useful for creating one large file from several smaller, related files.

COPY adds the source files to the end of *source_1* in the order you specify them.

COPY considers this process successful if it finds at least one, but not necessarily all, of the specified source files. If COPY cannot find *source_1*, it adds the remaining source files to *source_2*. If it cannot find *source_2*, it adds the remaining source files to *source_3*, and so on. If COPY cannot find any source files, it displays an error message.

The following example adds a copy of B.DAT from drive B to the end of A.DAT, and the result is left in A.DAT.

```
A:>COPY A.DAT+B:B.DAT
```

Combining -

```
A.DAT
B:B.DAT
```

1 file combined.

A:>_

The message 1 file combined. means that COPY found at least one source file to add to A.DAT.

The remaining examples illustrate more sophisticated uses of the COPY command.

The next example adds all the DAT files, except BASIC.DAT itself, to the end of BASIC.DAT:

```
A:>COPY BASIC.DAT+*.DAT
```

Combining -

```
AREA.DAT
VOLUME.DAT
WIDTH.DAT
HEIGHT.DAT
```

1 file combined.

A:>_

The following example creates multiple target files. Each PRT file is a list of part-time employees for a different work shift: DAY.PRT, SWING.PRT, and GRAVEYRD.PRT. Each FUL file is a list of full-time employees for each shift. By using global file name characters in the file names, COPY adds each FUL file to the end of the corresponding PRT file. The resulting file is a list of all the employees on a shift. This file is copied to an EMP file on drive B. This example creates three target files: DAY.EMP, SWING.EMP, and GRAVEYRD.EMP.

A:>COPY *.PRT+*.FUL B:*.EMP

Combining -

DAY.PRT
SWING.PRT
GRAVEYRD.PRT

3 files combined.

A:>_

Using device names with COPY

When you copy a file to or from a device, the file is copied in ASCII (-A) mode.

The following example copies information that is entered at the keyboard into a file. The name of the source file is CON, which is another name for your keyboard and screen.

After each line of text you type, press **Enter**. When you have typed your last line of text, press **Ctrl+Z** to insert an end-of-file character in the file. COPY puts the text in the target file and returns you to the system prompt. In this example, the target file is the SCREEN.TXT file.

A:>COPY CON SCREEN.TXT
I am using the COPY command
to copy text from my keyboard
to a file. [Ctrl+Z]

1 file copied.

A:>_

DEFINE command

Purpose

To enable you to substitute a logical name for a more complex name.

Formats

Create a logical name:

(1) `DEFINE logical_name=value`

List the logical names that have been created:

(2) `DEFINE`

(3) `DEFINE -N`

(4) `DEFINE -S -N`

(5) `DEFINE -N logical_name`

(6) `DEFINE -S -N logical_name`

Delete a logical name:

(7) `DEFINE -D logical_name`

Type

Internal

Type can be used on a LAN (MCF Network) or on a non-LAN system.

Description

The DEFINE command lets you substitute a logical name for a more complex device name, directory name, file name, or other logical information.

In addition, DEFINE lets you:

- List your logical names and the names they replace
- List any drive assignments you have changed with the ASSIGN command
- Delete any of the logical names you created
- Define the error message help level (see Table 27 on page 190)

The help level you specify determines the amount of information the operating system includes with its error messages. The higher the help level number, the more detailed the error message. The help level can have a value between 1 and 4.

Table 27 summarizes the type of information you get with each help level.

Table 27. Error message help levels

Help level	Description
1	Displays the operating system function, the error source module, and the return code.
2	Identifies the command and type of error in one sentence. An example of a level 2 error message is COPY: Write error.
3	Expands on the level 2 message and includes more specific information. An example of a level 3 message is COPY: An error occurred writing report.txt on A:.

Table 27. Error message help levels (continued)

Help level	Description
4	Expands on the level 3 message and often suggests a possible solution to the error. An example of a level 4 message is COPY: An error occurred writing report.txt on a: The disk A: is full. You can erase unnecessary files to free up space.

You can create a logical name at the process level. A process-level-logical name can be used only by the particular program for which it was created. The name would not have any meaning to programs that you are running on other screens, commands you have already started, or programs that are being run by other users.

A system-level logical name can be used by all the programs and users on the system.

You can specify only one logical name on a command line.

Examples

Format 1

Format 1 defines a process-level-logical name. The following example defines the filespec A:/WORK/WORDPROC.EXE as WP. This logical name can be used only by the program that is running in the current window. WP does not have any meaning to programs that are running in other windows.

```
A:>DEFINE WP=A:/WORK/WORDPROC.EXE
```

```
A:>_
```

To set the error message help level, set the logical name HELPLVL to a number from 1 to 4. The following example of format 1 sets the error message help level to 2:

```
A:>DEFINE HELPLVL=2
```

```
A:>_
```

Format 2

Format 2 causes all system-level and process-level names to be listed.

Format 3

Format 3 displays all the process-level logical names, their equivalent names, and any drive assignment names created with the ASSIGN command. The following example lists three process-level-logical names and one drive assignment:

```
A:>DEFINE -N
```

```
Process-level logical names:
```

```
WP = A:/WORK/WORDPROC.EXE
INT = A:/PROGRAMS/BASIC/INTEREST.BAS
AVE = A:/PROGRAMS/BASIC/AVERAGE.BAS
```

```
A:>_
```

Format 4

Format 4 displays all the system-level logical names and their equivalent names. The following example lists four system-level logical names:

```
A:>DEFINE -S -N
```

System-level logical names:

```
DATA = C:/CLIENTS/DATABASE.EXE
A: = FLPY0:
B: = FLPY1:
C: = HD1:
```

```
A:>_
```

Format 5

Format 5 displays the specified process-level-logical name and its equivalent name. This format also displays any drive assignments you have created with the ASSIGN command.

The following example displays the equivalent name of the process-level-logical name WP:

```
A:>DEFINE -N WP
```

```
WP = A:/WORK/WORDPROC.EXE
```

```
A:>_
```

Format 6

Format 6 displays the specified system-level logical name, for example:

```
A:>DEFINE -S -N DATA
```

```
DATA = C:/CLIENTS/DATABASE.DAT
```

```
A:>_
```

Format 7

Format 7 deletes the specified process-level-logical name. This example deletes the logical name WP:

```
A:>DEFINE -D WP
```

```
A:>_
```

DIR (directory) command

Purpose

To list files in the current or specified directory.

Note: When issuing a DIR command across a LAN (MCF Network) from a personal computer that is operating under DOS (that is, a non-4690 Store System LAN controller), use a fully-qualified path name.

Formats

List all the files in a directory:

- (1) DIR
- (2) DIR
- (3) DIR *-parameter*

List specific files in a directory:

- (4) DIR *filespec*
- (5) DIR *filespec -parameter*

Type

Internal

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the controller RAM disk.

Description

Depending on the format and parameters you use, you can list some or all of the files that are in a directory, and any hidden or system files.

Regardless of the format you use, DIR displays the name of each file, volume identification, and the amount of free space that is left on the disk. If the drive is formatted for a partition larger than 4GB, then the free space is shown as xxx Kbytes free. If the drive is formatted for a partition of 4GB, then the free space is shown as xxx bytes free.

In each format of the DIR command, entries that refer to subdirectories are identified with <DIR> in the file size column.

If you do not specify the -W (wide display) parameter, the listing includes the size of each file in decimal bytes and the date and time you last modified it.

If you do not specify the -H (hidden files) parameter, DIR does not list hidden and system files, even if they are present.

You can use global file name characters with DIR.

Unless you specify a path with DIR, the command lists the files in the current directory.

If either file name or file extension is omitted, an * is assumed.

You can also use the DIR parameters in any combination. Table 28 on page 194 summarizes these parameters.

Table 28. DIR parameters

Parameter	Description
-D	Shows the distribution attributes of each file (1=local, 2=mirrored at update, 3=mirrored at close, 4=compound at update, and 5=compound at close).
-H	Displays all hidden and system files in addition to the usual listing.
-P	Pauses after each panel of directory listings. Press any key to display the next panel of data.
-S	Displays all file names throughout the subdirectories of the specified drive that match the file specification given on the command line.
-W	Produces a wide display of the directory listing.

Examples

Format 1

Format 1 lists the current directory of the current drive. Format 1 is the equivalent of entering DIR *.*.

A:>DIR

Volume in drive f0: is DISKETTE1
Directory of f0:

```
FILE1   DAT      10368   9-04-1985   1:19p
FILE3   DAT      1613   6-27-1985   12:14p
BUDGET  DAT        31   2-15-1985    8:52a
LEVEL2  <DIR>           5-09-1985   12:10p
FILE1           2288   7-02-1985    3:25p
```

5 Files 141312 bytes free

A:>_

Format 2

Format 2 shows an example of using the DIR command on a system with long file name support enabled.

This example lists the directory entries in a specified subdirectory on a system that has an NFS mount group with mount point G: configured. Mount point G: is accessing a system that has long file name support enabled. The example shows the entries in the ExampleFiles subdirectory.

G:ExampleFiles/> dir

DIR:

The utility completed successfully.

Directory of n0:ExampleFiles/

```
2-04-2000  2:24p  <DIR>  .
2-04-2000  2:05p  <DIR>  ..
2-04-2000  2:08p           7  startpage.html
2-04-2000  2:08p          10  EAMITEMR.DAT
2-04-2000  2:21p  <DIR>    MoreFilesHere
12-03-1999  2:59p           4  HelloWorld.class
2-04-2000  2:19p          37  FileNumber1
```

7 Files 404619264 bytes free

G:\ExampleFiles/>

Format 3

Format 3 uses one or more parameters to modify the listing that is produced by format 1.

The following example uses the -W parameter to produce a wide display of the current directory. The wide display lists only file names and directory names. Each line contains five file names. You should use this parameter only if you have an 80-column display.

A:>**DIR -W**

Volume in drive f0: is DISKETTE1
Directory of f0:

FILE1	DAT	FILE3	DAT	BUDGET	DAT	LEVEL2	FILE1
5 Files 141312 bytes free							

A:>_

The next example uses the -D parameter to display the distribution attributes of each file. As shown in the listing, file ADXC SouF.DAT is compound at close, file TESTFILE.DAT is local, and so on.

C:\>**DIR:-D**

Volume in drive h0: is T4690C
Directory of h0:

.	<DIR>	6-17-1987	5:30p
..	<DIR>	6-17-1987	5:30p
ADXC SouF	DAT 68 5	10-24-1987	3:57p
EAMITEMR	DAT 8192 4	1-12-1988	3:09p
EAMTENDV	DAT 4096 4	11-28-1987	1:43p
EAMILIST	DAT 8192 3	1-12-1988	3:09p
EAMDMCTL	DAT 10752 3	8-06-1987	10:45a
TESTFILE	DAT 65518 1	7-20-1987	2:44p
6 Files 2885632 bytes free			

Format 4

Format 4 lets you list directory entries for a specified drive or for one or more specified files.

The following examples are all variations of format 4. The following example lists the files in the current directory of drive B:

A:>**DIR B:**

Volume in drive f1: is DISKETTE2
Directory of f1:

EXPENSES	DAT	24	5-05-1985	12:24p
SCORES	<DIR>		3-27-1985	9:10a
ABC	DAT	25	5-18-1985	4:30p
LEVEL2	<DIR>		5-15-1985	2:11p
LEVEL1	<DIR>		5-15-1985	2:06p

5 Files 151040 bytes free

A:>_

The next example lists the files in the current directory with the specified file name and file extension:

A:>**DIR BUDGET.DAT**

Volume in drive f0: is DISKETTE1
Directory of f0:

BUDGET	DAT	31	2-15-1985	8:52a
--------	-----	----	-----------	-------

1 File 141312 bytes free

A:>_

The next example uses a global file name character to list all the DAT files in the current directory of drive B:

A:>**DIR B:*.DAT**

Volume in drive f1: is DISKETTE2
Directory of f1:

EXPENSES	DAT	24	5-05-1985	12:24p
ABC	DAT	25	3-18-1985	4:30p

2 Files 151040 bytes free

A:>_

Entering a file name without a file extension produces the following result. In this case, the file extension defaults to *.

A:>**DIR FILE1**

Volume in drive f0: is DISKETTE1
Directory of f0:

FILE1		2288	7-02-1985	3:25p
FILE1	DAT			

2 Files 141312 bytes free

A:>_

To display the entry for a file that has no file extension, enter the file name followed by a period. In this case, the file extension does not default to *. The following example displays the entry for FILE1, but not for FILE1.DAT:

A:>**DIR FILE1.**

Volume in drive f0: is DISKETTE1
Directory of f0:

FILE1		2288	7-02-1985	3:25p
-------	--	------	-----------	-------

1 File 141312 bytes free

A:>_

The following example lists the files in the subdirectory LEVEL2 on drive A:

A:>DIR LEVEL2

Volume in drive f0: is DISKETTE1

Directory of f0:/LEVEL2/

.	<DIR>		5-09-1985	11:45a
..	<DIR>		5-05-1985	9:00a
MYPROG	COM	2463	7-30-1985	8:55a
HOMEWORK	TXT	1	9-18-1985	2:20p

2 Files 141312 bytes free

A:>_

The previous example includes two special entries: a single period and a double period. These entries are found in all subdirectory listings. The single period represents the subdirectory that is being listed (LEVEL2). The double period represents the directory that is one level above LEVEL2. This latter directory is called the *parent* directory. In the previous example, the parent directory is also the root directory.

If your current directory is LEVEL2 and you want to see the files in its parent directory, enter this command:

A:>DIR ..

Volume in drive f0: is DISKETTE1

Directory of f0:/LEVEL2/..

FILE1	DAT	10368	9-04-1985	1:19p
FILE3	DAT	1613	6-27-1985	12:14p
BUDGET	DAT	31	2-15-1985	8:52a
LEVEL2	<DIR>		5-09-1985	12:10p
FILE1		2288	7-02-1985	3:25p

5 Files 141312 bytes free

A:>_

Format 5

Format 5 is identical to format 4 except that you specify one or more of the DIR parameters.

LAN (MCF Network) Format

This example uses format 4 of this command on a LAN system. It lists the directory entries of the specified subdirectory on the C drive of store controller CC. The list is redirected to DIR.LST on the local store controller.

A:>DIR ADXLXCCN::C:\ADX_SDT1 > DIR.LST

DISKCOMP command

Purpose

To compare diskettes.

Note: This command is used only for comparing diskettes. This command cannot be used for individual files, RAM disks, hard disk drives, or optical cartridges.

Formats

- (1) DISKCOMP
- (2) DISKCOMP *drive_1*

Note: DISKCOMP is limited to the A: drive only.

Type

External

Description

DISKCOMP compares the contents of two entire diskettes.

Use the COMP command to compare individual files. If you specify a fixed disk drive letter with DISKCOMP, you receive an error message.

DISKCOMP determines the number of sides and sectors per track that are compared based on the diskette in *drive_1*.

DISKCOMP prompts you to insert the diskettes into drive_1 at the appropriate time.

DISKCOMP waits for you to press any key before it continues.

If an error occurs while DISKCOMP is comparing the diskettes, it issues a message that indicates the track and side on which the error occurred. Then DISKCOMP continues to compare the rest of the diskette.

When DISKCOMP is finished, it prompts:

Compare more diskettes (Y)es?_

If you press **Y**, DISKCOMP prompts you to insert another pair of diskettes. It compares these diskettes on the drives you originally specified.

If you do not want to compare any more diskettes, press **N**.

Examples

Format 1

Format 1 performs a single-drive comparison on the current drive. Depending on the size and availability of your store controller's memory, DISKCOMP might prompt you to insert your diskettes more than once.

A:>DISKCOMP

Insert first diskette in drive f0:
Press any key when you are ready ...

Comparing 9 sectors per track, 2 sides

Insert second diskette in drive f0:
Press any key when you are ready

Comparing

Insert first diskette in drive f0:
Press any key when you are ready ...

Diskettes compare ok

Compare more diskettes (Y)es?_

Note: Messages can vary slightly depending on the format of the diskettes that are being compared.

Format 2

Format 2 specifies *drive_1* but not *drive_2*. DISKCOMP uses the current drive as the second drive. If you specify the current drive for *drive_1*, DISKCOMP performs a single-drive comparison.

A:>DISKCOMP B:

Insert first diskette in drive f1:

Insert second diskette in drive f0:
Press any key when you are ready

Comparing 9 sectors per track, 2 sides

Diskettes compare ok

Compare more diskettes (Y)es?_

Note: Messages can vary slightly depending on the format of the diskettes that are being compared.

Format 3

Format 3 compares the diskette in *drive_1* with the diskette in *drive_2*. If you specify the same drive twice, DISKCOMP performs a single-drive comparison.

The following example compares the diskettes in drives A and B:

A:>DISKCOMP A: B:

Insert first diskette in drive f0:

Insert second diskette in drive f1:
Press any key when you are ready

Comparing 9 sectors per track, 2 sides

Diskettes compare ok

Compare more diskettes (Y)es?_

Note: Messages can vary slightly depending on the format of the diskettes that are being compared.

DISKCOPY command

Purpose

To copy the entire contents of one diskette to another.

Note: This command is supported in Classic Mode only.

Formats

- (1) DISKCOPY
- (2) DISKCOPY *source*
- (3) DISKCOPY *source destination*
- (4) DISKCOPY -M *source destination*

Type

External

Description

Source is the drive that contains the diskette you want to copy; *destination* is the drive that contains the diskette you are copying to.

Attention: DISKCOPY writes over any information already on the destination diskette.

DISKCOPY copies only diskettes. If you specify a hard disk drive, optical drive, or RAM disk letter, DISKCOPY displays an error message. If you want to copy individual files, use the COPY command.

Notes:

1. Both diskettes must be formatted with the same number of sides and sectors per track. If the destination diskette has never been formatted, you must format it before using DISKCOPY.
2. DISKCOPY is only supported in V6 Classic Mode.

Note:

DISKCOPY prompts you to insert the diskettes at the appropriate times. DISKCOPY waits for you to press any key before continuing. After copying, DISKCOPY prompts:

Do you want to copy another diskette (Y/N)?_

If you press **Y**, DISKCOPY prompts you to insert another pair of diskettes. The copy is performed on the drives you originally specified.

Press **N** to end the command.

Examples

Format 1

Format 1 performs a single-drive copy on the current drive. Depending on the size of your store controller's memory, the operating system might prompt you to insert your diskettes more than once.

Note: Messages can vary slightly depending on the format of the disks that are being copied.

A:>**DISKCOPY**

Insert source diskette in drive f0:

Press any key when you are ready

Copying 80 tracks, 18 sectors per track, 2 side(s)

Insert destination diskette in drive f0:

Press any key when you are ready

Insert source diskette in drive f0:

Press any key when ready

Copying 80 tracks, 18 sectors per track, 2 side(s)

Insert destination diskette in drive f0:

Press any key when ready

Do you want to copy another diskette (Y/N)?_

Format 2

Format 2 specifies the source drive but not the destination drive. DISKCOPY uses the current drive as the destination drive. If the source drive is also the current drive, a single-drive copy takes place. The following example copies the diskette in drive B to the current drive, drive A.

Note: Messages can vary slightly depending on the format of the disks that are being copied.

A:>**DISKCOPY B:**

Insert source diskette in drive f1:

Insert destination diskette in drive f0:

Press any key when you are ready

Copying 80 tracks, 18 sectors per track, 2 side(s)

Do you want to copy another diskette (Y/N)?_

Format 3

Format 3 specifies both the source and the destination drives. If you specify two different drives for the source and destination, the contents of the first diskette are copied to the second. If you specify the same drive for the source and destination, DISKCOPY performs a single-drive copy.

Note: Messages can vary slightly depending on the format of the diskettes that are being copied.

A:>**DISKCOPY A: B:**

Insert source diskette in drive f0:

Insert destination diskette in drive f1:

Press any key when you are ready

Copying 80 tracks 18 sectors per track, 2 side(s)

Do you want to copy another diskette (Y/N)?_

Format 4

Format 4 lets you make multiple copies of the source diskette. If you specify the -M option, DISKCOPY prompts you for another destination diskette after each successful copy.

ECHO Command

Purpose

To determine whether command lines in a batch file appear as the batch file runs.

Formats

- (1) ECHO
- (2) ECHO ON
- (3) ECHO OFF
- (4) ECHO *message*

Type

Internal

Description

The ECHO command is a batch command that determines whether the lines in a batch file appear as the batch file runs.

Any output and messages that are produced by a command always appear, whether echo is ON or OFF.

Note: To echo any of the reserved symbols (>, <, !, |), you must precede them with a quote mark (").

Examples

Format 1

Format 1 displays whether ECHO is ON or OFF.

```
A:>ECHO
```

ECHO is ON.

```
A:>_
```

Format 2

Format 2 (ECHO ON) displays the command line of each command as it runs.

Format 3

When you use format 3 (ECHO OFF), the command lines in the batch file do not appear as the file runs. In addition, any remarks in the file that are made with the REM command do not appear. This example illustrates format 3:

```
A:>ECHO OFF
```

—

The cursor sits suspended below the command line and no system prompt appears. The system waits for you to enter another command.

Format 4

Format 4 (ECHO *message*) consists of the ECHO command followed by a message. This message always appears when you run your batch file, whether echo is ON or OFF. The message is a string of ASCII characters up to 127

characters long. For a list of the ASCII characters, see the *4680 BASIC: Language Reference*. The following example shows a batch file, called ECHOTEST, that illustrates format 4:

```
ECHO OFF
ECHO Although ECHO is OFF, this message appears.
REM This remark is not displayed.
ECHO ON
REM ECHO is now ON.
REM Now remarks are displayed.
```

When you run ECHOTEST, the following output appears:

```
A:>ECHOTEST
```

```
A:>ECHO OFF
```

```
Although ECHO is OFF, this message appears.
```

```
A:>REM ECHO is now ON.
```

```
A:>REM Now remarks are displayed.
```

```
A:>_
```

ELSE Command

Purpose

To provide conditional alternatives for IF commands.

Format

ELSE *command*

Type

Internal

Description

The ELSE command provides conditional alternatives for IF commands. ELSE can only be used following an IF statement. Nested IF commands can appear within ELSE statements if the ELSE statement is enclosed in curly braces ({}).

Example

```
ECHO OFF
IF EXIST TEST.DAT {
    ECHO The file TEST.DAT exists. It will be erased.
    ERASE TEST.DAT
}
ELSE {
    ECHO The file TEST.DAT is missing. It will be created.
    ECHO JUNK > TEST.DAT
}
```

ERASE Command

Purpose

To delete the specified file.

Formats

- (1) ERASE *filespec*
- (2) ERASE *filespec* -Q

Type

Internal

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the store controller RAM disk.

Abbreviation

DEL

Description

You can use the alternate form DEL instead of ERASE. ERASE and DEL function the same way.

If you enter ERASE by itself, you receive an error message. If you do not specify a drive with the ERASE command, ERASE deletes the file from the current drive. If you do not specify a path, the ERASE deletes the file from the current directory.

If you use the filespec *.* to erase all the files in a directory or on a disk, the ERASE command issues the following message to make sure that you want to erase them:

Are you sure (Y/N)?

—

If you want to erase all of the files on the disk, type **Y** and press **Enter**; otherwise, type **N** and press **Enter**.

You cannot erase the system file and the two special entries (. and ..) in each subdirectory.

The operating system does not automatically put an * on the end of the file specifications. You **must** specify the extension. For example, ERASE ADX* will cause the operating system to erase only the ADX* files without an extension.

You can use the global file name characters *, ?, and ^ in the file name and file extension. You should use global file name characters with caution, however, because several files can be erased with a single command. For more information on global file name characters, see Chapter 5, "Working in Command Mode," on page 133.

Examples

Format 1

Format 1 erases one or more files from the specified or current drive. The next three examples show different ways of using format 1. The first example erases MYFILE.1 from the current directory of drive A:

```
A:>ERASE MYFILE.1
```

```
A:>_
```

The next command erases all the files in the current directory of drive B:

```
A:>ERASE B:*. *
```

```
Are you sure (Y/N)?
```

```
Y
```

```
A:>_
```

After you answer the prompt, ERASE carries out your instruction and returns you to the system prompt. If you answer **N**, the operating system cancels the ERASE command and returns you to the system prompt. None of your files are erased.

The following example erases all the files in a specific directory:

```
A:>DEL B:/ACCOUNTS
```

```
Are you sure (Y/N)?
```

```
Y
```

```
A:>_
```

Format 2

Format 2 uses the -Q parameter to erase each of the files you specified. Press **Enter** after you type each response. The -Q parameter helps you avoid accidentally erasing important files.

```
A:>ERASE *.DAT -Q
```

```
FLPY0:LIST.DAT, delete (Y/N)?
```

```
N
```

```
FLPY0:MARKET.DAT, delete (Y/N)?
```

```
Y
```

```
FLPY0:REPORT.DAT, delete (Y/N)?
```

```
Y
```

```
A:>_
```

LAN (MCF Network) Format

This example uses format 1 of this command on a LAN system. It erases file FIX.1.BAT from the root directory of the C drive on store controller CC.

```
C:>ERASE ADXLCCN::C:\\FIX1.BAT
```

The following example uses the DEL form of this command on a LAN system:

```
C:>DEL ADXLCCN::C:\\TEST\\TST1.BAT
```

This is an abbreviated form of format 1 of the ERASE command. The command is issued to a remote store controller.

EXIT Command

Purpose

To return from the command shell to the SYSTEM MAIN MENU or to return from a secondary shell to the primary shell.

Format

EXIT

Type

Internal

Description

When you start Command Mode from the SYSTEM MAIN MENU, you create a primary shell. EXIT removes the primary shell from memory and returns you to the SYSTEM MAIN MENU.

If you are in Command Mode and invoke a secondary shell with COMMAND, EXIT returns you to the primary shell.

Example

In the following example, EXIT returns you to the primary shell and the SYSTEM MAIN MENU. First, you start a Command Mode window:

```
C:>COMMAND
```

```
C:>EXIT
```

```
C:>EXIT
```

The second EXIT returns you to the SYSTEM MAIN MENU.

FCOPY Command

Purpose

To execute file-to-file copies faster.

Formats

- (1) FCOPY *source destination* -S
- (2) FCOPY *source destination* -Q

Type

Internal

Description

FCOPY lets you execute file-to-file copies faster than the COPY command.

Note: FCOPY does not create a temporary file during the copying process, so if you end the process you could lose your destination file (if it already exists). You can replace the existing batch-file or command-line COPY command with the FCOPY command. To replace the command, type the following statement on the command line or add it to your AUTOEXEC.BAT file:

```
DEFINE COPY = FCOPY
```

Note: If you use the full expression COPY.286 in your batch file or on the command line, the COPY command is executed instead of FCOPY.

Your source and destination files can contain drive names, device names, and directory paths. Global file name characters can only be used in the source file, and if they are used, the names of the files are listed as they are copied. See Chapter 5, "Working in Command Mode," on page 133 for information on global filename characters.

Read-only attributes are not copied with a file. If a file is marked read-only, the destination copy does not have the read-only attribute.

Examples

Format 1

Format 1 lets you copy system and hidden files.

Format 2

Format 2 displays a message that asks you if the file is to be copied.

FIND Command

Purpose

To search files or command output for a specified character string.

Formats

Search one or more files:

- (1) FIND *"string" filespec*
- (2) FIND *-parameter "string" filespec*
- (3) FIND *"string" filespec_1 ... filespec_last*
- (4) FIND *-parameter "string" filespec_1 . . . filespec_ last*

Search command output:

- (5) COMMAND | FIND *"string"*
- (6) COMMAND | FIND *-parameter "string"*

Type

External

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the store controller RAM disk.

Description

FIND searches files or command output for a specified character string, and sends lines that include the string to the standard output device or to a file.

The string you want to search for must be enclosed in quotation marks. If the string begins with a quotation mark, enter two quotation marks. The FIND command is case-sensitive.

You can specify more than one filespec on the command line. Separate each filespec with a space. FIND searches for the string in each file, in the order the files appear on the command line.

Table 29 summarizes the parameters you can use with FIND.

Table 29. FIND Parameters

Parameter	Description
-C	Displays the file name in which the matched string was found, followed by the total number of lines in which the quoted string was found.
-N	Displays the line number of each match, followed by the line in which the string occurs.
-V	Displays each line that does not include the string.

Examples

Format 1

Format 1 displays each line of the specified file that contains the string. The following example searches the file STAR for the string "Twinkle, Twinkle,". The filespec of STAR appears below the command line, preceded by hyphens.

```
A:>FIND "Twinkle, Twinkle," STAR
```

```
----- STAR  
Twinkle, Twinkle, little star  
Twinkle, Twinkle, little star
```

```
A:>_
```

Format 2

Format 2 uses the -C parameter. The following example counts the number of times the string "Nevermore" appears in the file A:/POETS/POE/RAVEN.TXT:

```
A:>FIND -C "Nevermore" /POETS/POE/RAVEN.TXT
```

```
----- POETS/POE/RAVEN.TXT: 8
```

```
A:>_
```

The following example displays the line number of each match:

```
A:>FIND -N "Twinkle, Twinkle," STAR
```

```
----- STAR  
[1]Twinkle, Twinkle, Little star  
[5]Twinkle, Twinkle, Little star
```

```
A:>_
```

The following example lists the lines in AUTHORS.TXT that do not contain the string "BRONTE, EMILY":

```
A:>FIND -V "BRONTE, EMILY" AUTHORS.TXT
```

```
----- AUTHORS.TXT  
This semester you will be reading two works  
by each of these authors:
```

```
AUSTEN, JANE  
DICKENS, CHARLES  
JOYCE, JAMES  
STEVENSON, ROBERT LOUIS
```

```
A:>_
```

Formats 3 and 4

Formats 3 and 4 are identical to formats 1 and 2 except that you specify more than one filespec on your command line.

Format 5

Format 5 searches the output from a command for the specified string. The following example searches the output of the DIR command for the string ACCOUNTS PAY:

```
A:>DIR | FIND "ACCOUNTS PAY"
```

```
ACCOUNTS PAY 1429 1-18-1985 1:42p
```

Format 6

Format 6 is identical to format 5 except that you specify one of the FIND parameters.

LAN (MCF Network) Format

This example uses format 1 of this command on a LAN system. It searches file FIX.BAT in the root directory of the C drive on store controller CC for the character string "bat".

```
A:>FIND "bat" ADXLCCN::C:\\FIX.BAT
```

FOR Command

Purpose

To repeat another command several times for different files, commands, or applications.

Format

FOR %c IN (*item_1 ... item_n*) DO *command* %c

Type

Internal

Description

FOR can be used on the command line or in a batch file, but it is usually used in a batch file. Only one FOR command is allowed on a command line or on one line in a batch file.

The words FOR, IN, and DO are required. The *c* in the %c parameter can be any single character. You can specify two percent signs with the %c parameter if you want.

The items in the list can be the names of files (including other batch files), commands, applications, or text. Separate each item with one or more blank spaces. You can use global file name characters with the FOR command, but not path names. There is no limit to the length of your item list; however, the entire FOR command line must be 128 characters or less.

The command at the end of the FOR command can be the name of a command, an application, or a batch file. This command is run against each of the items in your list in left-to-right order.

Examples

This example runs the TYPE command against FILE1, FILE2, and FILE3.

```
FOR %v IN (FILE1 FILE2 FILE3) DO TYPE %v
```

This is the same as entering TYPE FILE1, TYPE FILE2, and TYPE FILE3 on successive command lines or on successive lines in your batch file.

The next example shows that you do not have to use file names for items; you can use any string provided it is meaningful in the context of the FOR command.

```
FOR %v IN (Gloria Rose Stephanie) DO ECHO %v
```

In left-to-right order, the names Gloria, Rose, and Stephanie are echoed to the panel.

FORMAT Command

Purpose

To prepare a disk in a recording format acceptable to the operating system. It analyzes the entire disk for any defective tracks and prepares the disk to accept operating system files.

Note: Formatting a RAM disk is not supported by this command.

Formats

- (1) FORMAT *drive*:
- (2) FORMAT *drive*: *-parameter*

Type

External

Description

Attention: Formatting destroys any existing data on the disk. Because of this, you must always specify the disk you want to format.

You must format all new disks before they can be used by the operating system. You can format your disk with the default values, or if you are formatting a diskette, you can also use the -P parameter to select a standard format from the FORMAT MENU. Table 30 on page 214 summarizes the parameters you can use with FORMAT. Unless otherwise noted, you can use these parameters in any combination on the same command line.

Note: When formatting diskettes with V6 Enhanced Mode, parameters are ignored and the diskette will be formatted as a 1,440k diskette.

Table 30. FORMAT parameters

Parameter	Description
-P	Displays the FORMAT MENU and prompts you for a diskette format. -P can be used only to format diskettes. -P overrides -1 and -8 if they appear on the same command line.
-Pn	Formats the diskette using the <i>n</i> th option from the FORMAT MENU.
-V	Prompts you for a volume label. A volume label must be 11 characters or fewer. All characters acceptable in file names are acceptable in volume labels. Volume labels are used to identify your disks.
-1	Selects a single-sided format from the internal library of standard formats. -1 can be used only with diskettes.
-8	Selects an 8-sector format from the internal library of standard formats. -8 can be used only with diskettes.
-Cn	Defines the sectors per cluster on a disk. <i>n</i> must be a power of 2.
-L	Specifies the long format of an optical cartridge, CD, or DVD media
-O	Needs to be used when running the format command in the background to format an optical cartridge.

The -1 and -8 parameters let you modify the default formatting values. When you specify -1, FORMAT prepares your diskette for single-sided use. FORMAT does not

change the default values for the number of sectors and tracks if it has a single-sided format with those values in its library.

When you specify -8, FORMAT prepares your diskette at 8 sectors per track. FORMAT does not change the default values for the number of sectors and sides if it has an 8-sector format with those values in its library.

When you use the FORMAT command to prepare a diskette, FORMAT prompts you to insert a diskette in the current or specified drive, warns you that any data already on the diskette will be lost, and asks you if you want to continue the FORMAT operation.

If you decide to continue, type **Y** and press **Enter**. FORMAT then displays the drive name, the formatted size of the diskette, the number of sectors, the total number of tracks, the number of sides, and the media size. The HEAD and TRACK fields are updated while the disk is being formatted.

When the formatting is completed, FORMAT displays the **C:** prompt.

All of the values that are filled in by Xs above vary according to the parameters you use to format your disk and the type of disk that you have.

Enter **Y** to format another diskette. Press **Enter** to end the FORMAT program.

Examples

Because of the variety of disks available, statistics displayed by the FORMAT command are represented by Xs in the following examples.

Format 1

Format 1 formats the disk in the specified drive with the default values.

The following example uses format 1 to format the diskette in drive A:

A:>FORMAT A:

Formatting drive A:

X,XXXk (XX sector, XX track, XXXX-sided, X.XX")

WARNING: All data on drive A: will be LOST.

Insert diskette to be formatted in drive A:

Continue (Y/N) ? Y

Formatting drive A:

X,XXXk (XX sector, XX track, XXXX-sided, X.XX")

HEAD XX TRACK XX

Formatted drive A:

XXXX total sectors at XXX bytes/sector, X sectors/cluster.

XXXXXX bytes total disk space.

XXX bytes in boot area.

XXX bytes in X FATs.

XXX bytes in XXX root directory entries.

XXXXXXX bytes available on disk.

Format another floppy diskette (Y/N)? N

A:>_

Format 2

These examples use format 2 and one or more FORMAT parameters.

The next example uses the -P parameter to display a list of formats. The letters in parentheses after each format are media codes. The operating system and certain programs use media codes to identify the disk format you are using. Type your selection at the cursor position and press **Enter**.

A:>FORMAT B: -P

Choose one of the following:

6. 720k (9 sector, 80 track, double-sided, 3.5")

7.1,440k (18 sector, 80 track, double-sided, 3.5")

8.2,880K (36 sector, 80 track, double-sided 3.5")

Press ESC to quit.

Enter choice:

Note: This table varies slightly depending on the type of diskette that you are formatting. Select the correct option from the menu. The -V parameter prompts you for a volume label.

A:>FORMAT B: -V

Formatting drive B:

X,XXXk (XX sector, XX track, XXXXX-sided, X.XX")

WARNING: All data on drive B: will be LOST.

Insert diskette to be formatted in drive B:

Continue (Y/N) ? Y

Formatting drive B:

X,XXXk (XX sector, XX track, XXXXX-sided, X.XX")

HEAD XX TRACK XX

Enter disk volume label (1-11 characters): DISKETTE2

Formatted drive B:

2880 total sectors at 512 bytes/sector, 1 sector/cluster.


```
1474560 bytes total disk space.
   512 bytes in boot area.
   9216 bytes in 2 FATs.
   7168 bytes in 224 root directory entries.
-----
1457664 bytes available on disk.
```

Format another floppy diskette (Y/N)?

In the previous example, DISKETTE2 was entered as the volume label. The FORMAT command does not allow you to enter a volume name that is longer than 11 characters.

Format 3

The following example uses format 3 to format the DVD in drive P (Classic mode):

```
C:\>format p: -L
```

```
FlexOS Format Version 2.02
Copyright (C) Digital Research, Inc 1985,91
```

Formattting drive P:

xxx percent of disk formatted

```
C:>
```

Note: where xxx is the percent of disk formatted from 001 to 100

Format 4

The following example illustrates problems that could occur when a short format fails (Classic mode). The following message is displayed:

```
C:\>format p:
```

```
FlexOS Format Version 2.02
Copyright (C) Digital Research, Inc 1985,91
```

Formatting drive P:

FORMAT: The media type is unknown and unsupported by the file system.

```
C:>
```

CHKDSK command must be used to get mode details about short format error.

```
C:\>chkdsk p: -F
```

```
Format Failed
Internal Error
Disk has unreadable media.
```

FSET Command

Purpose

To set file attributes.

Formats

View the attributes of a file:

(1) FSET *filespec*

Change the attributes of a file:

(2) FSET *filespec* -H=ON | OFF

(3) FSET *filespec* -A=ON | OFF

(4) FSET *filespec* -S=ON | OFF

(5) FSET *filespec* -R=ON | OFF

Type

External

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the store controller RAM disk.

Note: FSET supports -R on the P: drive.

Description

The FSET command lets you:

- View the attributes of a file.
- Change the attributes of a file. You can change certain characteristics of your file without changing the information in it.

You can use global file name characters with FSET to change the attributes of or set the protection on several files at once.

Changing the attributes of a file

Your operating system files can have one or more of the following attributes:

- -H (Hidden file). If you specify -H=ON in your FSET command, that particular file becomes a hidden file. It does not appear in your directory listing unless you use the DIR command with the -H parameter.

If you specify -H=OFF, the file appears in your directory listing. If you do not specify a value for -H, -H=OFF is the default. Note, however, that if your file is also a system file (-S=ON), it does not appear in your directory listing even if -H is off.

- -A (Archive file). If this attribute is off, it indicates that the file has been archived since it was created or last modified. -A is automatically turned on when you modify your file. Programs that back up your files can turn off archiving so they can back up files that have changed since the last backup.
- -S (System file). If you specify -S=ON, your file becomes a system file. A system file can be accessed by any user on the system; the user does not need a copy of the file in their own directory. Applications that are useful to many people on the system make good system files. A system file does not appear in directory listings unless you use the DIR command with the -H parameter.

If you specify `-S=OFF`, your file is not a system file, and it appears in the directory listing. If you do not specify a value for `-S`, `-S=OFF` is the default. Note, however, that if your file is also a hidden file (`-H=ON`), it is not shown in your directory listing even if `-S` is off.

- `-R` (Read-Only file). If you specify `-R=ON`, your file becomes a read-only file. You can only read the file, even if your user ID or group ID gives you other privileges. `-R=OFF` switches off the read-only attribute. If you do not specify a value for `-R`, `-R=OFF` is the default.

Examples

Format 1

Format 1 displays the file attributes of the specified file. The following example displays the file attributes of `PERSONAL.TXT`:

```
A:>FSET PERSONAL.TXT
```

```
A:PERSONAL.TXT      A=ON,0=RWED
```

```
A:>_
```

Format 2

Format 2 (the `-H` parameter) sets the hidden attribute on or off for the file you specify in the command line. Files with the hidden attribute on do not appear in a directory listing unless you use the `-H` parameter of the `DIR` command.

```
A:>FSET BUDGET85 -H=ON
```

```
A:>_
```

Format 3

Format 3 (the `-A` parameter) sets the archive attribute on or off for the file you specify in the command line. In the following example, the archive attribute is set to `ON` for the file `BUDGET85`:

```
A:>FSET BUDGET85 -A=ON
```

```
A:>_
```

Format 4

Format 4 (the `-S` parameter) sets the system attribute on or off for the file you specify in the command line. Files with the system attribute on do not appear in a directory listing unless you use the `-H` parameter of the `DIR` command.

```
A:>FSET BUDGET85 -S=ON
```

```
A:>_
```

Format 5

Format 5 uses the `-R` parameter to set the read-only attribute on or off for the file you specify on the command line. The read-only attribute overrides the access rights that are user and group based. A process cannot delete or write to a read-only file even if it has write and delete privileges for the file.

```
A:>FSET BUDGET85 -R=ON
```

```
A:>_
```

LAN (MCF Network) Format

This example uses format 1 of this command on a LAN system. It displays the file attributes of file FIX.BAT in the root directory of the C drive on store controller CC.

```
C:>FSET ADXLXCCN::C:\\FIX.BAT
```

GOTO Command

Purpose

To transfer control to the line that has the label that matches the one used on the GOTO line.

Format

GOTO *label*

:label

Type

Internal

Description

The GOTO command is a batch command that transfers control to the line that has the label that matches the one that is used on the GOTO line. GOTO lets you override the normal top-to-bottom running of commands. You can use GOTO only in a batch file.

Labels in a GOTO command appear in at least two places: after the word GOTO, and again on a separate line anywhere else in your batch file. The label name next to GOTO is separated from GOTO by a blank space.

Either label name can be in uppercase or lowercase, or a combination of the two cases. The label names are case-sensitive. The system treats the first eight characters in a label name as significant, and any subsequent characters are ignored.

The second label name (the point to which control is passed) must meet the following requirements:

- It must be on a line by itself. Any information after the label is treated as a comment.
- It must begin with a colon (:).
- The colon must be in column one of the new line.

If the label referenced by GOTO does not meet these conditions, the batch file ends with the message, "Label not found."

If you use a label without a corresponding GOTO, the system treats the labeled line as a comment line.

Example

In this batch file, GOTO2.BAT, the control passes from GOTO THREE to :THREE. The system ignores the statements between the GOTO statement and the label.

```
REM The name of this file is GOTO2.BAT.
REM GOTO2.BAT exits normally but skips
REM label :TWO.
REM
:ONE
REM I'm in label ONE.
DIR A:
GOTO THREE
:TWO
```

```
REM I'm in label TWO.  
DIR B:  
REM  
A:CHKDSK  
CLS  
:THREE  
REM I'm in label THREE.  
VER
```

IF Command

Purpose

To test a condition to see if it is true or false.

Formats

- (1) IF EXIST *filespec command*
- (2) IF NOT EXIST *filespec command*
- (3) IF *string1==string2 command*
- (4) IF NOT *string1==string2 command*
- (5) IF ERRORLEVEL *decimal_number command*
- (6) IF NOT ERRORLEVEL *decimal_number command*

Type

Internal

Description

The IF command can be used as a batch command that lets you test a condition to see if it is true or false. If the condition is true, the system runs the command specified at the end of the IF statement. If the condition is false, the system does not run the command.

The command on the IF/IF NOT line must be a system command or batch command. For example, you might want to know if a file is present or absent on disk before you take an action.

Examples

Format 1

Format 1 tests for the presence of a file in the current directory of the current or specified drive. In this case, the filespec is limited to a drive name, a file name, and a file extension; directory paths are prohibited. You can use global file name characters in the filespec. Directory names are not supported with the IF EXIST statement on the M: drive.

An IF EXIST statement might be used in a batch file as follows:

```
IF EXIST TEST.DAT GOTO DOCOPY
ECHO The file TEST.DAT does not exist.
GOTO EXIT
:DOCOPY
REM
REM We want to make a backup copy
REM of the file TEST.DAT.
REM Copy TEST.DAT to TEST.BAK.
REM
COPY TEST.DAT TEST.BAK
REM
REM We have finished the backup so exit.
REM
:EXIT
ECHO We are done.
```

Format 2

Format 2 inverts the logic of format 1. The following is how the backup task shown in format 1 might look using the IF NOT EXIST:

```
IF NOT EXIST TEST.DAT GOTO NOCOPY
REM
REM  We want to make a backup copy
REM  of the file TEST.DAT.
REM  Copy TEST.DAT to TEST.BAK.
REM
COPY TEST.DAT TEST.BAK
REM
REM  We have finished the backup so exit.
REM
GOTO EXIT
:NOCOPY
ECHO The file TEST.DAT does not exist.
:EXIT
ECHO We are done.
```

The preceding examples show that using IF or IF NOT is often a matter of style.

Formats 3 and 4

Formats 3 and 4 compare the two strings before running the command. The strings can be any alphanumeric characters. If the two strings are identical, even to the matter of uppercase and lowercase letters, the system runs the command at the end of the line for format 3, but does not run the command for format 4.

For example, assume that a batch file called APPAREL.BAT contains these lines:

```
ECHO
IF %1==Sweaters ECHO %1 are on sale today.
VOL
```

If your command line looks like this:

```
APPAREL Sweaters
```

then the system substitutes “Sweaters” for %1 in both places and tests to see if the parameter “Sweaters” matches the string on the right side of the double equal signs. If the test string in the batch file does match the string on the command line, the system runs the ECHO command.

This is the complete example:

```
A:>APPAREL Sweaters
```

```
A:>ECHO
ECHO is ON
```

```
A:>ECHO Sweaters are on sale today.
Sweaters are on sale today.
```

```
A:>VOL
Volume in drive A: is BATCHDISK
```

```
A:>_
```

If the two strings are different, the ECHO command is not run. The system runs the VOL command whether the strings match or not.

Formats 5 and 6

Formats 5 and 6 test exit codes in a program that you create and run against a batch file. If the exit code-decimal number in your program is equal to or greater than the decimal number in your batch file, the system runs the specified command. Otherwise, that command is not run and control in your batch file passes to the line following IF ERRORLEVEL.

The ECHO command is often used with the IF and IF NOT statements to inform you of the exit conditions the system encounters while running a program. Consider the following batch file, TESTPROG.BAT:

```
TESTHELP
REM
REM TESTHELP is a program you are running
REM
:CHKERRS
    IF ERRORLEVEL 2 GOTO WRITERR
    IF ERRORLEVEL 1 GOTO READERR
    IF ERRORLEVEL 0 GOTO NORMAL
:WRITERR
    ECHO TESTHELP had a WRITE failure.
    GOTO REALEXIT
:READERR
    ECHO TESTHELP had a READ failure.
    GOTO REALEXIT
:NORMAL
    ECHO TESTHELP finished normally.
    REM It wasn't condition 2 or 1.
    GOTO REALEXIT
:REALEXIT
    REM End of processing.
```

TESTPROG.BAT checks each IF ERRORLEVEL number value for a matching value in your program that is running. Of course, the messages you use with each ERRORLEVEL statement should reflect what is happening in the program.

LIST Command

Purpose

To display a list of the operating system commands that are internal to the system.

Format

LIST

Type

Internal

Description

Some commands are internal to the operating system and some are external. The internal commands are built into the system and are ready for use.

The LIST command lets you list the commands that are built into the system.

Example

The following example lists all of the commands that are built into the system.

```
C:>LIST
```

MKDIR Command

Purpose

To create a subdirectory on the current or specified disk.

Formats

- (1) MKDIR directory_path
- (2) MKDIR drive:directory_path

Type

Internal

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the store controller RAM disk.

Abbreviation

MD

Description

The number of subdirectories you can create is limited only by the available disk space. However, the maximum length of a directory path from the root directory to the level you want to access must be 128 characters or fewer.

Examples

Format 1

In the following example, Format 1 of the MKDIR command creates a new subdirectory called FOREST. FOREST is attached to the root directory.

```
A:>MD /FOREST
```

```
A:>_
```

If you want to create a subdirectory beneath FOREST, use one of the following two examples.

If your current directory path is root/FOREST, use this example:

```
A:>MD TREES
```

```
A:>_
```

The absence of a leading “ / ” tells the operating system to create a subdirectory TREES and add it to the current directory.

The following example also creates TREES and adds it to the path root/FOREST; you can use this command line regardless of your current directory path.

```
A:>MD /FOREST/TREES
```

```
A:>_
```

Note that in the previous example, the first “ / ” tells the operating system to begin its directory search with the root directory.

Format 2

Format 2 is an extension of Format 1. Format 2 creates a directory on a drive other than the current drive. The following example creates a subdirectory that is called PHONE in the root directory on drive B.

```
A:>MD B:/PHONE
```

```
A:>_
```

LAN (MCF Network) Format: MKDIR ADXLXCCN::C:\TEST

This example shows how to create a new subdirectory that is called TEST on the C drive of store controller CC.

MORE Command

Purpose

To display the contents of a file or the output from a command one panel at a time. This command should not be used with other interactive commands.

Formats

Display a file one panel at a time:

(1) MORE < *filespec*

Display command output one panel at a time:

(2) COMMAND | MORE

Type

External

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the store controller RAM disk.

Description

MORE displays the contents of a file or the output from a command one panel at time.

MORE displays a panel of data, and then displays the message:

```
-- More --
```

If you press any character key, another panel of data is written to the screen or file. This process continues until all input data is read. You can stop the MORE command by pressing **Ctrl+Break**.

Note: Some commands that work well with |MORE are the following:

- LIST
- TYPE
- DIR
- TREE
- SORT

Examples

Format 1

In Format 1, MORE uses a file for its input. The following example displays a panel of data from the file TEST.HIS.

```
A:>MORE < TEST.HIS
```

```
HISTORY 170 TEST ANSWERS
-----
```

```
1.
Louisiana Purchase
2.
Jefferson
3.
Monroe
.
```

```

.
.
20.
Lincoln

-- More --

```

Format 2

In Format 2, MORE uses the output from a command as its input. The following example displays a panel of output from the DIR command.

A:>DIR | MORE

```

Volume in drive A: is DISKETTE1
Directory of A:

SALES      JAN    2015   1-15-1985   1:19p
EXPENSES   JAN    1613   1-31-1985   12:14p
SALES      FEB    3256   2-15-1985   9:00a
EXPENSES   FEB    1300   2-21-1985   2:45p
SALES      MAR    8788   3-15-1985   10:21a
EXPENSES   MAR    1525   3-25-1985   12:14p
.
.
.
EXPENSES   OCT    2691   10-29-1985   8:30a
--- MORE ---

```

LAN (MCF Network) Format:

```

MORE<ADXLCCN::C:\\JWTST.BAT
rem this is jw
.
.
.
MORE

```

This example uses format 1 of this command on a LAN system. It displays the contents of JWTST.BAT in the root directory of the C drive on store controller CC.

ORDER Command

Purpose

To define the search order of specified file extensions.

Formats

- (1) ORDER
- (2) ORDER EXT
- (3) ORDER EXT;EXT; ... EXT;

Type

Internal

Type can be used with the store controller RAM disk.

Description

ORDER functions the same way as the PATH command except that it searches for file extensions rather than subdirectories.

ORDER is most often used to search for the file extensions of executable files when the operating system is loaded. These file extensions include 286.

If you specify a file extension that does not exist, the operating system displays an error message when it tries to load the file.

If you specify more than one file extension with ORDER, you must separate file extensions with a semicolon, blank, or tab.

Examples

Format 1

Format 1 displays the search order currently defined to the operating system. In the following example, if you do not specify a file extension with your file, the operating system looks to see if your file has a file extension of EXE.

```
A:>ORDER  
ORDER = EXE
```

```
A:>_
```

Format 2

Format 2 tells the operating system to look for a file with the specified file extension if an extension is not supplied by the user. The following example tells the operating system to try and match the EXE file extension if an extension is not supplied by the user.

```
A:>ORDER EXE
```

```
A:>_
```

Format 3

Format 3 specifies more than one file extension. Separate file extensions with a semicolon, blank, or tab.

PATH Command

Purpose

To define the search path of specified directories for commands or batch files that were not found by searching the current directory.

Formats

- (1) PATH
- (2) PATH *directory_path*
- (3) PATH *directory_path;directory_path...*
- (4) PATH *drive:*
- (5) PATH *drive:directory_path*
- (6) PATH *drive:directory_path;drive: directory_path...*

Type

Internal

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the store controller RAM disk.

Description

PATH defines the search path of specified directories for commands or batch files that were not found by searching the current directory. PATH does not change your current directory.

If the command you entered is not found in any of the directories you specified, you receive an error message.

If you specify a path that no longer exists, the system ignores that path and goes on to the next one, if one exists.

Note: When you specify one or more search paths, you must type the words DEFAULT: and SYSTEM: in front of the first path.

If you specify more than one path with the PATH command, you must separate paths with a blank, semicolon, or tab.

Examples

Format 1

Format 1 of the PATH command displays the search path currently defined to the operating system.

```
C:>PATH
PATH = C:/
A:>_
```

Format 2

Format 2 specifies a search path on the current drive. When you enter a command that is not found in the current directory of the current drive, the operating system searches the directory you specified with PATH. The current directory is not

changed. In the following example, the operating system first looks for a given command in the current directory of drive C, and then in the directory /COMMANDS.

```
C:>PATH DEFAULT: SYSTEM:/COMMANDS/
```

```
C:>_
```

Format 3

Format 3 specifies more than one path on the current drive. In the following example, the current directory is the root directory of drive A, and the BASIC program INTEREST.BAS is in the subdirectory /BASIC. When you run the program INTEREST, the operating system looks for it in the current directory and the subdirectories /ACCOUNTS and /BASIC.

```
A:>PATH DEFAULT: SYSTEM:/ACCOUNTS/;/BASIC/;
```

```
A:>_
```

Format 4

Format 4 searches the root directory on the specified drive.

Format 5

Format 5 lets you specify a search path on a drive other than the current one as follows:

```
B:>PATH C:DEFAULT: SYSTEM:/COMMANDS/
```

```
B:>_
```

Format 6

Format 6 lets you specify more than one search path on the specified drive.

LAN (MCF Network) Format: PATH DEFAULT: SYSTEM: ADXLXCCN::C:\TEST

This example uses the PATH command on a LAN system. It sets up PATH to search for commands or files first on the default drive of the local store controller, then subdirectory TEST on the C drive of store controller CC.

PAUSE Command

Purpose

To suspend batch processing so that you can evaluate the current situation and possibly take an appropriate action.

Format

PAUSE

Type

Internal

Description

The PAUSE command is a batch command that suspends batch processing so that you can evaluate the current situation and possibly take an appropriate action. For example, you might want to change diskettes between commands.

Insert PAUSE commands in your batch file where you want to end or continue batch processing. Each PAUSE command stops your system and gives you time to decide whether to end processing. If you decide to end processing, press **Ctrl-C**. To continue processing, press any other key.

Examples

The following example suspends processing and displays the message, Press any key to continue... If you press any key (except Ctrl+C), the batch file continues to run with the line following PAUSE. If you press Ctrl+C, batch processing ends. For example, suppose that MAYBE1.BAT file contains the following commands:

```
VER
DIR A:*.BAK
PAUSE
DIR A:*.TXT
PAUSE
COPY A:*. * B:
```

The two PAUSE commands let you examine your panel's contents before you proceed or stop.

PRINT Command

Purpose

To:

- Create jobs by sending existing files to be printed
- Work with or alter a job that is presently queued for printing
- Alter the entire print queue

Formats

Use file specifications and printer names to manage a specific device:

(1) `PRINT filespec_1[filespec_2 ... filespec_x-s][-d=device]`

Note: File specifications can be used to schedule existing files for printing or to obtain the status of any jobs that match the file specifications. The `-d` designates a specific queue, and must come at the end of the command string.

An example of a device is a printer with the name PRN1:. If a job has been placed on hold and the system re-IPLs for any reason, the device name will be HOLD.

Use job identifiers (IDs) to manage a specific job:

(2) `PRINT -j=jobid[-a, -c, -h, -p, -s, or -t]`

Note: Job IDs can be used to perform actions on jobs that are already scheduled for printing. They are the only method by which individual files can be worked with or altered. The `-j` designates a specific job ID, and must come at the beginning of the command string.

Use printer names to manage a specific device:

(3) `PRINT [-a, -c, -h, -l, -r, -s, or -t][-d=device]`

Note: Printer names are specified whenever the user wants to cause a command to refer to a printer other than the default printer.

Printer names are of the form `node::printer:`. The node portion is optional, but (if used) the two separating colons must be included. The printer portion can consist of the entire printer name (such as PRN1:) or only the printer number.

The colon following the printer name or number is also optional. To use the default printer, do not specify a printer name or number.

Valid printer numbers are 1 to 8 (if all eight possible printers are installed). The printer name PRN0: or number 0 can be used to indicate the default printer. The `-d` designates a specific device, and must come at the end of the command string.

An example of a device is a printer with the name PRN1:. If a job has been placed on hold and the system re-IPLs for any reason, the device name will be HOLD.

Type

External

Type can be used on a LAN (MCF Network) or non-LAN system.

Description

The PRINT command feature can only be used in Command Mode. PRINT gets all the information needed to execute the request from the command line in all cases, except for the job and queue transfer option -t. For both job and queue transfer requests, you are prompted for the required printer names. For queue transfer requests, you are prompted for both the source queue that is to be redirected and the destination queue that is to receive the jobs. For job transfer requests, you are prompted for the destination queue to which the job is to be routed.

The following are the options that are available for the PRINT command. Some options have multiple meanings, depending on whether they refer to a specific job or an entire queue.

Table 31. PRINT options

PRINT command option	Option name	Meaning
-a	Activate job or queue	Place job back in queue or start queue printing again.
-c	Cancel job or queue	Remove single job or all jobs in queue from system.
-h	Hold job or queue	Place job or queue on hold indefinitely.
-l	Load queue	Restart queue after an IPL.
-p	Priority job	Move job to top of queue.
-r	Resume queue	Resume a queue after a previous transfer.
-s	Job or queue status	List position of job in queue or list all jobs currently in queue.
-t	Transfer job or queue	Remove single job or all jobs from queue and send them to another queue.

PROMPT Command

Purpose

To set the system prompt.

Formats

- (1) PROMPT
- (2) PROMPT *text*

Type

Internal

Description

If you want to change the system prompt from the default setting, use the PROMPT command. PROMPT sets a new system prompt. Text for the system prompt must be 128 characters or fewer.

The first nonblank character after the word PROMPT is the first character of the prompt. The operating system considers all the text on the PROMPT command line to be the new system prompt.

If you enter the PROMPT command by itself, the operating system displays the default prompt. You can include special characters in the text in the form \$c, where c is one of the following:

t	time
d	date
p	current directory path
v	version number
n	current drive letter (a-p)
g	> character
l	< character
b	character
q	= character
h	backspace and erasure of the previous character
e	ESCAPE character
_	go to beginning of new line on the panel

Any other character is treated as a null character. The PROMPT command takes no action on it. See the *4680 BASIC: Language Reference* for the escape sequences that can be used with \$e.

Examples

Format 1

If you enter Format 1, the system returns to the default prompt.

For example if the current prompt contains the date, the current directory path, and the > character (Mon 06-10-1996C:\>) entering PROMPT results in the following prompt:

C>

Format 2

Format 2 changes your default prompt to the specified text. For example, if the current prompt is the default prompt C>, entering PROMPT \$p\$g results in the following prompt:

```
C:>
```

If the current prompt contains the current directory path and the > character, entering prompt \$p\$g\$ev results in the following prompt:

```
C:>
```

and now causes display output to go to the next line when end of line is reached.

The use of logical names affects the prompt differently. For example, if the current prompt contains the current directory path and the > character, entering ADX_SPGM: results in the following prompt:

```
ADX_SPGM:>
```

and changes the subdirectory to ADX_SPGM.

QUERY Command

Purpose

To return an error level value.

Format

(1) QUERY *option string*

Type

Internal

Description

The QUERY command returns an error-level value. Use QUERY with the IF ERRORLEVEL batch command. Table 32 lists the options available for the QUERY command, and gives a description of these options.

Table 32. QUERY Command Options

Option	Description
Y-N "Continue?"	Prompts for a Yes or No response (Y=1 and N=0 for the error level). This prompt is not case sensitive.
0-9 "Enter a number from"	Prompts for a value from 0 to 9. The range can be any value as long as the left value is less than the right. You can enter any single digit in the range.
a-z "Enter"	Prompts for a single alphabetic character in an expressed range. The left character in the range must be less than the right character. You can enter any single character in the range. This prompt is not case sensitive.

Examples

QUERY *option string*

The QUERY command example shows a query of a batch file for the device type to load.

```
ECHO OFF
ECHO Which directory would you like to list:
ECHO 1 - Root directory of C: drive
ECHO 2 - Root directory of D: drive
ECHO 3 - Root directory of A: drive
QUERY 1-3 "Select a number from"
IF ERRORLEVEL 3 DIR A:/
IF ERRORLEVEL 2 DIR D:/
IF ERRORLEVEL 1 DIR C:/
```

REM Command

Purpose

To let you put remarks in your batch file.

Formats

- (1) REM
- (2) REM *remark*

Type

Internal

Description

The REM command is a batch command that lets you put remarks in your batch file. These remarks appear on the panel when they are encountered in running batch files. Processing is not interrupted by a REM command; the remark appears and processing continues on the line that follows the REM command. REM lines do not appear when ECHO is OFF.

Examples

Format 1

Format 1 of the REM command inserts a blank line in your batch file. Careful use of blank lines makes batch files easy to read because of the visual (and hopefully logical) separation of one block of text from another. This example uses the REM command to separate the COPY commands from the ERASE commands.

```
REM
COPY A:*.BAK C:
COPY B:*.BAK C:
REM
REM
ERASE A:*.BAK
ERASE B:*.BAK
REM
DIR C:*.BAK
```

Format 2

Format 2 lets you put remarks in your batch file. A remark is a string up to 123 characters. Adding text to your REM line keeps you informed of actions your batch file took or is about to take. For example, suppose that ENDOFDAY.BAT contains the following commands:

```
REM      copy *.BAK files to drive C
COPY A:*.BAK C:
COPY B:*.BAK C:
REM
REM      Erase *.BAK files on A and B
REM
ERASE A:*.BAK
ERASE B:*.BAK
REM      Show current set of *.BAK files
REM      on drive C
DIR C:*.BAK
```

The meaning of each portion of text becomes clear. Using remarks in your batch files makes them easier to read and to work with.

RENAME Command

Purpose

To change the filespec of a file.

Format 1

RENAME *old_filespec new_filespec*

Format 2

RENAME *-D old_dirspec new_dirspec*

Type

External

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the store controller RAM disk.

Abbreviation

REN

Description

RENAME does not change the information in the file.

If you do not specify a directory path with *new_filespec*, the file stays in the same directory after it is renamed.

If *old_filespec* and *new_filespec* have different directory paths, the file is renamed and moved to the second directory path. The second directory path must already exist.

You cannot rename a file to a different drive.

You can use global file name characters in both filespecs.

On the VFS drive, you cannot rename a file to the same name with a different case structure. For example, the following command is not allowed:

```
m:>rename myclient.class MyClient.class
```

You must rename the file to a temporary name, then rename the file from the temporary name to the new name.

Note: For a VFS drive or CD-RW/DVD drives which use the UDF File System, the operating system does not support any asterisk (*) wildcards. For example, the following rename command is not allowed on a VFS drive.

```
m:>rename myApp* yourApp*
```

Likewise, the following rename command is not allowed on a UDFFS (UDF File System) drive.

```
p:>rename myApp* yourApp*
```

Examples

In the following example, Format 1 renames IN.TXT to OUT.TXT:

```
A:>RENAME IN.TXT OUT.TXT
A:>_
```

In the following example, the file REVIEW.Q1 on drive C: is renamed REVIEW.Q2. The global file name character tells the operating system not to change the prefix portion of the file name.

```
A:>REN C:REVIEW.Q1 C:*.Q2
A:>_
```

The following example renames the file LIST.TXT to LISTLESS.TXT and moves it to a different subdirectory.

```
C:>REN /WORK/LIST.TXT /LEISURE/LISTLESS.TXT
C:>_
```

The following example uses Format 2 to rename directory WORK to WORKDOCS.

```
C:>REN -D WORK WORKDOCS
```

LAN (MCF Network) Format:

The example below renames the file LOST.DAT on store controller CC to FOUND.DAT.

```
C:>REN ADXLCCN::C:\LOST.DAT ADXLCCN::C:\FOUND.DAT
```

RESTORE Command

Purpose

To restore one or more files that have been backed up from a source disk.

Note: This command is not supported on the DVD-ROM drive.

Formats

- (1) RESTORE *backup_diskette*:
- (2) RESTORE *backup_diskette*: *destination*:
- (3) RESTORE *backup_diskette*: *destination*: *filespec*
- (4) RESTORE *backup_diskette*: *destination*: *filespec* *-parameter*

Type

External

Type can be used with the store controller RAM disk.

If you are restoring to a system that did not previously have the operating system installed, you must IPL from the Supplemental Option using the CD-ROM. Then, run CPREP to prepare the hard disk drive before you perform the restore.

Note: This command cannot be used to restore the entire hard disk drive when the store system has been IPLed from hard disk drive. If you want to restore a single file or a group of files, you can do so while on line but the files must not be in use when you restore them. If you get an access error or you want to restore the entire hard disk drive, you must IPL from the Supplemental Option using the CD-ROM in order to perform the restore.

Description

RESTORE puts files back on the disk after they have been backed up. For instructions on how to back up files, see the BACKUP command in this section.

In the formats shown above *backup_diskette*: is the letter of the drive from which the system gets the files to be restored and *destination*: is the drive to which the files are to be written. *filespec*: is the file name, including the path (if required) of the files to be restored.

If you do not specify a path for the *destination*:, the files that are backed up from the directory whose name is the same as the current directory are restored to the current directory. Files can only be restored to a directory of the same name as that from which they were copied using BACKUP. Table 33 summarizes the parameters that you can use with the RESTORE command. You can use both parameters on the same command line.

Table 33. RESTORE Parameters

Parameter	Description
-P	Causes RESTORE to prompt you before restoring files that have changed since the last backup or files marked read-only.
-S	Restores files in the specified subdirectory and in all the subdirectories on the disk. This includes subdirectories at all levels beyond the specified directory.

Notes:

1. If the RESTORE command fails on one diskette in a series of diskettes that are being restored (for example, if the drive latch was not closed), re-issue the RESTORE command using the -P option. The -P option allows you to bypass the files which were previously restored successfully and to continue restoring where the previous attempt failed.
2. To restore a file to a directory whose name is different from that of the directory from which the file was backed up, restore the file to the original directory, and then use the RENAME command (specifying the directories) to switch the file to the new destination directory. The switch only changes the directory entries; it does not recopy the file.

When RESTORE prompts you to insert the backup disk, make sure that you insert the first disk that might contain the file you want to restore. If you are not sure, insert backup disk number 1. If the file is not on the backup disk you inserted, RESTORE prompts you to insert the next backup disk.

If you specify a global file name character in the file name, all of the files that match that file name are restored. After RESTORE restores all the files that match the global file name character name, RESTORE prompts you to insert the next backup disk.

Note: Backup files created with the BACKUP command on 4680 systems cannot be restored using the RESTORE command on the operating system. If you need to back up files from 4680 systems, copy the file ADX_SPGM:RESTORE.286 from the 4680 Operating System onto a diskette. Then, use this file to restore the files to the operating system.

Examples

Format 1

Format 1 does not specify the destination disk; therefore, the destination disk must be in the current drive. The following example restores all files on the diskette in drive A to the current directory of the disk in drive C. In this case, that is the root directory of drive C.

C:>RESTORE A:

Insert the backup diskette 01 in drive A:
Press any key when you are ready

S

*** Files were backed up 09-01-85 ***

*** Restoring files from disk 01 ***

/SALES.AUG
/INCOME.AUG
/EXPENSES.AUG

C:>_

Format 2

Format 2 restores all the files in the current directory of the backup disk to the current directory of the destination disk. The following example restores all files on the diskette in drive B to the hard disk drive C.

C:>RESTORE B: C:

Insert the backup diskette 01 in drive B:

```
Press any key when you are ready

*** Files were backed up 09-01-85 ***

*** Restoring files from disk 01 ***
/QUARTER4.REV
/STRATEGY.QT1
/FORECAST.QT1

C:>_
```

Format 3

Format 3 restores one or more specific files from backup to the destination disk. The following example restores LETTER.TXT to the directory CUSTOMER on drive C.

```
C:>RESTORE A: /CUSTOMER/LETTER.TXT

Insert the backup diskette 01 in drive A:
Press any key when you are ready
*** Files were backed up 09-10-85 ***

*** Restoring files from disk 01 ***
/CUSTOMER/LETTER.TXT

C:>_
```

Note that it is not necessary to specify drive C because drive C is the current drive.

The following example restores each file from backup with a file extension of MEM that was backed up from the current directory.

```
C:>RESTORE A: C:*.MEM

Insert the backup diskette 01 in drive A:
Press any key when you are ready

*** Files were backed up 09-15-85 ***

*** Restoring files from disk 01 ***
/SCHEDULE.MEM
/REVIEW.MEM
/MEETING.MEM
/CHANGE.MEM

Insert the backup diskette 02 in drive A:
Press any key when you are ready

*** Files were backed up 09-15-85 ***

*** Restoring files from disk 02 ***
/REVIEW2.MEM
/MEETING2.MEM
/CHANGE2.MEM
/VACATION.MEM

C:>_
```

Format 4

Format 4 uses one or both of the RESTORE parameters. The -S parameter restores backed up files to their original directories on the destination disk. If a directory no longer exists on the destination disk, restoring files to it recreates the directory.

The -P parameter prompts you before restoring Read-Only files and files that have changed since the last backup. Operating System files are marked Read-Only when they are created by the FORMAT command. If you want to restore a diskette that has these files on it, use -P.

RMDIR Command

Purpose

To remove a subdirectory from the current or specified disk.

Formats

- (1) RMDIR *directory_path*
- (2) RMDIR drive:*directory_path*

Type

Internal

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the store controller RAM disk.

Abbreviation

RD

Description

You cannot remove a directory that has files in it, nor can you remove the root or current directories. RMDIR removes the last directory name in the specified path.

If you try to remove a directory that contains files, you receive an error message.

Examples

Format 1

In the following example, Format 1 removes the directory TREES from the directory path root/FOREST/TREES.

```
A:>RD /FOREST/TREES
```

```
A:>_
```

Format 2

Format 2 lets you remove a directory that is not on your current drive. Format 2 is identical to Format 1 except that you specify a drive letter with your command.

LAN (MCF Network) Format: RMDIR ADXLXCCN::C:\TEST

This example removes a subdirectory from drive C of store controller CC.

SECURITY Command

Purpose

To view or change current file access privileges.

Formats

View your current file access privileges:

(1) SECURITY

Change your current file access privileges:

(2) SECURITY -O=[R][W][E][D]

(3) SECURITY -G=[R][W][E][D]

(4) SECURITY -W=[R][W][E][D]

Type

Internal

Description

The file access privileges that SECURITY lets you view or change are read (R), write (W), execute (E), and delete (D). You can change the level for all three user classes: owner (-O), group (-G), and world (-W). See the FSET command for an explanation of file access privileges and user classes.

You can combine formats 2 - 4, above, to change the file access privileges for any combination of user classes—user, group, and world—on a single command line.

Privileges set with SECURITY are in effect for the duration of a terminal session. You can invoke SECURITY from the command line.

Example

The following command sets the file access privileges for group and world users to read access.

```
A:>SECURITY -G=R -W=R
```

SHIFT Command

Purpose

To let you use more than ten parameters on one command line.

Format

SHIFT

Type

Internal

Description

The SHIFT command is a batch command that lets you use more than ten parameters on one command line. SHIFT does this by shifting all the words on a command line one word to the left. SHIFT can only be used with batch files.

Remember that you can only put ten placeholders in a batch file (%0 through %9). However, SHIFT lets you use different parameters for the same placeholders in a given batch file.

Because placeholders in a batch file depend on a parameter's position on a command line for placeholder substitution, SHIFT changes the relationship of the parameters to the batch file placeholders in the following way:

1. The parameter that is leftmost on the line before SHIFT was run is no longer available to the running batch file after a SHIFT is run.
2. The remaining command line parameters decrement their ordinal values by one; so that the eleventh parameter becomes the tenth, the tenth becomes the ninth, and so on.

Compare the relationship of the command line parameters to the batch file placeholders in the following example:

```
A:>SHIFT1 FIRST SECOND THIRD
```

FIRST, SECOND, and THIRD are the first, second, and third parameters to be used with SHIFT1.BAT, which follows:

```
ECHO %0 %1 %2
SHIFT
ECHO %0 %1 %2
SHIFT
ECHO %0 %1 %2
```

The batch file has three placeholders: %0, %1, and %2. Consequently, SHIFT1 substitutes for %0, FIRST for %1, SECOND for %2, and THIRD for %3 before any shifting occurs.

When you run SHIFT1 by using the following command line, this output appears on the screen:

```
A:>SHIFT1 FIRST SECOND THIRD
```

```
A:>ECHO SHIFT1 FIRST SECOND
SHIFT1 FIRST SECOND
```

```
A:>ECHO FIRST SECOND THIRD
FIRST SECOND THIRD
```

```
A:>ECHO SECOND THIRD  
SECOND THIRD
```

```
A:>_
```

Each leftmost parameter from the original command line disappears after each SHIFT execution. Notice also how the values for %0, %1, and %2 change after each SHIFT: FIRST substitutes for %0, SECOND for %1, and THIRD for %2 after the first SHIFT. Finally, notice that the parameter THIRD was not available to the running batch file until the parameter list had shrunk by one word — that is, after the first SHIFT.

By structuring your command line parameters and carefully synchronizing them with the placeholders in your batch file, SHIFT lets you use as many parameters as you can fit on a command line.

SORT Command

Purpose

To rearrange the input from a file or the output from a command.

Formats

Sort the input from a file:

- (1) SORT <filespec
- (2) SORT -parameter < filespec

Sort the output from a command:

- (3) COMMAND | SORT
- (4) COMMAND | SORT -parameter

Type

External Can be used on a LAN (MCF Network) or non-LAN system.

Description

The SORT filter rearranges the input from a file or the output from a command. SORT does not change the contents of the file; SORT changes the way the information appears.

Data is sorted using the ASCII collating sequence. See the ASCII table in the *4680 BASIC: Language Reference* for information on collating sequence. Table 34 summarizes the parameters you can use with SORT.

Table 34. Sort Parameters

Parameter	Description
-R	Sorts lines in the reverse order of the ASCII collating sequence. Z comes before Y, which comes before X, and so on.
++N	Sorts lines into the ASCII collating sequence based on the character in column N. N must be a whole number.

Note: To pause and restart the display of sorted information, press **Ctrl+S** or **Ctrl+Q**. Ctrl+S pauses the display, while Ctrl+Q restarts the display.

Examples

Format 1

Format 1 sorts the input from a text file into the ASCII collating sequence and displays it on your screen. Lines are sorted beginning with the character in column 1.

The following example sorts a guest list. Notice that the heading "Guest List," the line under the heading, and the blank line in the file are also sorted.

```
David Doe
Phillip Johnson
Michele
Smith
```

Denise
Smith

The following command sorts the list into alphabetical order by the first name of each guest:

A:>SORT < GUEST.LST

David Doe

Denise
Smith

Michele
Smith

Phillip Johnson

A:>_

Format 2

Format 2 uses one or both of the SORT parameters listed in the preceding table. The guest list was prepared so that the last name of each guest begins in column 10. The following example uses the +N parameter to sort the list alphabetically by last name.

A:>SORT -+10 < GUEST.LST

David Doe

Phillip Johnson

Denise
Smith

Michele
Smith

Guest List **A:>_**

Format 3

Format 3 sorts the output of a command. The following example sorts the root directory of drive A. The files are listed alphabetically, and the remaining three lines, "Directory of...", "Volume in drive...", and "3 Files..." are also arranged in the ASCII collating sequence.

A:>DIR

FOG	TXT	64	2-01-1985	2:53a
SAND	TXT	3768	6-18-1985	12:17p
BEACH	LST	81	3-01-1985	3:08p

Volume in drive f0:\ has no label
Directory of f0:

3 Files 241947 bytes free

The following is an example of Format 3.

A:>DIR | SORT

```
          5 Files      241947 bytes free
Directory of f0:
Volume in drive f0: has no label
%P00900      0  6-10-1985   4:30p
BEACH      LST      81  3-01-1985   3:08p
FOG        TXT      64  2-01-1985   2:53a
SAND       TXT     3768  6-18-1985  12:17p
```

A:>_

Format 4

Format 4 uses one or both of the SORT parameters. The next example does a reverse sort of the directory that is shown above.

A:>DIR | SORT-R

```
%P00900      0  6-10-1985   4:30p
SAND      TXT     3768  6-18-1985  12:17p
FOG        TXT      64  2-01-1985   2:53a
BEACH     LST      81  3-01-1985   3:08p
Directory of f0:
Volume in drive f0: has no label
          5 Files      241947 bytes free
```

A:>_

The next example sorts the lines according to the character at column 10 (the column the file extension starts in), and sends the output to the screen. The directory is now sorted alphabetically by file extension.

A:>DIR | SORT-+10

```
          5 Files      241947 bytes free
%P00900      0  6-10-1985   4:30p
BEACH      LST      81  3-01-1985   3:08p
FOG        TXT      64  2-01-1985   2:53a
SAND       TXT     3768  6-18-1985  12:17p
Volume in drive f0: has no label
Directory of f0:\
```

A:>_

The next example combines the -R and -+N parameters to do a reverse listing of the characters in the specified column number. The following example sorts the directory by file extension in reverse order.

A:>DIR | SORT -R-+10

```
Directory of f0:\
Volume in drive f0: has no label
%P00900      0  6-10-1985   4:30p
SAND      TXT     3768  6-18-1985  12:17p
FOG        TXT      64  2-01-1985   2:53a
BEACH     LST      81  3-01-1985   3:08p
          5 Files      241947 bytes free
```

A:>_

The following example uses redirection symbols and a filter together to perform a more sophisticated task.

PARTTIME.LST is a list of part-time employees' names. FULLTIME.LST is a list of full-time employees's names. The following example sorts PARTTIME.LST

alphabetically, and adds it to the end of FULLTIME.LST. FULLTIME.LST now contains a list of all the employee's names.

A:>SORT < PARTTIME.LST >> FULLTIME.LST

A:>_

LAN (MCF Network) Format: SORT < ADXLXCCN::C:\FIX.BAT

This example uses format 1 of this command on a LAN system. The records contained in file FIX.BAT in the root directory of drive C on store controller CC are sorted in ascending alphanumeric order, and then displayed.

TREE Command

Purpose

To display all the directory paths on the current or specified drive. TREE also gives you the parameter of listing the files in each subdirectory.

Formats

- (1) TREE
- (2) TREE *drive*:
- (3) TREE -F
- (4) TREE *drive*: -F

Type

External

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the store controller RAM disk.

Description

The TREE command displays all of the directory paths on the current drive or a specified drive. The command also lists the files in each subdirectory.

Examples

Format 1

Format 1 displays the full path of each directory name on the current drive along with the names of any directories defined within it.

```
A:>TREE
```

```
Directory Path Listing.  
The volume is INVEST
```

```
Path: f0:/
```

```
Subdirectories: STOCKS  
              BONDS  
              COMMODITIES
```

```
Path: f0:/STOCKS/
```

```
Subdirectories: BLUECHIP
```

```
Path: f0:/STOCKS/BLUECHIP/
```

```
Subdirectories: None
```

```
Path: f0:/BONDS/
```

```
Subdirectories: None
```

```
Path: f0:/COMMODITIES/
```

```
Subdirectories: None
```

```
A:>_
```

Format 2

Format 2 is identical to Format 1 except that you specify the drive:

```
A:>TREE f0:
```

Format 3

Format 3 uses the -F parameter to list all the subdirectories on the current drive and to list the files in each subdirectory. In the following example, there is one subdirectory called MYTEST on drive A. MYTEST contains the files GEORGE, THOMAS, and ABRAHAM.

```
A:>TREE -F
```

```
Directory Path Listing.  
The volume is NAMES
```

```
Path: f0:/
```

```
Subdirectories: MYTEST
```

```
Files           : NAMES  
                  DATES  
                  SCORES
```

```
Path: f0:/MYTEST/
```

```
Subdirectories: None
```

```
Files           : GEORGE  
                  THOMAS  
                  ABRAHAM
```

```
A:>_
```

Format 4

Format 4 lists all the subdirectories and files on the specified drive.

LAN (MCF Network) Format: TREE ADXLXCCN::C: > TREE.LST

This example uses Format 2 of this command on a LAN system. It redirects a list of the subdirectories that are contained on drive C of store controller CC to file TREE.LST on the default drive of the local store controller.

TYPE Command

Purpose

To display the contents of a specified text file on the screen. TYPE also enables you to print the contents of a file.

Format

TYPE *filespec*

Type

External

Type can be used on a LAN (MCF Network) or on a non-LAN system.

Description

You must enter the file name and the file extension, if there is one. You cannot specify global file name characters with TYPE. If you do, you receive an error message.

Text files appear in a legible format. Other files, such as object program files, might be unreadable because they contain special characters. TYPE is intended for text files.

Note: The TYPE command interprets line feed characters literally. Therefore, if a line contains only a line feed as opposed to a carriage return/line feed, the line might not be viewable.

Examples

In the following example, TYPE displays the contents of FUNCTION.LST. The file is in the current directory of drive B.

```
A:>TYPE B:FUNCTION.LST
```

```
This month I will attend the following functions:
```

```
November 11 - Junior League  
November 15 - 4H Club  
November 28 - Rotary Club
```

```
A:>_
```

By adding a parameter to the specification, you can use TYPE to print the contents of a file. The example below shows how to print the contents of FUNCTION.LST.

```
A:>TYPE B:FUNCTION.LST >PRN
```

LAN (MCF Network) Format: TYPE ADXLXCCN::C:\JWTST.BAT

This example displays the contents of file JWTST.BAT (in the root directory of drive C on store controller CC) at the local store controller.

UNLOCKP Command

Note: This command is not required in Enhanced Mode.

Purpose

When an attempt is made to write to a DVD media, the optical unit's tray locks. You run UNLOCKP to commit the data that resides in the controller's memory and unlock the unit's tray

Formats

UNLOCKP

Type

External

Description

Write all data from the buffer to the DVD media.

This command only affects DVD media formatted with UDF File System, It takes no arguments.

Examples

Format 1

In the following example, UNLOCKP commit the data that resides in the controller memory to the media and unlock the unit's tray.

```
P:\>UNLOCKP
```

```
Unlock:
```

```
The utility completed successfully.
```

VER Command

Purpose

To display the version number of the base FlexOS operating system.

Format

VER

Type

External

Description

The VER information is set automatically in the operating system.

Example

The following example shows the message that is printed on the screen after executing the VER command.

```
C:>VER
```

```
4690 OS V5 FlexOS-386 Version 2.3
```

Note: This message will vary depending on the version of the operating system that is installed on your store controller.

VERIFY Command

Purpose

To verify that your files are written correctly to a disk.

Formats

- (1) VERIFY
- (2) VERIFY ON
- (3) VERIFY OFF

Type

External

Description

The VERIFY command ensures that files are written correctly to a disk. For example, you can use this command to ensure that data is not written to a bad sector.

Note: To perform a disk verify, that a disk driver must be installed using DVRLOAD with the V option.

Examples

Format 1

Displays whether the VERIFY command is turned on or off.

```
C:>VERIFY
```

```
VERIFY is on
```

Format 2

Format 2 specifies that VERIFY is on and write operations are done correctly.

```
C:>VERIFY ON
```

Format 3

Format 3 disables the VERIFY command.

```
C:>VERIFY OFF
```

VOL Command

Purpose

To display the volume label of a disk in the current or specified drive.

Formats

- (1) VOL
- (2) VOL *drive*:

Type

External

Type can be used on a LAN (MCF Network) or on a non-LAN system. It can be used with the store controller RAM disk.

Description

You can write a volume label on your disk using either the DISKSET or FORMAT commands.

Examples

Format 1

Format 1 displays the volume label of the disk in the current drive. In the following example, the current drive is drive A.

```
A:>VOL
```

```
Volume in drive f0: is RECORDS
```

```
A:>_
```

If there is no volume label on the disk, VOL displays:

```
A:>VOL
```

```
Volume in drive f0: has no label
```

```
A:>_
```

Format 2

Format 2 is identical to Format 1 except that you specify the drive letter.

LAN (MCF Network) Format: VOL ADXLXCCN::A:

This example displays the volume label of the diskette in drive A on remote store controller CC.

Creating an image file of a diskette

From Command Mode, you can use the ADXNDSKL command to create an image file of a diskette and store it on a hard drive or to use a diskette image file to build the diskette in the target drive. The source or target drive must be A:, and the other must be a file specification on a drive large enough to hold a diskette image file. If the source is A:, then the command creates an image file of the diskette and stores

it in on the target drive in the specified file. If the target is A:, then the source must be a diskette image file that is then used to build the diskette in the target drive.

Note: When creating a diskette from an image file, the target diskette must be the same size and type as the diskette that was used to create the image file. This utility cannot be used to transfer data from one size or type of diskette to another.

The following example creates an image file of the diskette and stores it in a file that is called image.dat:

```
C:> ADXNDSKL A: image.dat
```

This example uses the image.dat file to build the diskette in the A: drive:

```
C:>ADXNDSKL image.dat A:
```

Burning an ISO image on a CD or memory key

Purpose

From Command Mode on an Enhanced controller, you can use the **BURNISO** command to burn an ISO image stored on the hard drive of the Enhanced controller to either a CD or a memory key.

Formats

BURNISO *SourceISOName DestinationMedia*

- (1) BURNISO *f:/example.iso* CD
- (2) BURNISO SUPPS CD
- (3) BURNISO SUPPS MemKey

Description

From Command Mode on an Enhanced controller, the **BURNISO** command can be used to burn an ISO image stored on the hard drive of the Enhanced controller to either a CD or a memory key.

The source ISO image either must be located on the F: drive or must be the 4690 OS Enhanced Supplementals ISO image. When the source of the ISO image is located on the F: drive, the fully qualified path name must be used to define that ISO image source. When the source of the ISO image is the 4690 OS Enhanced Supplementals, use SUPPS in place of the *SourceISOName* variable.

Note: The only ISO image that you can burn to a memory key is the Supplementals image (SUPPS). CD media can be used for burning both Supplementals (SUPPS) and other ISO images.

The destination of the ISO image must be either a CD or a memory key. To denote a CD, use CD in place of the *DestinationMedia* variable. To denote a memory key, use MemKey in place of the *DestinationMedia* variable.

Notes:

1. If the destination media is a memory key, only a portion of the memory key is used to burn the Supplementals image. The remainder of the memory key that is not needed for the Supplementals image remains available for regular usage and is formatted as FAT32. Manipulating files in this FAT32 portion does not

change or damage the Supplementals portion. If the memory key is reformatted after burning the Supplementals image to it, however, the Supplementals image is lost.

2. If the destination media is a CD, it must be a CD-RW or a blank CD-R. If the media is not capable of being blanked or if the media has been damaged, a message explaining the issue is written to the log file; try again with another CD.

All output messages will appear both on the command window and in the log file for this command, which is located on the 4690 OS Enhanced controller F: drive and is named burniso.log. Each new execution of the **BURNISO** command creates a fresh F:\burniso.log file, so that everything in that log file represents the data from the most recent execution of the **BURNISO** command.

The following is a list of some error messages that you might see:

Too few parameters

Missing parameter - Provide these parameters: **f:/path/
imagename.iso|SUPPS CD|MemKey**.

Too many parameters

Invalid number of parameters - Provide these parameters:
f:/path/imagename.iso|SUPPS CD|MemKey.

Invalid parameters

Invalid parameter: [Invalid source ISO name or Invalid Destination] -
Provide these parameters: **f:/path/imagename.iso|SUPPS CD|MemKey**.

Source ISO Image is not present on the F: drive

Invalid parameter: Image could not be found - ISO image burning did not complete.

A CD drive capable of burning CDs could not be found on this controller

No writable CD drive was located - ISO image burning did not complete.

No CD was found in the CD drive

No disc in drive - ISO image burning did not complete.

CD was not blank and was not capable of being blanked

Disc is not blank - ISO image burning did not complete.

CD-RW failed the attempt to blank

Failed to reformat disc - ISO image burning did not complete. Note that this may also sometimes appear if no CD is in the drive.

Media other than a CD was in the drive (such as a DVD)

Disc media not supported - ISO image burning did not complete.

No memory key was found attached to this controller

No memory key was located - ISO image burning did not complete.

Only Supplementals (SUPPS) can be burned to the memory key

Only 4690OS Supplementals (SUPPS) allowed for memory key - ISO image burning did not complete.

Examples

Format 1

This will burn the source ISO image f:/example.iso to a CD:

```
C:>BURNISO f:/example.iso CD
```

The resulting message:

Success: The ISO Image was burned to the CD

Format 2

This will burn the source 4690 OS Enhanced Supplementals to a CD:

C:>**BURNISO SUPPS CD**

The resulting message:

Success: The ISO Image was burned to the CD

Format 3

This will burn the source 4690 OS Enhanced Supplementals to a memory key:

C:>**BURNISO SUPPS MemKey**

The resulting message:

Success: The ISO Image was burned to the Memory Key

Chapter 8. Creating and editing files

This chapter shows you how to use the text editing programs, DR EDIX and XE Editor. These programs let you create memos, letters, brief reports, or computer programs. The printed page that you get using these text editors looks exactly as it appears on your panel.

Unlike some text-editing programs, DR EDIX and XE Editor do not read from and write to files automatically. Everything you type is stored in a block of memory called a *buffer* until you tell the text editors to do something with it. The beginning and end of the buffer are equivalent to the beginning and end of your file.

Using DR EDIX

To use DR EDIX, you must be in Command Mode. When you select Command Mode from the menu, a system prompt appears, and you can enter DR EDIX commands directly. See “Starting DR EDIX” to begin entering DR EDIX commands.

Most DR EDIX commands are invoked by pressing the **Ctrl** key and one or two letter keys in combination. In this guide, the **Ctrl** key is indicated by a caret (^). (Letters corresponding to letter keys are printed in bold text.) These key combinations are used to invoke a DR EDIX command. For example, the characters **^G** mean to press the **Ctrl** key and the **G** key, while **^Z** tells you to press the **Ctrl** key and the **Z** key.

DR EDIX has an online help facility that allows you to get helpful instructions whenever you need them. At any time, you can press **^J** to find out what you should do and what commands are available to you.

For example, when you are editing text, press **^J** to get a summary of all the DR EDIX commands. After you choose a command and a prompt appears, press **^J** to get specific instructions on what to do next. Press the **Space bar** to return to your session.

Whenever you are unsure what to do next, press **^J**.

Attention: When using DR EDIX with the 3151 Display Station, you should set the Forcing Insert option for the 3151 to LINE to enhance usability. See the *3151 ASCII Display Station Reference Manual* for details of this parameter.

Note: This program follows the 8.3 naming convention and saves file names in uppercase characters. It is only recommended for use on the C: or D: drives. There are limitations when DR EDIX is used on other drives and file systems.

Starting DR EDIX

To start DR EDIX, you must be in Command Mode. From the SYSTEM MAIN MENU, select the Command Mode option by typing **7** and pressing **Enter**.

When the prompt appears, you can begin entering commands from the command line.

Entering DR EDIX commands

To enter DR EDIX commands, at the system prompt type this command, and press **Enter**:

```
C:>DREDIX
```

Your panel is blank except for the cursor and the DR EDIX status area at the top of the panel. (The status area is discussed in “Status area” on page 271.) DR EDIX is ready for your input.

Entering text

To begin your first document, just start typing. Press any of the alphanumeric keys on the keyboard and a character appears on the panel in the current buffer. The blinking cursor indicates where the next character you enter will appear.

When you enter text, you can also use the Tab and Enter keys. The Tab key puts the cursor at the next tab stop. Tab stops are initially set eight spaces apart in columns 9, 17, 25, and so on.

To end a line, press **Enter**. This action inserts a carriage return at the end of your line. The cursor moves to the start of the next line.

Instead of typing a character, you can issue a DR EDIX command.

Inserting and overtyping

DR EDIX has two modes of entering text: insert and overwrite. Insert mode is the default when you start DR EDIX. You can alternate between insert and overwrite modes by pressing **^V**.

In insert mode, the character you type is inserted at the cursor position and the cursor moves to the right. The rest of the characters on the line also move to the right.

In overwrite mode, the character you type replaces the character at the cursor position and the cursor moves to the right. The rest of the characters on the line do not move. When you are in overwrite mode, the status line at the top of your panel displays the letter O in reverse video.

Moving the cursor

The space you have to enter text is much larger than the size of your panel. You can move the cursor anywhere within the buffer, and DR EDIX scrolls the text left, right, up, or down so that you can see the part of the buffer that contains the cursor.

Table 35 summarizes the cursor control keys. Most of the keys that position the cursor have arrows on them and are on the keypad on the right side of the keyboard. The function keys on the keyboard are programmed with the most frequently used cursor commands.

Table 35. DR EDIX cursor control keys

To move the cursor	Use
Left one space	^S
Right one space	^D
Left one word (or tab stop)	^A
Right one word (or tab stop)	^F

Table 35. DR EDIX cursor control keys (continued)

To move the cursor	Use
To the beginning of the line	^QS
To the end of the line	^QD
Up one line	^E
Down one line	^X
Up one page	^R
Down one page	^C
To the top of the buffer	^QR
To the bottom of the buffer	^QC
To the top of the window (see “Working with windows and buffers” on page 272)	^QE
To the bottom of the window (see “Working with windows and buffers” on page 272)	^QX
To the left and erase character (backspace)	^H
Go to line (prompts)	^ZG

Editing text

Most of the DR EDIX editing commands are invoked by pressing a combination of the **Ctrl** key and another key. The editing commands let you delete characters or add, split, recall, or rejoin lines.

Deleting characters

You can delete individual characters by using **^G** or the **Backspace** key.

^G deletes the character at the cursor.

The **Backspace** key moves the cursor one position to the left and deletes the character there. It is most useful when you notice an error as soon as you type it; you can backspace over the incorrect character and type another one. If the cursor is already under the first character of the line, the **Backspace** key deletes the carriage return of the previous line.

Deleting lines

^Y deletes the current line. You can recall a deleted line with **^ZU**, the UNDO command.

Delete to end-of-line

^QY deletes the current line from the cursor to the end of the line. You cannot recall characters deleted with this command.

Recalling lines

^ZU recalls the last line you deleted. If you delete two lines, press **^ZU** twice to recall both of them. The lines reappear in the opposite order that you deleted them. DR EDIX remembers the last 50 nonblank lines you delete.

Splitting lines

To split a line of text, move the cursor to where you want to split the line, and press **Enter**. The text to the right of the cursor moves to the next line. Any lines below the split line move down too.

Joining lines

To join two consecutive lines, put the cursor at the end of the upper line and press **^G**, or put it at the beginning of the lower line and press **Backspace**. DR EDIX joins the two lines and puts a blank space between the two pieces. If you do not want the extra blank, press **^G** again.

Adding a blank line

^ZI inserts a blank line before the current line; **^ZA** inserts one after the current line. The cursor is positioned at the beginning of the new line.

Inserting control characters

To insert a control character into the text, enter **^P** followed by the character.

Saving your work

When you finish entering and editing text, and want to save your text in a file, press **^KS**. **^KS** writes your buffer to a file.

If your buffer does not have a name, DR EDIX prompts you for a file name. Type the name of the file you want to store your text in and press **Enter**. Use the same rules to name DR EDIX files that you use for other files you create in Command Mode (see Chapter 5, "Working in Command Mode," on page 133). Do not use a file name that already exists unless you want to replace the contents of the old file with new material.

^KS does not change the current buffer in any way. You can continue editing that buffer and press **^KS** again later. You can begin editing a different file with **^KE**. You can exit DR EDIX with **^KQ**.

Note: Write your file to disk often during a lengthy editing session. If the power goes off or your store controller fails, your edits up until the last **^KS** will be saved. No editing is saved until you press **^KS**.

If you forget to press **^KS** before exiting or before editing a different file, DR EDIX gives you a second chance by asking you to confirm the **^KQ** or **^KE** command.

If you press **^KS**, and then decide not to save your file, press **^KQ**. It cancels the command you entered. You are now back editing your document. Anytime you want to cancel a command, press **^KQ**.

Note: This program follows the 8.3 naming convention and saves file names in uppercase characters. It is only recommended for use on the C: or D: drives. There are limitations when DR EDIX is used on other drives and file systems.

Understanding DR EDIX backup files

Before DR EDIX writes a file to disk, it takes the previous version of the file and gives it the file extension BAK. This is called a *backup* file.

This feature guards against losing the file because of an error during the write operation. If DR EDIX detects an error while writing your file, it displays a message and restores the original file. If DR EDIX does not detect the error, as in the case of a power failure, you have to carefully inspect both the new and old files when your store controller is running again and do the appropriate renaming yourself.

You never lose data because of a write error, provided you pay attention to messages from DR EDIX and use **^J**, the HELP command, whenever you are unsure of what to do.

Backup files are also valuable if you accidentally erase the current version of your file. On the other hand, backup files take up valuable disk space, so you might want to copy your current version onto a diskette and erase the others to free disk space.

Editing another file

After you have written your text into a file, you can immediately begin working with a different file. To edit another file, press **^KE**, the EDIT command. (If you did not press **^KS** to save the previous file, DR EDIX asks you if it is all right to lose the changes.)

When you press **^KE**, DR EDIX displays the following prompt at the top of the panel followed by a blinking cursor:

File:

Type the name of the file you want to edit and press **Enter**. If you want to work with a file that is on a different drive, include the drive letter and a colon (:) before the file name. DR EDIX reads this file into the current buffer.

You can work with the file using any of the DR EDIX commands described in this chapter. When you finish editing the file, use **^KS** to save your edits.

Exiting DR EDIX

Use **^KS** to exit from DR EDIX. If any buffers have been changed but not yet written, you must confirm that you want to exit. Usually, you should answer N to the confirmation to cancel the exit.

Use **^ZS** to review the status display to see which buffers have unwritten changes in them. The status display is described in this chapter. If you want to write the file to disk, press **^KS**. To exit DR EDIX, press **^KQ**.

If you accidentally press **^KQ** while you are editing, DR EDIX displays the following prompt:

OK to lose changes (y/n)?

If you press **Y**, the DR EDIX program ends without writing your changes to disk. You return to the system prompt. Press **N** to continue editing or to choose another DR EDIX command.

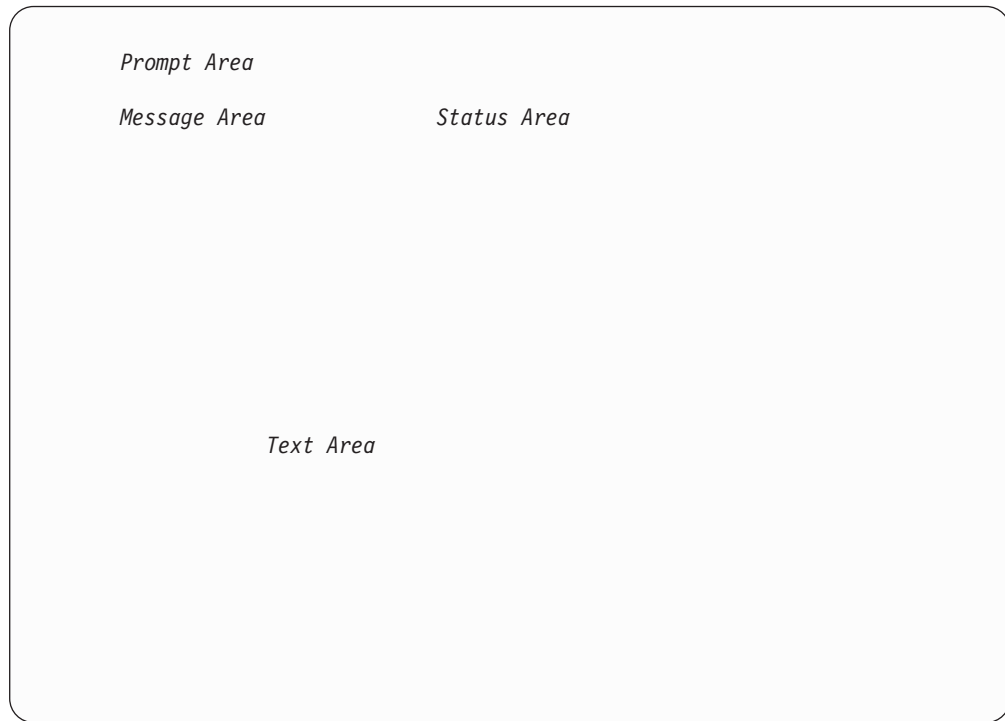
Understanding the panel

This section describes how DR EDIX uses the panel and explains how you interpret the information that DR EDIX places there. You might want to read this section with the DR EDIX panel in front of you.

The panel is divided into four areas:

- Text area
- Prompt area
- Message area
- Status area

The following figure shows the location of the four areas:



Text area

The text area is where you enter text, move your cursor, and enter DR EDIX commands.

Prompt area

Some DR EDIX commands prompt you for additional information. For example, when you use **^KE** to read a file into your current buffer, DR EDIX prompts you for a file name.

DR EDIX prompts appear in the upper left corner of your panel.

When you respond to a prompt, you can use the conventional alphanumeric keys, plus **Tab** and **Enter**. **Enter** has special meaning here; it tells DR EDIX you have finished your response.

If your response is longer than the space provided, DR EDIX scrolls the prompt window to the left. However, your response cannot be longer than 50 characters.

If you make a mistake when you type your response, you can use most of the commands for entering and deleting text to correct it.

If you want to cancel a command after getting a prompt, press **^KQ**. Note that **^KQ** has a special meaning here; rather than exiting DR EDIX, it exits from the command that caused the prompt.

If you need help after getting a prompt, press **^J**.

Message area

The message area is the left half of the top row of the panel. DR EDIX messages appear in reverse video. See “DR EDIX error messages” on page 280 for a complete list of these messages.

Status area

The DR EDIX status area is in the upper right corner of your panel. The status area gives you continuous information about the current window. A typical DR EDIX status message looks like this:

```
REVIEW.MEM
B1      48  16/18
```

In the preceding example, the status area gives you the following information:

- REVIEW.MEM is in the current buffer.
- Buffer 1 (B1) is the current buffer.
- The cursor is on column 48.
- The cursor is on line 16.
- The buffer is 18 lines long.

When the cursor changes windows, the status area is immediately updated to reflect the status of the new current window.

Status table

You can obtain additional status information by using **^ZS**, the DR EDIX STATUS command. When you press **^ZS**, you get a table of information about each of your files, windows, and buffers. Table 36 is a sample status table.

Table 36. DR EDIX status table

File	Bfr	Lines	Wnd	Chg	Mrk
REVIEW.MEM	1	18	1	Y	
EXPENSES.MEM	2	13	2		
GUESTS.LST	3	25		Y	Y
	4	15			

This status table lists buffers 1 through 4. The first column shows the file name associated with each buffer, if there is one. The status table also displays the total number of lines in each buffer, the window number in which it appears, whether the buffer has unwritten changes, and whether the buffer contains a marked block.

REVIEW.MEM is in Buffer 1. It has 18 lines of text and appears in Window 1. It contains changes that have not been written to disk.

EXPENSES.MEM is in Buffer 2. It has 13 lines of text and appears in Window 2.

GUESTS.LST is in Buffer 3. It is 25 lines long, and it does not appear in a window. GUESTS.LST contains changes that have not been written to disk. It also contains a block of text that has been marked for copying, moving, or deleting. (Block commands are described in “Using block commands” on page 273.)

Buffer 4 is empty. The status table displays one line of information for each buffer, even if the buffer is empty. A buffer is always at least one line long, even if it is empty.

Working with windows and buffers

When you work with a file in DR EDIX, it is stored in a block of memory called a *buffer*. For all practical purposes, once you read a file into a buffer, or create new text in an empty buffer, you can treat the buffer as a file, moving from the beginning to end, adding and deleting lines, writing it to a disk, and so on.

A line in a DR EDIX buffer can be up to 255 characters long. Because some of these characters can be tabs, you can actually enter lines that take over 429 columns to display. This is useful for editing very wide files such as spreadsheet reports.

There is no fixed limit to the number of rows in a buffer. DR EDIX adds lines to a buffer until the available memory is filled. If you fill a buffer, use the **^KS** command to write the buffer contents to a file.

DR EDIX can divide its memory into as many as four buffers; therefore, you can work on up to four documents at once. You can switch back and forth between all four buffers.

Using windows

DR EDIX lets you split your panel horizontally into two *editing windows*.

Note: Do not confuse editing windows with the windows used on your operating system. Editing windows described here are applicable only when using DR EDIX.

Using editing windows is the same as if you had two physically separate panels.

You can display any buffer on the panel by assigning it to a DR EDIX window. By moving the cursor, you can position a window anywhere within a buffer to view the buffer's contents. In each window you can view and edit the contents of any of the four buffers.

When you begin a DR EDIX session, Window 1 (W1) occupies the entire panel, and it is the only window allowed to do so.

If you press **^Z2**, the text area is split horizontally to make room for Window 2 (W2). Window 2 can occupy only the bottom half of the panel.

The DR EDIX commands **^Z1** and **^Z2** move the cursor to windows W1 and W2, respectively. The **^Z1** and **^Z2** commands are assigned to function keys **F1** and **F2**.

The **^ZO** command restores the full text area to Window 1.

Table 37 shows how to manipulate windows.

Table 37. DR EDIX window commands

Command	Meaning
^Z1	Window 1. Switches cursor to W1. W1 is always present on the panel.
^Z2	Window 2. If necessary, splits the panel to show W2, then switches cursor to W2.
^ZO	One window. Restores the panel to one window (W1). Use ^Z with the letter O, not zero.

Using buffers and files

When you start DR EDIX, Buffer 1 is assigned to Window 1, and Buffer 2 is assigned to Window 2 by default, even though Window 2 is not showing.

A buffer might or might not have an associated file name. That file name might or might not be the name of an existing disk file. DR EDIX creates a file when the contents of a buffer are written to a disk. If you specify a file name when you start DR EDIX, that file name is associated with Buffer 1.

If you start DR EDIX without specifying a file name, the buffer you use is empty and has no name. This buffer does not have an associated file name until one of the following occurs:

- You write its contents to a disk using **^KS**. In this case, DR EDIX prompts you for a file name.
- You assign the buffer a file name using **^KF**.
- You purposely read a file into the buffer by using **^KE**. You must specify the name of the file you want to read in. The name of your buffer is the name of the file you read in.

You can assign any buffer to any window by using the **^ZB** command. When you choose **^ZB**, DR EDIX displays the following prompt:

Buffer number or file:

^ZB displays the specified buffer in the current window. If you give a file name, the buffer associated with that file name appears in the current window.

Using block commands

A *block command* is a command that affects a user-defined number of lines. You can mark a single line or a group of lines. You cannot mark only a portion of a line. You can use block commands to:

- Copy a block of text within a buffer or between buffers.
- Move a block of text within a buffer or between buffers.
- Delete a block of text from a buffer.
- Locate a block of text in (or move the cursor to) any buffer.

Marking a block

To use any of the DR EDIX block commands, you must first mark a block. If you try to perform a block command without marking a block, DR EDIX displays the following message:

No source block

The procedure for marking lines has to be started and completed in the same buffer, and the buffer must be visible in a window. You can mark a block only in a buffer you can see. However, once this block is marked, you can put this buffer in the background.

If the cursor is positioned in a particular window and you press **^KB** once, DR EDIX displays a message indicating that it is expecting the second **^KB** command. If you move the cursor to another window, the message area clears.

To mark a single line, press **^KB** twice while the cursor is positioned anywhere within the line.

To mark a group of lines, press **^KB** on the first line of your block. After you issue the first **^KB** command, DR EDIX displays the following message:

Waiting for second mark ...

DR EDIX expects a second **^KB** command to complete the marking of the block.

Move the cursor to the last line of the group and press **^KB** again. DR EDIX highlights the marked lines by displaying the first character of each marked line on the panel in reverse video.

At this point, you can execute one of the DR EDIX block commands.

Unmarking a block

You can unmark a block of lines by pressing **^KB** a third time while the cursor is positioned anywhere in the visible buffer that contains the marked block. A block remains marked until you unmark it with a third **^KB**, or **^KY** or **^ZG**.

You can have only one marked source block at any time. If you have a marked block in one buffer and you press another **^KB** in a different buffer, DR EDIX displays the following message in the message area:

Source block marked in some other buffer

To mark a second block of lines, you must unmark the previously marked block. If a marked block does not appear on the panel, use the **^ZS** command to get the status panel. The buffer containing the marked block has a Y in the Mrk column of the table. You need to assign the buffer with the marked block to a window so you can see it, then press **^KB** to unmark the block.

Copying a block

After you have marked a block of lines, position the cursor on the line where you want the block to be copied, and press **^KC**. You can copy a block of lines to the same or any other buffer that is in view in one of the windows. The marked block need not be in view.

You can copy a block only to a buffer that you can see. You must see the cursor on the text you are copying to.

After you copy a block, the block stays marked. You can use the **^KC** command repeatedly to copy the same block to several locations.

Moving a block

Moving a block of lines is similar to copying it, except that the lines are deleted from their original location.

You can move a block of lines within the same buffer or to any other buffer with the **^KV** command. The buffer to receive the block must be in view in a window. The buffer containing the marked block need not be in view. The marked block is inserted after the line where the cursor is positioned.

After you move a block, the block stays marked. You can use **^KV** to move the block again.

Deleting a block

You can delete a block of lines in a buffer with the **^KY** command.

After you enter **^KY**, the cursor is positioned on the first nonblank character of the first line following the deleted block.

Because the marked block disappears from the buffer, **^KY** changes the status of the buffer from marked to unmarked.

You can undo a block deletion line by line using the **^ZU** command for every line that was deleted.

Note: You cannot recall an entire block of deleted lines with one command.

Searching for and translating text

DR EDIX has two commands, **^QF** and **^QA**, for searching for and translating text. You can search for and replace a single character or a string of characters like a word or sentence. The string you search for must be 50 characters or fewer. You can create complex search patterns by using the global file name characters ***** and **?**.

Press **^E** to recover the pattern from a previous **^QF** or **^QA** command. Press **^X** to bring the current pattern to the prompt area.

Searching for a text string

Use **^QF** to search for a text string in a buffer. When you press **^QF**, DR EDIX prompts you for a pattern. To search for the string *Buffalo Bill* in a buffer, press **^QF** and at the prompt type the following:

Pattern: **Buffalo Bill**

DR EDIX displays the following message while it searches for a match:

Searching

DR EDIX searches the entire buffer. When it is finished, it displays one of the following messages:

Found

Not found

If a match is found, you can look for the next match by pressing **^L**, the REPEAT-SEARCH command. Press **^L** repeatedly to find all occurrences of the match in the buffer.

You can do anything you want after pressing **^L** without losing the pattern, unless you issue another command in which you specify another pattern.

Searching with global file name characters

You can use global file name characters in the search pattern. DR EDIX uses the asterisk (*****) and question mark (**?**) global file name characters.

The ***** global file name character stands for any string of any length in a line. For example, the pattern:

a*p*

matches the strings:

application

astute programmer

antipasto

Note that * finds blank lines created with the **Enter** key, but ?* and *? look for at least one character, including the space character. The * global file name character can appear any number of times in a pattern. However, the asterisk cannot be the first nonblank character or the only character in a search string.

Note: If you specify a pattern with several asterisks, the number of possible matches that need to be searched for might be very large. This causes a long searching time.

The ? global file name character stands for any single character (including a tab or a space), but not the end-of-line character. For example, the pattern:

```
c?re
```

matches all of the following:

```
care
c re
c:re
```

The ? global file name character can appear any number of times in a pattern.

Searching for global file name characters

If you want to search for an asterisk or question mark in your text, you have to precede the asterisk or question mark with a backslash (\). The \ tells DR EDIX to treat them as regular characters, not global file name characters.

Similarly, if the string you are searching for contains a backslash, you must precede it with a second backslash. To search for the string:

```
hello\goodbye
```

use the pattern:

```
hello\\goodbye
```

Translating text strings

^QA translates matched strings into other strings.

After you press **^QA**, DR EDIX prompts you for a pattern and then a replacement string. Suppose you want to translate the string *DR EDIX is easy to learn*. Press **^QA** and at the pattern prompt type:

Pattern: **DR EDIX is easy to learn.**

Press **Enter** and at the replacement prompt type:

Replacement: **DR EDIX is fun to use.**

Press **Enter** again. The **Enter** key is not part of the replacement text. The replacement text can be shorter or longer than the matched text, but it cannot be longer than 50 characters.

DR EDIX scans the buffer while the following message appears:

```
Searching
```

DR EDIX scans the entire buffer and returns to the original cursor position.

If no match is found, DR EDIX displays the following message:

```
Not found
```

If a match is found, DR EDIX displays this message:

```
l=local; g=global; KQ=exit; other=skip
```

If you press the letter **l**, only that match is replaced by the replacement string and DR EDIX searches for the next match. Press the **Space bar** or any other key to skip to the next matched string. **^KQ** cancels the search-and-replace operation. You can continue to scan the buffer by pressing the **Space bar** or the letter **l** each time DR EDIX pauses for your input. The **^QA** command ends automatically when the buffer has been completely scanned.

If you press **g**, DR EDIX replaces all matched strings with the replacement pattern. Once you press **g**, you cannot cancel the search-and-replace operation. Null strings are valid replacements. A null string replaces the text with nothing. To change a text pattern to a null string, simply press **Enter** at the replacement prompt.

You can use the ***** and **?** characters in the replacement pattern. They are placed in the text as an asterisk and question mark, not as global file name characters.

Just as with search patterns, DR EDIX remembers the pattern and replacement string from the previous **^QA** command. If a previous pattern or replacement string exists, you can bring it to the prompt response area by pressing the up-arrow key.

Going to a line by number

You can use the **^ZG** command to go to a specific line. When you press **^ZG**, DR EDIX prompts you for a line number.

The line number you specify must be between 1 and the number of lines in the buffer. If you enter a number that is out of this range, DR EDIX prompts you again.

After you enter the line number, DR EDIX moves the cursor to the first column of that line. If you enter 0 for the line number or press only **Enter**, DR EDIX returns your cursor to its location before you pressed **^ZG**.

DR EDIX does not number lines on the display, but the current line number is always shown in the status area. When you use DR EDIX commands, you do not need line numbers. Line number support is provided for use by other programs such as language compilers.

Merging files

To merge a file with a file already in a buffer, put the cursor one line above where you want to insert the second file and press **^KR**. To add a file to the end of a buffer, insert the file after the last line of the buffer.

When you press **^KR**, DR EDIX prompts you for a file name. Enter the file name you want to read in and press **Enter**. DR EDIX inserts a copy of this file into the buffer after the current line. You still have a copy of the second file on your disk.

The current file name, displayed in the status area, remains the same.

If you want to read in only part of a file into a buffer, you have two choices:

- Read in the whole file and then delete the part you do not need.
- Read the file into another buffer and use **^KV** to move the lines you want into the target buffer.

Pressing **^KE** and entering the name of the file is nearly the same as pressing **^KR** when you want to read a file into the buffer. The difference is that **^KE** erases the old contents of the buffer while **^KR** appends the new file to the existing contents of the buffer.

Changing the file name

You can change the current file name with **^KF**. DR EDIX prompts you to enter a new file name. After you enter a file name, it appears in the status area. The text in the file is not changed by **^KF**.

Summary of DR EDIX commands

This section provides a quick reference to DR EDIX commands. It contains each command and a brief description of its function.

Most DR EDIX commands are invoked by pressing a combination of **Ctrl** key and another key.

Table 38. DR EDIX commands and error messages

Task	Command	Description
Switch to window 1	^Z1	Switches to window 1 and its associated buffer. (Use the numeral 1, not the keypad 1.)
Split panel to show window 2	^Z2	If necessary, splits the panel to show window 2. If window 2 is already showing, this command switches to window 2 and its associated buffer. (Use the numeral 2, not the keypad 2.)
Add blank after current line	^ZA	Adds a blank line after the current line.
Prompt for buffer number or file name	^ZB	Prompts for a buffer number or file name and brings into the current window the specified buffer, or if a file name was given, the buffer associated with that file.
Insert copy of marked lines	^KC	Inserts a copy of the marked lines after the current line.
Delete current line	^KY	Deletes the current line.
Reinitialize current buffer	^KE	Prompts for a file name and reinitializes the current buffer so that editing on that file can begin.
Remember and display file name	^KF	Prompts for a file name and then makes it the remembered file name that appears in the status area.
Search for match on pattern	^L	Searches for the next match on the currently defined pattern.
Search for line number	^ZG	Prompts for a line number and places the cursor at the start of that line. Specifying 0 or Enter moves the cursor to the start of the marked block. This unmarks the block. If necessary, DR EDIX switches panels and buffers to find the block.
Provide help	^J	Provides help information.
Insert blank line	^ZI	Inserts a blank line before the current line.

Table 38. DR EDIX commands and error messages (continued)

Task	Command	Description
Mark first/last line of block	^KB	Marks the first or last line of a block. Blocks can be copied, deleted, moved, or located. If a block is already marked, this command clears the marks.
Move marked lines	^KV	Moves the marked lines to follow the current line. Places the cursor at the first character of the moved lines.
Restore panel to one window	^ZO	Restores the panel to one window (window 1).
Display status panel	^ZS	Displays the status panel.
Read file into current buffer	^KR	Prompts for a file name and reads that file into the current buffer to follow the current line. Places the cursor at the first character of the inserted text.
Prompt/search for pattern of text	^QF	Prompts for a pattern and then searches for text that matches it. Places the cursor at the start of the matched text.
Prompt for pattern/replacement string	^QA	Prompts for a pattern and then prompts for a replacement string. Searches for the pattern, places the cursor there, and then asks you if you want to perform a local translation or global translation, cancel the translation, or skip to the next match. If you request an option other than cancel, then searches for the next match and repeats.
Undo previous ^Y	^ZU	Undoes the previous ^Y by inserting the last deleted line before the current line.
Prompt for file name	^KS	If there is no file name in the status area, prompts for a file name. Then writes the buffer to that file.
Exit DR EDIX	^KQ	Exits from DR EDIX. System returns to the system prompt. If issued in response to a prompt, cancels the command that caused the prompt. If issued during a translation, cancels the translation.
Put cursor at first character of line	^QS	Places the cursor at the first nonblank character of the current line.
Put cursor at end of line	^QD	Places the cursor at the end of the current line.
Erase marked block	^KY	Deletes the marked block of lines.
Put cursor at last window line	^QX	Places cursor at the first nonblank character of the last line of the window.
Put cursor at first window line	^QE	Places cursor at the first nonblank character of the first line of the window.
Put cursor on previous word	^A	Places cursor on first character of previous word.
Put cursor on next word	^F	Places cursor on first character of next word.
Insert control character	^P	Places the next typed character into the text as a control character.

Table 38. DR EDIX commands and error messages (continued)

Task	Command	Description
Backspace	^H	Moves the cursor one character to the left and deletes the character that was there.

DR EDIX error messages

This section describes the error messages that you might receive from DR EDIX. Each description tells you how to correct the condition.

BAD PATTERN

The specified pattern contains a combination of characters that are not valid, usually a special character. See “Searching for and translating text” on page 275, or press ^J for help.

BAD TARGET

The target line for a copy or move, indicated by the current cursor position, is located within the marked lines. Move the cursor out of the marked block.

CAN'T CREATE FILENAME.EXT

A new file cannot be created to perform a ^KS command, probably because of a full directory or insufficient disk space. Try writing the file to another drive or change disks and reissue the command.

CAN'T DELETE FILENAME.EXT

The system cannot delete the specified file because the disk is defective, there is a problem with your hard disk drive hardware, or you have protected your diskette with the FSET command.

CAN'T OPEN FILENAME.EXT

CAN'T OPEN FILENAME.EXT; NEW FILE ASSUMED

The file you are trying to read with ^KR or ^KE cannot be opened. If you have spelled the file name correctly, the file might exist in a different directory. In the case of the second message, DR EDIX assumes that you are opening a new file.

DEVICE FULL

The disk to which you are writing with ^KS is full. The file has been restored to the way it was before the write. You must write to a different drive or make room on the disk by erasing unneeded files. Press ^J for instructions.

EMPTY BUFFER

You have attempted to delete the only line in the buffer when the line consisted of the end-of-line character only. DR EDIX buffers can never get smaller than a single end-of-line character.

EMPTY DELETE STACK

You have issued the **^ZU** (UNDO) command when there are no deleted lines in the undo stack.

INVALID COMMAND

You have issued a command that is not valid in this context. An example is **^ZG** given in response to a Pattern: prompt. To cancel the prompt, press **^KQ**.

INVALID RESPONSE

Responses to a (y/n)? query must be an uppercase or lowercase y or n.

NO BUFFER HAS THAT FILE

No buffer is associated with a file of the name you entered. Press **^ZS** to see which files are associated with which buffers, or invoke a buffer by number.

NO HELP FILE

DR EDIX cannot find the file named HELP.EDX. To receive help, this file must be on the current drive or drive A.

NO PATTERN

You have pressed **^L** (CONTINUE-SEARCH), but there is no previous pattern stored.

NO SOURCE BLOCK

You have issued a block-oriented DR EDIX command, but there is no marked block. You must first use **^KB** to mark the lines to be copied or moved.

NOT FOUND

The pattern you entered could not be matched.

NUMBER REQUIRED

After pressing **^ZG** (GO), you must type a number following the Line: prompt. To cancel the **^ZG**, press **^KQ**.

OUT OF RANGE

The line number you gave in response to a **^ZG** (GO) command must be between 1 and the number of lines in the buffer. Press **^KQ** to cancel the **^ZG** command, or type an existing line number.

READ ERROR

An error occurred while DR EDIX was reading a file. This message can occur if a bad filespec is specified or if the file that DR EDIX is reading is not terminated with an end-of-file character (decimal 26).

RESPONSE TOO LONG

The response you typed after a prompt has too many characters for that particular prompt. Press **^J** to see which responses are appropriate.

S_MALLOC

There is not enough memory to edit the file, or the file you are editing exceeds the maximum size limit (approximately 150 KB) supported by the DR EDIX editor.

SOURCE BLOCK MARKED IN SOME OTHER BUFFER

Use **^ZS** to see which buffer is marked. Make that buffer current (with **^ZB** or with a window-switching command) and press **^KB** to clear the marks. You can also use **^ZG** to go directly from your current position to the start of the source block.

TEXT CAPACITY EXCEEDED

You have filled all allocated memory with text. At this point, write all buffers to disk, exit from DR EDIX, and invoke DR EDIX again. Although DR EDIX usually maintains the integrity of internal storage structures when memory is exhausted, it is safer to get a fresh start.

UNKNOWN COMMAND. PRESS ^J FOR HELP.

You have typed a command that is unknown to DR EDIX.

WRITE ERROR

An error occurred while DR EDIX was writing a file. This is probably the result of a defective disk, and is different from the device full condition.

Using XE Editor

To use the XE Editor, you must be in Command Mode. When you select Command Mode from the menu, a system prompt appears, and you can enter XE commands directly. See “Starting XE Editor” on page 282 to begin entering XE commands.

XE Editor has an online help facility that allows you to get helpful instructions whenever you need them. You can press **F1** to find out what you should do and what commands are available to you.

Starting XE Editor

To start XE Editor (XE), you must be in Command Mode. From the SYSTEM MAIN MENU, select the Command Mode option by typing **7** and pressing **Enter**. Change directories (CD) to the directory where the file that is to be edited or created is located.

Note: You can also enter a path or file name when starting the editor or edit a file from the command line within the editor.

To create a file or edit a file, at the Command Mode prompt type the following command, and press **Enter**:

C:>**XE** *filename*

where *filename* is the name of the file to edit or create and can be a path to another drive or subdirectory.

Note: In most instances, XE Editor ignores case when you name your file. However, if you create a file on the M: drive, XE Editor is case-sensitive.

When XE is started, the cursor is always on the command line at the bottom of the panel. To move the cursor to the text area, press **Esc**.

Entering text

To begin your first document, move the cursor to the text area and start typing. Press any of the alphanumeric keys on the keyboard and a character appears on the panel in the current file. The blinking cursor indicates where the next character you enter will appear.

When you enter text, you can also use the Tab and Enter keys. The Tab key puts the cursor at the next tab stop. Tab stops are initially set eight spaces apart in columns 9, 17, 25, and so on.

To end a line, press the **Enter** key. This action inserts a carriage return at the end of your line. The cursor moves to the start of the next line.

Inserting and replacing

XE Editor has two modes of entering text: insert and replace. Replace mode is the default when you start XE. You can alternate between insert and replace modes by pressing the **Insert** key.

In insert mode, the character you type is inserted at the cursor position and the cursor moves to the right. The rest of the characters on the line also move to the right. When the editor is in insert mode, the status line at the bottom of your panel displays **INS**.

In replace mode, the character you type replaces the character at the cursor position and the cursor moves to the right. The rest of the characters on the line do not move. When the editor is in replace mode, the status line at the bottom of your panel displays **REP**.

Moving the cursor

The space you have to enter text is much larger than the size of your panel. You can move the cursor anywhere within the file, and XE Editor scrolls the text left, right, up, or down so that you can see the part of the file that contains the cursor. Table 39 summarizes the cursor control keys.

Table 39. XE Editor cursor control keys

To Move the Cursor	Use
Left one space	←
Right one space	→
To the beginning of the line	Home
To the end of the line	End
Up one line	↑
Down one line	↓
Up one page	Page Up
Down one page	Page Down
To the top of the file	Ctrl-Home

Table 39. XE Editor cursor control keys (continued)

To Move the Cursor	Use
To the bottom of the file	Ctrl-End
To the left and erase character (backspace)	Backspace
Go to line (nnn)	Enter goto <i>nnn</i> on the command line, where <i>nnn</i> is the line number.

Editing text

Most of the XE editing commands are invoked by pressing a combination of the **Alt** key and another key. The editing commands let you delete characters or add, split, recall, or rejoin lines.

Deleting characters

You can delete individual characters by using the **Del** key or the **Backspace** key.

The **Del** key deletes the character at the cursor.

The **Backspace** key moves the cursor one position to the left and deletes the character there. It is most useful when you notice an error as soon as you type it; you can backspace over the incorrect character and type another one. If the cursor is already under the first character of the line, the **Backspace** key is ignored.

Deleting lines

Ctrl-Backspace deletes the current line.

Delete to end-of-line

Ctrl-E deletes the current line from the cursor to the end of the line. You can undo the delete using **F9**.

Splitting lines

To split a line of text, move the cursor to where you want to split the line, and press **Alt-S**. The text to the right of the cursor moves to the next line. Any lines below the split line move down too.

Joining lines

To join two consecutive lines, put the cursor at the end of the upper line and press **Alt-J**. XE Editor joins the two lines and puts a blank space between the two pieces.

Adding a blank line

Ctrl-Enter inserts a blank line after the current line. The cursor is positioned at the beginning of the new line.

Inserting control characters

To insert a control character into the text, press **Alt-X** followed by the character.

Saving your work

If you want to save your file to a disk and continue editing, press **F2**. If you want to save your file to a disk and exit the XE Editor, press **F4**.

Note: Save your file to disk often during a lengthy editing session. If the power goes off or your store controller fails, your edits up until the last **F2** will be saved. No editing is saved until you press **F2** to save or **F4** to save and close.

Exiting XE Editor

If you have made changes to the file and you want to save the changes and exit XE, press **F4**. If you have not made changes, press **F3** to exit XE.

If you accidentally press **F3** while you are editing and you have made changes to the file, XE displays the following prompt:

Throw away changes (y/n)?

If you press **Y**, XE ends without writing your changes to disk and returns you to the Command Mode prompt. Press **N** to continue editing.

Editing multiple files

When multiple files are open for editing in the editor, use the **F10** key to switch between the documents. You can open multiple files by:

- Using the XE *filename* command
- Using the XEDIT *filename* command
- Specifying file names as parameters to the editor on a 4690 command line

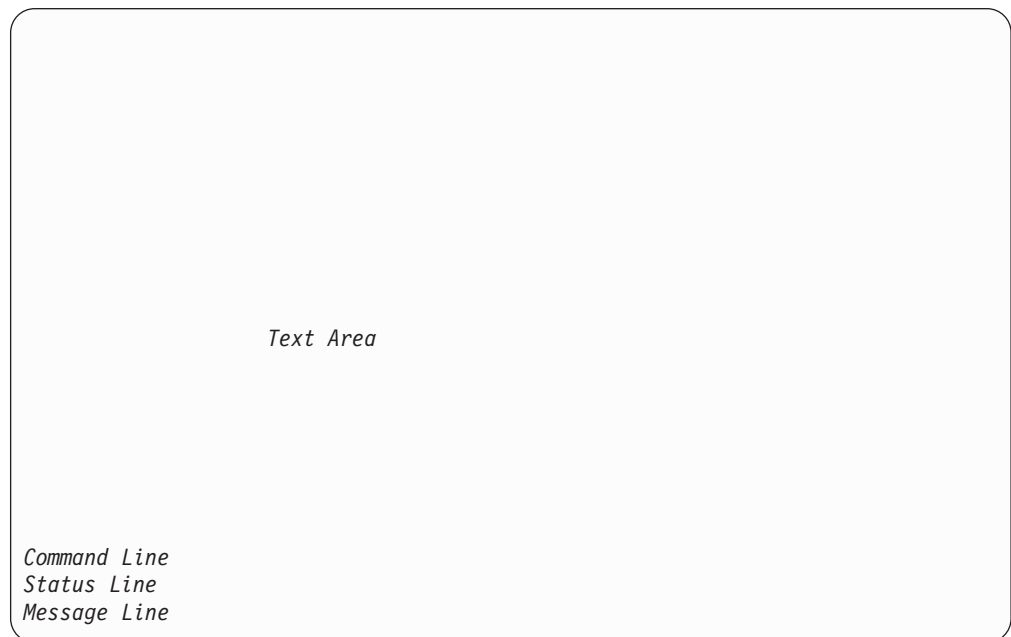
Understanding the panel

This section describes how XE Editor uses the panel and explains how you interpret the information that XE places there. You might want to read this section with the XE Editor panel in front of you.

The panel is divided into four areas:

- Text area
- Command line
- Status line
- Message line

The following figure shows the location of the four areas:



Text Area

The text area is where you enter or change your text. This area displays an extra line above and below the file data. These lines contain the Top Of File and Bottom Of File messages. While the cursor is in the text area, you can use key commands to control the text. Table 40 shows the special keys that you can use within the text area.

Table 40. Text control keys

Key	Description
Esc	Toggle between the command line and the data area
Ctrl-Enter	Add a new line
Ctrl-Backspace	Delete a line
Alt-C	Copy marked lines
Alt-D	Delete marked lines
Alt-J	Join
Alt-L	Mark lines
Alt-M	Move marked lines
Alt-R	Repeat line
Alt-S	Split
Alt-U	Unmark lines
Alt-Y	Find mark
Alt-Xnnn	nnn represents ASCII code entered from the numeric keypad only. All characters must be entered with the Alt key held down.
Ctrl-E	Truncate line
Ctrl-F	Repeat find

In addition to the keys to control the text, you can also draw boxes using the **Alt-1** through **Alt-9** and **Alt--** and **Alt=** command keys. These numbers and special characters must be entered from the top row of the keys of the keyboard.

Command line

The command line is where you enter XE Editor commands. For example, from the command line you can locate a text string or change a text string to a different text string. The background of the command line is a different color than the text area.

The command line is needed for commands that do not use control keys. You can switch the cursor position between the command line and the text area using the Esc key. The following commands are available on the command line:

Command	Description
EDIT [filename]	Switch to another document
E [filename]	Switch to another document
FILE [filename]	Save document then quit (same as F4)
NAME [filename]	Rename document
QQ	Quit without saving changes
SAVE [filename]	Save document to disk (same as F2)
GOTO linenum	Go to line number

Command	Description
GET [filename]	Read another file into document
PUT [filename]	Write marked block to file or PRN
L /target [/options]	Locate a string
C /target [/replace[/options]]	Change a string

Press **Enter** to perform the command after typing it. The commands are not case-sensitive. The command line will appear cyan on a color display and white on a monochrome display.

Message line

The bottom line of the panel is the message line. This is where error messages or prompts are displayed. When there is no message to display, the message line shows the meanings of the function keys.

Status line

The line below the command line is the status line. This line shows the name of the file being edited on the left side. On the right side are three fields. The first field is the row number at which the cursor is located. The second field is the column number at which the cursor is located, and the third field is the mode of the insert key. The insert key mode will either be “Ins” for insert mode or “Rep” for replace mode.

Using block commands

A *block command* is a command that affects a user-defined number of lines. You can mark a single line or a group of lines. You cannot mark only a portion of a line. You can use block commands to:

- Copy a block of text
- Move a block of text
- Delete a block of text
- Locate a block of text

Marking a block

To use any of the XE Editor block commands, you must first mark a block. If you try to execute a block command without marking a block, the command is ignored.

To mark a single line, press **Alt-L** while the cursor is positioned anywhere within the line.

To mark a group of lines, press **Alt-L** on the first line of your block. Move the cursor to the last line of the group and press **Alt-L** again. XE Editor highlights the marked lines by changing the color of the lines to red or white.

At this point, you can execute one of the XE block commands.

Unmarking a block

You can unmark a block of lines by pressing **Alt-U**. You can have only one marked source block at any time. To mark a second block of lines, you must unmark the previously marked block.

Copying a block

After you have marked a block of lines, position the cursor on the line where you want the block to be copied, and press **Alt-C**. After you copy a block, the block stays marked. You can use the **Alt-C** command repeatedly to copy the same block to several locations.

Moving a block

Moving a block of lines is similar to copying it, except that the lines are deleted from their original location.

You can move a block of lines with the **Alt-M** command. After you move a block, the block stays marked. You can use **Alt-M** to move the block again.

Deleting a block

You can delete a block of lines in a file with the **Alt-D** command. Because the marked block disappears when deleted, there is no need to unmark it.

Searching for and translating text

XE Editor has two commands, **L** and **C**, for searching for and translating text. You can search for and replace a single character or a string of characters like a word or sentence.

Searching for a text string

Use **L** on the command line to search for a text string. For example, to search for the string *Buffalo Bill*, type **L /Buffalo Bill/** on the command line. If you are not sure of the case, type **L /buffalo bill/c**. If you want to search for Buffalo Bill only in the marked block, type **L /Buffalo Bill/m**.

Note: The **L** is optional; you can also use **/buffalo bill/**.

If a match is found, you can look for the next match by pressing **Ctrl-F**, the REPEAT-SEARCH command. Press **Ctrl-F** repeatedly to find all occurrences of the match in the file.

Translating text strings

Use **C** on the command line to translate matched text strings into other text strings. For example, to replace the string Buffalo Bill with the string Annie Oakley, type **C /Buffalo Bill/Annie Oakley/**. To replace the matched strings everywhere in your text, type **C /Buffalo Bill/Annie Oakley/***. To replace the matched text string only within the marked block, type **C /Buffalo Bill/Annie Oakley/m**.

Going to a line by number

You can use the **GOTO** command to go to a specific line. For example, to place the cursor on line 100, type **goto 100** on the command line.

The line number you specify must be between 1 and the number of lines in the file. If you enter a number that is out of this range, the cursor will be positioned at the bottom of the file.

XE Editor does not number lines on the display, but the current line number is always shown in the status line.

Importing files

To import a file, put the cursor one line above where you want to insert the second file and type **get filename**, where *filename* is the name of your second file.

The current file name, displayed in the status line, remains the same.

If you want to import only part of a file into a file, you have two choices:

- Import the whole file and then delete the part you do not need.
- Import the file into another file and use **Alt-M** to move the lines you want into the target file.

Changing the file name

You can change the current file name by typing **name** *filename* on the command line. *filename* is the new name of the file

Command line logging and the XE editor

The 4690 operating system logs the read/write access from the editors, but not each change. The DREDIX editor creates a BAK file that can be used to compare the changes made to the files. The XE editor does not currently create a BAK file. If both editors are available, and you need the security of command-line logging, it is recommended that you disable the XE editor.

Chapter 9. Using Java with the operating system

This chapter explains the use of Java with the operating system. This chapter contains information on the operating-system graphics support for Java and also information on mouse support.

Java 2 at the Java Development Kit (JDK) level 1.4.2 runtime is supported by 4690 Version 4 or higher. JDK level 1.4.2 should be compatible with Java programs written and compiled with earlier Java levels. Any incompatibilities between various Java levels are documented on Oracle's Java Web site. As announced on Jan. 11, 2005, 4690 OS support of Java 1 (1.1.8 JVM) ended on Aug. 31, 2006. Beginning with V6 of 4690, the 1.1.8 JVM is no longer included with the OS. Java 1 applications should be compiled with a currently supported version of the compiler for use with 4690 OS V6. All Java 2 JVM executable programs are located in the new logical directory named `java2bin:`. Support is no longer available for Java JVM 1.1.8; however, you can continue to use programs written for Java 1.1.8.

Note: The JAR and APPLETVIEWER tools are part of the Software Development Kit (SDK) and are not part of the Java Runtime Environment (JRE). All executable files that are part of the SDK are placed in the `M:/java2/bin` directory; thus, you need to specify this path name directly or the new logical name, `java2sdk:`, to reference these files.

For information on configuring your system to enable Java support, see the *Planning, Installation, and Configuration Guide*.

Java 6 at the Java Runtime Environment (JRE) level 1.6 is supported by 4690 Version 6 Release 2 or later, Enhanced Mode. JRE level 1.6 should be compatible with Java programs written and compiled with earlier Java levels. Any incompatibilities between various Java levels are documented on Oracle's Java Web site. All JRE level 1.6 executable programs are located in the `java6bin` logical directory.

Software Development Kit (SDK) functions are not included at the 1.6 level; SDK functions are only available as part of the 1.4.2 SDK.

For information on designing an application with Java, see the *4690 OS: Programming Guide*. See the `READJPOS.DAT` file in the `JAVA` directory of your 4690 OS Version 4 or higher system for information on 4690 JavaPOS support.

Understanding Java with the operating system

The operating system allows you to enable Java graphics support for your system and run a Java application from a 469x or SurePOS 700 Series terminal or controller. This Java support is also available on the SurePOS 350 terminal and TCxWave 6140 Series terminals. The Java environment works so that existing 4690 applications can run concurrently with the Java application. Also, new applications using the Java language can be added to the operating system environment. There are three options for a Java application to run in existing 4690 environments:

- Independent
- Concurrent
- Java only

With the independent option, you can write a new Java application for the terminal that will run independently of the existing POS applications. If you have Java

graphics enabled, you have access to a Java console using the terminal video. You can switch between the terminal console and the Java console using the **Alt+SysRq** key sequence. To display the Java console, press **Alt+SysRq J**. To display the terminal console, press **Alt+SysRq T**. The key sequence switches between the Java and terminal consoles. If a Java 6 graphical application is running, the system displays the enhanced mode graphical extensions screen. The enhanced mode graphical screen can be displayed on controllers when Java 6 is not running by pressing the key sequence **Alt+SysRq X**.

The concurrent option enables you to write a new Java application for the terminal that can run concurrently with the existing POS applications. With this option, modifications would be required for the existing CBASIC application to use C access routines to interface to the Java application using 4690 pipes. See the *4690 OS: Programming Guide* for more information on designing applications for the operating system.

The Java only application allows you to write a Java application for the terminal that can replace the existing 4690 POS application.

The operating system provides Java classes that enable access to various features on store controllers and terminals. These features include POS files, keyed files, pipes, system status, and system controls. See the *4690 OS: Planning, Installation, and Configuration Guide* and to the *4690 OS: Programming Guide* for information on configuring classpaths.

Only the Java runtime environment (JRE) is available on the operating system. Java development must be performed on another system. The operating system supports the use of long file names used by Java. Long file names are supported in both the controller and the terminal. This support allows Java programs written on other systems to be run with the operating system. The operating system uses a virtual file system (VFS) to translate long file names into file names that work with the existing file system and data distribution. You must enable long file name support through system configuration. You can use the 4690 Network File System (NFS) to point your system to remote systems that have long file name support enabled. See “Network file system data” on page 53 for an explanation of how 4690 VFS works with 4690 NFS and how to configure NFS within the operating system.

Logical file names can be created to help with long file paths. For example, a logical file name can be created that includes the path and the file name. If a file object is created with a file name that contains logical names, the `getAbsolutePath` and `getCanonicalPath` methods might return the name strings containing unexpected characters.

See the *4690 OS: Planning, Installation, and Configuration Guide* for worksheets to configure long file name support. See the *4690 OS: Programming Guide* for information on how VFS is implemented within the 4690 OS environment.

Note: A Java application in the terminal cannot use the `R::M:\filespec` method to access long file name support on the controller. The application must access long file name support by using a drive letter that has been defined in the terminal as being part of an NFS mount group. The mount group should point to the remote controller's exported M or N drives, which contain the files the application needs.

There is no system-configured primary or backup controller for a terminal's Java application. Instead, it is recommended that you use the Terminal Offline Function

(TOF) as described in the *4690 OS: Programming Guide*, or the Enhanced terminal preload and Java configuration as described in the *4690 OS: User's Guide*. Both TOF and the Enhanced preload functionality allows you to load into terminal memory all of the Java classes and application resources required by a Java application. Therefore, if the terminal loses its connection with the controller, the Java application can continue to operate in an offline mode.

Note: TOF is not supported for Java 6 on 4690 OS Version 6 Enhanced Mode terminals.

The system Java directories JAVA2BIN for the Java 2 JVM and JAVAEBIN for the Java 6 JVM must remain on their installed locations. User .ZIP or .JAR files can be located on the D drive if the classpath is changed appropriately.

Notes:

1. Any file compression program can be used to compress (zip) class files for use in the Java classpath. User .Jar files are supported and follow the same requirements as the .ZIP files. When zipping Windows NT files, be sure to use the -x flag. Be careful when putting the same class file in multiple .ZIP files. This action can result in your using an outdated version of the class file.
2. The controller classpath is the same for all controllers on an MCF system. When placing user .ZIP or .JAR files on the D drive, ensure that all controllers on the MCF system have D drives.

The maximum number of running threads is 63. Attempting to start additional threads will cause an OutOfMemoryError exception.

Configuring your system

The operating system contains menus to enable you to specify Java configuration options. The Java configuration data contains information such as whether a terminal will be enabled to support Java and controller and terminal classpath information. From the SYSTEM CONFIGURATION panel, you can specify whether Java will be enabled for your store system. See the *4690 OS: Planning, Installation, and Configuration Guide* for assistance with configuring your system.

If you are using Java 2, see the "Java problem resolution" section and the "Additional hints and tips" section in the *4690 OS: Programming Guide*.

The operating system allows a batch file to be configured as a primary or secondary application, which allows a Java application to be run from the 4690 menu. For example, when the batch file, ADX_UPGM:JAVAPRIM.BAT, contains the line, java SalesApplication, the primary application name is configured as ADX_UPGM:JAVAPRIM.BAT. Select the primary application from the menu starts Java with the SalesApplication class. Other supported batch file commands are also allowed. Secondary applications use batch files in a similar manner. The command session ends when the configured batch file completes.

The operating system allows Java to be configured as a background application. See the *4690 OS: Planning, Installation, and Configuration Guide* for information on configuring a Java application as a background application.

Configuring terminal functions for Java

From the TERMINAL FUNCTIONS panel, you can define Java application functions such as displaying Java application status, starting Java applications, and stopping a Java application. See "Use Java application functions" on page 97 for more information on Java terminal functions.

If you are using Java 2, see the "Java problem resolution" section and the "Additional hints and tips" section in the *4690 OS: Programming Guide*.

Configuring controller functions for Java

From the CONTROLLER CONFIGURATION panel, you can define the Java video attributes. These video attributes indicate whether the controller video display uses monochrome support only, or if the controller will use Java graphics. If the controller uses Java graphics, you must define the color palette, which is the number of colors your monitor supports.

Note: Before performing any Java-based configuration utilities, enable Java graphics in the controller configuration. To enable Java graphics, select **Controller Configuration**; then select **Video Attributes**. When asked if Java graphics will be used by the controller, select **Yes**. Then, activate the change and IPL your system before Java graphics are enabled.

If you are using Java 2 or Java 6, see the "Java 2 problem resolution" section and the "Additional hints and tips" section in the *4690 OS: Programming Guide*.

NFS backup for Java 2 terminal applications

By default, Java 2 programs can only be loaded if the NFS drive (L:) referred to by the JAVA2BIN: logical name is available. Because the JAVA2BIN: logical name, typically, points to the VFS drive (M:) on the primary controller, Java 2 is not able to load if the primary controller is down and even if there is a backup controller available. To address this issue, configure a list of alternate locations for the Java 2 executable programs by creating the file, ADX_IDT1:ADXTRMUF.DAT, if this file does not already exist. Add a line to the ADX_IDT1:ADXTRMUF.DAT file containing the list of the JAVA2BIN: locations you want to use (including the default JAVA2BIN: setting 1:\java2\jre\bin), for example: ADXJ2TPTH=k:\java2\jre\bin;l:\java2\jre\bin .

Set ADXJ2TPTH to a semicolon delimited set of path names that refer to a Java 2 installation on the primary or backup controllers that are used for terminal loading. The path names should refer to the directory containing the JAVA.386 executable file. **There must not be any blank or null spaces in the line.**

When the terminal application loader attempts to execute a Java 2 program, the terminal application loader first checks to see if the Java executable file exists in the current JAVA2BIN: directory. The initial setting for the JAVA2BIN: logical name is 1:\java2\jre\bin\. If this file exists or if ADXJ2TPTH is not set, the terminal application loader attempts to start the Java 2 program, logging any errors that it receives. However, if the program file does not exist and ADXJ2TPTH is set, the terminal application loader looks for the program file in each directory in the list. If the terminal application loader finds the file, the terminal application loader **redefines the JAVA2BIN: logical name** to be equal to that directory and then executes the program. If none of the listed directory names contains the Java

executable file, the JAVA2BIN: logical name is not changed. However, even in this case, the terminal application loader still attempts to start the Java program and an appropriate error message is logged.

There are several important points to be aware of when using the capability described above:

- The terminal application loader does not try to emulate RIOAM (R::) file access. In particular, files opened using the R:: prefix are opened on the acting file server, including the program names used when loading Java programs and classes. Therefore, if a terminal configured to run a Java program is initially started when the backup controller is the acting file server, the Java program is loaded from the backup controller. If the primary controller is then activated and the Java program is restarted, the Java program is loaded from the primary controller. The Java 2 backup capability, as described above, does not take the location of the acting file server into consideration. The Java 2 backup capability only looks at whether the Java 2 executable file is accessible at a given directory location. In addition, when reloading a Java 2 program, the Java 2 backup capability does not check to see if the original JAVA2BIN: location is available. The Java 2 backup capability continues to use an alternate load directory until that directory is no longer available.
- The version of Java 2 files at every backup location listed in ADXJ2TPATH *must be identical*. This essentially means that the locations must be on backup file servers for the same controller. When using Java TOF support, many of the TOF files are stored in the m:\Java 2 directory on the primary controller and distributed to the backup controllers. These TOF files must also be available at each alternate location. For additional information regarding TOF, see the *4690 OS: Programming Guide*.
- If your current Java 2 classpath setting refers to files on the L: drive, be aware that those files on the L: drive will be inaccessible if the primary controller goes offline. There are several ways to address this issue; however, in all cases ensure that the file is correctly distributed to all backup controllers. If the file is in the m:\java directory on the controller, you can use the JAVA2BIN: logical name to refer to the file, for example, JAVA2BIN:..\..\my.jar. Another method is to place the file on the C: drive on the controller and distribute the file. The terminal classpath could then use R:: to access the file.
- The user.home property in Java 2 is currently hardcoded to l:\java2 on terminals and the user.dir property is hardcoded to L:\. These values are not modified if Java 2 is loaded from an alternate location. If all of your data files are distributed to all controllers or if you always want them to refer to valid disks, it is advisable that you change the default values to be relative to java2bin:. For example, change the current directory for Java applications, as specified by the TJAVA2DEF logical name, from L:\ to java2bin:../../../. Making this change, basically, means that the default directory for terminals would be the root directory of the drive where Java 2 is loaded from, which would still default to L:\. You could also change the names of either of these properties using the -D command line parameter to override the system property values. Specifying the names this way ensures that the current directory and the user home directory are still valid if Java 2 is loaded from a backup controller. See the *4690 OS: Programming Guide* for additional information on how to change the home and default directory settings.
- The property, java.home, is based on the location that the JVM is loaded from and is set to the directory immediately before the location of the Java executable file. By default, the java.home property is set to l:\java2\jre on terminals. If the first alternate path in the above example was used, the java.home property would be set to k:\java2\jre. If your application depends on the existence of user files

relative to this directory, ensure that the files are correctly distributed to all controllers that Java 2 might be loaded from.

Installing and running a Java application

Note: Before performing any Java-based configuration utilities, enable Java graphics in the controller configuration. To enable Java graphics, select **Controller Configuration**; then select **Video Attributes**. When asked if Java graphics will be used by the controller, select **Yes**. Then, activate the change and IPL your system before Java graphics are enabled.

When installing a Java 2 package that results in expanded directories with long file names, install the package on the M or N drive. When installing a Java 6 package, all files directly accessed by the JVM (jar files, properties files, etc) need to be placed on the F: drive.

To install and start a Java application from a terminal or controller, perform the following steps:

1. Enable Java graphics during controller configuration.
2. Enable VFS for the C: drive.

Note: In 4690 OS V6, VFS is enabled for the C: drive by default.

3. Copy the .ZIP or .JAR files into a directory on the 4690 system (do a `chkdsk -F`, if necessary).
4. Add the .ZIP or .JAR files to the terminal or controller classpath.

Notes:

- a. When running a Java application from a controller/terminal, add the .ZIP or .JAR file to the controller classpath.
 - b. Ensure that the full path name for the terminal classpath, which refers to an RIOAM file, adheres to the RIOAM file-naming restrictions. See “NFS backup for Java 2 terminal applications” on page 294 for information regarding file-naming restrictions.
5. Distribute the files using the Distributed File Utilities.
 6. Activate the system configuration.

Notes:

1. On a terminal that has a Java application loaded, pressing **Alt+SysRq J** or **Alt+SysRq T** before the terminal is fully loaded results in an inability to access the Java console. Java runs but is not accessible.
2. On a controller, you cannot toggle between a Java window and the command line that started the Java application. To return to that specific command prompt, the Java program must be exited.
3. All printing from Java must be done using a PostScript-type printer.
4. For versions of the 4690 OS prior to Version 5, the LAN Disk Rebuild Utility could not handle complete file names that were longer than 47 characters. For example, `ADXLXTCN::H0://ADX_UDT1/LEVEL1/LEVEL2/THEFILE1.TXT` exceeds the limit starting with TXT. Version 5 adds support for complete file names that are a maximum of 128 characters, which is the limit set by the OS shell exceeds the 47-character limit beginning with the TXT.

Running Java 2 JVM applications from the command line

Starting with 4690 OS Version 6, Java 2 is the default JVM. The default value of the logical name "path" is "default: system: java2bin:". The logical name java2bin: points to the location of the Java 2 JVM. If the path has been modified from this default either in configuration or BAT files then java2bin: will need to be added to it if you wish to use Java.

Running Java 6 JVM applications from the command line

There are several methods to start an application with the Java 6 JVM. The new logical name, javaebin:, points to the location of the executable programs for the Java 6 JVM. The javaebin: logical name is used in the following methods, although using the explicit path name is also valid.

To run Java 6 JVM applications from a command line, use one of the following options:

- Prefix the name of the Java 6 JVM application that you want to use with an explicit path name by entering:
`javaebin:java.386`
- Make the current directory *javaebin:* before running the Java 6 JVM application by entering:

```
cd javaebin:
```

Because the current directory is stored in the logical name default, the Java 6 JVM applications appear in the path before the Java JVM 1.4.2 applications.

- Using the define command, include the javaebin: logical name in the path by entering:

```
define path=default: system: javaebin:
```

Running Java applications from a terminal

To run a Java application from a terminal, use the Java Application Functions option from the TERMINAL FUNCTIONS menu.

A Java application in the terminal cannot use the R::M:\filespec method to access long file name support on the controller. The application must access long file name support by using a drive letter that has been defined in the terminal as being part of an NFS mount group. The mount group should point to the remote controller's exported M or N drives, which contain the files the application needs.

When accessing long file name support on a controller from either a terminal or controller you cannot use the ADXLXccN:: NetBIOS node name convention. To access long file name support on a remote controller, an application must use a drive letter that has been defined as being part of an NFS mount group. The mount group should point to the remote controller's exported M or N drives, which contain the files the application needs.

Note: If a terminal Java application indicates that a file is not found on the controller but the file does exist, make sure that the terminal application is requesting the file in the R::XXXXXXXX:XXXXXXXX.XXX format. If the file name does not adhere to the RIOAM naming restrictions, the controller indicates that the file cannot be found. See "NFS backup for Java 2 terminal applications" on page 294 for information regarding file-naming restrictions.

Configuring multiple graphical Java applications

Configuring multi-app	298
Property file	298
Configuring multi-app on a terminal	299
Running multi-app on a controller	299
Configuring classpath	299
Window Manager Dialog	299
Accessing the Window Manager Dialog	299
Closing the Window Manager Dialog	300
Using the Window Manager Dialog window operations	300
Using the Window Manager Dialog application operations	300
Errors when starting applications	301

This section explains how to configure and run multiple graphical Java applications on 4690 controllers and terminals. The phrase *multiple graphical Java applications* will be referred to as *multi-app*.

Some of the multi-app capabilities are made directly available to Java applications. The Java API for using multi-app is documented in the *4690 OS: Programming Guide*.

4690 OS Version 4 added enhancements to Java multi-app configuration which allow setup to be done using Generic Terminal Configuration. See Chapter 10, “Enhanced terminal preload and Java configuration,” on page 305 for more information.

Note: MultiApp is not supported for Java 6.

Configuring multi-app

Up to four graphical Java applications can be configured to run at the same time when using multi-app. See “Using graphics with the operating system” on page 301 for more information on how to configure graphics on 4690 OS. One of these applications must be designated as the primary (parent) application. The other applications, up to three applications, are then designated as secondary (child) applications. The significance of an application's role is that if the primary application terminates, then all of the secondary application terminate also. Although, the opposite is not true. That is, if a secondary application terminates, then no other applications are automatically terminated. Typically, the primary application would be a Point-of-Sale application and the secondary applications would be applications of lesser importance, such as a Web browser.

Property file

Configuring multi-app is done via the multi-app property file. The property file provides information such as what applications should be run, what the applications' roles are (such as primary or secondary), and what arguments should be passed to the application. The sample file, C:/JAVA/MULTISMP.PRP, is provided to assist in getting started. The sample file should not be used to run multi-app in your store. Instead, copy the sample file to a new location and edit the copy. The name of the property file has no restrictions beyond those of the 4690 file system requirements, for example, 8 dot 3 name on the C or D drives. The properties that can be defined in the properties file are described in the *4690 OS: Programming Guide*.

Note: Although a maximum of four applications can run at one time, any number of applications can be defined in the property file. By defining an application as

GUIStartable, the application is made available for the user to start. Any number of applications can be defined as GUIStartable.

Configuring multi-app on a terminal

After a property file has been created for multi-app, the terminal can be configured to run the multi-app startup class. The multi-app startup class simply reads the property file passed in as an argument and starts the parent application and child applications with `onStartup` set to true. To configure the multi-app startup class on a terminal, define the Java class and parameters in the terminal's load definition as follows:

```
com.ibm.OS4690.multiapp.MultiAppStartup <property file>
```

Note: If the property file resides on a drive that is accessed through RIOAM (C and D drives) then an `R::` must be appended to the name of the file as follows:

```
com.ibm.OS4690.multiapp.MultiAppStartup R::<property file>
```

Running multi-app on a controller

MultiApp is not necessary on a controller because a controller can already run several graphical applications at one time. However, multi-app can be run on the controller, allowing several graphical applications to be seen at one time. To run multi-app on a controller, type the following text in a command session:

- For Java 2:

```
java2bin:java com.ibm.OS4690.multiapp.MultiAppStartup <property file>
```

Note: No additional configuration is needed on a controller to run multi-app.

Configuring classpath

For the Java 2 Virtual Machine, all classes are contained in `M:/JAVA2/JRE/LIB`. This means that no classpath changes need to be made specifically for multi-app.

Configuring the classpath for multi-app can be very error prone. It is not recommended that the classpath be configured for all the applications being run with multi-app. For the parent application, the standard classpath configuration mechanism should be used. That is, the Java configuration utility under system. For secondary applications, the classpath should be configured by explicitly specifying the classpath in the "systemArgs" property. For example, if the application needs to have a classpath such as:

```
.;c:/java/lib/classes.zip;c:/java/lib/OS4690.zip;c:/classes/other.zip
```

Then its `systemArgs` property in the property file appears as follows:

```
app1.systemArgs=classpath
.;c:/java/lib/classes.zip;c:/java/lib/OS4690.zip;c:/classes/other.zip
```

Window Manager Dialog

The Window Manager Dialog provides basic window operations, such as minimize, maximize, restore, raise, and close. The Window Manager also provides functions related to multi-app management, such as starting and stopping applications. The Window Manager Dialog is only available for Java 2 applications.

Accessing the Window Manager Dialog

The Window Manager Dialog can be accessed in the following three ways:

- Press the key combination: **Ctrl + Esc**
- Click the mouse on the wallpaper

- Press the system sequence: **S1 4 S2**

Notes:

1. The **Ctrl + Esc** key combination does not work on an ANPOS keyboard if the keyboard is in POS mode, such as, when sending keystrokes to JavaPOS in a POS application. If you find that **Ctrl + Esc** is not showing the Window Manager, try using the **S1 4 S2** sequence.
2. If **Ctrl + Shift + Esc** is pressed in error, the system will switch to underlying console with stdout and stderr. To revert back to graphics mode, press **Ctrl + Shift + Esc** a second time.

Closing the Window Manager Dialog

The Window Manager Dialog can be closed using either of the following methods:

- Perform one of the access methods, for example, **Ctrl + Esc** a second time.
- Click **Cancel** located in the lower right-hand corner of the dialog box.

Using the Window Manager Dialog window operations

In the center of the Window Manager Dialog there is a list of all windows that are currently open on the screen. A row of buttons along the top of the Window Manager provides actions that can be performed on each of these windows. To perform an action on a window, select the appropriate window from the windows list and press the button for the desired action. The behavior of each action button is as follows:

Minimize

Shrinks the selected window into an icon, if the window is resizable.

Restore

Restores the window to its last non-iconized/non-maximized state.

Maximize

Makes the window the size of the screen, if the window is resizable.

To Front

Raises the window above all other windows.

Close Closes the window.

Note: The **Close** action from the Window Manager Dialog behaves the same way as the close action from the Window's menu. If the window does not close when the Window Manager's **Close** action is performed, then the application is choosing not to honor the close request.

Using the Window Manager Dialog application operations

The Window Manager allows you to start and stop applications using the two buttons at the bottom of the screen that are labeled "Start Application..." and "Stop Application...".

Press **Start Application** to bring up the Start Application Dialog. The Start Application Dialog contains a list of all applications available for you to run. These applications are specified as GUIStartable in the properties file. To start one of these applications, select the application from the list and press **OK**. Press **Cancel** to close the Start Application Dialog without performing any actions.

Press **Stop Application** to bring up the Stop Application Dialog. The Stop Application Dialog contains a list of all currently running applications that you are allowed to stop. These applications are specified as GUIStoppable in the properties

file. To stop one of these applications, select the application from the list and press **OK**. Press **Cancel** to close the Stop Application Dialog without performing any actions.

Errors when starting applications

If an error occurs when starting an application, a dialog providing information on the error appears. If the new JVM was able to successfully initialize and begin invoking the main method of the application, the following errors can be determined and shown in the dialog.

- Exit Code 311: Specified class could not be found.
- Exit Code 312: Class did not succeed to initialize.
- Exit Code 313: No main() method defined for the specified application.
- Exit Code 314: Graphics failed to initialize.

When an error occurs, check stdout and stderr for information that might help determine the problem. By default stdout and stderr are written to the console. Otherwise, if redirection is being performed, check the files that contain the redirection.

Using graphics with the operating system

The operating system provides a graphics capability that is accessible only through Java. Java controls for a graphical interface with windows, dialog boxes, and pull-down/pop-up menus are part of the graphics capability.

Notes:

1. Systems with Super VGA (SVGA) graphics adapters and monitors support these graphics levels. However, not all hardware platforms support these graphics levels, so it is very important to configure your system correctly.
2. Java can run on monochrome displays in a non-graphics mode. Graphics is not supported on monochrome monitors.

When enabling Java, graphics support must be specifically requested, and verify that your monitor supports the required number of colors. If graphics is not configured, or if you indicate that your hardware does not support the required graphics level, graphics will not start when requested. See the *4690 OS: Planning, Installation, and Configuration Guide* for more information on configuring Java graphics.

Java graphics uses a combination of user-configured parameters and hardware-detected values to ensure that you will not damage your monitors accidentally by running at unsupported graphics levels. A pointing device (mouse) is supported while using graphics at both the controller and the terminal.

To use Java graphics:

1. Configure Java graphics, and also specify the screen resolution and color format for your video display.
2. During the IPL process, the Java graphics driver queries the video hardware and confirms that the graphics adapter and attached monitor support the requested level.
3. Start the Java graphics application.
4. The Java graphics driver puts the system into graphics mode as requested.
5. Enhanced-mode terminals support a unique resolution and color format for a secondary video display, if attached, or they can use the same settings as the primary video display.

If Java graphics does not start on your system, there could be several causes. First, you might not have configured Java graphics. If Java graphics is not configured and you start a Java graphics application, the system logs an error message and does not put the system into graphics mode. (Because Java graphics cannot confirm that your hardware can handle the required graphics level, it will not start rather than damage the attached monitor.)

Second, if you configure Java graphics and indicate that the system supports 256 colors and during the subsequent IPL the Java graphics driver detects that the required graphics level is not supported, the Java graphics driver logs a warning message. If you attempt to start a Java graphics application after the IPL, the application does not start and an additional warning is logged.

Finally, if you do not configure Java graphics and your hardware correctly, the Java graphics driver might attempt to put the system into graphics mode when you try to run a Java graphics application. In this scenario, a message is not logged because the Java graphics driver does not detect a problem. For example, if your system has a monochrome monitor and you configure Java graphics and indicate that the system supports 256 colors, the system might display a blank application screen when you attempt to run a Java graphics application.

See the *4690 OS: Messages Guide* for information on Java graphics messages.

Using multiple video displays

This section discusses the use of multiple video displays by graphical Java applications on a terminal.

A video display is referred to as either the primary video display or the secondary video display. The primary video display is that which is plugged into the built-in video card. The secondary video display is that which is plugged into the PCI video card. Each application using graphics is assigned a default display, which can be either the primary video display or the secondary video display. The default display for an application can be configured on a per application basis.

A second video adapter is not required by all supported system types. The SurePOS 700 Series Models 7x3, 7x4, and 7x5, as well as the SurePOS 300 Series model 350, provide two built-in video ports and do not support the external video adapters discussed in the next section. In this case, the secondary video display is the display connected to the secondary built-in video card or port (port B).

An external video adapter is also not supported by TCxWave 6140 Series systems. TCxWave 6140 Series systems provide a built-in video display, which always acts as the primary video display. A Display Port is also provided on TCxWave 6140 Series systems for connecting a second video display. Video displays connected to the Display Port will always act as the secondary video display.

Supported video cards

A limited set of video cards are supported for the secondary display. See `c:\java\dualvid.txt` for a list of the supported video cards.

SurePOS 700 Series Models 7x3, 7x4, and 7x5, the SurePOS 300 Series model 350 and the TCxWave 6140 Series do not support external video adapters. Instead, they provide a built-in second video port. These models do not require a change to the BIOS setting.

Supported video dongles

TCxWave 6140 Series systems provide a Display Port for connecting secondary video displays. A Display Port Dongle can be used to convert the Display Port output into VGA signals. The TCxWave 6140 Series supports a limited list of Display Port Dongles.

Display Ports and Display Port Dongles are only supported for the TCxWave 6140 Series systems.

Configuring dual display

Configuring dual display requires a change in the terminal's BIOS settings. In general, it is a matter of specifying the "boot", "primary", or "default" display as the AGP or built-in video card. For specific information on this procedure, see `c:\java\dualvid.txt`.

Configuring the default display

Unless explicitly configured otherwise, an application's default display is always the primary video display. The multi-app function is required to configure an application's default display to be the secondary video display. For information on how to configure the default display using multi-app, refer the Startup Property File section in the *4690 OS: Programming Guide*.

Secondary video display function limitations

The secondary video display does not support all of the functions that are available on the primary video display. Touch input is supported, but input from a mouse or keyboard is not supported on the secondary video display.

Note: Any pointing device attached to the terminal will effect the cursor on the primary display including a mouse attached to the secondary display.

Accessing multiple displays from a single application

Both Java 2 and Java 6 provide a standard API for accessing multiple displays. Refer to the method `java.awt.GraphicsEnvironment.getScreenDevices()` as a starting point.

Chapter 10. Enhanced terminal preload and Java configuration

Overview

4690 OS Version 4 added several improvements in terminal loading and configuration. Most of these changes are available in Generic Terminal Configuration. This list of improvements is described in this chapter:

- **Enhanced RAM disk** – You can configure an enhanced RAM disk for both the controller and terminal. The enhanced RAM disk (drive letter Q:) can be much larger than the legacy RAM disks and also supports long file names. The disk is configurable on the same screens as current controller and terminal RAM disks are configured. For a controller/terminal, the size of the enhanced RAM disk in the terminal device characteristics is ignored.
- **Long file name support for terminal hard disks** – With 4690 OS Version 4 or higher, when a terminal hard-disk drive C: is formatted, a VFS drive supporting long file names (M:) is automatically created on the terminal. This is similar to how long file name support works on the controller; the C drive actually holds the data for the M: drive. If the terminal hard disk drive is unformatted or was not formatted with 4690 OS Version 4 or higher, then the hard disk drive is not modified and an M: drive is not created. With 4690 OS Version 6 Release 2 or later, a terminal loaded in Enhanced mode will also have an F: drive that supports long file names for use with Java 6 or later. The F: drive is a RAM disk that requires no configuration on terminals without a hard disk, or is a separate partition on the hard drive if one is present. This means that data on the F: drive may or may not be persistent after reboot, depending on whether or not the terminal has a formatted hard disk.
- **Java Application Configuration** – You can configure Java Terminal Offline support (JavaTOF) and multiple Java application support (MultiApp) within Generic Terminal Configuration. The generation of the TOF and MultiApp properties files and the terminal response file is handled automatically and transparently. All required files (including the TOF archives and JVM) are transferred automatically to the terminal, if requested. This is available only for Java 2 terminal applications.
- **Enhanced Disk Preloading** – It is possible to cause files to be preloaded to the legacy terminal RAM disks (X: and Y:). The existing support is unchanged, however these enhancements are available in Generic Terminal Configuration:
 - Files can be preloaded to any available terminal drive (C:, M:, Q:, X:, or Y:). However, on a controller/terminal, files can only be preloaded to RAM disks.

Note: F: is a terminal preload target (for Enhanced Terminal only). F: is not supported for an Enhanced Controller/terminal.

 - Multiple bundles can be configured. The same bundles do not need to be preloaded to every terminal or to the same drive on each terminal. (It is controlled on a load definition basis).
 - Wildcards are supported.
 - Files can be marked as “optional”.
 - Subdirectories are supported (also on legacy RAM disks).
 - Long file names are supported, if the target drive supports them.

- **Caching of data on the terminal** – Data is transferred to the terminal only as needed. If the terminal has a hard disk, the TOF archives and preload bundle data files are saved on the hard disk. When the terminal is reloaded, the files are only transferred if they are changed.

No existing capabilities have been removed. Current configurations will continue to work as they did before. Java application configuration and terminal preloading can still be done in the manner they were done previously. JavaTOF and MultiApp can still be used manually as well. The only functional change will be the automatic creation of the M: drive when the C: drive on a terminal is formatted.

One requirement to be aware of: A terminal must be configured to use TCP/IP in order to take advantage of some of the enhancements. Specifically, when preload bundles are configured or an enhanced Java application is configured, TCP/IP is used to transfer additional files to the terminal. An enhanced RAM disk can be configured and used (along with the M: drive) without TCP/IP support.

Java application setup

Java terminal applications can now be configured completely in Generic Terminal Configuration. Applications are configured independently and then added to a specific set of terminal load definitions. The Java version to use is specified in the terminal load definition. This allows several applications with a complex set of parameters to be configured at one time and easily changed back and forth. If desired, the use of JavaTOF can also be enabled when setting up an application to be run using Java 2.

Basic application configuration is similar to how legacy configuration is done. However, instead of providing a single command line with both JVM and application arguments, multiple fields are used to prompt the user for information. When using legacy application configuration, a maximum of 64 characters can be entered for the entire Java command line. If the user exceeds this limit, the command line must be placed in a response file and the name of the response file added to the load definition instead. This limit is not present for new applications that use the enhanced JavaTOF in 4690 OS V4 and later releases. A response file is still used, but it is generated automatically. The following fields have been added:

Classpath

This is where the program's classpath is specified. When configuring an application with this method, the classpath is required and the terminal classpath in system configuration is ignored. Normally, the JAR files and directories in the classpath must be accessible from the terminal (using the terminal drive letters and logical names). However, if JavaTOF is enabled, these files and directories must be accessible from the controller since TOFCreate reads this data to produce TOF archive files.

Note: JavaTOF is not supported in Java 6.

Other JVM options

This field is where other JVM options are specified, such as setting properties and Java heap size. All other switches that would normally appear before the application classname should be placed in this field.

Classname

This is the name of the application class to start. If you are using the JavaTOF or MultiApp capabilities built into enhanced JavaTOF configuration, do not specify TOFStartApp or MultiAppStartup classes as

your application name. These classes will be called automatically if needed. Instead, you should put the name of your actual application class in this field. The application class name will be placed in the correct response or property file as needed.

Note: JavaTOF and MultiApp are not supported in Java 6.

Application arguments

The values typed in this field are passed to the main method of the application.

The advanced section of Application Configuration allows for some portions of JavaTOF and MultiApp setup to be done. There are also separate sections for JavaTOF resource creation and dependency checking. The advanced application setup screen allows the use of JavaTOF to be selected. See the *4690 OS: Programming Guide* for more information on what JavaTOF is and why it can be useful. However, when JavaTOF is enabled in configuration, the generation of the TOF property file and the running of TOFCreate are done automatically. In addition, any class and resource archive files are automatically transferred to the terminal, if desired.

You do not have to use JavaTOF to run Java applications on a terminal in offline mode when using the enhanced terminal preload capabilities of 4690 OS Version 4 or higher. The only requirement is that all JVM and application files are copied to a hard disk or RAM disk on the terminal. You copy the JVM-related files by preloading the JVM to the terminal (see “Terminal load definitions” on page 309). You copy application data files, which includes JAR files, by creating bundles for those files and preloading them to the terminal (see “Preload bundles” on page 309).

The other items on the advanced configuration screen actually control Multiple Java Application support (MultiApp) capabilities. See the *4690 OS: User's Guide* for more information on MultiApp. As with enhanced JavaTOF support, the MultiApp property file is generated automatically, if needed. The additional fields are documented more fully in the online help in Generic Terminal Configuration, however if you are converting existing MultiApp property files, this is how each field maps to a MultiApp property:

GUI Startable

Selecting this is equivalent to setting `appl#.GUIStartable` to true.

GUI Stoppable

Selecting this is equivalent to setting `appl#.GUIStoppable` to true.

Launch on Startup

Selecting this is equivalent to setting `appl#.onStartup` to true. The primary application is always launched on startup.

Run on a secondary display

Selecting this is equivalent to setting `appl#.defaultDisplay` to Secondary. Configuration activation will fail if this option is selected for the primary application in a load definition.

Maximum number of instances

Setting this is equivalent to setting `appl#.maxInstances` to the same value and limits the total number of running instances of that application. There is a slight difference in behavior with respect to MultiApp. MultiApp doesn't consider the primary application as being an “instance” of that program. Thus, if a property file contains the property “`appl0.maxInstances=1`” and `appl0` is the primary application, the user will be able to start an additional

instance of the primary application (for a total of 2). If the same value was set for a secondary application, the user would be limited to starting a single instance. When the maximum number of instances on this screen is set to 1, however, the user will only be able to start a single instance of an application regardless of whether it is a primary or secondary application.

JVM ID

Setting this to anything other than “None” is equivalent to setting `appl#.jvmID` to the same value. MultiApp startup will fail if more than one application configured to launch at startup has the same JVM ID.

Redirect ... normal output

Setting this is equivalent to setting `appl#.stdOut` to the same value.

Redirect ... error information

Setting this is equivalent to setting `appl#.stdErr` to the same value.

The rest of the screens in Java Application Setup are related to JavaTOF. The online help in GTC provides more information for each of these screens. The information that follows is provided to aid users that are migrating TOF property files.

When running TOFCreat manually, the four properties in the property file are used to specify list files. These files contain lists of classes or JAR files that are used during TOF archive creation. In many cases, these files might contain hundreds of entries, which can be cumbersome and error prone to enter manually. For this reason, each of the TOF lists described below allow the use of list files. If a value in the list starts with an at-sign character (@), then the file will be opened during the preload build process activation and the contents of the file will be used as input to TOFCreat. Each list can contain a mixture of normal class/jar entries and list files. This list capability can be useful if an application has specific TOF setup requirements that may change due to application maintenance. If an application is configured to use list files for TOF related information, those list files can be updated by software maintenance as needed without requiring configuration changes.

The TOF Resource Creation screen allows the specification of a list of JAR files from which TOFCreat will extract resources. Adding JAR files to this list is equivalent to adding them to the file specified by the `resource.creation.use.resourcefilelist` property in the TOF property file.

The TOF Dependency Checking main screen allows for miscellaneous options to be passed to TOFCreat.

- Selecting the Output debug information... checkbox is equivalent to setting the properties `depchecking.debugon` and `resource.creation.debugon` to true in the TOF property file. Debug output for a TOFCreat run is written to the file `C:/ADX_STLD/INACTIVE/AppName.TLG`, where `AppName` is the name of the application in the Java Applications configuration screen.
- Setting the JavaPOS version on this screen to anything other than None is equivalent to setting the property `depchecking.javapos.version` to the same value.

The TOF Dependency Checking Include List allows the specification of a list of classes to include in dependency checking. Adding JAR files to this list is equivalent to adding them to the file specified by the `depchecking.use.includefile` property in the TOF property file.

The TOF Dependency Checking Exclude List allows the specification of a list of classes to exclude from dependency checking. Adding JAR files to this list is equivalent to adding them to the file specified by the `depchecking.use.excludefile` property in the TOF property file.

The TOF Dependency Checking JAR List allows the specification of a list of JAR files from which TOFCreate will include dependencies. Adding JAR files to this list is equivalent to adding them to the file specified by the `depchecking.use.jarfilelist` property in the TOF property file.

Preload bundles

A *preload bundle* is a group of files that can be transferred to a terminal as a single compressed unit and then extracted to a specific drive. The selection of which bundles are transferred to a specific terminal and the drive to which they are transferred is done in the load definition screen. Bundles can be preloaded to a terminal even if Java is not configured for that terminal.

When a bundle is created or edited, one or more files are added to the bundle. Each file specification in the list can contain wildcard characters. The supported wildcard characters are “?”, which matches exactly one character, and “*”, which matches zero or more characters. Directory recursion is also supported. This allows the user to include, for example, all files matching the specification in a subdirectory tree.

Each file specification can be marked as optional. If a file is not optional and the file does not exist or no matches are found for a wildcard specification, then an error is generated when the terminal preload files are being built.

When a file is added into a bundle and that file is in a subdirectory, the file will by default be placed in the same subdirectory on the terminal, with directories being created as needed. The file may be optionally placed in the root directory. This option should be selected if you wish to emulate how legacy RAM disk preloading works. See the online help for more information about this option and how it interacts when directory recursion is selected.

Note: Files loaded using enhanced terminal preload are loaded prior to files loaded using the legacy preload method described in Chapter 21, “Using Terminal RAM Disk Preload Utilities,” on page 419. Thus, if a file with the same name is loaded to the root directory of a disk using both methods, then the file loaded by the legacy preload will be the final copy of the disk after the machine is fully loaded.

Files within a bundle are always compressed before being transmitted to a terminal. Preload bundle configuration does not have the capability to store the files in uncompressed form, as did the legacy RAM disk preload facility. With the form of compression being used, however, this will not cause a problem even if the data being transferred is already compressed.

Terminal load definitions

The load definition screen is where a set of terminals can be configured to run one or more Java applications and to use preload file bundles. To select one of the newly configured Java applications, you must deselect the Use the legacy screen checkbox. You may then select one primary application and one or more secondary applications from the list. You may select any number of secondary applications.

However, there is a MultiApp limitation that you can only start a maximum of four applications, including both primary and secondary applications, at any one time.

The Preload to Drive option is important primarily for applications for which JavaTOF has been configured. This option is used to control where TOF archives are placed. However, it is also important for applications that do not have JavaTOF configured because response and MultiApp properties files are also preloaded to this location. When JavaTOF is not being used for the applications, the total size of the files copied to the terminal is very small (less than 1KB per application).

Use the Preload Bundles screen to associate defined bundles with a terminal load. This screen allows a bundle to be preloaded to a specific drive for this load definition. Because the bundle is removed from the available list when it is added to the list for a specific drive, a bundle can be loaded to no more than one drive on the terminal.

The JVM Preload screen allows selection of the drive letter to which JVM-related files are preloaded. Although None is an option, it is not recommended because the JVM cannot tolerate losing access to its files, especially the font files. (JVM can lose access if a terminal is accessing files and the terminal goes offline.) If the JVM is not preloaded to a terminal drive, JVM files will be accessed through the NFS drive L: as with legacy Java 2 applications.

In Enhanced Mode, the Java 6 JVM is preloaded only when a Java 6 Application is specified in the Terminal Load Definition record.

When the Java 2 JVM is preloaded to all terminals, it is not necessary to set the TERMJAVA logical name to 2. If you do set the logical name to 2, Java DLLs will be added to the loadshrink file as well as preloaded to the terminal using the preload code. However, when TERMJAVA is set to 2, JVM files that are part of loadshrink will be used to run Java programs. If the JVM is being preloaded to a RAM disk, any JVM files already loaded by loadshrink will not be extracted to the RAM disk. This allows the required size of the RAM disk to be reduced. To provide as much memory as possible to Java 2 applications, you should set TERMJAVA to 2 even if the JVM is being preloaded to the terminal.

Note: TCP/IP is required in order to use advanced Java application definitions and preload bundles.

No checking can be done to ensure that the drive letter selected for JVM, application, or bundle preload actually valid on the terminal. There is no way to know whether any of the terminals in the list will have a C: or M: drive available or whether the terminal will have enough memory to allocate the requested RAM disks. These types of errors are detected when the terminal is loaded.

Migrating legacy setups to 4690 OS Version 6 Release 2 or later

You do not have to make changes unless you want to use new or enhanced functionality. For example, you can take advantage of some functionality, such as RAM disk preloading, while continuing to use legacy JavaTOF. This section lists various hints and other information that may be helpful in migrating existing configurations.

RAM disk preloading

If you are using RAM disk preloading:

1. Create one preload bundle for each legacy drive being preloaded (X: and/or Y:).
2. Add entries for each file name listed in either ADX_IDT1:ADXTRMXF.DAT or ADX_IDT1:ADXTRMYF.DAT to the appropriate bundle file. Type in the full path name for each file.
3. If the file is optional (the line contains the “-O” flag), then select the bundle checkbox that indicates the entry is optional.
4. For each entry, select the checkbox that indicates the file should be placed in the root directory.
5. If any of entries in the legacy list file are related to JavaTOF, then they should only be added to the bundle if you are planning to continue to use legacy TOF. If you will be using enhanced JavaTOF, it will transfer the files it requires automatically.
6. When you define terminal load definitions for your terminals, you should preload the new preload bundles from the appropriate drive.
7. After completing your configuration tasks, delete or rename the legacy RAM disk preload list files if they will not be used. Also, delete the RAM disk archive files that get appended to loadshrink (ADX_SDT1:ADXTRM1F.DAT and ADX_SDT1:ADXTRM2F.DAT).

MultiApp

The primary application selected in a load definition cannot run on a secondary display. Other than this restriction there is no real difference in how a primary application is configured versus a secondary application.

Current MultiApp configurations are probably using a terminal response file in addition to a MultiApp property file. Any JVM options from the response file, such as options that set properties or change the heap size, should be copied to the JVM Options field in Java Application Setup for the primary application. Any JVM options in the MultiApp property file should be copied to the JVM Options field in Java Application Setup for the appropriate secondary applications.

Setting fields on the Java Application Setup Screen

One potentially error-prone task during migration of an existing setup is determining how to create and set the fields for new applications. The method for doing this depends on how the existing application is started. One thing to remember in all cases is that the terminal classpath defined in system configuration is not used by enhanced application setup. The classpath must be explicitly set for each application. If the system arguments for the application being migrated do not contain a classpath, the currently configured Java 2 terminal classpath should be copied into the classpath field.

If JavaTOF and MultiApp are not being used, then all parameters for the application and JVM are either in the load definition or in a response file. In this case, you should create a single application and set the fields as follows:

- In most files, the -cp or -classpath JVM options are used to set the classpath. The value after this option (the list of file and directory names) should be copied into the classpath field in Java Application setup. The copied string should be copied as it is, including the semi-colons.

- All JVM options start with a dash; the application class name does not. Therefore, except for the classpath option and its value, you should copy these values to the JVM Options field:
 - All arguments that start with a hyphen, until the first one that does not start with a hyphen
 - All parameters prefixed with -D, which is used to set JVM system properties
 - Any options prefixed with -X
- The first argument after the JVM options is the application classname.
- All remaining arguments, if any, should be copied to the application arguments field.

If legacy JavaTOF is being used to start the main application, then TOFStartApp will be the application classname listed in the load definition or response file. The property file passed to that class will contain additional information needed for migration. In this case, you would create a single application and set the fields as follows:

- The classpath and other JVM options will be set by the contents of the response file or load definition. Refer to the previous paragraph for information on how to set these.
- The application name is set in the TOF property file using the property `application.classname`.
- The application arguments are set in the TOF property file using the `application.args` property.

If MultiApp is being used, then MultiStart or `com.ibm.OS4690.multiapp.MultiAppStartup` will be the application classname listed in the load definition or response file. The property file passed to that class will contain additional information needed for migration. In this case, you would:

- Create one application for each application listed in the property file. The `appl#.displayName` and `appl#.details` properties would probably be good starting points for choosing the values for the application name and description fields.
- For the primary application (the one with the `appl#.parent` property set to true), the system arguments will be in the load definition or property file. Refer to the paragraph above for information on how to set the classpath and JVM Options in this case.
- For other applications, the system arguments will be specified in the property file using the appropriate `appl#.systemArgs` property.
- The application name and arguments are set in the property file using the appropriate `appl#.classname` and `appl#.appArgs` properties, respectively. If the application class name is TOFStartApp, however, then TOF is being used to start the application. In this case, the TOF property file specified by the `appl#.appArgs` property contains the application information. Specifically, the `application.classname` and `application.args` properties in the TOF property file contain these values.

JavaTOF

Legacy JavaTOF configurations may change the location of the Java extension directory in the terminal response or MultiApp property file (via the JVM option `"-Djava.ext.dirs=X:/"`). The Java extension JAR files were then loaded to the location and the property change allowed the JVM to locate them. When configuring an application using Generic Terminal Configuration that uses JavaTOF, this option should not be specified in the JVM options field. When TOFCreate is run on the

controller, it will look in this location for Java extensions. Because the files will not be there, TOFCreat will either fail due to missing classes or will not include the classes the application expects. You should only have to set the `java.ext.dirs` property if it was passed to the JVM in the command that ran TOFCreat in the legacy setup or if the property was listed in the legacy TOF property file.

Note: JavaTOF is not supported in Java 6.

All files in the classpath must be accessible from the controller (that is, contain controller drive letters) when JavaTOF is enabled. If JavaTOF is not enabled, the paths must be accessible from the terminal.

Controller/terminal setup

These are important facts to remember:

- The JVM can only be preloaded to the Q: drive.
- Application files, such as TOF archives, can only be preloaded to RAM disks.
- The allocation of an enhanced RAM disk (Q:) must be done in controller configuration. The RAM disk configuration done in the terminal device characteristics for the terminal side of a controller/terminal is ignored.

Terminal Configuration Activation

Activation of a terminal configuration containing enhanced configuration options can include the following activities, each of which might take a long time to complete depending on the quantity of data being processed:

1. Removing outdated files from the inactive directory.
2. Reading and verifying enhanced configuration.
3. Building the preload bundle list file. If a bundle contains a non-optional file entry and that file cannot be read, an error will be detected.
4. Running TOFCreat on any applications that have JavaTOF enabled. This step will report errors if TOFCreat cannot find all dependent classes.
5. Creating archive files for all preload bundles and TOF applications.
6. Distributing all created files.

Note: The preceding steps are only done if needed. The TOFCreat step, for example, is skipped if none of the application JAR files has changed. Bundles are also not rebuilt unless the files contained in the bundle have changed.

Preload bundles and Java applications that are not referenced in any load definition are ignored. No file checking is done for unreferenced bundles. TOFCreat is not run for unreferenced JavaTOF-enabled applications.

During the file distribution step above, the preload file server (`mtftpdpl`) on the controller is paused temporarily. This allows any rebuilt preload files to be replaced. Any terminal that is currently preloading files from the controller will wait until the rebuild is complete (and the server has resumed) before continuing. If any changes were made, the terminals will reload to ensure everything is preloaded correctly. A successful activation will be followed by a rebuild of the terminal loadshrink file. All terminal loads should be complete before running activation or the manual rebuild so that all terminals get the updated loadshrink.

Normally, when there are preload-related configuration activation errors, the error messages displayed by the configuration program should be sufficient to explain the problem. However, if this is not the case or if you need more information while attempting to fix the problem, errors are also logged to the ADX_SPGM:ADXPLDMN.LOG file. This file contains a lot of additional information, so finding the error messages can be difficult. The error messages are prefixed with the <<ERR>> string.

Note: Before activating configuration on a system that has a controller configured as a controller/terminal, you might need to stop any Java terminal applications running on the controller/terminal. A running terminal application might have one or more files locked that would be replaced by the activation process, which would cause an error. This happens if JavaTOF is enabled for the application and the application files are preloaded to the C: drive.

Terminal Preload Rebuild Utility

The terminal preload process described in “Terminal Configuration Activation” on page 313 normally only runs during the activation of terminal configuration. However, the terminal preload files are rebuilt when software maintenance is applied. In this case, the rebuild is done to ensure that any files that are referenced by applications that have JavaTOF enabled or are in preload bundles get properly packaged for use by the terminals. The rebuild is done during the first reboot after software maintenance has been applied. After the rebuild is complete, all terminals are reloaded if any preload files changed. If the rebuild fails, the message W692 is logged.

The terminal preload rebuild utility is run as a background process. You can determine the status of this process by viewing the background process screen.

In many cases, users may want to update files in a preload bundle or in JAR files referenced by a JavaTOF-enabled application and cause the related preload files to be rebuilt. For this reason, the terminal preload utility can be run manually. To rebuild any terminal preload files that need to be rebuilt, run the ADX_SPGM:ADXPLDRB.386 program. On a multiple controller system, this program can only be run on the acting master controller. Status messages are written to the standard output stream for the program; detailed messages are written to the ADX_SPGM:ADXPLDRU.LOG file. If the program fails, the message W692 is logged.

When running the terminal preload rebuild utility, there are some additional optional parameters. These are the optional parameters:

-force When this parameter is specified, all preload-related files are rebuilt even if they appear to be up to date.

-loadshrink

When this parameter is specified, the terminal loadshrink file is rebuilt upon successful completion of the rebuild process. The rebuild is only done if one or more loadshrink-related files were rebuilt.

-reload

When this parameter is specified, all terminals are reloaded upon successful completion of the rebuild process. The reload is only done if one or more preload-related files were rebuilt. If the terminal loadshrink is being rebuilt, the reload will be done after the loadshrink is complete.

When the Terminal Preload Rebuild utility is run either manually or after ASM, the utility uses the currently active terminal configuration. When the rebuild is done during terminal activation, the currently inactive terminal configuration is used; the configuration is then made active if there are no errors.

It is possible to configure a controller to have a terminal session. Normally the terminal session starts early during OS load process. However, when the terminal preload rebuild utility is run after system software maintenance has been applied, the terminal session is not started until after the rebuild process is complete. If a non-acting master controller with a terminal session is initialized while the rebuild is in progress, the terminal session on the non-acting controller is not started until the rebuild process is complete on the acting master.

Terminal loading

When a terminal is loaded, it first determines whether it has files that need to be preloaded. These files can include preload bundles, JVM archives, and TOF archives. Each of these files is transferred from the controller, if needed, and copied or extracted to the correct drive.

If the terminal has a hard disk, the hard disk is used to store cached copies of preload files that have been successfully received. During a terminal load, information about the current versions of these files is retrieved from the mtftp server on the terminal's acting controller. If the version of a file on the server matches the locally saved version, the file is not transferred and the local copy is used instead. If the local version of a file is missing, out of date, or the terminal does not have a hard disk, then the file is transferred from the server.

When more than one terminal requests the same preload file from a controller, the controller multicasts the file, which means sending the file to all terminals at the same time. This is similar to how the OS is loaded into terminals and can reduce the amount of time it takes to load the terminals.

If one of the steps in a file preload fails, the error message W065 is logged indicating the cause of the error. In most cases, an entry is also added to the ADX_STLD:ADXTM###.LOG file, where ### is the terminal number.

As a terminal is being preloaded, several messages appear on the terminal's system display. These messages only appear if the terminal is configured to use some of the advanced Java or preload bundle configuration functionality. Message W064 appears when the terminal is attempting to contact its controller. After preload is complete, W065 or W066 appears, depending on the results of the preload.

Because a preload might take a long time, additional messages are also displayed on the system display. Because there is very little room on the display, symbols are used to indicate what is happening. The first two characters on the first line of the display are used to indicate the specific task that is occurring. The remainder of the first line and the second line are used to further explain that task. Most messages are only displayed until the next update occurs and can be replaced almost immediately after being displayed. The primary purpose of these messages is to indicate progress. Any important events, such as preload failure, are logged as error messages on the controller.

These character prefixes are used when displaying preload task messages:

- !!** Displays the controller that is being used for preload. The prefix is followed by the logical name of the controller.
- <<** Indicates a file is about to be received from the server. The prefix is followed by the filename being transferred. The second line of the display indicates the transfer progress for this file. When the transfer is complete, the second line will display the internal code of the transfer. After the transfer begins, the prefix is replaced by a different prefix.
- >+ or >*** Indicates this terminal is the active client for a multicast transfer. The active client is responsible for controlling the transfer of the file. There is only one active client for a specific file being transferred from a given controller. The second character of the prefix alternates between the given values for extremely large files as long as progress continues to be made.
- >- or >| or >/** Indicates this terminal is a passive client for a multicast transfer. Passive clients receive file data from the server but do not participate in the transfer, thus they may miss parts of the file if they requested the file after the transfer has begun or due to network congestion. If a passive client misses parts of a file, it will request the parts again after the active client is done. There may be multiple passive clients for any given transfer. The second character of the prefix alternates between the given values as long as it continues to receive data blocks from the server.
- >>** Indicates a file is about to be sent to the server. The prefix is followed by the file name being transferred. The second line displays the percentage of the file that has been transferred. When the transfer is complete, the second line displays the internal code of the transfer. This message is displayed when log files are being transferred back to the server, which typically only happens due to an error condition. If the file is large and the percentage value is not being updated frequently enough, the second ">" will blink to indicate that progress is still being made.
- =x** Indicates data is being copied to a file. The letter x in the prefix is set to the drive letter to which the data is being copied. The prefix is followed by the name of the file being transferred. The second line of the system display contains a percentage indicating how much of the data has been copied. This message is displayed when an application JavaTOF archive is being copied to a RAM disk. The message is also displayed in some cases when a terminal log file is being copied to the controller (in which case the drive letter will be 'R').
- <>** Indicates the terminal received a recoverable error when attempting to receive a file from the server. The terminal will wait for a little while before attempting the transfer again. The prefix is followed by the name of the file being transferred. The second line displays the internal reason code for the error followed by the number of seconds left before the transfer is reattempted. Recoverable errors can be caused when the network is congested, or the server is too busy and cannot respond to requests quickly enough. A recoverable error is also generated when the server has been paused by the terminal preload rebuild utility; in this case, however, the terminal reloads when the server is resumed in order to ensure the terminal receives any updated files.
- ++** Indicates room for a preload file is being allocated on the hard disk or RAM disk of the terminal. For large files, this process can take some time, so a

message is used to indicate what is happening. The prefix is followed by the name of the file for which space is being allocated.

***x** Indicates an archive file (JVM or preload bundle) is being extracted. The letter *x* in the prefix is set to the drive letter to which the file is being extracted. The prefix is followed by the name of the file being extracted. The second line of the system display indicates the index of the file entry within the archive that is being extracted followed by the name of that file. The index is displayed in the format *x/y*, where *x* is the index of the file within the archive and *y* is the total number of files in the archive. When large file entries are being extracted, the name of the archive (on the first line) is followed by the percentage of the current entry that has been extracted. For example, if half of the large file *jvm.dll* is being extracted to the M: drive from the JVM archive file *ADXJ2BIN.BZP*, the following may appear on the system display:

```
*M adxj2bin.bzp 50%  
  3/24          jvm.dll
```

When the name of the entry being extracted is too long to fit on the display, only part of the name will be displayed. Also, depending on the capabilities of the display being used, the message might always appear in uppercase characters. Until extraction is started, the drive letter might be briefly replaced with an asterisk (*).

ADXPLDL

Used to display the final return code for the program. This message is usually displayed only briefly and is immediately replaced with a W065 or W066 message.

RP A program or other action is being run. The remainder of the message indicates the action that is being performed.

When a terminal is receiving a preload file from the controller, the second line of the system display is used to display an indication of which parts of the file have been received. Because of the nature of the multicast transfer, a terminal does not necessarily receive all parts of the file in order. If a passive client joins the transfer after it has already begun, the first block of file data it receives will be in the middle of the file. Due to network congestion or other issues, passive clients may also miss some of the data blocks that have been transmitted. Because of this, a simple percentage indicating the amount of data received is insufficient to describe the transfer progress. Thus, the following method is used.

A file is transmitted from the server as a series of data blocks. As data blocks are received from the server, the terminal code keeps track of what parts of the file have been received. The file is logically divided into 20 sections corresponding to the minimum of 20 characters available on all system displays. Each character on the second line of the system display represents the percentage of data blocks from that section that have been received as follows:

- **(blank space)** – No data for that section has been received.
- **0** – Between 1 and 9 percent of data for that section has been received.
- **1** – Between 10 and 19 percent of data for that section has been received.
- **2** – Between 20 and 29 percent of data for that section has been received.
- **3** – Between 30 and 39 percent of data for that section has been received.
- **4** – Between 40 and 49 percent of data for that section has been received.
- **5** – Between 50 and 59 percent of data for that section has been received.

- **6** – Between 60 and 69 percent of data for that section has been received.
- **7** – Between 70 and 79 percent of data for that section has been received.
- **8** – Between 80 and 89 percent of data for that section has been received.
- **9** – Between 90 and 99 percent of data for that section has been received.
- ***** – All data for that section has been received.

If a single active client transfers a large file from start to end, the second line is originally blank. Each character on the line displays the digits 0 through 9 and then finally changes to an asterisk. This happens for each character from left to right until the line is filled with asterisks, indicating that the transfer is complete. For smaller files, not all digits necessarily appear and not all characters in the line might be used.

Passive clients typically start receiving a file in the middle, meaning that nonblank characters start appearing in the middle of the second line. After the active client has completed the transfer, one of the passive clients is chosen to be the new active client and the rest of the passive clients remain passive. The new active client requests just the data blocks it has missed. If a passive client has received most of the file but is missing a few blocks, it might have to wait a while before it receives the blocks that it is missing. In this case, the second line of the display is not updated very frequently. For this reason, the second character of the first line is updated occasionally to indicate that progress is still being made and that the passive client is not hung.

File naming conventions

This section describes where preload data files are located and how they are named. This is for informational purposes only and to aid in migrating existing JavaTOF and MultiApp configuration purposes. The information presented here is subject to change as needed. Modification or deletion of any of these files is not supported and may cause problems. Users should not place any of their files in the preload directories.

All active preload files are placed in the directory C:/ADX_STLD, which is also accessible via the ADX_STLD: logical name. The files in this directory are distributed to other controllers; they are the files accessed by the terminals when they load.

Inactive preload files are placed in the C:/ADX_STLD/INACTIVE directory. Files are only placed in this directory during the preload data build process. If preload data building is successful, the files in this directory are moved into the C:/ADX_STLD directory and distributed. Any existing files of the same name are deleted and any obsolete files are removed.

The file name of most preload files is chosen by the user indirectly in Generic Terminal Configuration. For example, the user gets to choose a 1-8 character name for an application or preload bundle. The configuration program only allows valid file names and also prevents the user from starting the name with ADX. Data related to a given application or bundle is stored in one or more files with the corresponding file name. These files are automatically created by the Terminal Preload Build process. The file extension of the files is used to indicate the type of data they contain. This is the current list of file extensions and naming conventions:

BLS Preload bundle list file. The file contains an expanded list of all files to add

to that preload bundle. One file is created for each bundle. The file name is the same as the bundle name typed in by the user.

- BZP** Preload bundle archive file. The file name is the same as the BLS file. These archives are extracted on the terminal in order to be used.
- LMP** MultiApp properties file. The file name has the format ADXLDxxx.LMP, where xxx is the index value for this load extension (as described for LRS files). The preload code copies this file to the same location as the LRS file. This file is only created if the use of MultiApp is required.
- LRS** Terminal load response file (one per unique load definition). The file name will be of the format ADXLDxxx.LRS, where xxx is an index value indicating the load definition. This value is assigned by the preload code based on the position of a load definition in the configuration file. The preload code copies this file to the terminal (to the same location the application files are to be preloaded) if needed .
- LZP** Load file archive file. This is an archive file containing the LMP and LRS files. This file is used to package the files for transmission to the terminal, because they need to be in ZIP form for the preload header to be added and used to determine when they are outdated.
- TPR** TOFCreat Properties file. One file per TOF-enabled application. The file name is the name of the application. It is currently only needed when running TOFCreat on the controller, because TOFStartApp is not run on the terminal.
- TAC** TOF Application class list generated by TOFCreat. The file name is the same as the TPR file. This file contains application class names.
- TAR** TOF Application resource list generated by TOFCreat. The file name is the same as the TPR file. This file contains application resource names.
- TLG** TOF debug log file. The file name is the same as the TPR file. If the debug TOF checkbox is set in configuration, then TOF creates debug output which is then stored in this file.

Note: This file is created in the inactive directory and is left there. The file is not distributed with the other files.
- TSC** TOF System class list generated by TOFCreat. The file name is the same as the TPR file. This file contains system class names.
- TSR** TOF System resource list generated by TOFCreat. The file name is the same as the TPR file. This file contains system resource names.
- TZP** TOF Archive file containing application classes and resources. The file name is the same as the TPR file. The name of the archive file is added to the Java classpath as appropriate. The file is not extracted on the terminal.

Other individual files are automatically created that do not necessarily follow the naming conventions, even though they contain the same type of data.

ADXBCxxx.TZP

Contains the boot classes and resources for a set of applications for which JavaTOF has been configured. One of these files must be created for each set of TOF applications that have a common boot classpath. The file contains the aggregate of all boot classes identified by TOF for each of the applications. The xxx value is an index value indicating for which boot classpath this file is used.

ADXBCxxx.TPR

A properties file with naming as per the ADXBCxxx.TZP file. This file is used to help determine when the boot class archive needs to be rebuilt. This file is not sent to the terminal.

ADXLDEXT.DAT

Load definition extension file used to determine what to preload for each terminal. This file is placed in loadshrink so the terminal can access it during loading without having to access the controller.

ADX_SPGM:ADXPLDMN.LOG

Log file containing all of the logging (debugging) messages created by the Preload activation and distribution code. This file is created when the terminal configuration is activated.

ADXTMxxx.DAT

A terminal load log file containing output from the terminal preload loader program. The xxx value is the terminal number.

ADXJ2JAR.BLS/BZP

Java 2 boot class JAR files from the m:/java2/jre/lib directory. These files are unpacked to the Java 2 installation drive on a terminal if any of the applications do not use TOF. These files and the other JVM-related bundles are only created if the user preloads a JVM.

ADXJ2REQ.BLS/BZP

Java 2 required files. These files include properties files and other files installed with the JVM that are required by the JVM. They include all non-DBCS TTF files from the java2\jre\lib\fonts directory. Any additional fonts, such as user fonts or DBCS fonts, must be added to a bundle file by the user. These files also contain directory entries so the m:/adxetc directory tree can be created on the terminal. The files are unpacked to the Java 2 installation drive.

ADXJ2BIN.BLS/BZP

Java 2 binary files. All DLLs required by the JVM and java.386 are included in this file. The files are unpacked to the Java 2 installation drive.

ADXJ2EXT.BLS/BZP

Java 2 extension JAR files and other files from the M:/JAVA2/JRE/LIB/EXT directory. These files are unpacked to the Java 2 installation drive on the terminal when needed. These files and the other JVM-related bundles are only created if the user preloads a JVM.

ADXTTxxx.BLS/BZP

Contains the JAR files in any directories that the user specifies by the -Djava.ext.dirs property when JavaTOF is enabled for an application. These files are created as needed and extracted to the application preload drive on a terminal. Any files included in these bundles may have their names modified to ensure they can be extracted to a non-VFS drive (the JVM does not care about the names of the JAR files in its extension directories, only about the contents of the files). The xxx value indicates the directory to which the bundle applies.

These are some additional files that are related to loading:

ADX_SPGM:ADXPLDRS.LOG

Log file created when the terminal preload rebuild utility is run due to system software maintenance being applied.

ADX_SPGM:ADXPLDRU.LOG

Log file created when the terminal preload rebuild utility is run manually due to a user's request.

<PLDHDRS>

A special file name which indicates that the terminal is contacting the controller to determine which preload-related files are out of date and need to be retrieved.

ADXVXEDF.DAT

Java 6 related data files

VX_EOSPJ.DAT

Java 6 JVM

ADXJ2IDF.DAT

Java IME (Input Method Extension) files

ADXJ2FWF.DAT

Java 6 SBCS fonts

ADXJ2FDF.DAT

Java 6 DBCS fonts

ADXXTD*.DAT

Extension data

VX_RT8LF.DAT

4690 OS load shrink file (for terminal boot)

VX_TINRD.DA*

4690 OS hard disk boot file

VX_TVMLZ.DAT

4690 OS hard disk boot file

ADXILI*.286

4690 OS hard disk boot file

Frequently asked questions (FAQs)

Table 41. FAQs about JavaTOF enhancements in 4690 OS Version 4

Question	Answer
Preloading files to a controller/terminal is failing. What are some common problems?	Only RAM drives can be selected for bundle preloading on a controller/terminal. Also, the RAM disk Q: for a controller must be configured within controller configuration. Ensure the configuration is correct.
How does the user know whether or not to preload the JVM into the terminal?	If the user wants to be able to start or use applications on an offline terminal, then the JVM should be preloaded.
How does the user decide whether to place the JVM on the M: or Q: drives on the terminal? (If the terminal has no hard drive, then M: is not an option.)	If there is enough free memory and the Q: RAM disk can be made large enough, then the Q: drive would provide a runtime performance advantage.
What changes are required by the user to take advantage of the JVM being preloaded into the terminal?	None. This is handled by the terminal preload builder code.

Table 41. FAQs about JavaTOF enhancements in 4690 OS Version 4 (continued)

Question	Answer
After files are preloaded to a terminal hard disk, are they ever deleted?	Usually, no. Obsolete preload files in the C:/ADX_STLD directory are removed to free up space. However, no other files on the C: and M: drives on the terminal are ever deleted by the preload code. The drive must be formatted or the files deleted by user code for old files to be deleted.
Certain preload files are always extracted on some terminals, but only extracted some of the time on other terminals. Why is this?	This is probably due to the target drive to which the files are being extracted. The files must be extracted on every load if the target drive is a RAM disk because the drives are empty. If the target drive is a hard disk (C: or M:), then the files are only extracted when the preload archive file is rebuilt.
What if the same target file name appears in more than one bundle that is being preloaded to the same terminal?	The file will be overwritten each time a preload bundle in which it exists is extracted. If the contents of the file are not identical among bundles, then the file contains the data of the last bundle that was extracted. This can happen if the file is being extracted to the root directory of the drive and files with the same names but different directories on the controller were added to the bundles. The order in which bundles are extracted is undefined.
How do I set the JavaTOF property <code>depchecking.tolerateMissingClasses</code> and do I need it?	This property is always set in the TOF property file that is created by the preload code. Even if a class cannot be found, the dependency check will continue until the list of all missing classes is generated.
When setting up a TOF-enabled application, what happens if the same class is listed in both the TOF Dependency Include List and the TOF Dependency Exclude List?	This behavior is the same as in legacy JavaTOF. The value in the include list takes precedence.
When a terminal is running a Java application configured using the new JavaTOF and its status is queried, the running Java application is reported as “@ADXJ2XLD:”. Why is this?	<p>The response gives the name of the response file that was used to start the application.</p> <p>The user may look in the ADXLDEXT.DAT file to determine which ADXLDxxx.LRS response file is being used. Also, the user can specify the desired response file name manually as <code>@ r::adx_stld:adxldxxx.lrs \$</code> when using OCF to start an individual terminal, if stdout or stderr redirection statements are in use.</p>

Chapter 11. Understanding USB

This chapter explains the capabilities and limitations of USB devices connected to 4690 controllers and terminals. The operating system supports a subset of common PC devices, like standard USB keyboards or mice, in addition to POS-unique hardware like receipt printers, pole displays, and scales. The USB specification provides the Plug-and-Play capability to discover attached devices at boot time and allows for devices to be hot plugged, or detached and attached, with the system fully operational. This hardware capability is limited, however, by the operating system and POS application architecture. Both 4690 OS and the applications that run on 4690 OS perform device discovery during system or application startup. Therefore, all required devices should be attached (and powered on, if self-powered) at initial load time. Devices attached after boot time may not be discovered or initialized properly. Hot plugging of USB devices is generally not supported. The only exception is for USB media devices like floppy drives and memory keys, which do support hot plugging.

Multiple Device Support

The operating system will support up to three USB VFD or LCD displays, up to two USB Touch displays, and multiple USB media devices. All other USB devices are limited to one. The order that VFD or LCD displays are discovered determines which one becomes ANDISPLAY, ANDISPLAY2, and so forth. Devices are discovered in alphabetical order within each row, but there is no guarantee of discovery order across rows or hubs. Therefore, all identical VFD, LCD, and Touch displays must be attached in the same row to ensure that they are consistently discovered the same way. Failure to do so may result in devices changing their assignment from one boot to the next or when switching between Classic and Enhanced modes.

Notes:

1. If ANDISPLAY or ANDISPLAY2 have been redirected to Java, the redirected name is skipped when the operating system assigns names to the physical displays. For example, if two displays are connected and neither display is redirected to Java, the first display becomes ANDISPLAY and potentially the system display, while the second becomes ANDISPLAY2. If two displays are connected and ANDISPLAY has been redirected to Java, however, the first display becomes ANDISPLAY2 and potentially the system display. The second display becomes ANDISPLAY3.
2. The primary video display may be configured to be the system display. This is controlled by the terminal device characteristics.
3. If a keyboard is attached via the PS/2 port, then that keyboard is used for controller input on a combination controller/terminal and is used for all Java input. A USB-attached POS keyboard in this instance is limited to interacting with the POS application. Otherwise, if the only keyboard attached is a USB ANPOS or later variant, then it will be shared for all uses.
4. Prior versions of 4690 OS supported a limited detach and attach of an identical USB POS device, but this capability is not consistent with the base unit hardware specification, and therefore it is no longer tested or supported. Power the base unit off prior to detaching or attaching any hardware other than a supported USB media device.

USB Memory Keys

The operating system's Enhanced mode provides support for USB-attached memory keys. Only keys formatted as FAT or FAT32 are supported. Most USB keys are preformatted this way to make them portable across multiple operating systems. Long file names are supported on memory keys. Each device appears under the E: drive in a subdirectory numbered 000 or higher. For example, the root directory of the first USB key will be E:\000\. A file named `list` in the root of the E: drive describes the attached memory keys and their directory association. This read-only file may be typed to the screen or may be read as a properties file for Java applications. The maximum size of a single file that the operating system can read or write on any device, including memory keys, is 2 GB. CHKDSK is not supported for memory keys. If you attempt to run CHKDSK, it will report Disk is either invalid or locked by another process.

A USB key can be reformatted from the 4690 command line via **FORMAT E:\000**(or replace 000 with the appropriate directory name). A 4690 bootable (Supplemental) memory key can be created from the Enhanced Options Menu.

On memory keys, the case-handling of names follows these rules:

- Within any directory, two files can not exist which differ only in case. Attempts to create or access a file which matches an existing file except for case are considered to refer to the existing file. The existing case is retained, even if a new file is being created to overwrite an existing file. If you want to rename a file to another name that differs only in case, it must first be renamed to an intermediate name that differs in more than case, and then renamed to the final target name.
- File names that fit into an 8.3 format and are all upper case are displayed as all lower case. All other filenames are displayed as exact case.

These rules also apply to subdirectory names.

Configuring your system

Note: Before running any Java-based configuration utilities, you must enable Java graphics in controller configuration. Then, you must activate the configuration change and IPL your system before Java graphics are enabled.

The operating system contains menus allowing you to specify configuration information for the SurePOS 300/700 Series (Models 7x3, 7x4 7x5 and 350) systems and TCxWave 6140 Series (Model 100) systems. Configuration options for these systems are provided through Generic Terminal Configuration (GTC). The configuration data contains terminal device characteristics and terminal load definition information. See the *4690 OS: Planning, Installation, and Configuration Guide* for assistance with configuring your system.

Using USB with the operating system

USB devices with 4690 OS

A default keyboard layout is loaded depending on the type of keyboard attached. Supported keyboards are:

- USB 50 key keyboard
- USB 50 key POS keyboard with JUCC MSR

- USB 50 key keyboard with MSR and display
- Modular 67 key keyboard
- Modular 67 key LCD keyboard
- USB keyboard V
- USB keyboard VI
- USB PLU keyboard
- Modular ANPOS keyboard
- USB ANPOS keyboard with MSR
- USB 133 key keyboard with MSR
- SurePoint Solution

Chapter 12. Enabling MBrowser

MBrowser is a web browser based on Mozilla® technology that is distributed with 4690 and available for use on hardware running in enhanced mode. MBrowser is shipped as an operating system extension. To use MBrowser, enable the extension in your configuration:

- To use MBrowser on a controller, select it from the Extensions screen of the controller configuration for that controller.
- To use MBrowser on a terminal, select it from the Extensions screen of the appropriate load definition record in the terminal configuration for that terminal. A diskless terminal must have at least 1Gb of RAM in order to load MBrowser. Additional memory may be required for other data or applications loaded onto the terminal, and for the operational requirements of MBrowser (if large, complex web pages are loaded).

Note: When using MBrowser on a controller or terminal, selecting MBrowser for use in controller configuration will not enable it for use on the terminal side (and vice versa); it is configured for each side individually.

The extension name for MBrowser is *adxmbrowser* and it is listed under the *System Code* category in extension configuration. The GTK extension is a prerequisite for MBrowser and you should select it when MBrowser is enabled.

Although MBrowser is an extension, it cannot be launched via the ADXXTRUN extension launcher. For more information on configuring extensions, refer to the *4690 OS: Planning, Installation, and Configuration Guide*.

Launching MBrowser

To launch MBrowser, run the MBROWSER.386 program. When the MBrowser extension is deployed, a copy of the MBrowser launcher is placed in `f:\adx_spgm\mbrowser.386`. When the extension is withdrawn, the launcher is deleted. This allows terminal applications to launch the browser without having to create a bundle to transfer the file.

Normally when MBrowser is launched, it will create a graphical window containing the browser. When this is done from a controller command line or Java application, 4690 will switch to the enhanced mode graphical extensions screen; this is similar to the screen switch that happens when a Java graphical window is created. When MBrowser is launched via other methods (such as the XML based launch method below), an automatic screen switch does not happen. If the MBrowser graphical window is launched from a non-enhanced telnet session, the telnet screen will go blank while the graphical window is displayed on the controller.

Note: MBrowser behaves in the same fashion as graphical Java applications. Graphical applications cannot be launched during the controller boot process (stage 3 command IPL processor), from background applications on the controller, or from enhanced telnet sessions.

When MBrowser is launched from the command line with invalid flags or with a profile maintenance command that doesn't launch MBrowser (such as **-delete-profile**), the screen will flash briefly to graphics mode before returning to text mode. MBrowser can be launched with these maintenance commands at any time after the MBrowser extension is deployed. On a controller, MBrowser is

deployed right before stage 3 of the IPL command processor. On a terminal, MBrowser is deployed during the preload process.

When the browser starts up, the default directory is `f:/` (which is considered `/opt/ibm/retail/vx4690/devices/f_drive` or `/cdrive/f_drive` from where the enhanced operating system is running).

The MBrowser browser supports the concept of separate operational profiles. A profile contains all of the browser settings, cookies, bookmarks, and cache information required to run the browser. On multi user systems, each user will use a different profile. The browser also supports the ability for a single user to use multiple profiles, as well as a profile manager to manage the profiles. On 4690, profiles are managed in a different manner and the profile manager is not directly accessible.

The primary reason to use multiple profiles is when you want to run more than one instance of the browser on a machine at a time. A single browser instance (process) can create multiple windows, but all windows use the same profile settings. Using a different profile is also useful if you wish to configure or skin a browser differently based on where it will be used (operator display, customer display, or store associate at the controller). Using multiple profiles is required if you want the browser to appear on more than one display at a time.

Note: On other platforms, MBrowser does not have this restriction. However on 4690, the implementation of dual independent touch requires it.

System flags

The following flags are system flags. These flags must be passed in before any other flags and parameters. All system flags start with **-X**:

-Xdisplay:2

Run the browser on the secondary display in a terminal.

-Xnoswap

Running MBrowser from the command line or from a Java application will cause the screen to switch to the enhanced mode graphical extensions screen. The **-Xnoswap** flag prevents this behavior.

Arguments starting with a dash (-) are considered to be flags, and other arguments will be treated as URLs and passed to the browser. The following flags are supported:

-language

Set the language passed to the browser. This controls the language used in the browser's user interface it also used to generate the *user agent string* sent to Web servers, which could affect the language used in Web pages sent back to the browser. See "Customizing MBrowser" on page 330 for additional information on setting the language.

-reset-profile

This flag removes all files in the profile named via the **-profile** flag and re-initializes them from the defaults before starting the browser. This may be useful in cases where you want to reset everything when different users are using the same profile.

-profile name

Set the profile to the given name. The valid characters for the profile name are: alphanumeric characters (a-z, A-Z, 0-9), the dash, and the underscore (however a profile name may not start with a dash). If this flag is given

more than once, the last one given is the one used. A profile name must be between 1 and 31 characters in length (inclusive). The name is case sensitive so the profile names *Browser* and *browser* are unique.

-delete-profile name

Delete the profile with the given name. Parameter processing stops when this flag is given (all other parameters are ignored).

-extract-profile name

Save the current user files from the profile with the given name. The file is saved as `profile_name.zip` in the root directory of the `f:/` drive (so that it does not overwrite the user provided file in `f:/adxetc/ext/adxmbrowser`). Parameter processing stops when this flag is given (all other parameters are ignored). See the MBrowser customization section for the list of files that are saved. This command is intended for use by customers doing the initial setup of MBrowser for a store. It can be used with the MBrowser menus (or `about:config`) to configure the browser and save those settings in a file. When the browser exits, it stores all current settings in a file called `prefs.js` (it does not modify `user.js`). Because of this, the **-extract-profile** command will save the browser's current `prefs.js` in the archive as `user.js`. To modify bookmarks in the browser and save them via this method, set the `browser.bookmarks.autoExportHTML` setting to true so the browser will update `bookmarks.html`. You can also save bookmarks as `name.zip` in the root directory of the `f:/` drive (which corresponds to `F:/` on 4690).

-delete-all-profiles

Delete all MBrowser profile directories. Parameter processing stops when this flag is given (all other parameters are ignored).

-list-profiles

List all MBrowser profile directories. Parameter processing stops when this flag is given (all other parameters are ignored).

-geometry

Sets the size of the client portion of the MBrowser main window (this is the portion of the window controlled by MBrowser itself and excludes the title bar and other components drawn by the window manager). The format of the geometry string is `<width>x<height>` where *width* and *height* are integers. The minimum number for these values is 100, and the maximum values are based on the screen resolution. Invalid integer values are rejected with an error message, but valid integer values are rounded to the nearest valid value. Any other values are passed to MBrowser, which can impose its own limits (such as minimum size). MBrowser currently displays a maximized window in safe mode ignoring any values passed by this flag. The first time a window is opened with a given profile, it is always maximized.

-help or -?

Prints a brief help description.

The following flags (and any corresponding parameter values) are passed through to the browser:

`-safe-mode`, `-jsconsole`, `-url`, `-new-window`, `-chrome`

When MBrowser is started with a specific profile name, all windows are managed by the same MBrowser instance (process). If you launch MBrowser with the name of a profile that is already in use, an error message is displayed.

Customizing MBrowser

MBrowser stores configuration information, bookmarks, and cache data in a subdirectory referred to as the profile directory. When the browser first starts, it requires a few initial files to be in this directory but it will create the others as needed.

To customize MBrowser, the profile files are tailored by the user and placed in a zip archive file. The zip file is then copied to the extension user data directory (f:/adxetc/ext/adxmbrowser) on the machine running MBrowser. This can be done directly on a controller or via preload bundles on a terminal.

The zip file should be named name.zip, where *name* is the name of the profile indicated by the -profile flag on the MBrowser command line. If the profile name is not set, the file should be called profile.zip. Like profile names, the name of the file is case sensitive; the extension must be .zip in lowercase.

When MBrowser is started, the file will be examined. If the file is new (or changed), the files listed below are extracted. Any other files in the zip file will be ignored. Other files might be extracted in the future, so the user should not include any additional files at this point.

bookmarks.html

Contains bookmarks. Toshiba will provide a default file containing the Toshiba Web site if one is not provided.

chrome/userContent.css

Style sheet for the browser. Toshiba will not provide a default.

chrome/userChrome.css

Style sheet for the browser user interface. Toshiba will provide a default that makes the user interface font slightly bigger. If you provide a style sheet, your copy will override the Toshiba style sheet. Information on changing menus can be found at http://kb.mozillazine.org/Firefox%3A_Tips%3A_Customize_context_menu

user.js

This file includes user preferences. The browser stores preferences in the file prefs.js, however it reads values from both prefs.js and user.js when it starts up. If the same preference is given in both the prefs.js and user.js files, the preferences from the user.js file takes precedence. When a profile is initially created, Toshiba will provide a default prefs.js file, which contains some default settings for the browser (setup for fonts, etc). If settings changes are made within the browser, it will save its current preferences (including any values merged from user.js) into the prefs.js file. A guide on the format of user.js can be found at https://developer.mozilla.org/En/A_Brief_Guide_to_Mozilla_Preferences. Information on values that can be set in this file can be found at http://kb.mozillazine.org/About:config_entries.

Note: When you change preferences in MBrowser (either in about:config or by using the **Edit->Preferences...menu** option) the changes are saved to prefs.js. The values in user.js only override any values in prefs.js when the values are not the default. After a non-default value is placed in prefs.js (by setting preferences or about:config) it cannot be set back to the default by a value set in prefs.js.

The default browser language is controlled by the `-l` language flag, as described above (or by the logical name `LC_ALL` if the `-l` language flag is not specified). Language names have the format `ll-RR` where `ll` is a language code and `RR` is a region/country code. The default value for language is based on the `LC_ALL` logical name in 4690. The portion of `LC_ALL` before the period is removed and the `_` replaced with a dash (`-`). The following languages are currently supported by the browser.

af	ar	as	be	bg	bn-BD	bn-IN	ca
cs	cy	da	de	el	en-GB	eo	es-AR
es-CL	es-ES	es-MX	et	eu	fa	fi	fr
fy-NL	ga-IE	gd	gl	gu-IN	he	hi-IN	hr
hu	id	is	it	ja	ka	kk	kn
ko	ku	lt	lv	mk	ml	mr	nb-NO
nl	nn-NO	oc	or	pa-IN	pl	pt-BR	pt-PT
rm	ro	ru	si	sk	sl	sq	sr
sv-SE	ta	te	th	tr	uk	vi	zh-CN
zh-TW							

Additional fonts might be required to support a given language. The language codes are ISO 639-1 codes. For more information on fonts, go to www.loc.gov/standards/iso639-2/php/code_list.php

The language setting is saved in the `general.useragent.locale` preference in `prefs.js` before the browser is started. If this preference is also set in the `user.js` file, that value will take precedence. This arrangement allows the language to be overridden on the command line or when `LC_ALL` changes, but it prevents any changes made directly in the `about:config` from having any effect.

When a download is complete, the browser pops up an alert window to notify the user. This window is displayed in such a way that it overlays the 4690 console window. To avoid this possible interruption, the following browser property is set automatically, although it can be restored to its default of 2000 in `user.js`.

```
user_pref("browser.download.manager.showAlertOnComplete", false)
```

Toshiba does not support browser update (the browser files are read-only). If you wish to prevent the browser from prompting the user to perform updates, the relevant settings for preventing updates are:

```
user_pref("browser.search.update", false );
user_pref("extensions.update.enabled", false );
user_pref("app.update.enabled", false );
user_pref("app.update.auto", false );
```

If the machine has a hard disk, the preferences database will persist between boots and between any updates due to software maintenance. The MBrowser profile directories will be removed if MBrowser is removed from the configured extension list for that machine.

There are several profile related command line flags that give you more control about how the profile is managed.

Launching MBrowser from a terminal

MBrowser can be launched programmatically on a terminal by running the mbrowser.386 program (see “Launching MBrowser” on page 327). However, it can also be configured to start automatically. This is accomplished through the use of XML configuration files.

The design of the XML configuration files is similar to that of Java applications and load definitions in terminal configuration. The files are also managed and read by terminal activation code in the same manner. Specifically, the user will upload or change inactive versions of these files (and possibly make changes to other parts of terminal configuration) and then activate terminal configuration. If there are errors in these files or other parts of the terminal configuration, then activation will fail without updating the active files. If activation succeeds, the inactive configuration files will be renamed/moved to their active names. If neither inactive file exists or if they exist but match the corresponding active file, the configuration will indicate that no changes are required when activating.

There are some differences with the MBrowser terminal configuration files and other configuration. There are no configuration user interface changes to 4690; configuration is done through these files only at this point. The Configuration Utility was not updated to enable the import/export of these files. The contents of these files are not listed in any configuration reports.

Similar to Java applications, the application definition configuration file is used to configure the parameters and other information used to launch MBrowser. The active filename is ADX_SPGM:ADXCFGAA.DAT and the inactive file is ADX_SPGM:ADXCFGAI.DAT. The DTD for this XML data is located in c:\java\cfgxml.jar in the entry com\ibm\OS4690\cfg4690\util\generic\dtd\Application.dtd. A sample file is included in ADX_SPGM:ADXCFGAI.SMP. Figure 20 an example of an application configuration file.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ApplicationFile SYSTEM "Application.dtd">
<ApplicationFile>
  <FileVersion version="v6r3"/>
  <Application applName="TGCS">
    description="Toshiba Global Commerce Solutions Contact page"
    programName="f:/adx_spgm/mbrowser.386"
    programArguments="http://www.toshibagcs.com/contact.html"
    autoStart="1"
    restartOnFailure="0"
    outputFilename="r::tgcs%m.out"
    outputAppend="0"
  </>
  <Application applName="Toshiba">
    description="Toshiba Facebook page on secondary display"
    programName="f:/adx_spgm/mbrowser.386"
    programArguments="-Xdisplay:2 -profile profile2 https://www.facebook.com/ToshibaGlobalCommerceSolutions"
    autoStart="1"
    restartOnFailure="0"
    outputFilename="r::tosh%m.out"
    outputAppend="0"
  </>
</ApplicationFile>
```

Figure 20. Example of application configuration file

Each application to be configured is represented by an Application node in the file. The following attributes can be set:

applName

The name of the application that is used to identify the application in the

application load definition file. Application names within this file must be unique. The length must be between 1 and 31 characters long (inclusive) and consist of alpha numeric characters only. The name is case-sensitive; however to prevent debug errors, do not have multiple applications whose names differ only in case. This attribute is required.

description

A description of the application. This attribute is not currently used, but in the future might be displayed in the configuration. This attribute is optional.

programName

The name of the program to start. This must be a full path name, including extension. Since MBrowser is always extracted to `f:/adx_spgm`, this should be `f:/adx_spgm/mbrowser.386`.

Note: F: is case-sensitive, so the path and file name must be in lower case. This name must be less than 128 characters in length. No other validation is done, however it must be a valid 4690 path name to actually work. The name can contain logical names. This attribute is required.

programArguments

These arguments will be passed as the command line to the program. The string can be from 0 to 4096 characters long. This attribute is optional.

autoStart

Determines whether the application is started when the terminal loads. If the value is 0, it is not possible to start the application. This attribute is optional; it defaults to 1. The following values are allowed:

- 0 - Do not auto start the application
- 1 - Always restart.

restartOnFailure

Determines whether the application is restarted if it fails. There is a delay between restart attempts to prevent too much CPU usage. This attribute is optional; it defaults to 0. The following values are allowed:

- 0 - Never restart
- 1 - Always restart
- 2 - Restart only if the return code is 0
- 3 - Restart only if the return code is greater or equal to 0. For most applications, only load failures cause return codes less than 0

outputFilename

If this attribute is non-empty, it will be used for the name of the standard output and standard error files for the program being run. The string can be from 0 to 255 characters long. In addition the "%" symbol is an escape sequence. A "%m" in the name will be replaced by the machine id (3 digit terminal number). A "%%" in the string will be replaced by a single "%". Any unrecognized escape sequences be considered an error. After escape sequence replacement, the filename is passed to the operating system when the program is launched. It must otherwise be a valid 4690 filename (using logical names and writing to R: : are okay, for example). If the operating system cannot open the file at runtime, the program will continue to run (but with no stdout/stderr output) and a different event code will be used for the start message. This attribute is optional.

outputAppend

Controls whether the output file is appended to or not; it has no effect if

there is no output file. If the value is 0 and the output file listed in the outputFilename exists when the program is started, it will be deleted first; otherwise, it will be appended to. This attribute is optional; it defaults to 0. The following values are allowed:

- 0 - Do not append to the output file
- 1 - Append to the output file

The application load definition file determines which applications should be made available for which terminals in the store. The active filename is ADX_SPGM:ADXCFG.LA.DAT, and the inactive file is ADX_SPGM:ADXCFG.LI.DAT. The DTD for this XML data is located in c:\java\cfgxml.jar in the entry com\ibm\OS4690\cfg4690\util\generic\dtd\ApplicationLoadDefinition.dtd. A sample file is included in ADX_SPGM:ADXCFG.LI.SMP. Figure 21 is an example of an application load definition configuration file .

Figure 21. Sample application load definition configuration file

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ApplicationLoadDefinitionFile SYSTEM "ApplicationLoadDefinition.dtd" >
<ApplicationLoadDefinitionFile>
  <FileVersion version="v6r3"/>
  <ApplicationLoadDefinition machineRange="001,003-005">
    <LoadedApplication applName="TGCS"/>
    <LoadedApplication applName="TGCSSP"/>
  </ApplicationLoadDefinition>
  <ApplicationLoadDefinition machineRange="002">
    <LoadedApplication applName="TGCSSP"/>
  </ApplicationLoadDefinition>
</ApplicationLoadDefinitionFile>
```

Each set of terminals to be configured is represented by an ApplicationLoadDefinition node in the file. Each ApplicationLoadDefinition node has one or more LoadedApplication nodes that indicate the applications to be loaded on the terminal. The attributes available in these nodes are:

machineRange

This indicates the terminals on which the listed applications should be configured. The rules on setting the attribute are modeled after those used by the terminalRange attribute created by the configuration in the Java Load Definition extension file. The value consists of a comma separated list of terminal numbers and/or terminal number ranges. A terminal number range consists of two terminal numbers separated by a dash. The terminal numbers must be decimal values between 0 and 999 (inclusive); prefixing the value with 0's is optional. An entry containing the terminal number 0 is considered the default load definition and will be used for all terminals that are not otherwise explicitly defined. The first number of a range must be less than or equal to the second number. Listing a terminal twice within a range string or in multiple ApplicationLoadDefinition records is invalid. This attribute is required.

applName

This is the name of the application to be loaded. It is required and must match the name (and case) of an application in the application configuration file. Listing an application more than once for a terminal or terminal range is not flagged as an error, however the outcome is no different than listing the application a single time.

When a terminal loads, all applications configured for it (for which `autoStart` is true) are started and W723 messages are logged at that time. When a program ends, a W724 message is normally logged, however, if the application fails to load due to a terminal offline condition, the application will be reloaded automatically until the offline error is resolved. Note that only one application at a time can be loaded from a controller (via R:). All others will get offline errors and will be retried. There is no way to independently stop an application once it has started or to determine if it is running. At most, 16 processes can be started at a time using this mechanism. An attempt to start additional processes will fail and a W724 message will be logged with event code 001, and the return code 0x80004180 (overrun). See the *Wnnn* message section of the *4690 OS: Messages Guide* for the W723 and W724 messages.

Trouble shooting: For command line errors, you can set the `outputFilename` to a filename on the controller (using R: :) and see what gets placed in the file.

Program Exit Code

The exit code of `mbrowser.386` indicates why the program ended. Most negative error codes indicate there was a failure launching the application; these may be found in the Application program status xxx* section of the *4690 OS: Messages Guide*. Negative return codes of the form 0x8000Y0XX indicate that the program exited due to a linux signal (where XX is the signal number and Y is 0 or 1). Positive error codes are returned either by the browser itself or the program used to validate parameters prior to launching the browser.

Table 42 on page 335 lists the decimal and hexadecimal form of the values returned from the launcher.

Table 42. MBrowser Launcher Exit Codes

Decimal Value	Hex Value	Meaning
0	0	Operation complete. This return code is used for successful operations that do not otherwise start MBrowser (such as deleting, listing, or extracting profiles and printing help).
50	32	Graphics is not configured or available in this context. This error code will be returned if 4690 graphics mode is not configured or if the program is launched from a context where graphics cannot normally be used (such as from an enhanced telnet command session, a 4690 background task, or the IPL command processor).
51	33	MBrowser running. Returned when a profile is in use and it cannot be deleted, extracted or used by a new MBrowser instance.
52	34	Invalid zip. Returned when a ZIP file to be imported is invalid.
53	35	Profile not found. Returned when a profile named on the delete or extract profile commands cannot be found.
54	36	Missing argument. Returned when a required parameter (such as the profile name expected with the <code>-profile</code> flag) is missing.
55	37	Invalid argument. Returned when a flag or argument is invalid.
56	38	Unknown error. Returned when an unknown or internal failure condition is encountered.

Operational Environment

MBrowser runs under the Linux user id *vxuser*. With some file dialogs, such as the **File->Save As** dialog, you can select various directories, such as *vxuser*'s home directory, as the location to save files. Although this is possible, any files saved in locations other than */cdrive/f_drive* are not normally accessible from the 4690.

MBrowser works in 640x480 resolution, but several of the dialogs displayed by MBrowser do not work very well at such a low resolution. In particular, the preferences and page information dialogs were designed for a larger minimum screen size. Most other dialogs will scale themselves properly when re-sized. You can remove access to these dialogs when running in a POS environment with smaller screens that require such a low resolution.

Chapter 13. The PDF document viewer

A PDF document viewer based on evince is distributed with 4690 and available for use on hardware running in enhanced mode. The program is shipped as an operating system extension. The extension name for the PDF viewer is adxpdf and it is listed under the System Code category in extension configuration. To use the viewer, enable it in configuration:

- For use on a controller, select it from the Extensions screen of the controller configuration for that controller.
- For use on a terminal, select it from the Extensions screen of the appropriate load definition record in the terminal configuration for that terminal. A diskless terminal must have at least 1GB of RAM in order to load the adxpdf. Additional memory may be required for other data or applications loaded onto the terminal and for the operational requirements of the viewer (if large, complex documents are loaded).

Note: When enabling an extension on a controller/terminal, selecting the extension for use in controller configuration will not enable it for use on the terminal side (and vice versa); it is configured for each side individually.

The GTK extension is a prerequisite for the adxpdf and must also be selected. Although the adxpdf is an extension, it cannot be launched via the ADXXTRUN extension launcher. For more information on configuring extensions, refer to the *4690 OS: Planning, Installation, and Configuration Guide*.

Launching adxpdf

To start the adxpdf, run the launcher program f:\adx_spgm\adxpdf.386. When the extension is deployed the launcher program is automatically created. When the extension is withdrawn, the launcher is deleted. This location was chosen because the F: drive is available on both the controller and the terminal and can be used by any type of application.

When the extension is launched, it will create a graphical window for the program. When this is done from a controller command line or Java application, 4690 will switch to the enhanced mode graphical extensions screen; this is similar to the screen switch that happens when a Java graphical window is created. When the extension is launched via other methods (such as the XML based launch method below), an automatic screen switch does not happen. If the extension is launched from a non-enhanced telnet session, the telnet screen will go blank while the graphical window is displayed on the controller.

Note: Graphical extensions behave in the same fashion as graphical Java applications. Graphical applications cannot be launched during the controller boot process (stage 3 command IPL processor), from background applications on the controller, or from enhanced telnet sessions.

When the extension is launched from the command line with invalid flags or flags that do not otherwise launch the program (like -?), the screen will flash briefly to graphics mode before returning to text mode.

If the MBrowser web browser and the PDF Viewer are both configured, PDF links may be opened from within the browser. In addition, links to PDF files will

automatically be displayed by the PDF viewer. The viewer will launch in a separate window and is not integrated into the web page. Alt-Tab may be used to switch between windows.

When adxpdf starts up, the default directory is f:/ (which corresponds to /opt/ibm/retail/vx4690/devices/f_ drive or /cdrive/f_ drive in the local linux filesystem).

Command line syntax

The command line syntax used to start adxpdf is as follows:

```
adxpdf [flags] (filename | URL)
```

All arguments starting with a dash (-) are considered to be flags and are used to control program behavior. The following flags are system flags. These flags must be passed in before any other flags and parameters. All system flags start with -X:

-Xdisplay:2

Run the program on the secondary display in a terminal.

-Xnoswap

Normally, starting the program from the command line or from a Java2 application will cause the screen to switch to the enhanced mode graphical extensions screen. The -Xnoswap flag prevents this behavior.

The following non-system flags are supported:

-fsm Start the program in full screen mode, removing the title bar and menu. Press F11 to toggle full screen and normal display mode.

-pm Start the program in presentation mode. In this mode, a full page is displayed at a time. Press the F5 key to toggle presentation and normal mode.

-page label

Position viewer on given page number or label of the document. The allowed value for label parameter is determined by the document being viewed. Most documents contain a label for each page that corresponds to the page number for that page.

-language lang

Set the language used in the user interface. See below.

-help or -?

Prints a brief help description.

Once the program starts it will load and display the document specified by the filename or URL, which is a required parameter.

- The program supports the display of non-password protected PDF documents.
- Currently only http: URLs are supported
- When displaying files, 4690 file drive letters may be used, but the program will only be able to display files accessible to linux. Specifically, the program uses the same rules as python for resolving 4690 paths to make them accessible for linux programs. See the section titled "Rules for Path Names in the 4690 implementation of Python" in the *4690 OS: Programming Guide* for more information on how path names are resolved.

As indicated above, the language used by the user interface in adxpdf may be specified via the language flag. Language names have the format ll-RR where ll is a

language code and RR is a region/country code. The default value for language is based on the LC_ALL logical name in 4690 (whose default value is determined by the keyboard layout chosen when 4690 was installed). The portion of LC_ALL before the period is removed and the underscore (_) is replaced with a dash (-). The following languages are currently supported:

af	ar	as	ast	be	bg	bn	bn_IN	br	ca
cs	cy	da	de	dz	el	en_CA	en_GB	es	et
eu	fa	fi	fr	ga	gl	gu	he	hi	hu
id	it	ja	ka	kn	ko	ks	ku	lt	lv
mai	mg	mk	ml	mr	nb	nds	ne	nl	nn
oc	or	pa	pl	ps	pt	pt_BR	ro	ru	rw
si	sk	sl	sq	sr	sv	ta	te	th	tr
uk	vi	wa	zh_CN	zh_HK	zh_TW				

Additional fonts might be required to support a given language. The language codes are ISO 639-1 codes. For more information on these codes, see http://www.loc.gov/standards/iso639-2/php/code_list.php

Launching adxpdf from a terminal

See the 4690 OS Users Guide chapter on the MBrowser web browser. The section titled “Launching MBrowser from a terminal” describes methods of launching the browser. The same methods can be used to launch adxpdf (except that the program name is adxpdf.386).

Program Exit Code

The exit code of adxpdf.386 indicates why the program ended. Most negative error codes indicate there was a failure launching the application; these may be found in the Application program status xxx* section of the *4690 OS: Messages Guide*. Negative return codes of the form 0x8000Y0XX indicate that the program exited due to a linux signal (where XX is the signal number and Y is 0 or 1). Positive error codes are returned either by the viewer itself or the program used to validate parameters prior to launching the viewer.

Table 43 on page 339 lists the decimal and hexadecimal form of the values returned from the launcher.

Table 43. Adxpdf Launcher Exit Codes

Decimal Value	Hex Value	Meaning
50	32	Graphics is not configured or available in this context. This error code will be returned if 4690 graphics mode is not configured or if the program is launched from a context where graphics cannot normally be used (such as from an enhanced telnet command session, a 4690 background task, or the IPL command processor).
54	36	Missing argument. Returned when a required parameter (such as the page number expected with the -page flag) is missing.
55	37	Invalid argument. Returned when a flag or argument is invalid.
56	38	Unknown error. Returned when an unknown or internal failure condition is encountered.
57	39	File not found. The filename given on the command line was not found (or wasn't accessible from linux).

Operational Environment

The operational environment of adxpdf is identical to that of the MBrowser web browser. For more information refer to the “Operational Environment” on page 336 section of the Enabling MBrowser Chapter.

Chapter 14. 4690 Directory Services

This chapter explains 4690 OS support for directory services. Directory Services has been added to 4690 OS controllers in order to support enterprise managed user IDs. The 4690 logon process has been enhanced so that it can access the LDAP directory for user ID information, including password and authorization attribute data. In simple terms, a directory is just a database. Directories are usually optimized for reading data. 4690 OS uses Lightweight Directory Access Protocol (LDAP) for accessing the directory. The 4690 OS implementation is provided by OpenLDAP, an open source LDAP implementation. For more information about OpenLDAP and directory services, go to www.openldap.org.

Directory Services requirements

Directory Services runs only on 4690 OS enhanced controllers. When using a multiple controller environment, to ensure that user IDs stored in the LDAP database will be available during controller down situations, every controller in the multiple controller environment must be enhanced.

Configuring Directory Services

Directory Services is configured in System Configuration. There is a panel labeled "Directory Services" in the System Security section. Select the check box to enable Directory Services, then activate System Configuration. Directory Services will be started the next time you restart 4690 OS provided you have a valid ADX_IDT1:ADXLDIFF.DAT file, as described below.

Deselecting the check box and activating System Configuration will disable Directory Services, and Directory Services will continue to operate until 4690 OS is restarted.

Warning - when the 4690 OS is restarted, all data stored in the directory is deleted.

Initializing the directory

After enabling Directory Services in System Configuration, additional configuration data must be provided. Directory Services looks for that data in ADX_IDT1:ADXLDIFF.DAT. Note that the size of ADXLDIFF.DAT is limited to 65,536 bytes. A sample file, ADX_IDT1:ADXLDIFF.SMP is provided. The sample file contains extensive commenting to help you create your own ADXLDIFF.DAT file. Some of those comments will be reproduced below.

When 4690 OS initializes LDAP, it saves a copy of ADX_IDT1:ADXLDIFF.DAT internally. On each subsequent reboot, the internal version number is compared to the version number in ADX_IDT1. If a version change is detected, the existing LDAP directory is erased, and LDAP is re-initialized using the data from ADX_IDT1:ADXLDIFF.DAT. The new version becomes the one saved internally.

Note: If LDAP has been initialized, and ADX_IDT1:ADXLDIFF.DAT is not found, by default there is no version change. For security reasons, we suggest that you remove user definitions from ADX_IDT1:ADXLDIFF.DAT after LDAP has been initialized. Do not remove the file, as it may be needed during a LAN Disk Rebuild to setup a new controller.

Verifying installation

During initialization, the LDAP utility *slapadd* is used to populate the directory with entries from ADXLDIFF.DAT. Output from *slapadd* is saved in F:\adxetc\logs\slapadd.log. The results from adding each entry are logged. If any of the add commands fail, initialization fails and the LDAP database remains empty. Correct the entry and restart the controller to rerun the installation process.

ADXLDIFF.DAT

ADXLDIFF.DAT has two sections. The first section is the header. The header contains configuration attributes that help 4690 OS setup the LDAP directory. The second section contains initial entries for your directory, specified in LDAP Interchange format. The entries are grouped by object class. Comments are provided in ADXLDIFF.DAT for each class, and for each entry. Entries required by 4690 OS are noted.

The ADXLDIFF.DAT Header

As mentioned above, the header contains configuration attributes used by 4690 OS to set up the LDAP directory. The required attributes are:

- **version: 001**

This is a character string indicating the current version of the ADXLDIFF.DAT file. Any change in this string signals a change of version. If a version change is detected, the existing LDAP directory is erased, and LDAP is re-initialized using the data from ADX_IDT1:ADXLDIFF.DAT. The new version becomes the one saved internally.

- **root: dc=myCorp,dc=com**

This is the top entry in the LDAP directory, also called the base or the suffix. The distinguished name of every directory entry contains the root. Here are two examples:

- a. dc=toshiba,dc=com
- b. l=raleigh,o=toshiba,c=us

- **admin: cn=administrators,ou=groups,dc=myCorp,dc=com**

This is the distinguished name of the Administrator group. The Administrator group is a special groupOfNames object in the 4690 OpenLDAP implementation. You **must** be a member of this group to have read/write access to the 4690 LDAP database. The Administrator group must be defined in this file, using the same distinguished name. See “The Initial Administrator” on page 345 for more details.

- **security: level=simpletls,strength=112**

This sets the security level for communications between the LDAP server and clients. The “level” can be set to one of three values.

1. “level=none” indicates no communication security will be used. This is not recommended.
2. Set “level=simpletls,strength=xxx” to use a simple TLS (Transport Layer Security) connection. xxx indicates the encryption key length. Set it to an integer between 40 and 256. 112 is equivalent to 3DES. When level is set to simpletls, 4690 Directory Services will generate self-signed certificates for the LDAP server. LDAP clients attempting to connect to the 4690 LDAP server must have a certificate so an encryption method can be negotiated, but no certificate checking is done.
3. Set “level=tls,strength=xxx” to use a TLS connection. xxx indicates the encryption key length. Set it to an integer between 40 and 256. 112 is

equivalent to 3DES. When level is set to `tls`, the user is expected to provide the certificates used during TLS authentication:

- `f:/adxetc/ext/adxslapd/cacert.pem` - filename of the Certificate Authority certificate. If the file cannot be found, initialization fails.
- `f:/adxetc/ext/adxslapd/xxcert.pem` - filename of the LDAP server certificate, where `xx` is the controller ID. If the file cannot be found, initialization fails.
- `f:/adxetc/ext/adxslapd/xxkey.pem` - filename that contains the private key matching the LDAP server certificate, where `xx` is the controller ID. If the file cannot be found, initialization fails.

- **defaultpp: `cn=default,ou=policies,dc=myCorp,dc=com`**

This is the distinguished name of the default password policy. You may specify a password policy to use in each user definition. If a user does not have a defined password policy specified in the `pwdPolicySubentry` attribute, LDAP will use the default password policy. If you do not want a default password policy, enter `defaultpp: none`. See “Password Policy Settings” on page 347 for more details.

- **clientcertificates: `no`**

This indicates whether the user is providing client certificates to the 4690 OS. Client certificates are typically configured in the `servers ldap.conf` file. 4690 Directory Services code that accesses the LDAP database could be required to authenticate using the client certificates, depending on your configuration settings. The `clientcertificates` attribute is ignored unless the security level is set to `"tls"`. The attribute has three possible settings:

1. `yes` - the user will provide client certificates. If the following file names are not found, initialization will fail:
 - `f:/adxetc/ext/adxslapd/xxclcert.pem` - Filename of the client certificate, where `xx` is the controller ID.
 - `f:/adxetc/ext/adxslapd/xxclkey.pem` - Filename that contains the private key matching the client certificate, where `xx` is the controller ID.
2. `no` - client certificates will not be used.
3. `server` - the server certificates configured above will also be used as client certificates.

- **verifyserver: `never`**

Must be set to `"never"`, `"allow"`, `"try"`, or `"demand"`, as described in the OpenLDAP Administrator's Guide. Set this to `"demand"` for the strictest security requirements.

- **verifyclient: `never`**

Must be set to `"never"`, `"allow"`, `"try"`, or `"demand"`, as described in the OpenLDAP Administrator's Guide. Set this to `"demand"` for the strictest security requirements. Note that setting `clientcertificates` to `"no"`, then setting `verifyclient` to `"demand"` is an invalid configuration and will cause initialization to fail.

The line `@@End of configuration` header signals the end of the configuration section. Do not remove this line from your `ADXLDIFF.DAT` file.

ADXLDIFF.DAT Directory Entries

After the header, you define the initial entries for your directory using LDIF, LDAP Data Interchange Format. LDIF is simply a way to specify a directory entry in a plain text format.

We suggest you limit the directory entries defined in ADXLDIFF.DAT to those entries required to get the directory up and running. The sample file includes the following types of entries:

- Root entries (required)
- An administrator group (required)
- An initial administrator (required)
- 4690 User Authorization roles and groups (recommended)
- Password Policy settings (optional)

Note: The contents of ADXLDIFF.SMP is subject to change. The most current version, with detailed documentation, can be found in the ADX_IDT1 directory.

Root Entries

LDAP directory entries are organized in a tree-like structure. You will need to define the root entries used by the other entries in the file. For example, the Figure 22 sample file defines the following tree structure:

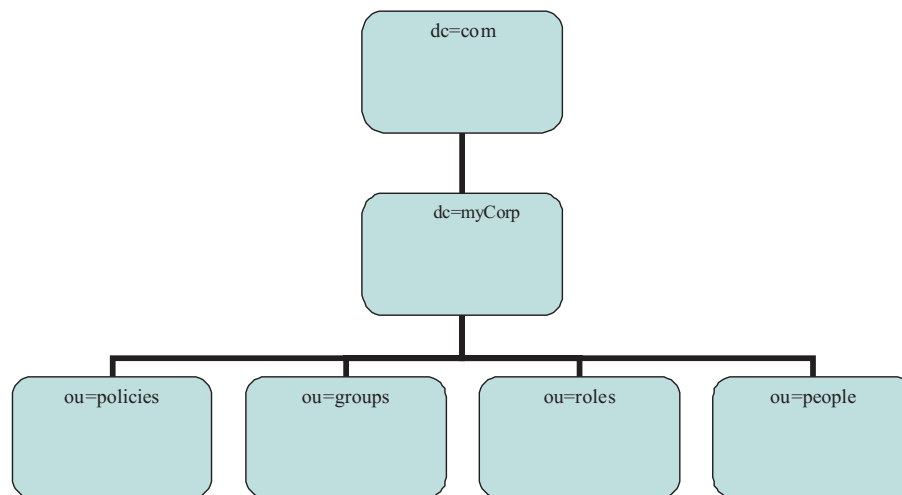


Figure 22. Root Entries

The entries in LDIF format, from the sample file:

```
## The root node
dn: dc=myCorp,dc=com
objectClass: dcObject
objectClass: organization
dc: myCorp
o: My Corporation

## The policies OU
dn: ou=policies,dc=myCorp,dc=com
objectClass: organizationalUnit
objectClass: top
ou: policies

## The groups OU
```



```
dn: ou=groups,dc=myCorp,dc=com
objectClass: organizationalUnit
objectClass: top
ou: groups
```

```
## The roles OU
dn: ou=roles,dc=myCorp,dc=com
objectClass: organizationalUnit
objectClass: top
ou: roles
```

```
## The people OU
dn: ou=people,dc=myCorp,dc=com
objectClass: organizationalUnit
objectClass: top
ou: people
```

The Initial Administrator

The 4690 OS limits LDAP access to members of a special "administrator" group. The group may have multiple members, but it must have at least one. Make sure the distinguished name (dn:) you specify matches the one you specified in the header. In the sample file, a user called "Default Administrator" is created to populate the administrators group.

```
dn: cn=administrators,ou=groups,dc=myCorp,dc=com
objectClass: groupOfNames
cn: administrators
ou: groups
member: cn=Default Administrator,ou=people,dc=myCorp,dc=com
```

The default administrator definition:

```
dn: cn=Default Administrator,ou=people,dc=myCorp,dc=com
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: Default Administrator
sn: dummy value
uid: admin
userPassword: {SHA}0DPiKuNIrrVmD8IUCuw1hQxNqZc=
destinationIndicator: group=2,user=1
pwdChangedTime: 20110101000000Z
ou: people
```

A note about inetOrgPerson:

The inetOrgPerson object class is a general purpose object class that holds attributes about people. Almost all LDAP distributions include this class. See RFC2798 for a full description. inetOrgPerson is based on the object class organizationalPerson, which is based on the objectclass person. An inetOrgPerson has two required attributes, cn (commonName) and sn (surname). There are many optional attributes available for your use.

4690 OS requires two of the optional attributes to be defined if the inetOrgPerson is going to logon to the 4690 OS. They are:

- uid - a 1 to 8 character user ID. The ascii characters 0x21 through 0x7E are valid. Every uid attribute in a store's directory must be unique. A inetOrgPerson object can have multiple user IDs defined, as long as they are unique. 4690 OS will not allow an ID to logon if that ID is found in more than one inetOrgPerson.
- 4690 Directory Services provides a way to support user IDs up to 72 characters long. See the section on "Support for Long User IDs" on page 348 below.

- userPassword - a 1 to 72 character password. The ascii characters 0x21 through 0x7E are valid. Your password policy settings may put additional restrictions on the password.

The default administrator's user password in the sample file is admin, the same as the user ID, encrypted using SHA1 and entered in Base64 format. 4690 OS will support the following encryption schemes:

- Clear Text
- SSHA
- CRYPT
- MD5
- SMD5
- SHA

See the *OpenLDAP Software Administrator's Guide's* section on password storage for more information.

4690 User Authorization Roles and Groups

In the 4690 OS, each ID is associated with a set of authorizations that indicate what operations that ID is allowed to do. For example, an ID might be authorized to use command mode, but not authorized to change configuration. In LDAP, group membership is used to represent these authorizations. For example, users that are members of the group ADX_CPW_ATTR_COMMAND_MODE have command mode access when signed onto the 4690 OS. Users that are not members of this group cannot use command mode.

There are over 75 4690 authorizations, each represented by a group. To help you manage these authorizations, 4690 OS supports the use of roles using the object class organizationalRole. A role is defined as a collection (or set) of authorizations. Users assigned to a role automatically receive all the authorizations assigned to that role. If you change the authorizations of a given role, it changes the authorizations for each user assigned to that role.

A role called Store Manager is defined in the sample file. We could have been more general and called it "full access" or "superuser", because this role is enabled for every 4690 authorization. We did this by placing the role in each of the 4690 authorization groups. A Default Store Manager user ID is included in the sample file, and it is a "roleOccupant" of the Store Manager role. This gives the ID all 4690 authorizations, just like the role. If you have 20 users that need full 4690 OS authorization, you simply add the 20 users to the Store Manager role, instead of adding all 20 users to every 4690 authorization group.

Roles and groups can be used in combination. For example, suppose role **x** is in 20 authorization groups. You can place a user in role **x** plus two additional authorization groups, for a total of 22 authorizations.

The roles and attribute groups required by 4690 OS are included in the sample file. They are not required to be in the ADXLDIFF.DAT file. However, they must be included in the LDAP directory before the directory IDs are used for 4690 OS logon.

4690 OS authorization records have a group field and a user field. The group field is an integer from 2 to 254. The user field is an integer from 1 to 254. These fields have the following documented meaning:

- Group 2, user 1 has owner access rights to application files (R, W, D)

- Group 2, user not 1 has group access rights to application files (R)
- Anything else has no access rights

If you have a need for these fields, you can set their values by using the attribute "destinationIndicator". For example:

- destinationIndicator: group=2,user=1

The default administrator in the sample file has the destinationIndicator set to group=2,user=1. Do not use a blank between the group and user.

If the destinationIndicator attribute is missing from a user record, the setting defaults to group=2,user=1.

Password Policy Settings

User passwords, like all other data in the LDAP directory, are managed at the enterprise level. However, you can have the LDAP server assist you in password management by specifying a password policy. The password policy is stored in the LDAP directory. It should follow the form provided below, with an updated root replacing the sample "dc=myCorp,dc=com".

The sample password policy:

```
## See the RFC draft titled draft-behera-ldap-password-policy-09 for complete
## details on the password policy.
dn: cn=default,ou=policies,dc=myCorp,dc=com
objectClass: pwdPolicy
objectClass: person
objectClass: top
cn: default
## This attribute indicates whether users can change their own passwords,
## subject to access control. 4690 OS does not currently allow users access
## to their own passwords.
pwdAllowUserChange: FALSE
## This holds the name of the attribute to which the password policy is
## applied. 4690 OS requires it to be set to 'userPassword'.
pwdAttribute: userPassword
## This attribute indicates how password quality will be verified. The
## following values are valid:
## 0 - quality checking is not enforced
## 1 - the server checks the quality, but if it is unable to check (due to
## a hashed password or some other reason) it will be accepted.
## 2 - the server checks the quality, but if it is unable to verify the
## quality, an error is returned.
## We recommend you set this to 1. You should be encrypting your passwords, so
## password length checks cannot be verified.
pwdCheckQuality: 1
## The number of seconds before a password expiration warning is returned.
## If not zero, it must be less than pwdMaxAge
pwdExpireWarning: 0
## LDAP keeps count of failed sign-on attempts. After this many seconds,
## the count is reset. If set to zero, the failure count is only reset by a
## successful sign-on
pwdFailureCountInterval: 60
## This is the number of times an expired password can still be used to sign-on.
## 4690 OS ignores this setting during sign-on.
pwdGraceAuthNLimit: 0
## This is the number of previous passwords stored. Previous passwords, if
## stored, cannot be reused.
pwdInHistory: 4
## This is a TRUE/FALSE setting indicating whether a user should be locked out
## after a certain number of unsuccessful sign-on attempts. That number is
## specified in another setting, pwMaxFailure. 4690 OS sign-on does not use
```

```

## this setting.
pwdLockout: FALSE
## If pwdLockout is TRUE, this is the number of seconds that must pass before
## a locked out user can retry their password. If set to zero, the password
## must be reset by an administrator.
pwdLockoutDuration: 0
## The number of seconds before a password expires. Zero means the password
## does not expire. pwdMaxAge must be >= pwdMinAge
pwdMaxAge: 0
## When pwdLockout is TRUE, this is the maximum number of sign-on attempts you
## can make before you can no longer sign-on. Zero means there is no maximum.
pwdMaxFailure: 5
## The number of seconds a password must be active before it can be changed.
pwdMinAge: 0
## If quality checking is enabled, this is the minimum number of characters
## allowed in a password.
pwdMinLength: 2
## A TRUE/FALSE settings indicating whether a user must change their password
## after it is set or reset by an administrator. Since 4690 does not allow
## users to change their passwords, set this to FALSE.
pwdMustChange: FALSE
## A TRUE/FALSE settings indicating whether password change requests must
## contain both the current and the new password.
pwdSafeModify: FALSE
sn: dummy value

```

If you are using a password policy, you need to include the attribute "pwdChangedTime" in your user definitions. It indicates the time the password was last changed. This time is used along with pwdMaxAge and pwdExpireWarning to determine when a password expires, and when password expiration warnings should be issued.

Support for Long User IDs

4690 Directory Services will allow you to logon with an ID up to 72 characters long, while providing a short ID to the 4690 OS. For example, you could logon using the ID dave@myCorp.com, and have the 4690 OS use the ID dave@.

To allow a user to logon using a long ID, define one or more long user IDs (greater than 8 characters) in that user's inetOrgPerson objectClass. Then define exactly one short user ID. The short ID will be used as the login ID for the 4690 OS. If multiple short user IDs exist, any long IDs will not be allowed to logon because they cannot be resolved to a specific short ID.

Here is a sample inetOrgPerson definition:

```

dn: cn=Dave Smith,ou=people,dc=myCorp,dc=com
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: Dave Smith
sn: Smith
## A single short ID. User can logon using dave@
uid: dave
## A long ID. Logon with dave@myCorp.com, 4690 OS will use dave@.
uid: dave@myCorp.com
## It is ok to have multiple long IDs, as long as there is just one short ID.
uid: DaveSmith@myCorp.com
userPassword: password
destinationIndicator: group=2,user=1
pwdChangedTime: 20110101000000Z
ou: people

```

Logging on to using Directory Services IDs

To logon using a short ID stored in the LDAP directory, enter the one to eight character user ID, followed by the '@' character, then press **Enter**. The '@' character is not considered part of the ID, it is just used to direct the OS to look for the ID in the LDAP directory.

To logon using a long ID stored in the LDAP directory, enter the 9 - 72 character user ID. Adding a trailing '@' character is optional, unless it is the 10th character. Trailing '@' characters are not considered part of the ID, but '@' characters in other positions are considered part of the ID. You are allowed to define the user ID "hastrailing@". Enter "hastrailing@" on the login prompt to use this ID.

If an ID is not found in the LDAP directory, the standard ID not found message is displayed.

If a password policy has been enabled for a user when signing on, the user could see a Password expires in xx days warning message, then they would press **Enter** to complete the signon process. The user could also see a Password has expired error message. In this case, the user can only return to the signon screen. Users cannot change the passwords of IDs stored in the LDAP directory. Changes must come from the Enterprise, or an Enterprise-provided application.

Updates to the Enhanced Security API

The Enhanced Security function added application programming interfaces (APIs) that allow more direct control over user authentication and security management. Several Enhanced Security API calls have been updated to support LDAP directory IDs. They are:

- **IsAvailable** – this call returns 0 if both Enhanced Security and Directory Services are unavailable. A positive value is returned if one or both are active.
- **GetCurrentUserID** – this call will return the current user's ID even if they logged on using an ID stored in the LDAP directory. An '@' character will be included at the end of the ID to indicate it came from the LDAP directory. If a user logged on with a long ID, the call returns the short ID that was passed back to the 4690 OS.
- **ValidateID** – if the ID ends with '@', it will be searched for in the LDAP directory instead of the enhanced authorization file
- **Validate** – validates the user ID and password of users in the LDAP directory when the ID ends with '@'.
- **GetAllAttributes** – returns an attribute string for IDs in the LDAP directory when the ID passed in ends with '@'
- **GetAllAttributesCurrentUser** – returns the attribute string of the current user, even if the current user ID came from the LDAP directory

Three API calls were modified to return a `ADX_CPW_ERR_NOT_SUPPORTED` error if the user ID contained a '@' character. They are:

- **StartSession**
- **StartSessionCurrentUser**
- **ChangePassword**

Additions to the Enhanced Security API

Two new API calls have been implemented specifically for Directory Services. They are:

IsMemberOf

IsMemberOf checks for group membership of a user with the provided ID and password. The group name provided is the common name (cn) of a LDAP groupOfNames object.

As described earlier, 4690 OS uses group membership to determine user authorization. For example, IsMemberOf("123","456", "ADX_CPW_ATTR_COMMAND_MODE") will tell you whether user 123 is authorized to enter command mode on the 4690 OS. You can create your own custom authorization groups and use IsMemberOf to determine which users have that authorization.

IsMemberOf supports the use of roles. If the user is a roleOccupant of an organizationalRole object and that role is a member of a group, then the user is considered a member of that group.

If the ID passed in is an empty string, IsMemberOf uses the ID of the current user. In this case, the password is ignored.

IsMemberOf is only valid for LDAP IDs. ADX_CPW_ERR_NOT_SUPPORTED is returned otherwise.

If the requested group does not exist, zero (false) is returned.

C Interface:

```
long ADX_CPW_IsMemberOf(const char *id, const char *pw, const char *group);
```

Java Interface:

```
(com.ibm.OS4690.security.EnhancedSecurity)  
boolean isMemberOf(String id, String password, String group);
```

Parameters:

Id – a non-null string that is either zero length (empty string), or from 2 to 72 characters in length.

The string must end with the '@' character.

An empty string is a special case that signals the ID of the current user should be used.

Password – a string from 1 to 72 characters in length.

If the ID is an empty string, meaning the current user ID is used, then the password parameter is ignored.

Group – a non-null string up to 256 characters in length.

Return Codes:

0 – false, the user is not a member of this group

1 – true, the user is a member of this group

ADX_CPW_ERR_SERVICE_NOT_AVAILABLE

ADX_CPW_ERR_ID_NOT_FOUND

ADX_CPW_ERR_PASSWORD_DOES_NOT_MATCH
ADX_CPW_ERR_INVALID_DATA
ADX_CPW_ERR_NOT_SUPPORTED

GetGroups

Returns a buffer containing the common name (cn) of every groupOfNames objectClass the user is a member of. The group names are separated by a null character and the end of the data is signified by two consecutive null characters. The user's ID and password are required. The data is returned in the location pointed to by the parameter buffer.

If skip4690 is greater than zero, group names starting with ADX_CPW_ATTR (the 4690 OS user attribute groups) are excluded. This can be useful when working with user defined groups.

GetGroups supports the use of roles. If the user is a roleOccupant of an organizationalRole object, and that role is a member of a group, then the user is considered a member of that group and the group will be included in the returned string.

If the ID passed in is an empty string, GetGroups uses the ID of the current user. In this case, the password is ignored.

The maximum size of the return buffer is 65535 for 32-bit C and Java. 16-Bit C code has a buffer limit of 65458. If the buffer size is too small to hold the data, ADX_CPW_ERR_INVALID_BUFFER is returned.

GetGroups is only valid for LDAP IDs. ADX_CPW_ERR_NOT_SUPPORTED is returned otherwise.

C Interface:

```
long ADX_CPW_GetGroups(const char *id, const char *pw, short skip4690,  
char *buffer, long bufsize)
```

Java Interface:

```
(com.ibm.OS4690.security.EnhancedSecurity) ArrayList getGroups(String id,  
String password, boolean skip4690)
```

Parameters:

Id – a non-null string that is either zero length (empty string), or from 2 to 72 characters in length. The string must end with the '@' character. An empty string is a special case that signals the ID of the current user should be used.

Password – a string from 1 to 8 characters in length. If the ID is an empty string, meaning the current user ID is used, then the password parameter is ignored.

Skip4690 – Zero indicates false, all other values indicate true.

Buffer – A non-null pointer to an area used to store the results

Bufsize – The size of the buffer you provided. Must be greater than zero and less than 65536. For 16-Bit C code, it must be less than 65459.

Return Codes:

0 – Success
ADX_CPW_ERR_SERVICE_NOT_AVAILABLE
ADX_CPW_ERR_ID_NOT_FOUND

ADX_CPW_ERR_PASSWORD_DOES_NOT_MATCH
ADX_CPW_ERR_INVALID_DATA
ADX_CPW_ERR_INVALID_BUFFER
ADX_CPW_ERR_NOT_SUPPORTED

Replication

On multiple controller systems, LDAP Sync Replication is used to maintain a copy of the LDAP database on each controller. Updates sent to one LDAP provider (master) will be distributed to every controller in the store.

MirrorMode replication is used on 4690 OS master and alternate master controllers. In this configuration, both the master and alternate master are LDAP providers (masters) and they replicate from each other. Updates should be sent to only one controller, normally the 4690 OS master. If that controller is unavailable, updates can then be redirected to the second controller (alternate master) until the first is back online. This redirection is not automatic. It is up to the enterprise to determine which of the two controllers will receive updates at any given time.

4690 OS subordinate controllers *listen* for updates by running LDAP Sync Replication using the `refreshandPersist` option. Subordinate controllers listen to both the master controller and alternate master controller, so updates will take place as long as one of the providers is available.

On multiple controller systems, 4690 OS creates an `inetOrgPerson` with the common name “`replicationID`” and saves it in the directory. This ID is used by the OS during the replication process. Do not erase or modify this ID.

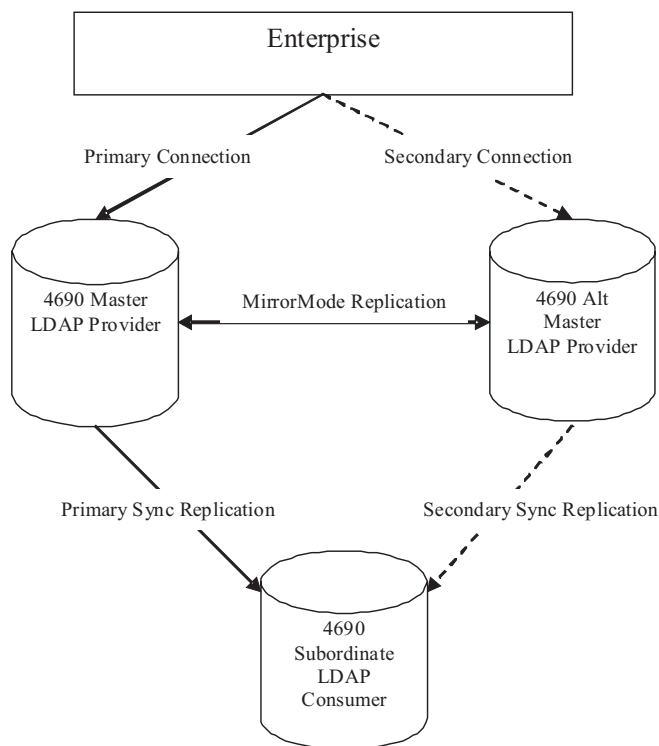


Figure 23. LDAP Sync Replication

Connecting to a 4690 OS Directory Server

To establish a connection with the LDAP server (slapd) on a 4690 controller, you need the following:

- The controller's ip address (or hostname)
- The distinguished name of a user in the controller's administrator group and the password for that user

The following sample code uses the LDAP API to connect to a 4690 controller that has the `simpletls` Transport Layer Security (TLS) option enabled:

```
LDAP *ldap;
int *ver = 3;
ldap_initialize(&ldap,"ldaps://lan0.adxlxcn.adxautonet:636");
ldap_set_option(ldap,LDAP_OPT_PROTOCOL_VERSION,&ver);
ldap_simple_bind(ldap,
"cn=Default Administrator,ou=people,dc=myCorp,dc=com","admin");
```

This example uses the `ldapsearch` utility to return all the `inetOrgPerson` entries in a database:

```
ldapsearch -x -ZZ -D "cn=Default Administrator,ou=people,dc=myCorp,dc=com" -w admin -h hostname -b "dc=myCorp,dc=com" "(objectclass=inetOrgPerson)"
```

See the `ldapsearch` documentation or man page for complete details.

Transport Layer Security

When you set `"level=simpletls,strength=xxx"` in the `adxdiff.dat` header, the 4690 OS LDAP server will use TLS for all communication. The Transport Layer Security provides encrypted communication between the 4690 OS LDAP server and client requests. The `"strength=xxx"` specifies the minimum key length that 4690 OS will accept, with `xxx` ranging from 40 to 256. Any system trying to connect to the 4690 OS LDAP server must support this key length.

4690 OS uses a self-signed certificate with a 2048 bit key to negotiate an acceptable TLS protocol. A new certificate is generated when a version change causes the LDAP database to be re-initialized from the current `adxdiff.dat` file.

If you have trouble getting your LDAP client to accept 4690 OS's certificate, add the statement `"TLS_REQCERT allow"` to the file `ldap.conf`.

Setting the security level to `"tls"` works much like `"simpletls"` except you provide the certificates used by the 4690 OS. You also provide additional configuration settings. Here is a summary of how this data is used:

The certificate `f:/adxetc/ext/adxslapd/cacert.pem` you provide is used to configure the `TLSCACertificateFile` directive in the `slapd.conf` configuration file.

The certificate `f:/adxetc/ext/adxslapd/xxcert.pem` (`xx` = controller ID) you provide is used to configure the `TLSCertificateFile` directive in the `slapd.conf` configuration file.

The certificate key `f:/adxetc/ext/adxslapd/xxkey.pem` (`xx` = controller ID) you provide is used to configure the `TLSCertificateKeyFile` directive in the `slapd.conf` configuration file.

The value you set the LDAP configuration attribute `verifyclient` to is used to set the `TLSVerifyClient` directive in the `slapd.conf` file.

The value you set the LDAP configuration attribute `verifyserver` to is used to set the `TLS_REQCERT` directive in the `.ldaprc` configuration file.

The certificate `f:/adxetc/ext/adxslapd/cacert.pem` you provide is used to configure the `TLS_CACERT` directive in the `.ldaprc` configuration file.

The `TLS_CERT` directive in the `.ldaprc` configuration file is set to `f:/adxetc/ext/adxslapd/xxclcert.pem` (`xx` = controller ID), a file you provide when you set the LDAP configuration attribute `clientcertificates` to `"yes"`. Setting `clientcertificates` to `"server"` sets `TLS_CERT` to `f:/adxetc/ext/adxslapd/xxcert.pem` (`xx` = controller ID).

The `TLS_KEY` directive in the `.ldaprc` configuration file is set to `f:/adxetc/ext/adxslapd/xxclkey.pem` (`xx` = controller ID), a file you provide when you

set the LDAP configuration attribute clientcertificates to "yes". Setting clientcertificates to "server" sets TLS_KEY to f:/adxetc/ext/adxslapd/xxkey.pem (xx = controller ID).

For 4690 Directory Services, the domain name of a 4690 controller is lan0.adxlx??n.adxautonet, where ?? is the controller's ID. This name must be used as the server's CN (common name) when creating certificates for the 4690 controller.

Additional information

The table below lists several sources of additional information for the topics discussed in this chapter:

Table 44. Additional information

OpenLDAP Main Site	www.openldap.org
RFC 2256 – A Summary of the X.500(96) User Schema for use with LDAPv3	www.faqs.org/rfcs/rfc2256.html
RFC 2798 – Definition of the inetOrgPerson LDAP Object Class	www.faqs.org/rfcs/rfc2798.html
RFC 4519 – Lightweight Directory Access Protocol (LDAP): Schema for User Applications	www.faqs.org/rfcs/rfc4519.html
Password Policy for LDAP Directories	http://tools.ietf.org/html/draft-behera-ldap-password-policy-09
RFC 2849 – The LDAP Data Interchange Format (LDIF) – Technical Specification	www.faqs.org/rfcs/rfc2849.html .
Enhanced Security API	<i>4690 OS: Programming Guide</i>

Part 2. 4690 Operating System Utilities

Chapter 15. Using the Display/Alter File Utility

This chapter explains how to use the Display/Alter File Data Utility provided by the operating system for displaying and altering data files. The utility is a menu-driven interface that allows you to display, scan, or alter file sectors on all supported diskettes and hard disk drives.

Note: The same functions can be performed on physical sectors of a disk or diskette, but only service personnel are authorized to alter them.

Accessing the Display/Alter File Utility

You can access the utility from Command Mode by entering:

ADXCSJ0L

You can display, alter, or scan files according to the access rights of each file.

You can use the utility in either *File Mode* or *Physical Sector Mode*.

Logging ADXCSJ0L Activity

If the logical name ADXCMDLG is defined, then 4690 OS will log a message to the system log whenever ADXCSJ0L opens or writes to a file.

Data will also be logged to a log file in the ADX_CLOG directory each time that ADXCSJ0L is invoked. Changes made to data using ADXCSJ0L are normally within a sector. 4690 OS will write an entire sector (in ASCII characters) when the WRITE function key is pressed.

The temporary log files for ADXCSJ0L have file names with the format of UUUUUUUU.Jnn, where UUUUUUUU is the one to eight character user ID and nn is the next sequential value (in the range 00-99) that is available in the directory. The logged information will include the operator ID, filename, sector number, and data.

Using File Mode

The first panel in File Mode, the SELECT FILE/DIRECTORY panel, contains the names of files and subdirectories contained in the current directory. From this panel, you can select a file or change to another subdirectory.

Selecting a Disk Drive

Use Physical Sector Mode to change the disk drive you want to access. Then you can return to File Mode to select a file.

Changing the Current Directory

To change the current directory, move the cursor to the appropriate directory entry shown on the panel. Press **F7** and the directory changes to the directory you selected. The parent directory entry of a subdirectory is denoted by two periods.

Selecting a File

Files will be in alphabetical order. To select a file, move the cursor to the appropriate name on the panel and press **Enter**. A panel appears that allows you to display or alter file sectors.

Displaying and Altering the File

The file modification panel is called the DISPLAY/ALTER FILE DATA menu. It displays one half of a file sector at a time. You can page up and page down to examine the first and second half of a sector or to get to the next or previous sector.

To alter the data in a sector, perform the following steps:

1. Type the appropriate hexadecimal data in a hexadecimal data field or the appropriate ASCII data in an ASCII data field in any location of either half sector.
2. Press **F6** to write the changes for that half sector to the disk. You can change the data in only one half sector at a time. Therefore, if you page to another half sector before you press **F6** for this half sector, the changes are discarded.
3. To discard the changes you entered on the menu before you pressed **F6**, press **F5**.
4. To select a new relative sector within the file, press **F9**, type the appropriate relative sector number, and press **Enter**.

The following table describes special functions that you can perform on the file modification panel:

Function	Key
Request the File Scan function.	F4
Retract changes entered on the screen but not written to the file for the sector being displayed.	F5
Write changes to a file.	F6
Save the current sector (SCS).	F7
Replace the current sector (RCS) with the sector last saved by the SCS key.	F8
Select a new relative sector.	F9

You can scroll through the sector using the PgUp and PgDn keys. The PgUp key moves you to the next half sector of data toward the beginning of the file. The PgDn key moves you to the next half sector of data toward the end of the file. When you press either of these scrolling keys, any data you entered on the panel but have not saved on disk is lost.

Scanning a File

Press **F4** to enter the File Scan function. The utility displays the FILE SCAN panel to obtain the information to control the scan.

On this panel, you specify whether to search the current file for an ASCII or hexadecimal string (A/H). You can enter up to 20 ASCII characters or up to 20 hexadecimal bytes. If the utility finds the string, it displays the relative file sector and the relative offset within the file sector. You can continue searching for the next occurrence of the string. If a new string is entered, the search begins at relative file sector zero.

To return to the file modification panel, press **F3**. To return to the SYSTEM MAIN MENU, press **F10**.

Using Physical Sector Mode

In Physical Sector Mode you can display and alter the contents of any data sector on a diskette or any data sector in the active partition on the hard disk drive. If you started the Display/Alter File Data Utility from Command Mode, you can only display sectors. If you started the Display/Alter File Data from the SYSTEM MAIN MENU, you can modify sectors if no other programs are using files on the selected drive. To modify sectors on drive C, you must use the Supplemental Option using the CD-ROM because the operating system always has files open on drive C.

If you want to modify sectors on the hard disk drive that are not in the current partition, select **Display/Alter Physical Sector Mode for the Fixed Disk** on the FILE UTILITIES menu on the Supplemental Option using the CD-ROM, and use Physical Sector Mode as described in this section.

Use Physical Sector Mode to switch the use of the Display/Alter File Data Utility from one disk drive to another.

To enter Physical Sector Mode, select option **2** from the FILE UTILITIES menu. Then select option **2** from the SELECT DISPLAY/ALTER FUNCTION menu.

Selecting a Disk Drive

To change the current disk drive, enter the disk drive number on the SELECT PHYSICAL SECTOR menu. You can enter a value from 0 to 4, which corresponds to drives A: to E:.

Selecting a Sector

The first panel in Physical Sector Mode is the SELECT PHYSICAL SECTOR panel. This panel lets you select the drive number and the head/sector/cylinder address of the sector to be used. Pressing **Enter** transfers to the sector modification panel.

Displaying and Modifying a Sector

The sector modification panel is called the DISPLAY/ALTER SECTOR DATA panel. It displays one half of a file sector at a time. You can page up and page down to examine the first and second half of a sector or to get to the next or previous sector. You cannot modify the first sector on a diskette.

To alter the data in a sector, perform the following steps:

1. Type the appropriate hexadecimal data in a hexadecimal data field or the appropriate ASCII data in an ASCII data field in any location of either half sector.
2. Press **F6** to write the changes for that half sector to the disk. You can change the data in only one half sector at a time. Therefore, if you page to another half sector before you press **F6** for this half sector, the changes are discarded.
3. To discard the changes you have entered on the panel before you press **F6**, press **F5** to undo the changes.

The following list describes the function keys, cursor movement keys, and scrolling keys available on the sector modification panel.

Function Keys

- F3** Transfers control to the SELECT PHYSICAL SECTOR panel.
- F4** Requests the Sector Scan function.
- F5** “Undo” any changes that were entered on the panel but not written to the file for the sector being displayed.
- F6** Writes changes to the sector. In Physical Sector Mode, this requires that no files are open on the drive containing the sector.
- F7** Saves the current sector (SCS).
- F8** Replaces the current sector (RCS) with the sector last saved by the SCS key.
- F10** Returns you to the Main Menu.

Cursor Movement Keys

Cursor Down

Moves the cursor down one field.

Cursor Up

Moves the cursor up one field.

Tab Moves the cursor horizontally to the next or previous hexadecimal or ASCII field.

Back Tab

Moves the cursor to the previous field.

End Moves the cursor to the bottom-right field.

Left Arrow

Moves the cursor to the previous position.

Right Arrow

Moves the cursor to the next position.

Home Moves the cursor to the top-left field.

Cursor Right

Moves the cursor one character right in the field.

Cursor Left

Moves the cursor one character left in the field.

Scrolling Keys

PgUp Moves to the next half sector of data toward the beginning of disk.

PgDn Moves to the next half sector of data toward the end of the disk.

Note: When a scrolling key is pressed, any data that has been entered on the panel and that has not been written to the disk is not saved.

Scanning a Sector

Press **F4** to enter the Sector Scan function. The utility displays the CYLINDER/SECTOR SCAN panel to obtain the information to control the scan.

On this panel, specify the range of the search and whether to search the current file for an ASCII or hexadecimal string (A/H). You can enter up to 20 ASCII characters or up to 20 hexadecimal bytes. If the string is found, the physical sector and the

relative offset within the file sector appear. You can continue searching for the next occurrence of the string. If a new string is entered, the search begins at offset zero of the starting sector of the range.

To return to the sector modification panel, press **F3**. To return to the SYSTEM MAIN MENU, press **F10**.

Using the Critical Sector Repair Function

To replace all of the data for a sector, use the Critical Sector Repair (CSR) function. This function is selectable from both File Mode and Physical Sector Mode. For example, you can save a sector of data in File Mode and restore the data in Physical Sector Mode. Also, you can transfer data sectors between diskettes and hard disk drives.

The function allows you to save the data from any sector in a memory buffer and copy the data from the memory buffer to the sector currently being displayed. Typically, this function is used to repair the boot record sector. Use the following Critical Sector Repair procedure when you are using two diskettes:

1. Insert a diskette with a good copy of the sector to repair.
2. Locate the good sector on the diskette.
3. Press the **SCS** key (**F7**) to save the current sector.
4. Insert the diskette with the defective sector. The two diskettes must be of the same type if the boot record sector is being repaired.
5. Locate the defective sector on the diskette.
6. Press the **RCS** key (**F8**) to replace the defective sector.

You can use a hard disk drive as the source of the good data or the destination of the repair data.

Chapter 16. Using the software maintenance utilities

This chapter describes the maintenance utilities of 4690 OS. The first part describes the Apply Software Maintenance (ASM) utility. The second part describes methods to remotely apply maintenance updates to 4690 OS. The third part describes the Build Software Maintenance Control File utility.

Apply Software Maintenance utility

Apply Software Maintenance (ASM) is a utility used to apply and manage changes to the operating system, the 4680 or 4690 application program products, and your application programs.

There are three procedures for using ASM:

- Transfer, test, and cancel maintenance
- Transfer, test, and accept maintenance
- Transfer and accept maintenance

ASM provides a two- or three-step procedure for applying these changes depending on whether you want to test the maintenance changes. You can choose not to test the maintenance if you are applying maintenance to many store controllers after a thorough test has been completed at the test site.

1. Transfer the Maintenance on the CD-ROM

ASM transfers replacement modules from the maintenance on the CD-ROM to a maintenance subdirectory. The maintenance subdirectory depends on which product is being transferred.

Note: Prior to 4690 OS V3R1, the maintenance diskettes were referred to as *corrective diskettes (CD)*.

Before transferring the maintenance on the CD-ROM, check to see if the software library is in a mode that will allow the transfer. Invoke ASM and select **Activate Maintenance**. If the words "in maintenance" are displayed beside the product that you want to transfer, maintenance has already been transferred and you must either activate or cancel to transfer the new maintenance. If the words "in backup" appear, you would normally accept the maintenance currently in test mode if it has been properly tested.

During the transfer of the maintenance on the CD-ROM, you might receive a message warning of possible insufficient disk space. ASM generates an estimate of the amount of disk space that is required for the transfer. This estimate is based on the sizes of the files currently active. Six percent is added to account for module growth and added modules. However, these factors can increase or decrease the actual amount of disk space that is needed differently than the six percent given in the estimate. Therefore, the estimate generated by ASM can be incorrect, and you can elect to continue despite the warning message.

2. Activate the maintenance

The maintenance transferred in Step 1 is not yet active. ASM refers to this state of transferred but not activated, as "in maintenance." To activate the maintenance, ASM needs to know if you want a backup copy of the present system saved in a backup subdirectory, in case, the maintenance must be removed. Keeping a backup copy is called *test mode*. This backup copy can

take up much disk space, and you can elect a different method of backing up the system. Test mode prevents further maintenance on that product until test mode has been exited.

- a. If you **test** the maintenance (apply the maintenance in test mode), ASM transfers the original, active modules that are being replaced from their program subdirectory to an appropriate backup subdirectory and moves the replacement modules in the maintenance subdirectory to the active program subdirectory. After the system IPLs itself, the system is running with the replacement modules in test mode.
- b. If you **cancel** the maintenance, ASM erases the new modules in the maintenance subdirectory that were transferred in step 1.
- c. If you **accept** the maintenance (apply the maintenance without test mode), ASM erases the original modules (now in the active program subdirectory) that are being replaced. After the system IPLs itself, both backup and the maintenance subdirectories are clear and ready for the next maintenance diskette or maintenance on the CD-ROM application. At this point, you have accepted the changes, and if they are unwanted, you must reinstall the original code.

A request that moves files to or from the active program subdirectory results in an IPL of the store controller in order to ensure that no files are open by an application. No files are moved in or out of the active program subdirectory except during the IPL. All verification has already been performed by ASM before the IPL so there should be no problems. However, if a problem is detected while moving the files, a brief message on the screen will indicate a failure, and an entry will be made in the System Event Log.

This IPL portion of ASM has automatic retry and recovery capabilities from power line disturbances. However, there can be problems that are encountered during the execution of ASM that the system cannot recover from automatically. Problems of this type include disk I/O errors due to hardware problems, distribution errors, or power line disturbances at critical times while activating operating system maintenance CD-ROMs. Any of these problems can result in incomplete maintenance to the operating system and possibly prevent a successful IPL of the store controller.

If, after you have applied maintenance to the operating system, the store controller does not complete IPL, use the Supplemental Option using the CD-ROM to try to restore the operating system. IPL with the Supplemental Option using the CD-ROM. If the failure was because a file could not be found, copy that file from the ADX_SBUL or ADX_SMNT subdirectory to the ADX_SPGM subdirectory. Then IPL your store controller again. Otherwise, if the problem is not due to a file not found, you should select ASM (Option 2, Activate Maintenance) and cancel maintenance for the operating system. If the ASM cancel completes successfully and the system successfully IPLs from the hard disk drive, the system has been restored to its previous operational level. If the system fails to IPL from the hard disk drive, you must reinstall the operating system and bring it up to the maintenance level prior to the one that caused the system failure.

3. Exit test mode.

If you put the maintenance in test mode in Step 2, you can accept or cancel these changes after the test is complete. ASM refers to this test mode as "in backup".

- a. If you **ACCEPT** the maintenance, ASM erases the original modules (now in the backup subdirectory) that have been replaced. Now both backup and the maintenance subdirectories are clear and ready for the next maintenance

diskette or maintenance on the CD-ROM. At this point, you have accepted the changes, and if they are unwanted, you must reinstall the original code.

- b. If you CANCEL the maintenance; ASM erases the new modules in the active program subdirectory, moves the original modules from the backup subdirectory to the active program subdirectory, and restores the system back to the level it was, before maintenance was transferred.

ASM verification of maintenance

ASM verifies the following:

- The new and current Product Identification (PID) numbers are the same.
- The new and current release numbers are the same.
- The checksum in the Product Control File (PCF) is the same as that is generated by ASM for each new module.
- If the checksum for the new module is different from the current module, the new module will be applied. Otherwise, ASM assumes that the module has not changed and will not apply the new module. The check can be bypassed using the force option when building the PCF. The Julian date will be ignored.

Note: If the module levels are equal but the checksums in the PCF are different, the module will be applied.

- The PCF is on the CD-ROM and has the same file name as the PCF of the same product in the ADX_nPGM subdirectory.
- There is no maintenance pending in the backup state.
- There is at least one module on the maintenance on the CD-ROM that has not been previously applied.
- If dealing with a LAN (MCF Network) system and running ASM on the master store controller, a check is made to verify that all store controllers on the LAN system are at the same level as the master store controller and are able to receive distributed maintenance.
- All modules specified in the PCF have been transferred from the maintenance the CD-ROM.
- All modules selected for add or replace exist in ADX_nMNT and are greater than zero bytes. If this check fails, the product is incomplete. The only valid action for an incomplete product, is a CANCEL. In some cases, ASM can determine that the transfer was done locally instead of through HCP. In this case, ASM automatically erases the incomplete product from the ADX_nMNT subdirectory.
- The following relationships must hold from maintenance on the CD-ROM and base level:
 1. The new base level must be greater than or equal to the current base level.
 2. If the new and current base levels are equal, the new maintenance on the CD-ROM must be greater than or equal to the current maintenance on the CD-ROM level.
 3. If the new maintenance on the CD-ROM is greater than the current maintenance on the CD-ROM, then one of the following conditions must exist:
 - a. The current base level must be 0000, or
 - b. The new base level must be 0000, or
 - c. The new base level must be equal to the new maintenance on the CD-ROM.

The following example describes a possible combination of base and maintenance levels that can be used to support a release of a licensed product. The restrictions

prevent incorrect or down-level maintenance from being applied. In the example, the product was installed as base level 9400. Two maintenance diskettes (9401 and 9402) were applied that were intended for base 9400. Then a new base level (9403) was applied that contained a complete distribution of licensed product code incorporating all maintenance from the previous base level. Next, a maintenance diskette or maintenance on the CD-ROM was applied that was intended for base level 9403. Because maintenance 9404 is intended for base level 9403, ASM prevents maintenance 9404 from applying to base level 9400.

Example:

Base	CD
9400	9400
9400	9401
9400	9402
9403	9403
9403	9404

ASM history and product-level files

ASM keeps a log of all actions that it attempts. Each ASM action attempted for any product creates a record in a history file. ASM also has a product-level file which contains information about the active level of a software product. These files are described in detail below.

ASM history file

ASM keeps a log of all actions that it attempts. Each ASM action attempted for any product creates a record in a history file.

The name of the history file is based on the name of its correlating Product Control File (PCF). The file is ADXLxHyF.DAT in the ADX_SDT1: subdirectory where x is an H for Toshiba products whose PCF file has an S as its 5th letter; otherwise, it is the 5th letter of the PCF file. The x can be 0 through 9 , and V through Z. The y can be 0 through 9, K through O, and V through Z. Characters A through J and P through U are reserved for Toshiba Product Control Files. There is one local history file for each product on each store controller.

Examples:

Product	Description	ASM History file
ADXCSTSD.DAT	4690 OS	ADXLHFSF.DAT
ADXCATGD.DAT	ACE	ADXLAFGF.DAT
ADXCITYD.DAT	Test Product YY	ADXLIFYF.DAT

Whenever ASM starts an action such as transfer or activate, ASM appends a record to the history file. However, when using the HCP to transfer maintenance from the host, an entry is not logged. Each record is 80 ASCII bytes plus a carriage return and line feed so that it can be easily printed. The history file contains the following data:

- Date and start time that the action was attempted.
- Product letter. This is the seventh letter of the PCF name.
- Controller node ID.

- Product level:
 - Release Level
 - Base Level
 - Maintenance on the CD-ROM
 - User Modification Level
 - PTF Number
 - Emergency Fix Number

An entry of 7 hyphens (-----) for the PTF number means that the file containing this information does not exist. An entry of 7 question marks (???????) means that the file containing this information exists but could not be read.
- Action that was attempted:
 - X** = Transfer
 - O** = Overlay (transfer with combine)
 - T** = Test
 - C** = Cancel
 - A** = Accept
- Mode before the action was attempted:
 - M** = In maintenance
 - B** = In backup
 - 2** = In maintenance and in backup
 - N** = Neither in maintenance or in backup
- Completion indication:
 - S** = Successful
 - F** = Failed
- Store number.
- Secondary product letter. This is the fifth letter of the PCF name. If the fifth letter of the PCF name is an S, this field will be blank.

Only the most recent 99 records are kept. Older records are erased. If these older records are important to you, you should periodically save the history file.

The product-level information comes from the maintenance diskette or from maintenance on the CD-ROM that is associated with the action.

- A TRANSFER or COMBINE action records the product-level information of the maintenance diskette or the maintenance on the CD-ROM in the drive.
- A CANCEL or ACCEPT action when the product is “in backup” records the product-level information of the maintenance diskette or the maintenance on the CD-ROM that had been activated.
- A TEST, ACCEPT, or CANCEL action when the product is “in maintenance” records the product-level information of the maintenance diskette or the maintenance on the CD-ROM that had been transferred.

Note: The product-level information is recorded before the files are moved.

ASM product-level file

The product-level file contains information about the active level of a software product. The name of the product level file is based on the name of its correlating Product Control File (PCF). The file is ADXLxLyF.DAT on the ADX_SDT1: subdirectory where x is a V for Toshiba products whose PCF file has an S as its 5th letter; otherwise, it is the 5th letter of the PCF file. The x can be 0 through 9, and V through Z. The y can be 0 through 9, K through O, and V through Z. Characters A through J and P through U are reserved for Toshiba Product Control Files. There is one local product-level file for each product on each store controller.

Examples:

Product	Description	ASM product-level file
ADXCSTSD.DAT	4690 OS	ADXLVLSF.DAT
ADXCATGD.DAT	ACE	ADXLALGF.DAT
ADXCMTND.DAT	Test Product MN	ADXLMLNF.DAT

The file can be retrieved by a host to determine the current level of the product. Actions that do not affect the ADX_?PGM level affect only the mode field of this file. This file always reflects the ADX_?PGM level.

Whenever ASM changes files in ADX_?PGM successfully, ASM reads the new level information from ADX_?PGM and writes it to the product-level file that overlays the record that was previously logged. The record is 80 ASCII bytes plus a carriage return and line feed so that it can be easily printed. The file contains the following data:

- Date and time of the last successful update to the ADX_?PGM level.
- Product letter. This is the seventh letter of the PCF name.
- Store controller node ID.
- Current product-level information:
 - Release Level
 - Base Level
 - Maintenance on the CD-ROM
 - User Modification Level
 - PTF Number
 - Emergency Fix Number

Note: An entry of 7 hyphens (-----) means the file containing this information does not exist. An entry of 7 question marks (???????) means the file containing this information exists but could not be read.

- Action that resulted in a change to the current product level:
 - T** = Test
 - C** = Cancel
 - A** = Accept
- Current mode:
 - M** = In maintenance
 - B** = In backup
 - 2** = In maintenance and in backup
 - N** = Neither in maintenance or in backup
- Indication of completion of the last successful update to the current level. This value is always “S” by definition. It is included so that the format will be consistent with the history file.
- Store number.
- Secondary product letter. This is the fifth letter of the PCF name. If the fifth letter of the PCF name is an S, this field is blank.

Format of history and product-level files

The records within the history and product-level files are 82 bytes long with blanks or periods separating each field as described in Table 45 on page 371.

Table 45. Format of history and product-level files

Offset	Description	Size	Data type	Format
0	Date	8 bytes	ASCII	mm/dd/yy
9	Time	5 bytes	ASCII	hh:mm
15	Product Letter	1 byte	ASCII	
17	Node ID	2 bytes	ASCII	
20	Release Level	3 bytes	ASCII	
24	Base Level	4 bytes	ASCII	
29	CD Level	4 bytes	ASCII	
34	User Mod Level	2 bytes	ASCII	
37	PTF Number	7 bytes	ASCII	URxxxxx
45	Emergency fix	7 bytes	ASCII	EFxxxxx
53	Action	1 byte	ASCII	X, O, T, C, or A
55	Mode	1 byte	ASCII	M, B, 2, or N
57	Completion	1 byte	ASCII	S or F
59	Store Number	4 bytes	ASCII	
64	Secondary Product Letter	1 byte	ASCII	
80	Carriage Return	1 byte	ASCII	X'OD'
81	Line Feed	1 byte	ASCII	X'OA'

LAN (MCF) system considerations

This section describes considerations for using the software maintenance utilities on a LAN (MCF Network) system.

Handling local files

Normally, all software maintenance is applied at the master store controller only. The maintenance files that are applied are distributed to the other store controllers on the network. The only exception to this is for local files. Because local files are not distributed on the network, they must be applied at the appropriate store controller.

Handling distribution exceptions

If a distribution exception occurs at any time while applying software maintenance, the exception is logged in the Distribution Exception Log. Normally, an exception record is logged for the particular file being updated, renamed, or deleted. However, because of the possibility of a large number of files that are being transferred by ASM, logging each file name that failed to be distributed can fill up the exception log. Instead, ASM logs a single exception record for the whole ADX_nMNT subdirectory and this subdirectory is copied to the store controller when it returns online. This is performed in cluster mode for better performance.

If the file delete is not successfully distributed while running ASM, only the specific file delete is logged.

Preparing for and applying maintenance

Before applying maintenance to the master store controller, clear the exception logs to remove possible conflicts to the maintenance process.

DDA automatically distributes the module replacements to all the store controllers on the LAN system. If an error occurs while transferring the module replacements, you should exit ASM, invoke Report Module Level on each store controller on the LAN system, and compare module levels to determine the cause of the error. When running on the master store controller, the Activate Maintenance commands TEST, CANCEL, and ACCEPT cause all store controllers on the LAN system to perform the same operation.

ASM performs several tests on all configured store controllers to ensure that all of the store controllers can properly receive the maintenance. If any of these tests fail, a panel appears with warning messages that describes the potential problem. ASM performs the following tests:

- *All store controllers have the same software level as the master store controller.* If they are not at the same software level, they can be made the same software level by applying the latest maintenance. Because ASM transfers only the files that are new for the master store controller, it can miss a file that is needed to bring a subordinate up to the latest maintenance level. Run Report Module Level after ASM completes and compare each module level to ensure that the subordinate is now at the same software level as the master store controller.
Another way that the store controllers could be at different software levels is if the store controllers are not in the same mode. One might be “in maintenance” or “in backup” while another is in a different mode. Try to bring the store controller to the same level before running ASM on the master store controller.
- *All store controllers are able to receive distributed maintenance.* ASM opens the PCF on each controller to verify the software level. If the PCF cannot be opened, either because it does not exist or because the store controller is down, ASM issues a warning. You can bypass the warning, but the store controllers end up at different software levels.
- *All store controllers have enough disk space to receive the maintenance.* ASM estimates how much disk space is required and compares it against how much is available on each store controller. This estimate might be inconsistent, and therefore, you can bypass a warning of insufficient disk space. It would be advantageous to erase unwanted files to make more disk space available before bypassing this warning.

Optional ASM indicators

The ASM utility (adxcst0l which is described in detail in the 4690 OS Communications Programming Reference) has three optional indicators: NI – the no IPL indicator, TL – the terminal load indicator and BY – the bypass LAN problems indicator. All of these indicators can be specified when using either method described in “Remote application of combined/compressed maintenance” below.

When the NI indicator is specified on the ASM command, ASM verifies checksums, distributes files, and creates the ASM activation file to get ready to activate the maintenance. The IPL of the store controller must be performed manually at a time that has the least impact on customer checkout.

This optional parameter provides greater control over the timing of the IPL of the store controllers to perform activation of maintenance. This option is useful only for stores that remain open for 24 hours a day because it requires that an extra step be performed manually at the store controller.

LAN (MCF Network) customers must ensure that the manual IPL is performed on all store controllers. Otherwise, the store controller that performed IPL will receive the

maintenance, but the other store controllers will continue running at the old level. A dump or an IPL of one store controller could cause this to happen, as well as store personnel failing to IPL all store controllers. For this reason, it is recommended that you do not operate in “ready for IPL” mode for a long period of time.

While in “ready for IPL” mode, the host must not send more maintenance to the ADX_?MNT: subdirectories because they are not checked by ASM before being activated.

To remove the store controller from “ready for IPL,” mode erase the ADX_SPGM:ADXCSTAF.DAT file.

Combining a delta with maintenance

ASM allows you to combine a delta diskette with maintenance. After you transfer the maintenance diskette or maintenance on the CD-ROM, transfer the delta. A warning screen appears, and ASM combines the delta and maintenance together. Modules on the maintenance will be replaced by the modules on the delta.

You do not need to combine two deltas together because the later delta includes all maintenance from the earlier delta. You do not need to combine maintenance with any previous level of code because the maintenance diskettes or maintenance on the CD-ROM includes all maintenance that has been changed since the base code.

Remote application of combined/compressed maintenance updates

Operating system software maintenance can be applied from a host site to controllers located at remote locations using a combine/compress utility. This utility decreases the amount of data that is sent to remote locations by compressing and combining the files of a maintenance update. This results in reduced cost and usage of your networks.

In order to perform a successful remote installation the following conditions must be met:

1. The same application prefix must be configured on both the controller that is located at your host site and the controllers that are located in your stores.
2. The systems at your host site and in your stores must be at 4690 OS Version 1 Base Level 9400 or later to use this procedure.

There are two methods for performing a remote installation:

1. The RMA and Director method which utilizes Remote Management Agent and Director to send RMA Software Distribution packages to the stores and execute actions specified in the packages (utilizing RCP and ASM for 4690 OS maintenance actions).
This method requires the use of RMA V2R4 and Director 5.20.2 update 2. For more information please refer to the appropriate RMA and Director publications.
2. The RCP method which utilizes the 4690 OS RCP command to perform maintenance actions on remote controllers. The choice of how the RCP commands and maintenance files are transferred to the remote controllers is left to the customer.

RMA and Director method

Step 1. Preparing your system

After you have signed on to your master host-site controller, you must end all store applications that are running, including windows and background applications.

To end window applications, perform the following steps:

1. Press **SysRq**.
2. Type **W**.
3. Press **F8** to end Windows Applications.
4. Type **Y** and press **Enter**.

Repeat steps 3 and 4 for each active window.

To end background applications, perform the following steps:

1. Press **SysRq**.
2. Type **B**.
3. Select an active background application.
4. Press **F8** to end background applications.
5. Type **Y** and press **Enter**.

Repeat steps 3 through 5 for each active background application.

After you have ended all background and window applications, press **F3** to return to the SYSTEM MAIN MENU.

Step 2. Transferring the maintenance update files

To transfer the Operating System Maintenance Package to your master host-site controller from the CD-ROM in the maintenance package, perform the following steps:

1. From the SYSTEM MAIN MENU, select **4** INSTALLATION AND UPDATE AIDS
2. From the INSTALLATION AND UPDATE AIDS panel, select **5** APPLY SOFTWARE MAINTENANCE
3. From the APPLY SOFTWARE MAINTENANCE panel, select **1** TRANSFER FILES
4. Follow the instructions as they appear on your screen.

Step 3. Combine/compress files using ASMBUNDL

Perform the following steps if you want to combine and compress the maintenance update files.

1. Ensure that you have at least 16 MB of free disk space before invoking ASMBUNDL. This is the minimum amount of disk space that is needed to allow ASMBUNDL to create the combined/compressed maintenance files.
2. Enter Command Mode and set the current directory to ADX_SMNT.
3. Type **ASMBUNDL**. ASMBUNDL is a BAT file that combines and compresses the files in ADX_SMNT into files with the name:
ADXHJxxF.DAT
Where xx is 01, 02, 03, and so on, depending on the amount of maintenance that was transferred to ADX_SMNT.
4. Record the number of ADXHJxxF.DAT files that were created.
5. Ensure that ASMBUNDL has completed without errors due to lack of disk space to create the combined/compressed files. If errors were encountered, delete any ADXHJ??F.DAT files that are created by ASMBUNDL and ensure that you have at least 16 MB of free disk space before invoking ASMBUNDL. You might have to delete some unnecessary files (for example, the controller dump file ADXCSLCF.DAT, which will be recreated during the next IPL).
6. When ASMBUNDL completes, transfer the following files from your master host site controller to your Director Server (using FTP, NFS and so on.):

Note: Pay particular attention to the source subdirectory from which the files will be transferred.

- ADXHJxxF.DAT from the ADX_SMNT subdirectory
- ADXHJLCL.286 from the ADX_SMNT subdirectory
- ADXNSXZL.286 from the ADX_SMNT subdirectory

Note: The xx in ADXHJxxF.DAT is a wildcard. Transfer all of the ADXHJxxF.DAT files.

Step 4. Transferring the maintenance to your stores

Use RMA and Director Console to create an RMA Software Distribution package for the 4690 OS maintenance, transfer the RMA Software Distribution package to your stores and execute the maintenance actions in the package. For more details on this step please refer to the appropriate RMA publications.

Note: It is required to have a RMA Master Agent running in each store for this step. The RMA Master Agent may run on a Windows or Linux machine or a 4690 OS enhanced controller.

Step 5. Completing the installation at the host site master controller

To complete the installation process at the host site master store controller, perform the following steps:

1. From the SYSTEM MAIN MENU, select the Command Mode option by typing **7** and pressing **Enter**.

Your screen will go blank for a moment, and then the system prompt appears.

2. When the prompt appears, enter:

```
ADX_SPGM:ADXCSH0L BACKGRND NNNN
```

Note: This step activates the same operating system upgrade that was applied to your remote store controllers. The activation of your operating system maintenance causes a re-IPL of the host site master store controller; if the store system at your host site is on a multiple controller system using the DDA, the subordinate store controllers at your host site will re-IPL as well.

3. Delete all ADXHJxxF.DAT files from the master host site controller.

RCP method

Step 1. Preparing your system

After you have signed on to your master host-site controller, you must end all store applications that are running, including windows and background applications.

To end window applications, perform the following steps:

1. Press **SysRq**.
2. Type **W**.
3. Press **F8** to end Windows Applications.
4. Type **Y** and press **Enter**.

Repeat steps 3 and 4 for each active window.

To end background applications, perform the following steps:

1. Press **SysRq**.
2. Type **B**.
3. Select an active background application.
4. Press **F8** to end background applications.
5. Type **Y** and press **Enter**.

Repeat steps 3 through 5 for each active background application.

After you have ended all background and window applications, press **F3** to return to the SYSTEM MAIN MENU.

Step 2. Transferring the maintenance update files

To transfer the Operating System Maintenance Package to your master host-site controller from the maintenance diskettes or the CD-ROM in the maintenance package, perform the following steps:

1. From the SYSTEM MAIN MENU, select **4** INSTALLATION AND UPDATE AIDS
2. From the INSTALLATION AND UPDATE AIDS panel, select **5** APPLY SOFTWARE MAINTENANCE
3. From the APPLY SOFTWARE MAINTENANCE panel, select **1** TRANSFER FILES
4. Follow the instructions as they appear on your screen.

Step 3. Combine/compress files using ASMBUNDL

Perform the following steps if you want to combine and compress the maintenance update files.

1. Ensure that you have at least 16 MB of free disk space before invoking ASMBUNDL. This is the minimum amount of disk space that is needed to allow ASMBUNDL to create the combined/compressed maintenance files.
2. Enter Command Mode and set the current directory to ADX_SMNT.
3. Type **ASMBUNDL**. ASMBUNDL is a BAT file that combines and compresses the files in ADX_SMNT into files with the name:

ADXHJxxF.DAT

Where xx is 01, 02, 03, and so on, depending on the amount of maintenance that was transferred to ADX_SMNT.

4. Record the number of ADXHJxxF.DAT files that were created.
5. Ensure that ASMBUNDL has completed without errors due to lack of disk space to create the combined/compressed files. If errors were encountered, delete any ADXHJ??F.DAT files that are created by ASMBUNDL and ensure that you have at least 10 MB of free disk space before invoking ASMBUNDL. You might have to delete some unnecessary files (for example, the controller dump file ADXCSLCF.DAT, which will be recreated during the next IPL).
6. When ASMBUNDL completes, transfer the following files on your master host site store controller to your host processor:

Note: Pay particular attention to the source subdirectory from which the files will be transferred.

- ADXCSCF.DAT from the ADX_IDT1 subdirectory
- xxxRCP1F.DAT (the RCP command file) from the ADX_IDT1 subdirectory (where xxx is the application prefix that is defined for the system in the system configuration)
- ADXHJxxF.DAT from the ADX_SMNT subdirectory
- ADXHJLCL.286 from the ADX_SMNT subdirectory
- ADXNSXZL.286 from the ADX_SMNT subdirectory

Note: The xx in ADXHJxxF.DAT is a wildcard. All the ADXHJxxF.DAT files should be transferred to your host processor.

7. Edit the xxxRCPIF.DAT file to set up the commands in this file to control the way you want to install the maintenance update.

xxxRCP1F.DAT is the command file that should contain the commands:

```
ADX_SMNT:ADXHJLCL N
ADXCSTOL N 1S
```

The first command decompresses the ADXHJ??F.DAT files in the ADX_SMNT subdirectory. The second command applies the operating system maintenance in TEST mode. If you want to apply the operating system maintenance in ADX_SMNT subdirectory in ACCEPT mode, change the second command to:

```
ADXCSTOL N 3S
```

The second command executes the ASM program. You can start the ASM program with a separate ASM command by deleting the second command in the xxxRCP1F.DAT file and building a separate RCP command file. Refer to the *4690 OS: Communications Programming Reference* for additional information about using RCP.

Step 4. Transferring files to your store controllers

1. Transfer the following files from your host processor to your remote store controllers:

Note: Ensure that the files are transferred to the correct subdirectories that are shown in the following list:

- ADXCSHCF.DAT to the ADX_IDT1 subdirectory
- xxxRCP1F.DAT (the RCP command file) to the ADX_IDT1 subdirectory (where xxx is the application prefix that is defined for the system in the system configuration)
- ADXHJ??F.DAT to the ADX_SMNT subdirectory
- ADXHJLCL.286 to the ADX_SMNT subdirectory
- ADXNSXZL.286 to the ADX_SMNT subdirectory

Note: The ? in ADXHJ??F.DAT is a wildcard

2. Invoke the Remote Command Processor (RCP) using the START USER PROGRAM feature of ADCS or INITIATE CLIST command of NETVIEW DM. Refer to the *4690 OS: Communications Programming Reference* for more information on running RCP commands.

The RCP command decompresses the files in the ADX_SMNT subdirectory, and then activates the operating system maintenance simultaneously on your remote store controllers.

You should plan to activate at a time when your store is not active, because the maintenance procedure will require a re-IPL of the controller and terminals. It is recommended that no other windows are open because that might create a file usage conflict.

RCP runs as a background process and then other background processes are started. Ensure at least four background slots are available before starting any RCP commands.

Attention: Your remote store controllers will IPL.

3. After the maintenance update has been activated, retrieve the ADX_SDT1: ADXCSHSF.DAT file from the remote store controllers to determine the success or failure of the install, and to see if any error messages associated with the RCP command were logged. This file is the status file for RCP commands that contains processing type messages as well as error messages.

Step 5. Completing the installation at the host site master controller

To complete the installation process at the host site master store controller, perform the following steps:

1. From the SYSTEM MAIN MENU, select the Command Mode option by typing **7** and pressing **Enter**.

Your screen will go blank for a moment, and then the system prompt appears.

2. When the prompt appears, enter:

```
ADX_SPGM:ADXCSH0L BACKGRND NNNN
```

Note: This step activates the same operating system upgrade that was applied to your remote store controllers. The activation of your operating system maintenance causes a re-IPL of the host site master store controller; if the store system at your host site is on a multiple controller system using the DDA, the subordinate store controllers at your host site will re-IPL as well.

3. Delete all ADXHJxxF.DAT files from the master host site controller.

Build software maintenance control file utility

The Build Software Maintenance Control File Utility is used to maintain files that identify all modules that are contained in a Toshiba or user-product diskette or CD-ROM. These files are called *Product Control Files* (PCFs). Product Control Files contain information that is required for application of maintenance through the ASM Utility and for obtaining product and module level detail through the Report Module Level Utility. Therefore, you need to use the Build Software Maintenance Control File Utility to create a diskette to supply maintenance (through ASM) to your own applications or 4680 or 4690 application program products. With the PCF, you can support your applications similar to the way 4680 or 4690 application-licensed products are supported.

You can use the utility by selecting the **Installation and Update Aids** option on the SYSTEM MAIN MENU. Then select the **Build Software Maintenance Control File** option from the INSTALLATION AND UPDATE AIDS panel. During a Build Software Maintenance Control File session, the utility presents a series of panels that prompt you for all required PCF information. The system also prompts you for data that is required for each module to be included in the maintenance activity. At the end of your input session, you can direct the PCF to a hard disk drive or diskette.

While you are building the Software Maintenance Control File, you can either enter data for each module individually, or let the utility automatically add or update module records for all of the files in a subdirectory or on a diskette.

When you choose the automatic option, you are asked to specify the path name of a subdirectory or diskette and to select the distribution attributes to be assigned to all modules in that subdirectory. You can modify those attributes later by editing the individual module records in the Software Maintenance Control File. If you group your modules into several subdirectories or diskettes according to their distribution attributes, you can set the correct attributes for each group as you add them to the PCF.

When you let the utility automatically process all modules in a subdirectory, it calculates a checksum for each module. ASM uses this checksum to verify that the transfer of maintenance is correct. When a module that is already listed in the Software Maintenance Control File is modified, a new checksum must be calculated for it. You should update its module record either with the automatic option, or by erasing the module record and appending a new one. Either of these methods will generate a checksum acceptable to ASM. Editing an existing module record without first deleting it will not update the checksum.

Each module in the PCF requires an action to be performed, such as a add, replace, or delete (see below). The following section shows the logic used to demonstrate how the three actions will be handled for each module:

For each record in an ADX_?MNT PCF

- Search the ADX_?PGM PCF for a matching record
- Handle ACTION (ADD: 'A', DELETE: 'D', REPLACE: 'R')
- IF ACTION IS 'R'
 - replace_case:
 - IF record's module name IS NOT in ADX_?MNT subdirectory

```

- Skip this module
-IF record IS NOT found in ADX_?PGM PCF
- Record will be treated as ADD
- goto add_case
-IF ACTION IS 'A'
add_case:
-IF record's module name IS NOT in ADX_?MNT subdirectory
- Skip this module
-IF record IS found in ADX_?PGM PCF
- Record will be treated as REPLACE
- goto replace_case
-IF ACTION IS 'D'
-IF record IS NOT found in ADX_?PGM PCF
- IF module IS already deleted from ADX_?PGM subdirectory
- Skip this module

```

Each product maintained by ASM in a software library must include a PCF that defines the product. To use a product in a library, the operating system must locate the PCF for that product. PCFs are named ADXCfTpD.DAT, where *f* is usually an *S* for Toshiba products. The *f* can be 0 through 9, K through O, and V through Z. The *p* can be 0 through 9, K through O, and V through Z. ***Characters A through J and P through U are reserved for Product Control Files.*** If multiple products are installed, they each must have unique PCF names. ***Characters V, H, and M are not allowed as the fifth letter of the Product Control File.***

A PCF contains product-identifying information such as the common name of the product, the release and maintenance level of the product, the date of the last change to the product, the product number (identifier), the user modification level, and the maintenance level number.

It also contains descriptions of all permanent files of the product for which maintenance is wanted.

The module-identifying information includes:

- File name
- Change level of the module
- Type of module (system, sales application, or user module)
- Module Integrity Value (checksum)
- Distribution attributes
- Action to be performed (add, replace, or delete)

ASM provides for management of cumulative maintenance levels. Cumulative maintenance levels allow you to apply the latest level of code to any lower level as long as the base level is the same and therefore minimize the amount of data that is transferred. All modules that were changed since the base level was changed are on the maintenance diskettes or on maintenance on the CD-ROM. From the module level (usually a Julian date) and from the checksum that is listed in the PCF, ASM knows whether the module at that level (or later) is already applied. ASM will not reapply a module that has already been applied, or one that is at a later level than the version that is about to be applied.

The module level is ignored except for the following cases:

- If the first letter of the module level is an F, the module is in “fix-test.”
- If the first letter of the module level is a T, the module is a “trap-patch.”

These two cases do not prevent the module from applying, but a warning will be issued when the modules are replaced.

To review the product and file information for each of your products, select the **Report Module Level Utility** option on the INSTALLATION AND UPDATE AIDS panel.

Each product maintained by ASM must have a PCF in the appropriate ADX_*n*PGM subdirectory, that describes the product and the modules that are supported by the product. Each set of diskettes must have a Product Control File on the first diskette describing which product is being corrected, and that lists each module that is to be applied.

Building a software library

The Store System uses three software libraries to separate, control, and protect the products on your system.

Each library is defined by three related subdirectories in the root on the store controller's hard disk drive. These subdirectories are an active program subdirectory, a maintenance subdirectory, and a backup subdirectory.

ASM uses the following naming convention for the subdirectories of each library:

ADX_*n*PGM The active program subdirectory
ADX_*n*MNT The maintenance subdirectory
ADX_*n*BUL The backup subdirectory

The character *n* can be 0 through 9, A through H, or V through Z for your libraries. The characters I through U are reserved for Toshiba-defined libraries. Each subdirectory within a library must reside on the same drive.

The names for the subdirectories of the three libraries that are created, defined in the System Logical Names file, and supported by the operating system are:

Library	Subdirectories
Operating System	ADX_SPGM, ADX_SMNT, and ADX_SBUL
Applications	ADX_IPGM, ADX_IMNT, and ADX_IBUL
User Exits	ADX_UPGM, ADX_UMNT, and ADX_UBUL

To add a library of your own to the system, use the MKDIR command to create a set of three new subdirectories on the root directory of the hard disk drive. Follow the naming conventions for library subdirectories defined above. You must include the names of each subdirectory in the User Logical Names file because ASM uses these names to access the subdirectories.

Installing a user's product

To install a user's product:

1. Using the Build Software Maintenance Control File Utility create a test PCF that contains just the product record (no module records) and write it to the appropriate ADX_*n*PGM subdirectory, where *n* is I, S, or U. This file can also be installed through HCP. HCP does not normally allow creates and writes to any ADX_*n*PGM subdirectories, but this is an exception.
2. Build a maintenance diskette that contains the modules of the new product.
3. Proceed normally to transfer the maintenance, testing and accepting it as necessary.

Applying and maintaining user modifications

The Build Software Maintenance Control File Utility is important for modifying program code. You should create a PCF that includes a record for each module to be modified and place it on your maintenance diskette.

Use the following procedure to build the diskettes for ASM to apply modifications to 4680 or 4690 application program products:

1. If you are applying Toshiba maintenance at the same time that you are applying user modifications to the 4680 or 4690 application licensed product, transfer the maintenance that is supplied by Toshiba for the 4680 or 4690 application program product. This action has no effect on the operational environment. **Do not activate the changes at this time.** Activating at this time changes your operational environment before the appropriate user changes have been made.
2. Copy object or source modules that are needed for the user modifications to the ADX_UPGM subdirectory. You must do this step so that the new load modules can be built.
3. Compile (if necessary) and link the changed source code to create a new load module.
4. Using the Build Software Maintenance Control File Utility, create and place on diskette a PCF named ADXCSTGD.DAT.

If you are applying Toshiba maintenance at the same time that you are applying user modifications to Toshiba application code, read the PCF from the ADX_IMNT path. Otherwise, read the PCF from the ADX_IPGM path. You will probably want to erase all module records, then append one module record for each module to be modified. When appending this record, ensure that the module name entered is identical to the one being replaced. The current date is the default module level.

5. Copy the newly created modules to diskette with Command Mode and the COPY command.
6. Maintenance Diskettes have sequence numbers that force the diskettes to be inserted in order starting with diskette #1, then requesting diskette #2, and so on until all files listed in the maintenance PCF have been copied. ASM uses the volume label to determine the diskette sequence number.

You can optionally use diskette sequencing by putting the appropriate volume labels on your maintenance diskettes by using the DISKSET command in Command Mode. The volume label of the first diskette is of the form ????????#01, where ? is anything that a file name can be and the last two characters are the sequence number. If the volume label of the first diskette is not of this form, ASM allows you to insert the diskettes in any sequence and keeps prompting you for the next diskette until all files listed in the Maintenance Diskette PCF have been copied.

7. Invoke ASM and transfer the maintenance. If you are applying Toshiba maintenance at the same time, the Maintenance Diskettes copied in Step 1 should still exist "in maintenance" because no TEST step should have been done. Step 7 replaces the modules with your changed ones.
8. Run Report Module Level to verify that pending maintenance is correct. Blanks in the type column indicate that the module has been replaced by a user module. All other fields remain unchanged.
9. Perform a TEST operation from the ACTIVATE MAINTENANCE FILES panel of ASM. After the store controller performs IPL, the current subdirectory contains the new user environment while the backup subdirectory contains the old. The

TEST operation allows you to test the changes. Upon completion of test, you can return to the prior environment by performing a CANCEL if there is a problem in the maintenance.

10. If there is no problem with the maintenance, perform an ACCEPT operation to make the changes permanent and clear the backup subdirectory of modules for the next change.

Note: If you are transferring maintenance from the host processor to the store controller, you must first follow the above procedures on a store controller at the host processor site before you transmit the maintenance to a remote store controller. The maintenance transmitted must be the final copy in the maintenance subdirectory as it appears after Step 7.

Your modifications to the 4680 or 4690 application licensed products will be erased when maintenance to the product affects those files modified. You must make the modifications again to the maintenance level of the modules.

For example, you apply maintenance level 9305 and rebuild one of the load modules at the 9305 level with user modifications. These user modifications are replaced by the version when maintenance level 9306 is applied. If you apply the user modifications using a maintenance diskette or maintenance on the CD-ROM as designed, you receive a warning that your user modifications will be replaced.

To use maintenance level 9306 and keep the user modifications, rebuild the module at the 9306 level. Combine this user modification with the maintenance level 9306 by using a maintenance diskette that was built by the Build Software Maintenance Control File Utility before activating maintenance level 9306.

User checksums

A Module Integrity Value (checksum) is calculated for each module that is shipped from Toshiba, and the values are put in the PCF. The checksum is re-calculated when the maintenance is activated and compared against the value in the PCF. If the comparison fails, the maintenance is not applied to any of the modules on that maintenance level and returns the error:

Module Integrity (Checksum) error on <fn>.

where <fn> is the module name.

Modules that are provided by the user and applied using a PCF (built by the Build Software Maintenance Control File Utility) do not have a checksum until they are transferred. The checksums are written to the user maintenance diskette the first time you transfer the maintenance. If the user maintenance spans more than one diskette, you are prompted to insert the first diskette again. If the diskette has a write-protect tab, you are prompted to remove it. If the same maintenance is transferred to another store controller, or the maintenance is uploaded to a host processor and downloaded to remote stores, checksums ensure that the modules on the remote store controllers are the same as those transferred to the first store controller.

Transferring modules to other subdirectories

Normally, ASM moves maintenance modules to the program subdirectory where the active PCF resides. For example, if you transfer a module for a 4680 or 4690 application program product, it is transferred to ADX_IMNT. When activated, it is moved to ADX_IPGM. You might want to have a module moved to a subdirectory

other than ADX_*n*PGM. For example, you might want to store a data file that your application needs in the ADX_IDT1 subdirectory. In this case, you can create a file for handling these cases.

To build the special cases file, create a file that is named ADXC*f**p*U.DAT, where *f* is the fifth letter and *p* is the seventh letter of the PCF used to apply your user files. Edit the file (which is a list of records). Each record must contain the following:

- A beginning semicolon
- A logical subdirectory name (must be eight characters)
- A colon
- A file name and extension
- An ending semicolon

Do not put spaces between the information in a record and the information between records. For example, carriage return and line feed characters are ignored. An example of a special cases file follows:

```
;ADX_IDT1:EAXFILE1.DAT;  
;ADX_IDT3:EAXFILE2.DAT;
```

Special case files are similar. They are named ADXC*f**pl*.DAT. Special case files are kept in a separate file so that user-special cases are not replaced when there is an update to the special cases.

Make an entry for the special cases file and all files that are listed in the special cases file in your PCF when you use the Build Software Control File Utility. All of these files are to be included on the maintenance diskette that you build. The entry for the special cases file in the PCF must indicate that the special cases file is to be compound and distributed on close.

Chapter 17. Using the streaming tape drive utility

This chapter explains the Streaming Tape Drive Utility, which enables you to transfer data between the store controller's hard disk drives and the Streaming Tape Drive's tape cartridges. This chapter contains information on selecting a tape drive and using tape utility functions.

Understanding the streaming tape drive utility

The operating system provides the Streaming Tape Drive Utility to enable you to transfer data between the store controller's hard disk drives and the Streaming Tape Drive's tape cartridges. In addition, special tape cartridge handling functions are provided. You can start the utility from the FILE UTILITIES panel, from the host processor, or from a 4680 BASIC application by executing in the store controller.

Note: This utility is only supported in 4690 OS V6 Classic Mode.

Before using the Streaming Tape Drive Utility, you must first follow the procedure to select which tape drive device driver the operating system will load at IPL. See the *4690 OS: Planning, Installation, and Configuration Guide* for information on selecting a tape drive device driver.

When you start the Streaming Tape Drive Utility from the FILE UTILITIES panel, you are using it in an operator-interactive mode. Starting the utility in this mode enables you to use multiple tapes. The tape utility executes as a foreground application. In this type of operation, the operator supplies selections through menus and views response messages from the tape utility on the screen.

When you start the utility from either the host processor or from a 4680 BASIC application, the utility executes as a background application. In these last two cases, input to the tape utility is by commands that are stored in files on the hard disk drive. Responses from the tape utility are also stored in files on the hard disk drive. When you start the utility from the host processor or an application, only one tape is supported.

The Streaming Tape Drive Utility is provided on the hard disk drive resident system on the Supplemental Option using the CD-ROM. Depending on the types of streaming tape operations that are wanted, you might need to use both forms of the tape utility. The general rule of which system to operate from mainly depends on the amount of data you want to process. If you want to perform backups or restores of an entire hard disk drive, you must use the tape utility on the Supplemental Option using the CD-ROM. If you want to process single files or groups of files, you can use the hard disk drive system or the Supplemental Option using the CD-ROM. If you want to initiate tape operations from the host or from an application program, you must use the tape utility on the hard disk drive system.

The Streaming Tape Drive Utility provides flexibility in the choice of how much data to operate on. The choice can range from an individual file to all files on a hard disk drive or tape cartridge. Selections that involve multiple files are processed as a series of individual file operations.

Accessing the streaming tape drive utility

The Streaming Tape Drive Utility is accessed using menus for the hard disk system or the Supplemental Option using the CD-ROM. From the SYSTEM MAIN MENU, select the **File Utilities** option. When the FILE UTILITIES panel appears, select the **Streaming Tape Drive Utilities** option. When you select it, the STREAMING TAPE DRIVE UTILITIES selection panel appears.

This panel, and the ones that follow it, guide you in performing various operations with the Streaming Tape Drive. Operations include back up files to tape, restore files from tape, list tape contents, erase tape, and adjust tape tension.

On completion of a requested tape operation, a status and message panel appears. Messages indicate the number of successful file operations that is performed and individual file status messages for file operations that could not be successfully performed.

Using the streaming tape drive utility from the host

The Streaming Tape Drive Utility can be invoked from the host by using the Remote Command Processor (RCP). An appropriate RCP selection file and RCP command file must be present on the hard disk drive. Specific commands must be present in the RCP command file to initiate the appropriate tape utility functions. These commands include backup, restore, list, erase, and adjust tape tension. After the required RCP selection and RCP command files are on the hard disk drive, follow standard procedures for invoking RCP from the host. Starting the utility from the host processor restricts you to the use of only one tape. See the *4690 OS: Communications Programming Reference* for a description of the procedures and associated host processing.

Using the streaming tape drive utility from an application program

The Streaming Tape Drive Utility can be invoked from a 4680 BASIC application program through use of the ADXSTART application service facility. The application that must be started is the Remote Command Processor (RCP), named ADXCSH0L. RCP in turn initiates the Streaming Tape Drive Utility. An appropriate RCP selection file and RCP command file must be present on the hard disk drive. Specific commands must be present in the RCP command file to initiate the appropriate tape utility functions. These commands include backup, restore, list, erase, and adjust tape tension. After verifying that the required RCP selection and RCP command files are on the hard disk drive, the BASIC application should issue the ADXSTART call. Starting the utility from an application restricts you to using only one tape.

See the *4690 OS: Communications Programming Reference* for information on using ADXSTART to initiate a background application and for information on the details of the tape utility RCP commands.

Status file definition

When the Streaming Tape Drive Utility is invoked from the host or from a BASIC application, a status file is accumulated. If the tape operation is a backup or restore, this status file contains a status of the results from the operation. If the tape operation is list tape contents, this status file contains the list of the tape contents. It is intended to be analyzed by appropriate application programs at the store controller or the host to determine the success or failure of the requested tape utility

operations. The status file is named ADX_SDT1:ADXCSVDF.DAT. The format of the status entries is fixed length, 82 bytes long, with the last two bytes being carriage return/line feed characters.

File data accessing considerations

The Streaming Tape Drive Utility accesses data on the hard disk drives in the store controller by using the file services that are provided by the operating system. The utility protects against writing dynamically changing disk data to a tape, or writing tape data to a disk file that is being used by some other application through checks that are performed by the file system. If the utility detects such a data accessing conflict for a file, the operation for that file is bypassed and an error message appears to the operator or an entry is inserted in a status file.

When the store controller is operating from the system that is loaded from the hard disk drive, the operating system and installed applications have numerous files in use. An attempt to process all files by the tape utility in this environment would lead to numerous files being bypassed because of file accessing conflicts. The result would be considerably less than an all files tape utility operation.

When the store controller is operating from the Supplemental Option using the CD-ROM, there are no hard disk drive files in use. Therefore, the tape utility will not encounter file data accessing conflicts. This ensures successful handling of any number of files, from one to all, by the tape utility.

The tape utility is intended to be used for individual files or groups of files when executing from the hard disk drive system. To ensure a successful operation, the tape utility should be executed only when the intended files are closed or when they are no longer in use by your application.

Tape cartridge identification and contents

Each tape cartridge, when used by the Streaming Tape Drive Utility, contains a tape identification record that specifies a user name for the tape, the date and time the tape was labeled, and an indication if this tape is one of a set of tapes used to create a multiple-tape backup. Whenever a tape cartridge is inserted in the tape drive, a search is made for the tape identification record, and if found, it appears on the screen.

When backups of files to tape are performed, a variable number of files can be processed during each backup command. Each group of files that are processed by a backup operation is called a *volume*. A volume record is written to tape before the first file of the volume. Volumes are identified by a sequential volume number that is automatically incremented by the Streaming Tape Drive Utility. This concept of multiple volumes on a tape allows backing up the same named file at different times to the same tape.

Each data file on a tape is preceded by a file header record that contains:

- File name and disk directory path
- Disk file creation date and time
- Tape backup date and time
- File size
- File attributes

This header record provides the necessary information for future attempts to locate the file on tape and the corresponding disk file information if a restore of this file to disk is necessary.

Using tape utility functions

The Streaming Tape Drive Utility supports these distinct tape functions or commands:

- Back up Files to Tape
- Restore Files from Tape
- List Tape Contents
- Erase Tape
- Adjust Tape Tension

The following sections describe each of these functions.

Backing up files to tape

This function lets you copy files from a hard disk drive to the tape. Options enable you to back up:

- All files on a drive
- All files in a subdirectory (empty or null subdirectories are not backed up)
- All files that are named in a list (see “Creating a selection list file” on page 389)
- One file

If the tape cartridge in the tape drive has been previously used for backup, an option enables you to erase the tape before performing the backup, overwrite from the beginning of tape, or add the backup files as a new volume beyond the last recorded volume on the tape.

You should erase a new tape before use and adjust the tension of a used tape before use.

Restoring files from tape

This function lets you copy files from a tape cartridge to a hard disk drive. Options enable you to restore:

- All files on a tape volume
- All files in a subdirectory (the subdirectory is created if it does not exist)
- All files that are named in a list (see “Creating a selection list file” on page 389)
- One file

An option also lets you restore selected files from a particular volume or from all volumes on the tape. If a given file is present in multiple volumes being restored, the copy of the file in the last volume that is selected is the resulting file when the restore is complete.

Listing tape contents

This function lets you list the contents of a tape cartridge. The options enable you to list:

- All files on a tape volume
- All files in a subdirectory
- All files that are named in a list (see “Creating a selection list file” on page 389)
- One file

An option also lets you list selected files from a particular volume or from all volumes on the tape.

The list function displays or writes to the status file the file header information for each file that is found on the tape that matches the selection criteria.

The selection options for the List function are similar to those for the Restore function. Therefore, the set of files that is listed would be identical to the set of files that are restored if the same options are chosen.

Erasing tape

This function lets you erase all data from a tape cartridge. Erase causes the tape to be moved to the end of tape, erasing all data. The tape is then rewound to the beginning of the tape. On some tape drive models, a verification of all tape blocks is also performed, which marks the bad blocks as unused.

Note: You should erase a new tape before use.

Adjusting tape tension

This function is similar to erase except that the data is preserved on the tape. When Adjust Tape Tension is selected, the tape is wound to the end and rewound to the beginning. The purpose of this operation is to even out the tension in a tape cartridge. You should perform this function on new tapes before you use them for backups and on any tapes that have been transported or stored for significant periods of time or affected by changes in temperature since last use. Failure to adjust tape tension can cause read and write problems.

Creating a selection list file

One method for specifying the set of files to back up, restore, or list is to name a file that contains the appropriate set of files. The selection list must be created by using the redirected output from the Command Mode DIR command. DIR permits use of global file name characters for grouping of names and concatenating of DIR lists into the same file (for example, DIR C:\ADX_IDT1\ea1*.dat > C:\ADX_IDT1\tape.lst).

LAN (MCF Network) system considerations

The Streaming Tape Drive Utility can be used at any store controller that is equipped with a streaming tape adapter card and attached Streaming Tape Drive. The tape utility accesses hard disk drive data by using the operating system's file and LAN (MCF Network) services functions. When operating from the Supplemental Option using the CD-ROM, the LAN facilities are disabled. In this case, the tape utility functions as if the store controller, on which the tape utility is executing, is not connected to any other controller.

When operating from the hard disk drive system, LAN facilities are available. The LAN facilities let the tape utility access files that are associated with the local store controller node or with other nodes on the LAN. By prefixing the node name to the normally specified path, the Streaming Tape Drive Utility can access files on any store controller on the LAN.

The Streaming Tape Drive Utility preserves the distribution attributes of all files it processes. In addition, the tape utility restores files to the proper file owner for all

distributed files. The standard LAN file distribution functions are then used to propagate the restored file to the appropriate LAN nodes.

Using the streaming tape drive

Streaming tape can provide a means leading to improved methods for system backup and system replication. Streaming tape offers better performance backup and restore functions compared to the performance of diskettes. It also allows efficient data exchange capability for large data files, compared to either host communications or diskette operations. Streaming tapes can also be used for fast reorganization of hard disk drive files that are burdened by fragmentation.

A fundamental distinction must be made before understanding how to most effectively use streaming tape in the system. That distinction relates to the volume of data that is being processed. Streaming tape can back up and restore files in these general categories of use:

- Operations that involve the entire contents of a hard disk drive
- Operations that involve individual files or groups of files

Backing up entire hard disk drives

The first category of use, entire hard disk drives, generally demands use of the Streaming Tape Drive Utility on the Supplemental Option using the CD-ROM. This is the only way to back up all the files from the hard disk drive, avoiding “file in use” access conflicts. Similar access conflicts associated with the restore of an entire hard disk drive can be avoided by using the Supplemental Option using the CD-ROM.

System replication and hard-disk drive error recovery are examples of this category of streaming tape use. The general procedure to follow is:

1. Create a backup tape or tapes by executing the Streaming Tape Drive Utility from the Supplemental Option using the CD-ROM. Select the option to back up all files to the tape. Depending on the relative size of the hard disk drives and the tape cartridges, multiple tape cartridges might need to be inserted in this process.
2. Store the tapes for future system restore purposes or distribute the tapes for system replication.
3. When the need to restore from tape arises, follow these steps:
 - a. IPL the system from the Supplemental Option using the CD-ROM.
 - b. Select the **CPREP** or **DPREP** procedure to initialize and set up the hard disk drive.
 - c. Select to restore all files from tape by using the Streaming Tape Drive Utility. If multiple tapes are required, appropriate prompts guide you in the tape handling.
 - d. Remove the Supplemental CD-ROM and IPL the system from the hard disk drive.

Backing up selected files

This category of streaming tape use can be performed using the hard disk drive system or the Supplemental Option using the CD-ROM. It should be noted that all the same “file in use” access conflicts can occur when dealing with an individual file or group of files, but that in this category of tape use you have a much better opportunity to control the application and system use of files.

Examples of reasons you might want to back up and restore individual files include protecting a particular file against hardware failure, archiving on a daily or weekly basis of particular files, retaining a copy of files before additional handling or processing such as host transmission, or using tape to distribute very large files.

The following are some typical ways you might want to use the Streaming Tape Drive Utility:

- If the store environment is one where you are involved in executing various procedures daily at closing, it is likely that the interface of using menus is appropriate for using the tape utility. You select the tape utility, supply the file name or name of the file containing the list of files to process, load the proper tape into the tape drive, and back up certain daily files.
- If the closing procedures are more or less automated, the invocation of, and input to, the tape utility can be supplied by a BASIC application program run as part of the store closing procedures. You only need to ensure that the proper tape cartridge is properly loaded in the Streaming Tape Drive.
- If host transmissions and interaction are part of daily store closing activities, the host can initiate the correct tape utility operations at the appropriate times, similar to the case for application initiation. Again, you only need to ensure that the proper tape cartridge is loaded.

Backing up files that applications are updating

In some cases, you need to back up files that applications are updating. For example, accounting totals are kept in several files on the store controller. As each transaction occurs, the files tracking accounting totals are updated. Transactions occurring during the time of backing up these files with the tape streamer might be reflected in some of the files that are backed up, but not all. The sequence of events is:

At 11:30 a.m., a backup of the C: drive begins.
At 11:32 a.m., the transaction log and tender file are backed up.
At 11:34 a.m., a transaction occurs at a register. The transaction
log and all accounting totals files are updated.
At 11:35 a.m., the remaining accounting totals files are backed up.

Notice that the transaction log and the tender file on the tape do not reflect the transaction that occurred at 11:34 a.m., but the remaining accounting totals files do reflect the transaction. If this tape is used to restore the backed up files, the accounting totals would be considered incorrect or confusing.

This complication is not isolated to accounting totals. A similar example can be given for the print spooler. Any update that affects more than one file, or even different records within one large file, is exposed to this problem.

You can reduce the chances of this occurring by excluding files that are being updated (open for unlocked write access). This selection can be made on the panel when the backup operation is selected or through an optional parameter when using RCP. If you can include these files, you should stop all activity that would cause updates to these files for the duration of the backup. The best way to stop all activity is to use the Supplemental Option using the CD-ROM for the backup. See the *4690 OS: Communications Programming Reference* for more information on backing up files that are being updated by applications.

If a file is locked (opened for exclusive access), it will still yield a "file in use" error. If a file is not locked, but a record is locked during the backup, the utility tries to read the record every second until the record becomes unlocked. If any other read

error occurs, the return code will be reported, and the remaining portion of the tape version of the file will be filled with Ps. The disk version of the file will not be filled with Ps.

The default selection is to exclude files that are being updated. To include these files, use the RCP and specify the optional parameter **IW** in the position where the label usually starts.

Chapter 18. Using the optical drive utility

This chapter describes the Optical Drive Utility provided by the operating system. It contains information about accessing the Optical Drive Utility and the Optical Drive Utility functions.

Introduction to the optical drive utility

The operating system provides the Optical Drive utility to enable you to transfer data between the store controller's hard disk drives and the 3½-inch Rewritable Optical Diskette Drive's disks. In addition, the utility provides special disk-handling functions. You can start the utility from the File Utilities panel, the host processor, a 4680 BASIC application that is executing in the store controller, or CD/DVD write-capable media.

Optical diskette drives are not supported in Enhanced Mode.

When you start the Optical Drive Utility from the File Utilities panel, you are using it in an operator-interactive mode. Starting the utility in this mode lets you use multiple disks. The utility executes as a foreground application, which supports rewritable DVD's and CD's. In this type of operation, the operator supplies selections through menus and views response messages from the utility on the screen.

When you start the utility from either the host processor or from a 4680 BASIC application, the utility executes as a background application. In these last two cases, input to the utility is by commands that are stored in files on the hard disk drive, which is not supported in rewritable DVDs/CDs. Responses from the utility are also stored in files on the hard disk drive. When you start the utility from the host processor or an application, it supports only one disk.

The Optical Drive Utility is on:

- The hard disk-resident system
- The Supplemental Option using the CD-ROM system

Depending on the types of optical drive operations wanted, you might need to use both forms of the utility. The general rule of which system to operate from depends on the amount of data you want to process. If you want to perform backups or restores of an entire hard disk drive, use the utility on the Supplemental Option using the CD-ROM. If you want to process single files or groups of files, you can use the hard disk drive system or the Supplemental Option using the CD-ROM. If you want to initiate optical drive operations from the host or from an application program, you must use the utility on the hard disk drive system.

The Optical Drive Utility provides flexibility in the choice of how much data you need to operate. The choice can range from an individual file to all files on a hard disk drive. The system processes multiple files as a series of individual file operations.

You cannot IPL from the optical drive because of hardware and software restrictions. This also means that a data disk can remain in the drive during an IPL without causing a non-system-disk error.

The optical diskette uses a standard File Allocation Table (FAT) and cannot be partitioned, whereas rewritable DVDs/CDs use the Universal Disk Format (UDF). Write recovery during a power loss is not in effect for the optical drive.

Comparison with streaming tape drive utility

This section does not apply to CD/DVD drives (P:) and media.

The Optical Drive Utility is similar to the Streaming Tape Drive Utility.

Advantages

These are some of the advantages over the Streaming Tape Drive Utility:

- The system stores files on the optical diskette in a format that File Services can use. For example, after a backup to optical disk, the system can read files by using the 4690 Command Mode, 4680 BASIC applications, and DOS without using the Optical Backup Utility. The OS/2® operating system can read the hard disk drive only after it performs a CHKDSK O:/F.
- You can bypass a corrupted file.
- There are no high-priority processes that impact PLU.
- The Optical Drive Utility uses only about 400 KB of memory.
- There are no delays resulting from *hitchbacks* (the media stopping, rewinding, and starting again).
- The Optical Drive Utility uses much less time preparing the media for access because there is no recalibration or rewinding of the media.
- There is improved reliability of the hardware, media, and data.
- There is an improved menu to select volume to list, restore, or erase.
- You can restore a file to a different drive from which you backed it up.
- The list function allows longer path names without having to use a second line.
- You can format the optical diskette through RCP.
- The system handles errors during operation through one-line messages.

Disadvantages

These are some of the disadvantages over the Streaming Tape Drive Utility:

- You cannot back up the C: drive and the D: drive in the same backup operation because the directory of the optical disk will not contain information that indicates the original drive that you backed up. They must be backed up to different volumes. The description for each volume should describe the drive that you backed up.
- A list does not verify that the system can successfully restore the file.

Formatting the optical disk

A 4690 controller must format all optical diskettes that are used by the operating system. The operating system, DOS, and OS/2 can use an optical diskette that is formatted by a 4690 controller. A 4690 controller cannot access an optical diskette that is formatted by DOS or OS/2. OS/2 might require that CHKDSK O: /F be run first because it does not recognize the extended attributes.

If the diskette has already been formatted, the FAT will be reset. If the diskette has not already been formatted or if a long format is selected, a diskette surface analysis is done, taking almost 20 minutes. You can format optical diskettes by either using the Optical Backup Utility or using Command Mode. If using Command Mode, you can select a long format by using the -L flag on the FORMAT command. For example:

```
FORMAT O: -L
```

Powering off the device interrupts a format. The diskette cannot be ejected during a format. Exiting the utility does not stop the format. During a format, other applications cannot access the optical drive.

For rewritable CD/DVDs, the Optical Drive Utility can also format the disks. Extended attributes are not supported in these media. The format given is UDF. If a long format is selected, a disk surface analysis is done, taking up to one hour, depending on the media type. Short format resets the UDF and takes considerably less time than a long format. Rewritable DVDs/CDs can also be formatted from command line, using the long format: `FORMAT P: -L`, or the short format: `FORMAT P: .` The first time that a particular optical disc is used for UDF on 4690 OS, the long format option should be used. Distribution attributes are retained for 4690 operating system files.

Accessing the optical drive utility

You access the Optical Drive Utility by using menu panels for both the hard disk drive system and the Supplemental system. From the SYSTEM MAIN MENU, choose the File Utilities option. When the FILE UTILITIES panel appears, select the Optical Drive Utility option. When you select it, the OPTICAL DRIVE UTILITY selection panel appears.

This panel, and the ones that follow it, guide you in performing various operations with the optical drive. Operations include:

- Backing up files to an optical disk
- Restoring files from an optical disk
- Listing the optical diskette contents
- Erasing files from an optical disk
- Formatting an optical disk

Upon completion of a requested optical disk operation, a status and message panel appears. Messages indicate the number of successful file operations that is performed and individual file status messages for file operations that the system could not successfully perform.

Using the optical drive utility from the host

Note: (This functionality is not supported for rewritable DVDs and CDs.)

You can invoke the Optical Drive Utility from the host by using the RCP. An appropriate RCP selection file and RCP command file must be present on the hard disk drive. The RCP command file must contain specific commands to initiate the appropriate Optical Drive Utility functions. These commands include BACKUP, RESTORE, LIST, ERASE, and FORMAT. After the required RCP selection and RCP command files are on the hard disk drive, follow the procedures for invoking RCP from the host. Starting the utility from the host processor restricts you to the use of only one optical disk.

Because the RCP command file allows room only for one description during a backup selection, that description will be used for both the disk description and the volume description.

The RCP command line parameters cannot exceed 46 characters, including the description.

These procedures and associated host processing are described in the *4690 OS: Communications Programming Reference*.

Using the optical drive utility from an application program

You can invoke the Optical Drive Utility from a 4680 BASIC application program through the ADXSTART application service facility. The application that must be started is the RCP, named ADXCSH0L. RCP, in turn, initiates the Optical Drive Utility. An appropriate RCP selection file and RCP command file must be present on the hard disk drive. To initiate the appropriate Optical Drive Utility functions, the RCP command file must contain specific commands. These commands include BACKUP, RESTORE, LIST, ERASE, and FORMAT. After the BASIC application ensures that the required RCP selection and RCP command files are on the hard disk drive, it issues the ADXSTART call. Starting the utility from an application restricts you to using only one optical disk.

See the *4690 OS: Communications Programming Reference* for information on using ADXSTART to initiate a background application and for information on the details of the Optical Drive Utility RCP commands.

Logical drive name

The drive name is O: unless O: is already defined as a user-logical name. Alternatively, you can access the drive as OPTIC00: but this name does not allow you to use the optical drive from applications that reference the optical drive as O:, which include Create Problem Analysis Diskettes and the operating system configuration.

For rewritable DVDs/CDs the drive name is P:. (Alternatively, UDDFS: can be used, but only in Classic mode 4690 OS.) Commands such as **copy**, **delete**, **dir**, **rename**, and others can be performed on the DVD media unit only if the media has been formatted to support the UDF file system and you are using writable CD/DVD drives. Read/write commands cannot be performed when using a writable CD/DVD drive when using a media with ISO data.

Using the optical drive from command mode

You do not have to use the Optical Drive Utility to use the optical drive or DVD-ROM drive. The optical drive or DVD-ROM drive can be used as a secondary disk to hold files that are accessed from Command Mode or from an application program. The optical drive or DVD-ROM drive can be formatted using the Command Mode FORMAT command.

Most Command Mode commands support the optical drive. Create Problem Analysis Diskettes can direct the output to O:. Display/Alter and Performance Monitor will not support the optical drive or DVD-ROM drive. The operating system configuration allows some system and application logical names to be assigned to O: and P: for the DVD-ROM drive.

In Classic mode, you can redirect CPAD to the P: drive using option 6.

Status file definition

When you invoke the Optical Drive Utility from the host or from a BASIC application, a status file is accumulated. If the optical diskette operation is a backup or restore, this status file contains a status of the results from the operation. If the optical diskette operation is to list the optical diskette contents, this status file contains the list of the optical diskette contents. It is intended to be analyzed by the appropriate application programs at the store controller or the host to determine the success or failure of the requested Optical Drive Utility operations. The status file is named ADX_SDT1:ADXCS6DF.DAT. The format of the status entries is 82 bytes long, with the last two bytes being CR/LF.

File data accessing considerations

The Optical Drive Utility accesses data on the hard disk drives in the store controller by using the file services that are provided by the operating system. Through checks that are performed by the file system, the utility protects against writing dynamically changing disk data to an optical diskette, or writing optical diskette data to a diskette file that another application is using. If the utility detects a data access conflict for a file, the operation for that file is bypassed and an error message appears, or an entry is inserted in a status file.

When the store controller is operating from the system that is loaded from the hard disk drive, the operating system and installed applications have numerous files in use. An attempt to process “all files” by the Optical Drive Utility in this environment would lead to numerous files being bypassed because of file accessing conflicts. The result would be considerably less than an “all files” operation.

When the store controller is operating from the Supplemental Option using the CD-ROM, there are no hard-disk files in use. Therefore, the Optical Drive Utility will not encounter file data access conflicts. This ensures successful handling of any number of files, from one to all, by the Optical Drive Utility.

Use the Optical Drive Utility for individual files or groups of files when operating from the hard disk drive system. Run the Optical Drive Utility only when the intended files are closed or when they are no longer in use by your application.

There is no attempt to back up the following files:

- Temporary file, ADXCS6TF.DAT
- Volume label
- Optical volume label, VOLINFO
- Optical disk label, DSKINFO
- Temporary file, %* (any file that starts with a percent sign)
- Temporary file, *.\$\$\$ (any file whose extension is three dollar signs)
- System files (determined by the system file attribute in the FAT)

You can back up hidden files, but you cannot restore them if the files already exist. A file-in-use error appears if you attempt to restore a hidden file and the file already exists. The error comes from File Services, not this utility.

You can back up read-only files, but you cannot restore them if the files already exist. A file-access error appears if you attempt to restore a read-only file and the file already exists. The error comes from File Services, not this utility. The read-only attribute becomes disabled on the optical diskette during backup.

Disk identification and contents

Each disk, when used by the Optical Drive Utility, contains the following:

- A media identification file that specifies a user name for the disk or media.
- The date and time the media was labeled
- An indication if this disk is one of a set of disk used to create a multiple-disk backup

Whenever you insert an disk in the optical drive, a search for the disk identification file occurs, and it appears to the user if it is found.

When you back up files to the optical disk, the system can process a variable number of files during each BACKUP command. Each group of files processed by a backup operation is placed in a separate volume (root subdirectory called O:\VOLUMExx or P:\VOLUMExx, where xx is the volume number). The subdirectory (but not the files) that the file came from and its attributes are preserved within the volume. Only 99 volumes per disk set are supported. A volume identification file is written to the disk for each volume. A sequential volume number that is automatically incremented by the Optical Drive Utility identifies volumes. This concept of multiple volumes on an optical diskette allows backing up the same named file at different times to the same optical disk.

You can list, restore, or erase files that are written to the optical diskette without using the Optical Drive Utility by selecting a volume on O or on P.

If the optical disk does not have enough disk space to back up a file, you are prompted to insert another disk. Multiple disks are not supported through the RCP interface. Only 99 disks can be spanned. A file cannot be larger than the disk capacity and files are not split when the disk becomes full. When a file fails to fit on a disk, there is no effort to fill in the remaining free space with files that do fit. For rewritable DVDs and CDs, a file called DISTFILE is created for each volume with every file's distribution attribute in the volume.

Optical drive utility functions

The Optical Drive Utility supports these distinct optical diskette functions or commands:

- Backing up files to optical disk
- Restoring files from optical disk
- Listing optical disk contents
- Erasing files from optical disk
- Formatting optical disk

For rewritable DVDs and CDs, if you write or erase something to the media through command line, you must use the `unlockp` command to make the changes permanent. For the Optical Drive Utility case, `unlockp` is called automatically after any backup or erase to these disks (not required in Enhanced mode).

Backing up files to optical disk

Use this function to copy files from a hard disk drive to the optical disk. Options provided in this function enable backup of:

- All files on a drive
- All files in a subdirectory
- All files that are named in a list (see "Selection list file creation" on page 400)
- One file

Decide what to do with data already on the optical disks. Choose one of the following options:

- Format, then back up to volume 1
- Back up to new volume

If you choose to format the optical disk before the backup is performed, the format takes little time if the disk is already formatted. After the format, the system puts the files in a subdirectory on the optical disk that is named `VOLUME01`. The original subdirectories are created within the `VOLUME01` subdirectory.

If you choose to back up to a new volume, a new subdirectory is created on the optical diskette that is named `VOLUMExx`, where `xx` is one higher than the highest numbered volume on the optical disk. All files backed up are in subdirectories within the `VOLUMExx` subdirectory with the original subdirectory tree intact.

You can choose to include or exclude files that are open for unlocked write access. Two examples are the accounting totals files and the print spooler queue files. If you back up these files, you should stop all activity that would cause updates to these files.

The files backed up on the optical diskette will not contain the original drive, only the subdirectory and file names. It is your responsibility to indicate the original drive in the volume description if more than one drive is backed up to the same optical disk. For example, if you back up the C: drive to volume 1, and the D: drive to volume 2, you should indicate in the volume description for volume 1 that it was backed up from the C: drive and indicate in the volume description for volume 2 that it was backed up from the D: drive.

Restore files from optical disk

This function lets you copy files from an optical disk to a hard disk drive. Options provided in this function enable you to restore:

- All files in a volume
- All files in a subdirectory
- All files that are named in a list (see “Selection list file creation” on page 400)
- One file

An option in this function also lets you restore selected files from a particular volume or from all volumes on the optical disk. If a given file is present in multiple volumes being restored, the copy of the file in the last volume that was selected is the resulting file when the restore is complete.

List optical disk contents

Use this function to list the contents of a disk. Options provided in this function let you list:

- All files in a volume
- All files in a subdirectory
- All files that are named in a list (see “Selection list file creation” on page 400)
- One file

An option in this function also lets you list selected files from a particular volume or from all volumes on the optical disk. The list function displays or writes, to the status file, the file header information for each file that is found on the optical disk that matches the selection criteria. The selection options for the list function are

similar to those for the restore function. For example, the set of files that is listed would be identical to the set of files that are restored if the same options are chosen. Lists are limited to 600 files.

Erase files from optical disk

Use this function to erase selected volumes or selected files from the optical disk.

Format optical disk

This function lets you choose between a short format and a long format.

A short format attempts to reset the disk format, which contains the location of all files and subdirectories. When this option finishes, the disk is functionally empty. If an error occurs on a short format of an optical diskette in Classic Mode or on a rewritable DVD/CD in Enhanced Mode, a long format occurs even though you selected a short format. If a short format of a rewritable DVD/CD fails in Classic Mode, an error message will be given. You should request a long format instead.

A long format performs a disk surface analysis and erases the data. A long format takes almost 20 minutes for optical disk and up to one hour for Rewritable DVDs/CDs (depending on its size and media type.) You must format a new optical diskette using a store controller before you can use it. You cannot use an optical diskette that was formatted using a different operating system because the cluster size will be incompatible.

Selection list file creation

One method for specifying the set of files to back up, restore, list, or erase is to name a file that contains the appropriate set of files. You must create the selection list by using the redirected output from the Command Mode DIR command. DIR permits the use of global file name characters for grouping names and concatenating DIR lists into the same file. The maximum number of files in the list is 337.

Note: The DIR command cannot be run on the optical drive itself.

LAN system considerations

Any store controller equipped with a SCSI adapter and an attached optical drive or a DVD-ROM drive can use the Optical Drive Utility. The Optical Drive Utility accesses the hard disk drive data by using the operating system's file and LAN services functions. When operating from the Supplemental Option using the CD-ROM (DVD/CD is not supported), the LAN facilities are disabled. The Optical Drive Utility functions as if the store controller, on which the Optical Drive Utility is operating, is not connected to any other store controller.

When operating from the hard disk drive system, LAN facilities are available. These facilities provide the Optical Drive Utility with the capability to access files that are associated with the local store controller node or with other nodes on the LAN. By prefixing the node name to the normally specified path, the Optical Drive Utility can access files on any store controller on the LAN.

Understanding file distribution and DISTFILE

Another aspect of files in a LAN system deals with file distribution. The Optical Drive Utility preserves the distribution attributes of all files it processes. In addition,

the Optical Drive Utility restores files to the proper file owner for all distributed files. Designating the node name during the restore process will only affect local files. The standard LAN file distribution functions are then used to propagate the restored file to the appropriate LAN nodes. The node and drive from which the file originally came are not remembered.

DISTFILE is created under the Optical Drive Utility for every volume of a DVD/CD. This file contains the original file's distribution attribute copied from the hard disk. So, when a file from a DVD/CD is restored, it applies the distribution attribute contained in DISTFILE and distributes the file to the LAN. However, if a file is copied from the command line to a DVD/CD, it loses its distribution attribute.

Note: In rewritable DVDs/CDs, the distribution attribute is not supported. If you use the List Files option in the Optical Drive Utility for a DVD/CD, every file appears as a local file.

Supplemental Option using the CD-ROM considerations

You should not use logical names when operating with the Supplemental Option using the CD-ROM because the logical names are different or unavailable.

The LAN (MCF Network) is not supported when running with the Supplemental Option using the CD-ROM. All files are backed up or restored to the local store controller. The attribute of the file is preserved, but there is no attempt to distribute the file.

When operating with the Supplemental Option using the CD-ROM, the work file that is used to display the results goes to the V0: RAM disk. If the RAM disk becomes full, many of the file descriptions or errors are omitted. A message warns you of the omission. The same action occurs if the C: drive fills up when operating without the Supplemental Option using the CD-ROM.

Recommended usage for optical drive

The optical drive can provide an improved method for system backup and system replication. The optical drive offers better performance backup and restore functions compared to the performance of diskettes. It provides efficient data exchange capability for large data files, compared to either host communications or diskette operations. You can also use the optical drive for fast reorganization of hard disk drive files that are burdened by fragmentation.

A fundamental distinction must be made before understanding how to most effectively use the optical drive in the operating system. That distinction relates to the volume of data that is being processed. The optical drive can back up and restore files in these general usage categories:

- Operations that involve the entire contents of a hard disk drive
- Operations that involve individual files or groups of files

Entire hard disk drives and CD/DVD media

Note: Optical diskettes are not supported in Enhanced mode.

Operations involving the entire contents of a hard disk drive generally demand use of the Optical Drive Utility on the Supplemental Option using the CD-ROM. This is the only way to back up all the files from the hard disk drive, and to avoid file-in-use

access conflicts. Use the Supplemental Option using the CD-ROM to avoid similar access conflicts that are associated with the restore of an entire hard disk drive.

System replication and hard disk drive error recovery are examples of this category of optical use. The general procedure to follow is:

1. Create a backup-optical diskette by operating the Optical Drive Utility from the Supplemental Option using the CD-ROM. Select the option to back up all files to the optical disk. Depending on the relative size of the hard disk drives and the optical diskettes, you might need to insert multiple diskettes during this process.
2. Store the optical disks for future system restore purposes or distribute the optical disks for system replication.
3. When the need to restore from the optical diskette arises:
 - a. IPL the system from the Supplemental Option using the CD-ROM.
 - b. Select the **CPREP** or **DPREP** procedure to initialize and set up the hard disk drive.
 - c. Select to restore all files from optical diskette by using the Optical Drive Utility. If multiple optical diskettes are required, appropriate prompts guide you in the optical diskette handling.
 - d. Remove the Supplemental CD-ROM and IPL the system from the hard disk drive.

Selected files

You can perform operations on selected files by using the hard disk drive or the Supplemental Option using the CD-ROM. All the same file-in-use accessing conflicts can occur when dealing with an individual file or group of files. In this category, you can better control the application and system use of files.

Examples of reasons to back up and restore individual files are:

- Protecting a particular file against hardware failure
- Archiving of particular files on a daily or weekly basis
- Retaining a copy of files before additional handling or processing, such as host transmission
- Using optical diskette to distribute very large files. This also applied to CD/DVD media.

Following are some typical ways you might want to use the Optical Drive Utility:

- If the store environment is one where the operator is involved in completing various procedures daily at closing, it is likely that the interface of using menus is appropriate for using the Optical Drive Utility. The operator selects the Optical Drive Utility, supplies the file name, or name of the file that contains the list of files to process, loads the proper optical diskette into the optical diskette drive, and backs up certain daily files. This also applies to CD/DVD media.
- If the closing procedures are more or less automated, the invocation of, and input to, the Optical Drive Utility can be supplied by a BASIC application program run as part of the store closing procedures. The only operator involvement required is to ensure at some time during the day that the appropriate disk is properly loaded in the optical drive.
- If host transmissions and interaction are part of daily store closing activities, the host can initiate the appropriate Optical Drive Utility operations at the appropriate times, similar to the case for application initiation. Again, the only operator involvement is the insertion of the appropriate disk.

Backing up files that applications are updating

The following example illustrates problems that could occur when you are backing up files that the applications are updating.

Accounting totals are kept in several files on the store controller. As each transaction occurs, the files tracking accounting totals are updated. Transactions occurring during back up of these files with the optical drive might be reflected in some of the files that are backed up, but not all.

At 11:30 a.m., a backup of the C: drive begins.

At 11:32 a.m., the transaction log and tender file are backed up.

At 11:34 a.m., a transaction occurs at a register. The transaction log and all accounting totals files are updated.

At 11:35 a.m., the remaining accounting totals files are backed up.

The transaction log and the tender file on the optical diskette do not reflect the transaction that occurred at 11:34 a.m., but the remaining accounting totals files do reflect the transaction. If this optical diskette is used to restore the backed up files, the accounting totals would be incorrect or confusing.

This complication is not isolated to accounting totals; a similar situation can occur with the print spooler. Any update that affects more than one file or even different records within one large file is exposed to this problem.

You can reduce the chances of this complication occurring by excluding files that are being updated (open for unlocked write access). You can make this selection on the panel when the backup operation is selected or through an optional parameter when using RCP. If you can include these files, you should stop all activity that would cause updates to these files for the duration of the backup. The best way to stop all activity is to use the Supplemental Option using the CD-ROM for the backup. See the *4690 OS: Communications Programming Reference* for more information on backing up files that are being updated by applications.

If a file is locked (opened for exclusive access), it gives a file-in-use error. If a file is not locked, but a record is locked during the backup, the utility tries to read the record every second until the record becomes unlocked.

Write recovery

A loss of power during the update of a file on the optical drive might cause an incomplete file and a corrupted FAT. CHKDSK O: -F might fix the FAT, but it will not recover the file. In some cases, a FORMAT might be necessary to fix the FAT. If the affected file is a keyed file, the chains might be corrupted and render the file useless.

For DVD media, CHKDSK P: -F might fix the FAT. If you use the COPY command in Command Mode, an UNLOCKP command must be issued in order to complete the write operation. If the store controller is turned off without the UNLOCKP command, it could result in an incomplete file copy and/or corrupted DVD media.

This only applies to Classic mode; for Enhanced mode, you should repeat the backup.

Alternative drives

You can use the Optical Backup Utility to operate on a drive other than optical drives if the same drive can be accessed in Command Mode. For example, you can use the D: drive as the backup drive instead of the optical drive.

The panels and messages will continue to refer to the selected drive as an optical drive and optical disk.

Note: When booted from the Supplemental Option using the CD-ROM system, the Optical Drive Utility is only available if an optical drive or rewritable DVD/CD drive is attached to the system.

RCP command interface

After running the Optical Drive Utility through RCP, you can retrieve the output file, which contains the results of the operation, to a host by using the HCP name CS6DF/ (binary file) or CS6DF&; (ASCII-to-EBCDIC translation). See the *4690 OS: Communications Programming Reference* for a description of these commands.

Cancelling a backup that is in progress

A backup operation can be cancelled by stopping the backup Window in the Window Control dialog (**Alt-SysRq-W**), but the state of the backup medium in use at the time is unpredictable. The medium should be considered potentially damaged, and not trusted for further backup operations (even if reformatted).

In Enhanced mode, activity to the medium might continue for some time after the backup window is stopped. System activity can continue up to the amount of time it would have taken to complete the backup operation. During this time, the Eject Button and the Enhanced Menu Eject function can not eject the medium. Dumping controller memory and Enhanced Menu Cold Restart also might not be possible during this time. The activity to the medium will end after some time (disregard regular blinking that can continue to occur on some drives from status polling). Dumping controller memory and the Enhanced Menu Cold Restart function will then be available, and the medium can be ejected with the Enhanced Menu Eject function. If waiting for activity to cease is not acceptable, you can power off and power back on the machine to avoid waiting.

Chapter 19. Using the File Compress/Decompress Utility

The operating system provides functions to compress and decompress files on your system. File compression causes a file to occupy a smaller amount of fixed disk or diskette space than the original form of the file, while retaining all of the original data. File decompression restores a file to its original form. These processes preserve the file attributes, as well as the time and date of the original file. The utility can be started from the SYSTEM MAIN MENU or remotely from the host processor.

When the File Compression/Decompression is started from the FILE UTILITIES panel, you are using it in an operator-interactive mode. In this mode, the compression utility executes as a foreground application. Selection are made through menus and view response messages on the screens.

The utility executes as a background application when the utility is started from the host processor. Input to the background application is through the Remote Command Processor interface. See the *4690 OS: Communications Programming Reference* for more information about File Compression/Decompression Utility commands.

Accessing the File Compress/Decompress Utility through Panels

The File Compress/Decompress Utility is accessed by using menus. From the SYSTEM MAIN MENU, select the **File Utilities** option. When the FILE UTILITIES menu appears, select the **File Compress/Decompress** option. When selected, the FILE COMPRESS/DECOMPRESS panel appears.

This panel and the ones that follow it provide guidance in performing various operations with the File Compress/Decompress Utility. Options available on the panels are:

Request

Files can be compressed or decompressed. File compression makes the file smaller while retaining all of the original data. File decompression restores the file to its original size. These processes preserve the file attributes, the time, and the date of the original files.

Drive All files on a hard disk drive or diskette can be compressed or decompressed. This includes all files in subdirectories and files on the root directory. When compressing or decompressing from drive-to-drive, a drive cannot be compressed or decompressed onto itself.

Subdirectory

All files in a subdirectory on a hard disk drive or a diskette can be compressed or decompressed.

File The named file is acted on. The file name can be a fully qualified name of the file to be acted on, or it can contain a global file name character (*). Global file name characters can appear in either the file name, the extension, or both parts of the name, for example, C:\ADX_IDT1\EALTS*.DAT. Global file name characters are not allowed in the hard disk drive or subdirectory portions.

List The list option allows specifying the name of a file that contains a list of file names to be acted on. Each entry in the list must contain the source,

destination, and name of the file to be acted on. See the *4690 OS: Communications Programming Reference* for more information.

Note: Logical names and node names can be used also.

Source and Destination

The source and destination of files to be compressed or decompressed can be files on a hard disk drive or on a diskette drive. If the destination does not exist, the file is created. The destination subdirectory is created if it does not exist.

Note: Files cannot be compressed into the ADX_SPGM subdirectory on drive C. This restriction prevents putting the store controller in a state in which it does not successfully IPL. Also, a drive cannot be compressed or decompressed onto itself.

Using the File Compress/Decompress Utility from Host Processor

The File Compress/Decompress Utility can be invoked from the host processor by using the RCP facility. An appropriate RCP selection file and RCP command file must be present on the hard disk drive. Specific commands must be present in the command file to initiate the necessary Compress/Decompress functions. After the required selection and command file are on the hard disk drive, follow the standard procedures for using RCP from the host. See the *4690 OS: Communications Programming Reference* for a description of these ADXCS30L procedures and the associated host processing.

On completion of a requested utility command, entries are made in an RCP status file. The host can retrieve this status file for analysis and to determine the status of the operations.

Neither source nor destination can reference a diskette drive when using this function through the RCP interface.

Using the File Compress/Decompress Utility with Diskettes

Diskettes can be used to compress or decompress files. Operations from hard disk drive to diskette, diskette to hard disk drive, and diskette to diskette are possible with the File Compress/Decompress Utility.

When files are compressed to a diskette, the resulting file might be split across two or more diskettes that are based on the amount of space that is needed by the compressed file. To ensure correct decompression, a control file is placed on each diskette that contains part of a compressed file. The control file contains a diskette number that enables the decompression utility to ensure that the proper diskette is in place. This data is used when operating from diskette to hard disk drive or diskette to diskette. The name of the control file is CNTRL\$@% (there is no file extension associated with the file name).

Care must be taken when handling compressed files on diskettes. A common action is to compress files to a diskette, transport the files to another location, and then decompress the files. If you copy the compressed files onto another diskette, you must also copy the control file that exists on each diskette. The control file contains diskette sequencing information that allows the decompression utility to correctly decompress each file and to ensure that parts of compressed files are processed in the correct order.

LAN System Considerations

The File Compress/Decompress Utility can be used at any store controller. The utility accesses hard disk drive data or diskette data by using the operating system's file and LAN (MCF Network) services functions.

The File Compress/Decompress Utility accesses files that are associated with the local store controller node or with other nodes on the LAN. By prefixing the node name to the normally specified path, the utility can access files on any store controller on the LAN.

The File Compress/Decompress Utility preserves the distribution attributes of all files it processes. In addition, the utility restores files to the proper file owner for all distributed files. The standard LAN file distribution functions are then used to propagate the restored file to the appropriate LAN nodes.

The File Compress/Decompress Utility can access files on LAN-connected controllers. The utility prefixes the file path information with LAN node names having the format, ADXLXxxN::, where xx represents the LAN node ID.

Also, the files accessed (whether on a local or on another node on the LAN system) have associated file types and modes. However, you must understand LAN file types and modes (often referred to as *file attributes*). The LAN file types are local, mirrored, and compound. The file modes are distribute at close and distribute per update. See "Defining File Types and Modes" on page 455 for more information.

System files reside in any ADX_?MNT, ADX_?PGM, and ADX_?BUL subdirectories and in the ADX_SDT1 subdirectory. Application files reside in any other subdirectory, including the root.

The master store controller in a LAN system contains the prime version of all system files and application compound files. The file server controller contains the prime version of the application-mirrored files.

LAN (MCF Network) Attributes and the File Compress/Decompress Utility

The File Compress/Decompress Utility can compress any type of file and the destination of that compress can be directed to any location except ADX_?PGM subdirectories.

Decompressing files is more restrictive. When compressing a file, the utility puts a header in the compressed file that indicates the following items:

- Name of the original file
- LAN (MCF Network) attributes of the original file
- Size of the original file
- Size of the compressed file

The LAN attribute in the header and the location of the compressed file determines where the prime version of the file must reside. For example, if a file is a mirrored file and the compressed file resides in the ADX_?MNT, ADX_?PGM, ADX_?BUL, or the ADX_SDT1 subdirectory, the destination of the decompression also must be one of those subdirectories. If the destination of the decompression is any other subdirectory, the decompression of the file cannot take place because the controller where the prime file resides changes with the new destination. This restriction also applies when you try to rename a file from ADX_?MNT, ADX_?PGM, ADX_?BUL, or ADX_SDT1 to any other subdirectory. A file cannot be moved from one subdirectory

to another subdirectory on the same controller if that movement will cause a change in ownership of the prime version of that file.

File Compress/Decompress Utility and Hard Disk Drive Space

This utility uses hard disk drive space on the controller to compress and decompress files, even if the source and destination of the request is to a diskette drive. If the hard disk drive lacks space to perform the compress or decompress, the utility ends with a message indicating insufficient disk space. If you receive this message, you must make room on the hard disk drive before you can attempt the function again.

Error Logging

The File Compress/Decompress Utility stores the following information at the beginning of all compressed files. This information is used for logging error messages.

- 4-byte compression flag signaling that the file is in compressed form
- 8-byte character file name that contains the name of the original file
- 2-byte LAN file attributes that contain the type of file that was compressed (for example, compound)
- 4-byte file size of the compressed file
- 4-byte file size of the original uncompressed file

Chapter 20. Using the disk rebuild utility

This chapter explains how to use the LAN Disk Rebuild Utility.

Introduction to the LAN disk rebuild utility

The LAN Disk Rebuild Utility (LDRU) is designed to help recover from a hard disk drive failure. The LDRU is usable only if you have the Multiple Controller Feature (MCF). This utility is available using the Supplemental Option using the CD-ROM. It can recover either a C or D disk by accessing the files across the LAN (MCF Network) from another store controller. To use this utility, you must know the LAN node ID of any other store controller on the LAN and the data path to be recovered. This path is usually either C or D.

If the controllers are configured for IP Based communications (CCC/IP), to use this utility and transmit data over IP, you must know the IP address for the master controller and the IP information for the controller that is being rebuilt, in addition to the LAN node ID of any other store controller on the LAN and the data path to be recovered. Refer to the *4690 OS: Planning, Installation, and Configuration Guide* for information about CCC/IP.

Note: This utility can be run on only one store controller on the LAN at a time because it uses a reserved LAN node ID for its operation. If a store controller has run this utility, that controller must be reloaded before another controller can run the utility.

To perform a LAN disk rebuild, both the master controller and the file server must be found on the LAN. If a new store controller is being installed or an existing store controller is being replaced, you must also know the LAN node ID of the store controller to be installed or replaced.

The LDRU clears certain portions of NVRAM (non-volatile RAM) that results in the following:

- The Backup Allowed flag contained in NVRAM is cleared. After LAN Disk Rebuild is run, backup is prevented for any terminals for which the controller provides LAN backup. Therefore, if a LAN disk rebuild is performed on a controller, the following three actions must be performed:
 1. Re-IPL the controller
 2. Select **TCC Functions** from the STORE CONTROLLER FUNCTIONS panel
 3. Select **ALLOW BACKUP**, as necessary

If these three actions are not performed, a backup does not occur should a primary controller fail, that in turn results in the terminals going offline until backup is allowed or the primary controller is restored.

- The controller no longer acts as the master or file server

Once a LAN disk rebuild is performed, the information determining that the controller was acting as the master or file server before the LAN disk rebuild is lost. When the controller is restarted, the controller must be reactivated as the master or file server.

- Event logs are reset

The pointers to the current location in the event log are reset. Therefore, after the controller is restarted, the event logs are reset and any events that had been in the logs are now unavailable on that controller.

- System Message Level is set to 3

If you have set the system message level to any level other than 3, it gets reset to 3. (This determines the minimum severity for messages to be displayed on the System Message panel when **Alt+SysRq+M** is pressed.) Once the controller is restarted, **Set System Message Level** must be selected from the SYSTEM FUNCTIONS panel and the message level must be set to the previously specified level.

Certain issues should be kept in mind when performing the LAN Disk Rebuild Utility. Some of these issues are related to IP Security and SSH.

- Processes that were started manually will need to be restarted on the rebuilt controller.
- If IP Based communications (CCC/IP) is configured **and** there is an IP Security manual tunnel defined between the acting Master and the controller that is to be rebuilt, then that tunnel needs to be deactivated (rmtun -t # where # is the tunnel number) before starting LDRU. Otherwise, LDRU will not be able to communicate with the acting Master.
- Local files that were unique to the failing controller will not be restored. Local files from the source controller will be placed on the target controller.
- The following SSH files are not copied from source to target Controller during LAN disk rebuild:

```
ADX_SDT1/ADXSSHAK.DAT
ADX_SDT1/ADXSSHBK.DAT
ADX_SDT1/ADXSSHKH.DAT
ADX_SDT1/ADXSSHK2.DAT
ADX_SDT1/ADXSSHDK.DAT
ADX_SDT1/ADXSSHRK.DAT
ADX_SDT1/ADXSSHDK.PBK
ADX_SDT1/ADXSSHRK.PBK
ADX_SDT1/ADXSSHRC.DAT
ADX_SDT1/ADXSSHRC.PBK
ADX_SDT1/ADXSSHDC.DAT
ADX_SDT1/ADXSSHDC.PBK
```

Note: These files contain the keys which are unique for each client. The keys for SSH must be regenerated after LDRU.

- The IPsec configuration files contain IP information that is unique to each controller and are not copied to the target controller during LDRU. IPsec manual tunnel and filter configuration must be reestablished on the target controller, if desired.

V6 LAN disk rebuild considerations

Using the Supplemental Option when booted from the first disc (Vol 01 of 02) of the V6 Base/Migrate package will use V6 Enhanced Mode. Using the LDRU to build a disk in Enhanced Mode will create a V6 Enhanced Mode system, even if the source system is a Classic Mode controller. To rebuild a V6 Classic Mode controller, you need to boot the V6 Classic Mode Supplementals.

Note: When you use the LDRU to build a C: drive from a V6 Classic Mode system all of the VFS files from the M: drive will be copied (as these files actually reside on the C: drive). On a V6 Enhanced Mode system, the Remote Management Agent (RMA) code will reside on the F: drive so you may wish to erase the M:\RMA file directory tree after the LAN disk rebuild is complete.

Restoring data on a hard disk drive

The following outline lists the steps to restore data on a hard disk drive from another store controller across the LAN (MCF Network):

1. On the store controller that is to be rebuilt, start the Supplemental Option using the CD-ROM.

Notes:

- a. For an Enhanced Mode controller, the V6 or Enhanced supplemental CD must be used. If you do not use the supplemental option of the Base/Migrate or install CD, then you must generate an Enhanced Mode supplemental OS CD from the Enhanced Options Menu from another Enhanced Mode controller.
 - b. For a Classic Mode controller, the Classic supplemental CD must be used.
2. Run CPREP or DPREP to format a new hard disk drive.
 3. Verify the presence of an acting master store controller and file server.
 4. Start the LAN Disk Rebuild Utility.
 5. Select Ethernet as the LAN Type..
 6. Select the transfer type:
If you select NetBIOS, go to step 7.
If you select IP Based, enter the IP and node ID information; then, go to step 8.
 7. Enter the LAN node ID of another store controller on the system, the name of the disk that is to be rebuilt (C or D), and the LAN node ID of the store controller to be rebuilt.
 8. Wait for the data to be recovered.
 9. If an error occurs, check the audit trail. See "Checking the audit trail" on page 416.

Note: If you are working with the Supplemental Option using the CD-ROM, place the CD-ROM in the CD-ROM drive (and, if using the 4690 CD-ROM boot diskette, place it in the A: drive) of the store controller to be rebuilt. If the store controller is powered on, simultaneously press **Ctrl**, **Alt**, and **Del**. If the controller is powered off, power On the controller and then, choose the Supplemental Option.

When the Supplemental Option using the CD-ROM has been loaded correctly, a menu appears. Select the **Command Mode** option, to enter Command Mode. When the T> prompt appears, the system is ready for a command.

Formatting a hard disk drive

This step is not required if the disk being rebuilt is already formatted and ready to accept data. In most cases, however, a newly installed hard-disk drive or an existing hard-disk drive must be formatted. To format the hard-disk drive being rebuilt, use either CPREP for the C drive or DPREP for the D drive. To run these from the Command Mode prompt, type either **CPREP** or **DPREP** at the T> prompt.

With Version 4 of the 4690 OS, CPREP and DPREP have been changed to support larger hard-disk drives . While CPREP or DPREP is running, if the hard-disk drive contains 500 megabytes or more, the user is prompted to select either the legacy 16-bit FAT file system or the 32-bit FAT file system.

Note: Systems that are migrated from 4690 OS versions prior to Version 4 will not be reformatted. They will continue to be 16-bit FAT file systems. If you want to change to 32-bit FAT file system, you must run CPREP or DPREP.

After CPREP or DPREP is completely finished, the T> prompt appears again.

Locating controller IDs for NetBIOS-based systems

The Disk Rebuild Utility requires an active master ID and file server ID on the system. The LAN (MCF Network) node ID is a two-character ID used to describe a store controller on the LAN.

To define the ID, you can invoke the Display Controller Status on another store controller on the LAN and follow these steps.

1. Press the System Request key (**Alt+SysRq** or **Alt+SysReq**).
2. Select option **C** (Store Control Functions).
3. Select option **2** (Controller Functions).
4. Select option **1** (Display Controller Status).

Note: Use the **PgDn** key to view the roles of the various node IDs.

The LAN (MCF Network) node ID of another store controller on the system is needed for the utility to access that node and receive a copy of the node list file created during configuration. The list file contains all of the node IDs and functions for store controllers on the LAN. The procedure can also determine the ID of other store controllers on the LAN.

The following panel shows where the LAN node IDs and path names should be entered.

NSL1S003

Enter NODE IDS AND PATH NAME

Type the necessary information.

LOCAL NODE ID - ADXLXDDN::	LAN node ID of this controller
OTHER NODE ID - ADXLX__N::	LAN node ID of any other controller on the system
PATH TO BE RECOVERED - _____	Disk or path name to be recovered (C: or D:)

Press Enter when complete.

F1HELP F2 F3QUIT F4 F5 F6 F7 F8 F9 F10

Determining IP configuration information

If you want the Disk Rebuild Utility to transfer information over IP and your system is configured for IP Based communications, the Disk Rebuild Utility needs to know the IP address of the master controller in order to access it. It also needs to know the IP information for the controller that is being rebuilt. The master controller has an application, ADXIPCFG.286, that can be run from the command line to help you determine the IP information that you will need. (Refer to the *4690 OS: Planning, Installation, and Configuration Guide* for information about Controller-to-Controller Communications over IP.)

Note: This utility will not work with BOOTP configurations.

To determine the IP configuration information:

1. Type **ADXIPCFG** from the command line and press **Enter**.
2. The following screen is displayed:

CFG1S002

TARGET CONTROLLER IDENTITY

Please enter the two letter node id of the target controller to be rebuilt.

Node ID : __

Press ENTER when complete

F1HELP F2 F3QUIT F4 F5 F6 F7 F8 F9 F10

3. Enter the Node ID of the target controller that you want to rebuild and press **Enter**.
4. A screen similar to the following screen is displayed. Record the IP information for use in the LAN Disk Rebuild Utility. Press **Enter** after you record the information.

CFG1S001

IP CONFIGURATION

Record the necessary information.

LOCAL NODE ID – ADXLXDDN::

LAN Node ID of the controller To be rebuilt.

LOCAL NODE IP ADDRESS 192 . 168 . 1 . 2

LOCAL NODE IP NETMASK 255 . 255 . 255 . 0

LOCAL NODE ROUTER 192 . 168 . 1 . 17

OTHER NODE ID – ADXLXCCN::

LAN Node id of this controller

OTHER NODE IP ADDRESS 192 . 168 . 1 . 1

Press ENTER when complete

F1HELP F2 F3QUIT F4 F5 F6 F7 F8 F9 F10

Using valid paths

In most cases, the path name to be recovered is C or D. These paths signify that the entire hard disk drive is to be recovered. To recover part of the hard disk drive or to recover specific files be more specific in the path field. No logical names can be in the paths, such as ADX_IDT1:. Following are some examples of valid and incorrect path names:

- **Valid Paths**

- C: or D:**

- Recover the entire hard disk drive.

- C:\ADX_SPGM**

- Recover all of the ADX_SPGM subdirectory.

- D:\ADX_IDT1\EAL*.***

- Recover all files starting with EAL in ADX_IDT1 subdirectory on D.

- **Invalid Paths**

- A: or B:**

- Only C and D disk files are allowed.

- ADX_SPGM:**

- Resolves to A disk on supplemental.

- EALTRANS**

- No logical name on supplemental.

- C:\ADX_SPGM**

- Viewed as a file. Add a slash at the end of the path for subdirectory names.

Loading the disk rebuild utility

The Disk Rebuild Utility can be loaded from the Supplemental Option using the CD-ROM.

To load the Disk Rebuild Utility by choosing the Supplemental Option using the CD-ROM, perform the following steps::

1. Verify that a T> prompt is on the panel.
2. Place the CD-ROM in the CD-ROM drive.

Note: If no CD-ROM or the wrong CD-ROM is inserted when ADXNSL0L is run, the system prompts you to enter the correct CD-ROM.

3. Type **ADXNSL0L** at the T> prompt.

Note: Be sure to enter a zero, and not the letter O.

T>ADXNSL0L

4. Press **Enter**.

Entering the LAN type

You are asked to select the LAN type. Select Ethernet and a panel is displayed asking you to select a transfer method (see “Selecting a transfer method” on page 415).

Selecting a transfer method

If you select Ethernet as the LAN type, the following panel is displayed.

```
NSL1S006                DISK REBUILD UTILITY

To rebuild the disk you must select a transfer method.
Indicate the type :
                        1 = NetBIOS
                        2 = IP Based

Press ENTER when complete

F1HELP F2      F3QUIT F4      F5      F6      F7      F8      F9      F10
```

If you enter **1** to rebuild the disk using NetBIOS, you are asked to enter the LAN node ID (see “Entering the LAN (MCF Network) node ID” on page 416).

If you enter **2** to rebuild the disk using IP as a transfer method, you are asked to enter the IP information (see “Entering the IP information”).

Note: If you select the incorrect transfer method and press **ENTER**, you will need to IPL the controller before you can continue using the LDRU. If you correct the selection before you press **ENTER**, the problem will not occur.

Entering the IP information

If you selected the IP Based transfer method, the following panel is displayed. Enter the required IP information and then press **ENTER** to start the rebuild.

```
NSL1S007                ENTER NODE IDS, IP AND PATH NAME

Type the necessary information.

LOCAL NODE ID - ADXLX_N::                LAN Node ID of this controller

LOCAL NODE IP ADDRESS  __ . __ . __ . __
LOCAL NODE IP NETMASK  __ . __ . __ . __

    *Please enter 0.0.0.0 for the router information if not applicable*
LOCAL NODE ROUTER      __ . __ . __ . __

OTHER NODE ID - ADXLX_N::                LAN Node ID of the master
OTHER NODE IP ADDRESS  __ . __ . __ . __                                controller on the system

PATH TO BE RECOVERED - _____      Disk or path name to be
                                           Recovered (C: or D:)

To QUIT, press F3. To start LAN Rebuild, press ENTER.

F1HELP F2      F3QUIT F4      F5      F6      F7      F8      F9      F10
```

Entering the LAN (MCF Network) node ID

The LAN (MCF Network) node ID is a two-character ID used to describe a store controller on the LAN. The ID resides, for example, in the store controllers NVRAM. The Disk Rebuild Utility displays the current node ID of the store controller that is being stored as a default.

In most cases, the default node ID does not need to be changed. However, certain cases, such as installing a new NVRAM card in a store controller that is being restored or adding a new store controller to the LAN, require another node ID. If the Disk Rebuild Utility is being used to rebuild a hard disk drive on a store controller that is being added to the LAN, first add the node ID through store controller configuration. Configuration is performed at the acting master store controller.

Attention: If the local LAN ID is accidentally typed over or lost and the correct ID is not known, exit the Disk Rebuild Utility by pressing **F3**. When the utility is restarted, the default LAN ID appears again.

Recovering data

Enter the node ID, the disk or path, and the local node ID. The Disk Rebuild Utility verifies that the nodes are correct and that the path is valid, and then begins data recovery. The Disk Rebuild Utility creates any subdirectories that it needs on the new hard disk drive. During the processing, a percentage appears to provide an indication of how much data has been transferred. The percentage gives an estimation of how much time is left in the recovery process. The amount of data transferred increases by 10% intervals.

The following panel is an example of the results from the Disk Rebuild Utility.

NSL1S005		LAN DISK REBUILD UTILITY PROCESSING							
OFFSET		FILE NAME							
8196		H0:\\ADX_SPGM\\ADXCCKF.DAT							
0		H0:\\ADX_APGM\\ADXLND0L.286							
230194		H0:\\ADX_SPGM\\ADXCSC0L.286							
1016		H0:\\ADX_SPGM\\ADXDEDDF.DAT							
13789450		Total bytes transferred (60%)							
F1	F2	F3QUIT	F4	F5	F6	F7	F8	F9	F10

Checking the audit trail

The Disk Rebuild Utility creates two audit trail files that track the data that was transferred and any problems that may have occurred. The files are saved on the acting master controller. File C:\\ADX_SDT1\\ADXNSL1F.DAT contains the Disk Rebuild Utility start and finish times, the number of files transferred, the number of bytes transferred, the applications ending with **F3**, and any other problems encountered with transferring files. Consult this log file if a problem occurs while transferring a file and the file is not present on the rebuilt hard disk drive.

File C:\ADX_SDT1\ADXNSL2F.DAT contains the Disk Rebuild Utility start and finish time and the current percentage of completion. Examine either of these files using the system editor, DREDIX, or enter the following command on the command line of the acting master store controller:

```
TYPE c:\adx_sdt1\adxns11f.dat  
Or  
TYPE c:\adx_sdt1\adxns12f.dat
```

Use the COPY command to recover the files that were not transferred.

If the error message Unable to write to event trail log, rc=xxxxxxx is seen on the screen during the rebuild, it means that the LAN Disk Rebuild Utility was unable to log data in the audit trail on the master store controller for the reason given by the return code (rc=). One or more files were probably not transferred because of the same root cause.

Nested subdirectories

The maximum length of the complete path name (for example, the extended path name) that the Disk Rebuild Utility supports is limited. Because of this, the number of levels of subdirectories that the Disk Rebuild Utility supports when using nested subdirectories is also limited. When attempting to transfer a file with an extended path name that exceeds the limit, the utility might end before all requested files are transferred. In this case, the Disk Rebuild Utility logs an error in the audit trail.

Note: For versions of the 4690 OS prior to Version 5, the LAN Disk Rebuild Utility could not handle complete file names that were longer than 47 characters. For example, ADXLXTCN::H0://ADX_UDT1/LEVEL1/LEVEL2/THEFILE1.TXT exceeds the limit starting with TXT. Version 5 adds support for complete file names that are a maximum of 128 characters, which is the limit set by the OS shell.

Chapter 21. Using Terminal RAM Disk Preload Utilities

The operating system provides terminal RAM disk preload utilities to efficiently and quickly load customer-supplied files onto terminal RAM disks. This improvement in performance is accomplished by:

- Compressing files that were previously copied as-is using the ADXCOPYF command.
- Transferring files as one image to the terminal during remote IPL (RIPL).
- Broadcasting the image to all terminals that are performing an IPL remotely.

Files may also be loaded to the terminal RAM disks (and hard disks) using the enhanced terminal preload facility. Files loaded using enhanced preload have many of the same advantages as above (the files are compressed and broadcast to terminals in large bundles). Additionally, enhanced preload allows these advantages:

- The configuration utility is used to interactively configure file preload.
- Files can be placed on any available terminal disk (C:, Q:, M:, X:, Y:, or F:).
- Files can be placed in subdirectories.
- Difference sets of files can be broadcast to different terminals.
- If the terminal has a hard disk, the files to be placed on the RAM disk are cached on the C: drive and only reloaded from the controller when they change.

See Chapter 10, “Enhanced terminal preload and Java configuration,” on page 305 for more information.

The following steps show the process of using the preload utilities to compress files and load the terminal RAM disk.

1. Create the list files with the files that are to be loaded onto the RAM disks.
2. Run the Image Build Utility to create image files.
3. Run ADXRTCCL to create the terminal load-shrink file.
4. Reload the terminal.

These steps must be run each time you want to update your terminal RAM disk with new files.

Notes:

1. These preload functions are not available on terminals that have only 2 MB of memory.
2. In MCF environments, the Image Build Utility and ADXRTCCL must be run from the master controller.
3. Files loaded using the enhanced terminal preload are loaded prior to files loaded using the legacy preload method described here. Thus, if a file with the same name is loaded to the root directory of a disk using both methods, then the file loaded by legacy preload will be the final copy on the disk after the machine is fully loaded.

Creating List Files

To create list files, the first step is to determine which files are to be loaded onto the RAM disk. Using any editor program, create two input files in the ADX_IDT1: subdirectory that contain the list of files to be loaded. The two input files are shown below:

- ADXTRMXF.DAT for files to be loaded on terminal RAM disk X:
- ADXTRMYF.DAT for files to be loaded on terminal RAM disk Y:

These files should be provided in the *subdir:filename.ext* format, where *subdir* is a logical name specifying the subdirectory and *filename.ext* is a file name with an 8.3-character format. The fully-qualified file name can also be used. For example:

```
*These files are copied to X: in all terminals at RIPL:
ADX_IDT1:ITEMFILE.DAT  -o
ADX_IDT1:OTHERFIL.DAT
ADX_IDT1:EXTRAS.DAT   -c
ADX_IDT1:ANYFILE.DAT
ADX_IDT1:ANYFILE2.DAT
C:/ADX_IDT1/APPL/SA/UPC.DAT
```

Notes:

1. Long file names are not supported with this utility. The file names must be in an 8.3-character format.
2. Wildcard characters (such as * and ?) in list file names are not supported. The characters * or ! act as comment delimiters. Within a line, any characters after these delimiters are ignored. If either of these delimiters is used as the first character on any line, the entire line is ignored.
3. A file can be marked as optional in the list file by adding one or more spaces and -o or -O following the file name. If the utility cannot open a file that is marked optional, that file is skipped.
4. A file can be marked so that it is not compressed but copied as is by adding one or more spaces and -c or -C following the file name. This can be used on files such as .JAR and .ZIP files that have no repeatable patterns in them and can increase in size after compression.
5. No checking is performed to indicate a multiple use of the same file name in either the X: or Y: drive.

Running the Image Build Utility

When the files to be loaded onto the terminal RAM disks are determined, use the Image Build Utility, ADXTRM0L, to pack up the list files and to create two output files, ADXTRM1F.DAT and ADXTRM2F.DAT, in the ADX_SDT1: subdirectory. The Image Build Utility combines and compresses the contents of the files listed in ADXTRMXF.DAT into ADXTRM1F.DAT and combines and compresses the contents of the files listed in ADXTRMYF.DAT into ADXTRM2F.DAT.

The Image Build Utility can be run either from a command line within a .BAT file, or can be run as a background task. When starting the utility, use the parameters -x and -y to indicate the RAM disks for the image files that are to be built. For example, to create an image file for drive X:, use the following command

```
ADXTRM0L -X
```

Note: No checking is performed to prevent you from specifying -X or -Y multiple times on the command.

If any files specified in ADXTRMXF.DAT or ADXTRMYF.DAT do not exist or cannot be completely read or compressed, an error message is logged. After loading the terminal, verify the contents of the RAMDISK to ensure that all files that were specified to be preloaded into the terminal's RAM disk were successfully preloaded. See the *4690 OS: Messages Guide* for more information on this message.

The Image Build Utility creates or appends a log file, ADX_SDT1:ADXTRMLF.DAT, which contains a summary line of information for each file included in either ADXTRM1F.DAT or ADXTRM2F.DAT.

Running ADXRTCCL

You must run ADXRTCCL in order to place the image files created by the Image Build Utility in the terminal load-shrink file, ADX_SPGM:ADXRT8GF.DAT. ADXRTCCL is also automatically invoked after running Apply Software Maintenance or when a terminal, controller, or system configuration change is activated. If ADXRTCCL is not run each time you run the Image Build Utility, subsequent terminal loads will contain prior contents of the image files, if any exist.

Note: If configuration is running, ADXRTCCL cannot be run from the command line.

Notes:

1. The maximum size of the ADXRT8GF.DAT file is limited to 71 MB. ADXRTCCL enforces the 71-MB threshold when building the file. If the result is greater than 71 MB, the X: drive image file is omitted and a null section is substituted in its place. A system event error message is logged and an error message is written to the terminal load shrink log file, ADX_SPGM:ADXRTCCL.LOG. Processing continues with the Y: drive image file with the same checks being performed.
2. The size of the terminal load shrink file, ADXRT8GF.DAT, must not exceed the memory size of any terminal that is connected to the same TCC-configured controller. Therefore, each terminal connected to the same TCC-configured controller must have sufficient memory to contain the entire terminal load shrink file, plus the allocated RAM disks, and an estimated 2 MB of memory for use by the operating system.

Completing the Terminal RAM Disk Load

The final step in performing the terminal RAM disk load is to reload the terminals. During RIPL, the operating system automatically decompresses and writes the files to the root directory of drive X: or Y:.

Note: After loading the terminal, verify the contents of the RAMDISK to ensure that all files that were specified to be preloaded into the terminal's RAM disk were successfully preloaded.

Using a .BAT File

After creating the list files, a .BAT file can be created to automate the creation of the image files and the terminal load-shrink file, ADXRT8GF.DAT. The following example shows all of the steps necessary to build the terminal RAM disk preload file.

Note: Statements are numbered for reference purposes only and do not appear in the .BAT file.

1. ! This BAT file runs the build procedures for Terminal RAM disk Preload
2. XE ADX_IDT1:ADXTRMXF.DAT
3. XE ADX_IDT1:ADXTRMYF.DAT
4. ADXTRM0L -X -Y
5. XE ADX_SDT1:ADXTRMLF.DAT

```
6. PAUSE PRESS ANY KEY TO RUN ADXRTCCL
7. ADXRTCCL
8. XE ADX_SPGM:ADXRTCCL.LOG
9. DIR ADX_SPGM:ADXRT8GF.DAT
```

Line 1 is a comment to state the function of this .BAT file.

Lines 2 and 3 run the XE editor on each of the two input files, which allows you to easily change their contents.

Line 4 rebuilds the preload files for X: and Y: drives.

Line 5 runs the XE editor on the utility log file so that you can determine the file sizes.

Line 6 inserts a pause so that you can exit from the process.

Line 7, ADXRTCCL, rebuilds the ADXRT8GF.DAT file.

Line 8 runs the XE editor on the log file from ADXRTCCL to determine if the terminal RIPL file was rebuilt successfully.

Line 9 shows a directory listing of the ADXRT8GF.DAT file showing its total size.

Chapter 22. System Services Process Utility

This chapter explains how to use the System Services Process Utility.

Using the System Services Process Utility

Included in 4690 OS V5R2 is a new System Services Process as well as the System Services Process Utility. The System Services Process (SSP) is an internal OS function to monitor the RMA, SSH, and INETD processes. These services are optionally started at IPL time based on configuration. The SSP monitor will:

- Log an event when the service starts (as well as other services).
- Log an event if the service ends or is canceled by the user.

The System Service Process Utility is the user interface to control these specific controller system services. The following functions are available to the user:

- Stop a service
- Restart a previously running service
- Query if a service is active

The utility is a command line program, ADXSSP0L.286 and is invoked as:

```
ADXSSP0L ssssssss -X
```

Where:

ssssssss is the service to be processed, one of the following (must be upper case):

- RMA
- SSH
- INETD

-X is the command:

- C = Cancel
- R = Restart (only services that were started at IPL time and then ended can be restarted)
- Q = Query

The utility will display a status message on display and log that message to the log file, ADX_SDT1:ADXSSP0L.LOG.

In addition to the status message, the utility will also exit with a return code from the list shown below.

The ADXSSP0L.286 utility may also be run as a background application from RCP in which case the status line will not be displayed, but will be written to the log file. RCP will get the utility return code and include it in the RCP status file.

Note: When run as a background process, the BACKGROUND APPLICATION CONTROL screen (CSO5S001) will indicate a STATUS of ABENDED for any non-zero return code, including the positive return codes which are used to specify the state of the service in response to a query command. Check the ADX_SDT1:ADXSSP0L.LOG file or RCP log file for the actual completion code of the utility.

The format of the status line is:

TIMESTAMP [UUUUUUUUU]<SSSSSSSS> -c RRRRRRRR

where:

- *UUUUUUUUU* is current user
- *SSSSSSSSS* is the service
- *c* is the command
- *RRRRRRRRR* is the return code (hexadecimal)

Return codes

0 = success

1 (0x00000001) = starting at IPL time (response to query command)

2 (0x00000002) = running (response to query command)

3 (0x00000003) = stopped (response to query command)

4 (0x00000004) = ended (service ended programmatically)

-1 (0xFFFFFFFF) = bad command

-2 (0xFFFFFFF0) = no services found

-3 (0xFFFFFFF1) = specified service not found

-4 (0xFFFFFFF2) = service table full

-5 (0xFFFFFFF3) = service was not stopped (response to restart command)

-6 (0xFFFFFFF4) = service was already running (response to restart command)

-64 (0xFFFFFFF0) = too few command line arguments

-65 (0xFFFFFFF1) = invalid service requested

-66 (0xFFFFFFF2) = invalid command requested

-67 (0xFFFFFFF3) = request pipe open error

-68 (0xFFFFFFF4) = response pipe open error

-69 (0xFFFFFFF5) = request pipe read error

-70 (0xFFFFFFF6) = response pipe read error

-71 (0xFFFFFFF7) = request pipe write error

-72 (0xFFFFFFF8) = response pipe write error

-73 (0xFFFFFFF9) = no services found (on query command)

Note: Errors -64 through -73 are from the utility and the others are from the OS monitoring process.

Chapter 23. Controller ASM preload

Overview

The Apply Software Maintenance function of OS4690 allows product files to be extracted to and maintained on the C: and D: disk drives on 4690. Products that use Java6 in enhanced mode may require that files be placed on the F: drive to be available for the JVM. Although it's possible to use various other methods to copy files, the process is not well integrated into the software maintenance process. The "Preload F" program (ADXPLDPF) allows program products to copy files to the F: drive and perform other operations as part of the ASM process. Note that the program is not specifically restricted to the F: drive or to preloading files nor does it have to run only during ASM processing.

An overview of the steps the OS takes to enable this is as follows:

1. During controller IPL, the ADXPLDPF program is run at step W555:F92. This happens after Java 2 has been updated due to OS ASM (if needed). This step happens before step W555:X92, which extracts and enables OS extensions (such as MBrowser), and before step W555:093 (the stage 3 IPL command processor). This should allow preload command files to run Java2 if needed and to copy the files required by extensions to the F: drive before the extensions are initialized.
2. The ADXPLDPF program searches for directories in C:\ with the name ADX_?PGM (where ? is any letter). Each directory is searched for command files. The directories ADX_SPGM, ADX_IPGM, and ADX_UPGM are searched first (in that order); the other directories are searched in alphabetical order by directory name.
3. Command files have names of the format ADXC?T?Z.DAT. Within a given directory, the command files are processed in alphabetical order with one exception. With the ADX_SPGM directory, the file ADXCSTSZ.DAT (which is reserved for OS use) will be processed first if it exists.
4. Each command file found is then processed individually:
 - a. A severity message (W725) is logged indicating processing has started.
 - b. The file is parsed to ensure it's valid. All sections are parsed, including disabled conditional sections (see below).
 - c. If the file is not valid an error (W727) is logged.
 - d. If the command file is valid, all enabled conditional sections (trigger blocks) are executed. Processing does not stop if there are errors.
 - e. If there were errors during command file processing, an error message (W727) is logged.
 - f. If there were no errors during command file processing, a success message (W726) is logged.
5. The results of the processing is written to ADX_SPGM:ADXPLDPF.LOG

To use this functionality, a product should create a file named ADXC?T?Z.DAT (the command file) that contains commands used to extract files to the F: drive. The 5th and 7th letters of the filename should correspond to the same letters in the name of the product's product control file. (However, there is no requirement that the corresponding product control file exist). This file should be shipped with the product along with the other files and should be marked to be distributed to all

controllers that need to run the commands (compound or mirrored as desired). The file should be placed in one of the ADX_?PGM directories in the root of the C:\ drive.

The format of the command file (ADXC?T?Z.DAT) is as follows:

- Leading and trailing spaces on lines are ignored.
- Lines starting with # are comments and ignored. A # that is not at the beginning of a line is not a comment indicator.
- Blank lines are ignored.
- Any non-blank, non-comment line is considered a command.
- The entire file is processed before any commands are executed. Any errors (invalid commands, etc) will cause processing for that file to stop and an error to be logged. Any remaining command files will continue to be processed.
- Once commands in a file begin to be processed, execution of the remaining commands in that file will continue even if an earlier command fails. This is done in an attempt to get as much as possible set up even in the case of failures.
- The commands, flags, and command parameters are case sensitive.

Parameter processing for commands is identical to how arguments are broken up to be passed to 32-bit C/C++ programs in 4690. Specifically:

- Arguments are delimited by tab or space characters.
- Double quotes must be used to put spaces into an argument. To do this, begin and end the argument with double quotes.
- In order to put double quotes or a backslash within an argument, they must be escaped with the backslash character. Only \ is recognized as an escape sequence. For any other use of the backslash, both characters are added.
- Non-escaped backslashes within an argument are not made part of the argument; they are only used to determine the end of the argument. Specifically, the “end” of an argument is indicated by an even number of non-escaped double quotes followed by a tab or space character.

Some commands include filenames.

Notes:

1. 4690 logical names and other device letters are allowed in filenames, however the use of remote files (LAN or NFS) could negatively impact IPL speed.
2. It is currently possible to read from files on remote nodes (e.g. files containing a node or role name followed by “:.”), however writing to files on remote nodes is currently disallowed.
3. File and directory names on the F: drive are case sensitive.
4. During IPL, the current drive is C:. Non-absolute filenames are considered relative to the root directory of the drive on which they're located.
5. Filenames cannot contain semicolons.

Command Processing

By default all commands in the command file are executed; however, the “trigger” command can be used to indicate specific conditions upon which to execute the commands following it. The most common trigger condition will cause command execution when the product control file changes, thus copying files to F: when a new version of the product is installed as well as extracting old files to F: if maintenance is cancelled. However, the trigger file does not always have to be the

product control file; see the “Trigger – Control command execution” for more information. The trigger file must exist on all machines that the command file is on to avoid errors. A given trigger file may be used more than once in a given command file and will evaluate to the same condition in each command.

In order to determine when a trigger file changes, a hash of the trigger file's contents is stored in the file `ADX_SPGM:ADXPLDPF.$CC` where `CC` is the current controller id. Once a given command file is read in and executed with no errors, the state file is updated so the commands are not triggered on a subsequent IPL. This means that multiple command files may depend on the same trigger file; commands from both files will run when the trigger file changes. It's considered an error if a trigger file is missing. It is suggested that a file used as a trigger file be distributed, but this is not a requirement. Files matching the pattern `ADX_SPGM:ADXPLDPF.$??` will be excluded from a LAN disk rebuild. This will cause all “changed” trigger conditions to evaluate to true when the rebuilt machine is first loaded. In addition, if a command file is deleted and the system rebooted, all information related to trigger conditions for that command file is removed; thus if the command file is later restored to the system, all “changed” trigger conditions will evaluate to true.

When a search “spec” or specification is mentioned below, this means that wildcard characters are allowed in the string. The characters `*` and `?` are wildcard characters where `*` matches any number characters (including zero) and `?` matches any character. For most commands (such as `delete` and `rmdir`), the wildcard characters can only be given in the last path in a filename and thus the wildcards are only used to match the filename. The extraction of archive files is different. In this case, the search spec matches the entire path stored in the archive.

Most commands allow optional flag parameters before the main parameter (file name, directory, or program name). Each of these commands also supports an additional flag (`--`). This flag indicates that flag processing is complete and that the next parameter is not a flag. The `--` flag is required when the main parameter starts with a dash. Some commands (such as `mkdir`) do not currently have any other flags, however the `--` flag is still required (if the parameter starts with a dash) in order to support any future changes. Currently only `trigger` and `msg` do not support `--`, primarily because their initial parameter can never start with a dash.

Available commands

Trigger – Control command execution

Syntax:

`trigger condition ...`

This command determines whether the commands following it are executed. The trigger command should be followed by a list of one or more conditions. If all of the conditions are true, then all commands following the trigger command (up to the next trigger command or the end of the file) are executed. The following conditions are available:

- `changed=fn` – This command sets the trigger file name and is true when the contents of the trigger file changes (see above). Some special values of `fn` are:
 - `*pcf` – Represents the name of the product control file. This is only valid when the source filename has the name `ADXC?T?Z.DAT` and is located in one of the `ADX_?PGM` directories. The name of the product control file is generated

using the 5th and 7th letters of the command filename; it is assumed to be in the same directory as the command file.

- *self – The trigger file is the same as the current command file.
- mode=[cle] - Only run if OS mode matches the given value.
 - c – classic
 - e – enhanced
- role=[list...] - Only run if machine role matches at least one in the list. If more than one role is needed, separate them with commas. For example, the filter “role=m,am” will cause commands to be run on the master and alternate master on both classic and enhanced controllers. The list of role strings is:
 - cs – Configured subordinate on a LAN system
 - cafs – Configured alternate file server
 - cfs – Configured file server
 - cam – Configured alternate master
 - cm – Configured master
- true – This condition is always true. It can be used to always trigger the execution of a set of commands.
- isfile=path – This condition returns true if a file with the given path exists.
- isdir=path – This condition returns true if a directory with the given path exists. Check for the root directory of a drive (e.g. f:/) to determine if a drive exists)
- !condition – Adding “!” in front of a condition negates it.

The default trigger condition set is “file=*pcf mode=e role=all”. This condition will cause the commands in the file to be executed on all enhanced mode controllers when the contents of the corresponding product control file changes.

Extract – Extract an archive file

Syntax:

```
extract [flags] archiveFn targetDir [extractSpec ...]
```

Extract the contents of the archive file to the specified target directory. The directory and any parent directories are created if needed. The file to extract must be in zip format. By default all files in the archive are extracted, however one or more extraction specifications may be provided after the target directory. In this case only the file(s) matching these specs are extracted. Wildcards are allowed in the file specifications, however note that the spec is used to match the entire path. Valid flag:

- -X excludeSpec – This specifies a list of file specifications indicating files to not extract. This flag can be repeated multiple times to add more exclusions to the list.

Delete – Delete file(s)

Syntax:

```
delete [flags] searchSpec ...
```

Delete the file(s) matching the search specification; wildcards are allowed. Flags:

- -c – Respect case in when matching file names to the search specification. Normally case is ignored.
- -r – Delete file(s) recursively

- -rallowroot – Same as the -r flag, but will allow for recursive deletion of files starting at the root directory. If the directory portion of a search spec is the root directory (e.g. “q:\test.dat”) then -rallowroot is required.

Rmdir – Remove a directory

Syntax:

```
rmdir [flags] searchSpec ...
```

Remove the directory(ies) matching the search specification; wildcards are allowed. Unless the -r flag is given, the command will fail if the directory is not empty. Flags:

- -c – Respect case in when matching directory names to the search specification. Normally case is ignored.
- -r – Delete the directory(s) and all files/directories within them recursively.
- -rallowroot – Same as the -r flag, but will allow for recursive deletion of directories starting at the root directory. This is only required if the directory portion of a search spec is the root directory and the filename portion contains wildcard characters. For example, removing “q:\tmp” is allowed with just the “-r” flag, but removing q:\tmp* or q:* requires -rallowroot.

Copy – Copy file(s)

Syntax:

```
copy [flags] srcFn [destFn | destDir(:|\|/)]
```

Copy files from one location to another. The destination can be a filename or a directory name (a directory name must end in a slash, backslash, or colon). If the target directory doesn't exist, the directory (and any parent directories) will be created. The srcFn can contain wildcards; in this case the destination must be a directory. Flags:

- -r – Copy files recursively.
- -c – Respect case when matching file names to the search specification. Normally case is ignored.

Msg - Display messages

Syntax:

```
msg [flags] [message string]
```

When ADXPLDPF runs during IPL, it displays messages on two lines of the screen. By default the command being executed from the file is displayed on the first line and any additional useful data is displayed on the second line. This command can be used to replace either of the two lines with custom text. If the message is not given or is an empty string, the default message will be displayed (as described above). To prevent any messages from being displayed, provide a message consisting of one or more spaces to the msg command. Screen messages also appear in the log file to allow for debugging. Note that the messages within a trigger block are only in effect until the end of the trigger block. Messages specified before the first trigger block will stay in effect for the entire file (unless overridden in a trigger block). Flags:

- -1 – Set message for line 1
- -2 – Set message for line 2
- -L – Left justify the message

Mkdir – Create a directory

Syntax:

```
mkdir [flags] dir ...
```

Create one or more directories (and any required parent directories). The command is ignored if the given directory already exists. Flags:

- (none)

Run – Run a program

Syntax:

```
run [-i[p]] pgmname [pgmargs ...]
```

Run a program, passing the given arguments. The standard output and standard error from the program is captured and placed in the ADXPLDPF log file. Unless changed by a flag (see below) any non-0 return code from the program is considered an error. The full path name and extension of the program must be provided. Flags:

- -i – Ignore all return codes.
- -ip – Ignore positive return codes only. Typically the OS will return negative return codes if it cannot run a program.
- -ts – Timestamp program output. Normally program output is not timestamped (to make it a little more readable). Specifying this option causes the timestamps to be added.
- -uu – Update user. As the program runs and produces output (via stdout or stderr) the last line of output is written to the second line of the boot console.

Sleep – Add a delay

Syntax:

```
sleep timeout[ms]
```

Used to delay command processing for a brief time. The sleep command is intended for use in debugging (can be used with msg to determine what's happening for example). The default timeout is in seconds; add the “ms” suffix to specify a value in milliseconds. The timeout may range from 1 to 60 seconds (1ms to 60000ms) inclusive.

Command Tips

To completely erase a drive, you'll need to run two commands, rmdir and delete. The rmdir command (e.g. “rmdir -rallowroot q:\”) will recursively delete all files and “delete -rallowroot q:\”

Messages and error logging

Three new messages related to this functionality were added. The message text for each message contains the name of the file being processed, thus you will get one message per file. The message doesn't contain enough extra data for the full pathname of the file, so only the base filename will be logged. To help determine which ADX_?PGM directory the file is located in, the extra data for each message will contain the string “LIB=?” where “?” is replaced by 5th letter of the directory name (e.g. 'S' for ADX_SPGM, 'I' for ADX_IPGM, etc). If the filename is not in one of the ADX_?PGM directories, '-' will be used instead. The log file (see below) will contain the full path name.

- W725 – This message will be logged when processing for a command file is started. It will be logged as severity 5.
- W726 – This message will be logged when a command file is successfully processed. This will be logged as a severity 5 message.
- W727 – This message will be logged when there was a problem processing a file (errors in the file, errors extracting, etc). This will be logged as a severity 3

message like Java2 installation failures. The event code will indicate the failure reason. An event code of 0 indicates an ADXPLDPF execution failure. The extra data for the message indicates the line number as well as a return code that may provide more information for support.

The ADXPLDPF command will be run once during the controller boot sequence to process all files (it will be run at W555:F92). The output of this command will be written to ADX_SPGM:ADXPLDPF.LOG. Since it ends in .LOG it will be automatically collected by the Create Problem Analysis Diskette utility.

The ADXPLDPF program can also be run manually to debug command files. As stated above, a command file with a name of the form ADXC?T?Z.DAT which exists in one of the ADX_?PGM directories (on C:) is processed at machine IPL (i.e. it's an IPL command file). To do the same processing as is done during IPL (process the IPL command files in the same order), run the command:

```
ADXPLDPF [flags] *
```

In this case the output will be written to ADX_SPGM:ADXPLDPU.LOG and messages will be logged as they would during IPL. The program may also be run against an individual command file for debugging just that file. To do this run the command:

```
ADXPLDPF [flags] fn
```

Where fn is the name of the file to process. In this case output will go to standard output and messages will not be logged. When running the program against an individual command file, the trigger information database is updated for any "changed" trigger conditions that are processed. If the command file is an IPL command file, the database update may prevent the trigger blocks from running if a machine is subsequently IPLed (unless the trigger files also change). If the test file is not an IPL command file all trigger information related to the file will be removed at the next IPL.

The ADXPLDPF program supports the following flag:

- -? - Display additional help

Chapter 24. Directory Backup and Restore Utility (ADXZUDIR)

Overview

The ADXZUDIR tool is intended for use by store administrators to backup and restore the files on a 4690 system. The backup and restore may be done for individual directories or entire drives. There is also a flag which allows an entire system to be backed up or restored. Because the system will have files in use (or be modifying files) during normal operations, a system backup and restore must be run after booting the 4690 supplemental OS from CD or flash drive.

Files sizes on 4690 are limited to 2GB, however the drives on a 4690 system can hold a lot more data than this. Even with compression, this can be more data than will fit in a 2GB file. Because of this, if the size of the ZIP file is greater than 1GB it will be broken into multiple files, each approximately 1GB in size. The extension of the additional files will have the format .z## where each # is replaced with a decimal digit (i.e. .z00, .z01, etc). If more than 100 additional files are needed, the extension will expand to 3 or more digits (i.e. .z100, .z101). Before compression starts, all files that look like "extra" files (i.e. fn.z## where ## is **two or more** digits) will be removed. Note that if the name of the ZIP file ends in ".zip", the numbered extension (.z##) will replace ".zip". If name of the file does not end in ".zip", the numbered extension is appended to the name of the ZIP file provided. When listing, extracting or testing an archive, all additional files must exist. The only exception is when listing an archive file in manual mode. In this case, the tool will attempt to list the contents of the archive using the information from the ZIP file only.

Note that it's possible to create files on the F:/ drive that contain characters that are not valid in 4690 path names. Some of these characters include the colon (:), backslash (\), and the wildcard characters (? and *). The ADXZUDIR tool will not typically be able to backup these files (or if it does, it will not be able to restore them) depending on the behavior of 4690 when it encounters the invalid character. In addition, some programs like Lotus Expeditor will create symbolic links to process information (files in the linux /proc filesystem) on the F: drive. When Expeditor is not running, these links point to files that do not exist and thus cannot be accessed from 4690. These will not be backed up. The files typically have names like stat and statm.

Due to limitations with CD/DVD drive support in classic mode systems, ADXZUDIR will behave differently than in other cases. Specifically:

- When extracting or testing an archive, ADXZUDIR normally displays the name of the file being extracted or tested as well as the total number of files and percentage complete. When reading archive files from a CD in classic mode, the total number of files and percentage complete is not known until the archive is complete and therefore is not displayed.
- The entire ZIP file must be read from CD/DVD in order to perform the extract, test, and list operations. This can make the list operation (which is usually very rapid) take a lot longer than expected and increase the time required to test/extract the archive files since the ZIP file must be read twice.

To avoid these limitations, copy the entire backup set to the hard disk (including the .ZIP file and any additional .Z## files if the backup exceeds 1Gb) before performing the operation.

General Syntax

The general syntax for ADXZUDIR is:

```
adxzudir [flags] zipFnOrBaseName [...]
```

The flags for ADXZUDIR must come before any other parameters and must start with the dash character ('-'). In general, flags can appear in any order (with any exceptions noted below). The first parameter after the flags indicates the archive filename or base name to use. One of the following "operation" flags must be specified, which indicates the operation you wish to perform. The allowed flags are:

- -c - Create a ZIP archive (create a backup)
- -x - Extract a ZIP archive (restore a backup)
- -l - List the contents of a ZIP archive
- -t - Test the integrity of a ZIP archive.

Additional valid flags are:

- -log:fn - Log messages to the given filename (fn). The file is created if needed and truncated if it exists. By default, output goes to the program's standard output. Note that the log file is not created until all flags and parameters have been successfully processed.
- -loga:fn - Identical to the -log flag, except that if the file exists, messages will be appended to it. (It is not possible to append to log files on UDF formatted media in classic mode.)
- -- - This flag indicates the end of the flags. It may be used if the first parameter (ZIP filename) starts with a dash.
- -auto - Do an automatic backup (see below)

General notes on ADXZUDIR's operation:

- When a ZIP file is being created, file distribution attributes are automatically saved in the file (if the file system supports 4690 file distribution).
- If a ZIP file contains file distribution attributes and is restored to a file system that supports it, the attributes are restored. Distribution attributes are only restored on a LAN system if it is valid to do so (following ADXCSUOL's rules); they are always restored on a non-LAN system or when booting the supplemental OS.
- During restoration, files are distributed immediately if the store controller was booted normally and is configured for LAN mode. This means that as soon as the program ends, the files will have been distributed to other online controllers.
- Empty directories are saved and restored.
- Existing files are overwritten during archive extraction. The timestamps on any successfully restored files will match the timestamps read when the files were saved. The timestamps on newly created directories will match as well. However timestamps on already existing directories (e.g. C:\ADX_SPGM on a CPREPed system) are not updated.

File Exclusion

File exclusion specifications (exclude specs) are used to exclude files from being added to a ZIP file during creation or to prevent files from being extracted. To determine whether to exclude a file, the spec is compared against the entire path that is (or would be) stored in the ZIP file; case is ignored. In an exclude spec, '*' matches any character in a file's path and '?' matches a single character. Slashes

are not special so '*' and '?' will match them like any other character. Drive letters and device names are not stored in a ZIP file and are ignored during the comparison.

ADXZUDIR attempts to canonicalize and resolve a path name so that it will match the name of the file as it is stored in a ZIP file. If an exclude spec is a logical name (e.g. "hosts") it is resolved before being used for comparison. Drive letters and logical names containing colons (such as ADX_SPGM:) are also expanded to generate the final spec. To prevent logical name expansion (if, for example, you wish to exclude a file that is also a logical name such as "path"), enclose the spec in double quotes.

When file exclusion specs are read from a file, each line in the file is treated as a separate spec. White space at the beginning and end of each line is ignored. Any lines starting with '#' are considered to be comments and are ignored. To start a spec with '#' or to include leading or trailing spaces, enclose the spec in double quotes.

In automatic mode (described in the "Automatic Mode" on page 436 section, a set of built in file exclusion lists are used to minimize the size of an archive and ensure the machine boots properly. The exclusion list can be used to tailor the backup to the reason it's being created (i.e. whether it's being used as a full backup, being restored to the same controller or being restored to another controller). The name of the exclusion list may be specified on the command line to override the default one (which is described below). The available lists are as follows. Currently each list is a superset of the preceding one:

- *none - This list excludes nothing and is used by default during extraction. The `adxzudir.xla` file can still be used to exclude user specific files. The *none list typically shouldn't be used when creating an archive that you intend to restore because the *auto and other lists do additional processing to ensure the restored system reboots properly.
- *auto - Copies everything except those files required for successful restoration of a backup. Using this also triggers a terminal load file rebuild after IPL of the restored system (by creating the loadshrink trigger file). This list includes:
 - Enhanced terminal load files (which are automatically rebuilt)
 - Files on F: which are symbolic links
- *backup - Excludes files typically not needed when a backup is restored. This list includes:
 - The files in the *auto list
 - Controller and terminal dump files (if the files exist when a backup is created, a zeroed out 512 byte file will be added to the ZIP as a placeholder to ensure the files exist when the new system reboots)
 - Controller and terminal enhanced mode ZIP files
- *ldr - Exclude the same files as a LAN disk rebuild of C/D would. This list includes:
 - Files in the *backup list
 - The system trace file (an empty 512 byte file will be placed in a backup as a placeholder).
 - SIOAM spool file
 - Performance data file
 - SSH key and identity files
 - IPSEC

- Command line logging files
- Other system specific trigger files
- RMA log files on F: are also excluded

The exclusion list mechanism above was intended to be generic and used both during archive creation and during extraction. The *auto, *backup, and *ldr exclusion lists are intended to be used during archive creation only because they add additional entries to the archive file to accomplish their goal.

Note that the *ldr list doesn't completely emulate a LAN disk rebuild. Since the target machine's controller ID is not known, some ID specific changes are not done. For example, LAN disk rebuild will copy the master's version of the system logical names file (ADX_SPGM:ADXDA??.DAT) if the node being rebuilt doesn't have a node-specific version yet. LAN disk rebuild also skips ADX_IPGM/ADXCCCCF.DAT and changes the node id in ADX_SPGM:ADXLNDF.DAT. ADXZUDIR simply copies files (excluding the ones requested) and doesn't do any node-specific operations.

To add file specifications to the exclusion list used in automatic mode, create the file adxzudir.xla in the same directory that the ZIP archive(s) are located. If the name is not enclosed in double quotes, logical names are expanded above. If the final spec starts with a letter followed by a colon, it will be used to exclude matching files on that drive alone. To exclude files on any drive use '*' instead of a drive letter, e.g. "*:*.*tmp".

The final file exclusion list used during the operation of ADXZUDIR is written to the log file. The exact list isn't documented here because it can and will change as needed.

Automatic Mode

In an automatic backup or restore, the contents of all local disks is backed up or restored. In this case, local disks means specifically, the C:, D:, and F: drives accessible by 4690. Due to limitations on the number of files in a ZIP file, the contents of each drive are saved in a separate set of ZIP archive files. In automatic mode, ADXZUDIR checks each of these drives and will backup and restore files for each available drive. The list and test operations are also supported in automatic mode.

The syntax for an automatic mode command is:

```
adxzudir [flags] -auto[:driveList] baseFn
```

As indicated above, one of the flags must indicate the operation to be performed (-c, -x, -t, or -l).

The baseFn parameter is used to generate the output filenames when creating an archive and for searching for files for the other operations. To generate the ZIP filename for a given drive, first any extension in baseFn is removed then _?.zip is appended to baseFn, where '?' is the (lowercase) drive letter. For example, if baseFn is e:\000\backup, the ZIP filename for the C: drive will be e:\000\backup_c.zip.

The driveList is an optional parameter for the auto flag. If it is provided, it must consist of one or more letters. It is used to generate the list of drives for the operation. Duplicate letters are ignored. The driveList is used in the following ways depending on the operation:

- When doing a backup, it will determine the drives that are backed up and the order in which they are backed up. If any of the drives are not accessible, the operation will fail. The command assumes if the drive list is provided, then all drives are expected to be there.
- When restoring an archive, the drive list will determine which drives are restored and the order in which they are restored. If the backup file for a specified drive is missing, the command will fail. In addition, when driveList is NOT specified, the tool looks for other backups in the same directory. If a ZIP file for a given drive exists but the drive itself does not, the tool will fail (it assumes that you intended to restore a drive but it was not available). Setting the driveList will cause the tool to look for ZIP files for only the given drives.
- By default, when testing or listing files in auto mode, all files named baseFn_?.zip (where ? is a letter) are processed. Setting driveList modifies this behavior so that only files for the listed drives are processed. If the ZIP file for a given drive is missing, the command will fail.

The drive list is most useful when you want to back up a single drive on a system. For example, you could use it if the D: drive is present but you use it for your own backups and don't want to copy those over. The driveList is also useful if you want to copy files from a classic system (which doesn't have an F: drive) to an enhanced system (which does). For example, if ADXZUDIR were run on a classic mode system that contained only a C: drive, it would create a set of backup files for only that drive. To restore that backup to a classic mode system, you would use the flag -auto:c to indicate that the tool should only look for backup files for C: (and not both C: and F:).

In order to properly backup or restore all files on a system, ADXZUDIR must have full access to all files and the files must not be modified while the operation is occurring (otherwise the file data may be in an inconsistent state). To ensure that files are processed, the supplemental OS must be loaded by booting a CD or flash drive containing the supplemental OS image or booting the supplemental OS over the network.

It is possible to use ADXZUDIR when the supplemental OS image is loaded from the hard disk, however doing a full system restore from this environment is not supported. A CPREP may only be done when the non-hard disk version of the supplemental OS is loaded.

Creating a backup

To backup a system, run ADXZUDIR with the -c (create archive) flag:

```
adxzudir -c -auto[:driveList] baseFn [*xlist]
```

Unless driveList is specified, the C:, D:, and F: drives (if they exist) are backed up (i.e. driveList is essentially set to "cdf" with the drives that are not present removed from the list).

As described above, some files will be excluded from the backup automatically. By default the tool will exclude the same files as a LAN disk rebuild (using the *ldr exclusion list) however the list used may be overridden by providing an exclusion list name (*xlist) on the command line. The file adxzudir.xla (if it exists in the same directory as the ZIP file(s) being created) will be used to exclude additional, user specified files from the backup.

Status will be displayed on the screen. In addition, unless overridden with the -log or -loga flags, messages will be logged to the file adxzudir.lzc. This file will be placed in the same directory as the ZIP files.

See the section below, “Restoring a backup” for the additional conditions and steps required if the backup will be used for a full system restore.

Restoring a backup

To restore a backup, run ADXZUDIR with the -x (extract archive) flag:

```
adxzudir -x -auto[:driveList] baseFn [*xlist]
```

Unless driveList is specified, the C:, D:, and F: drives (if they exist) are restored (i.e. driveList is essentially set to “cdf” with the drives that are not present removed from the list).

By default the tool will extract all files from the backup (using the *none exclusion list) however the list used may be overridden by providing an exclusion list name (*xlist) on the command line. The file adxzudir.xla (if it exists in the same directory as the ZIP file(s) being restored) will be used to exclude additional, user specified files from being restored.

Status will be displayed on the screen. In addition, unless overridden with the -log or -loga flags, messages will be logged to the file ?:/adx_spgm/adxzudir.lzx where ‘?’ is the drive letter of first drive backed up (C: by default).

After the restoration is complete, if the log file location hasn’t been overridden, the tool will search the directory where the ZIP files are located. If the file adxzudir.lzc is found in that directory, it will be copied to the same location as the restore log file.

See the section below, “Testing a backup” for the additional conditions and steps required to restore a system.

Testing a backup

To test the validity of a backup, run ADXZUDIR with the -t (test archive) flag:

```
adxzudir -t -auto[:driveList] baseFn
```

Unless driveList is specified, all files matching the pattern baseFn_?.zip, where ‘?’ is a drive letter) will be processed. Each file will be tested in turn.

Status and any error messages will be printed to the screen.

Listing the contents of a backup

To list the contents of the ZIP files in a backup, run ADXZUDIR with the -l (list archive) flag:

```
adxzudir -l -auto[:driveList] baseFn
```

Unless driveList is specified, all files matching the pattern baseFn_?.zip, where ‘?’ is a drive letter) will be processed. Each file will be listed in turn.

Status and any error messages will be printed to the screen.

Manual Mode

Manual mode is used when you wish to backup or restore a single drive or directory or test or list a single ZIP archive.

The syntax for a manual mode command is:

```
adxzudir [flags] zipFn [directory]
```

As indicated above, one of the flags must indicate the operation to be performed (-c, -x, -t, or -l).

The name of the ZIP file comes immediately after the last flag. The filename is used as-is (the .zip extension is NOT added). The directory parameter is only allowed and is required when creating or extracting an archive file. No additional parameters are allowed for testing or listing archive contents.

In manual mode, status and error messages are displayed on the screen (i.e. written to the program's standard output stream). Use the -log or -loga flags to log error messages to a file instead.

Creating a ZIP archive

To create an archive, run ADXZUDIR with the -c (create archive) flag:

```
adxzudir [flags] -c zipFn directory
```

When creating a ZIP archive, the program recursively compresses all files in the given directory and adds them to the archive.

Extracting a ZIP archive

To extract files from an archive, run ADXZUDIR with the -x (extract archive) flag:

```
adxzudir [flags] -x zipFn directory
```

During extraction, all files/directories in the archive are extracted to the given target directory, replacing any existing files. The target directory is created if it doesn't exist.

Testing a ZIP archive

To test the integrity of files in an archive, run ADXZUDIR with the -t (test archive) flag:

```
adxzudir [flags] -t zipFn
```

This ensures all parts of an archive exist and tests the integrity of all files in an archive without extracting anything. This ensures the decompressed contents of each entry matches the CRC values stored in the archive.

Listing the contents of a ZIP archive

To list the contents of an archive, run ADXZUDIR with the -l (list archive) flag:

```
adxzudir [flags] -l zipFn
```

Information about each file and directory in the archive is displayed.

Using a response file

For running in an environment where command line length is limited (such as a background task) a response file may also be used to specify arguments. To use a response file with ADXZUDIR, start it with the command:

```
adxzudir @rspFn
```

where rspFn is the name of the response file. The response file should contain all flags and parameters to ADXZUDIR on the first line of the file. Any subsequent lines are ignored.

When using a response file, only a single argument to ADXZUDIR is allowed. If there is more than one argument, the command line is processed normally. If ADXZUDIR is launched from the background applications screen the first parameter (BACKGRND) is automatically ignored.

Creating an Image of a 4690 Store Controller

The following process can be used to create an image of the store controller. The image can later be used to recreate a replica of the system using a USB flash drive (memory key). The size of the flash drive will depend on the amount of data being backed up; a 4GB or larger drive is recommended.

1. Boot the store controller with the Supplemental OS CD that matches the OS level on the controller. (Classic Systems must use a classic CD, Enhanced Systems must use an Enhanced CD) Matching OS level is important. Loading the supplemental OS from the network, flash drive, or hard disk is also possible.
2. Select the option to boot the Supplemental OS
3. Select main menu option 5 to open a command window.
4. Attach the flash drive to the store controller.
5. Run the following command to backup the system:

```
adxzudir -c -auto e:\000\xxx
```

The above command emulates a LAN disk rebuild between the controllers (but in addition back up the contents of the F: drive). See above for a list of files excluded. If you wish to copy more files (for example to backup a system for restoration to the same system), use the backup exclusion list instead:

```
adxzudir -c -auto e:\000\xxx *backup
```

6. A log file named adxzudir.lzc is written to the same directory where the archive file(s) are stored.
7. Optionally, to test integrity of the archive to ensure it was written to the media correctly, run the following command:

```
adxzudir -t -auto e:\000\xxx
```
8. When this command completes remove the flash drive and the CD
9. Reboot the controller

The archive creation command above creates one or more ZIP files containing the contents of the local system drives (C:, D:, and F:) if they exist. The filename provided is used to determine the directory where the ZIP files are located as well as the filenames used. The filename portion of the provided path (represented by xxx above) is used to generate the filenames. For each drive compressed, a name of the format xxx_?.zip will be created, where '?' represents the drive letter. Any extension in the filename (anything following the first period) will be ignored.

When compression starts in auto mode, archive files for all drive letters (*xxx_?.z**) are removed from the directory to avoid any possibility of a mixed backup set.

If more than 1GB of compressed data is required to backup the files on a drive, additional files with the extension *.z##* will be created (where # is a digit from 0-9). If you wish to transfer the files on the flash drive to another location for backup or duplication, then all files must be copied. For example, in the above scenario you would want to copy *xxx_*.z**.

Restoring an image of a 4690 Controller

To restore a system from a set of backup files perform the following steps. When extracting files, replace the archive filename and directory (e.g. *e:\000\xxx*) with the same name you chose when running the backup procedure (or to the new location of the files on the backup media).

Files for all existing drives (C:, D:, F:) must be present in the given directory and no extra archive files (using the same base name and naming conventions used by ADXZUDIR) may be present. This check is done to ensure that everything you intended to restore is actually restored (to ensure you didn't forget to run DPREP for example). The *driveList* parameter (see above) can be used to override these checks and allow you to restore specific drives only.

1. Boot the store controller with the Supplemental OS CD that matches the OS level you will be restoring to the system. It is important that the image and the supplemental are the same level. Loading the supplemental OS from the LAN, or flash drive is also possible, however do not load the supplemental OS from the hard disk as the CPREP will not function as expected.
2. Select main menu option **5** to open a command window.
3. Run CPREP to format the C: disk. This will create an empty F: drive on enhanced mode as well.
4. If you will be restoring files on the D: drive, run DPREP as well.
5. Set the store controller node ID by running the following command:

```
adxnsx01 xx
```

where *xx* is the NODE ID

6. Insert a flash drive containing the files from the backup process.
7. To restore files, run the following command:

```
adxzudir -x -auto e:\000\xxx
```
8. The extraction log file will be written to *adx_spgm\adxzudir.lzx* on the first drive restored (C:).
9. When the extraction completes remove the CD and the flash drive.
10. Reboot the store controller.
11. You may receive the **W900** and **W901** messages. Press **F1** and **F2** at these messages.
12. Sign on to the store controller.
13. 13. If you received the **W900** and **W901** message, do the following to activate this controller:
 - a. Press **Alt-SysReq** then **C** to display the store control functions screen
 - b. Select option **4** (Multiple Controller Functions)
 - c. Select option **1** (Activate Master) and answer **Y** when prompted
 - d. Select option **2** (Activate File Server) and answer **Y** when prompted

- | 14. Verify all the boot completes and store controller functions as expected.
- | 15. Verify a terminal can load successfully if that was part of your original
- | configuration.

Part 3. Multiple Controller Feature Network

Chapter 25. Using the Multiple Controller Feature

This chapter describes local area networks (LANs) that are using the Multiple Controller Feature (MCF). The MCF operates on Ethernet LANs.

A LAN provides the hardware and access protocols necessary for connecting and communicating between multiple controllers.

The MCF is the software necessary for sharing functions and some I/O devices between store controllers and for updating and distributing files on the LAN. MCF is required if LU 6.2 communication is being used.

Data Distribution Application (DDA) is included in the MCF. DDA provides file synchronization during normal operations and file reconciliation when a disabled store controller is brought back online. This facility creates and updates multiple versions of files over the LAN.

DDA runs all the time on all the store controllers that use MCF. As the prime version of a distributed file is updated, renamed, or deleted, these changes are distributed to all store controllers that have a version of that file. These changes are distributed immediately if the file mode attribute is *per update*. These changes are distributed when the file is closed if the file mode attribute is *at close*.

Do not confuse the MCF with controller backup as they are not related. For information about store controller backup, see Chapter 31, "Store controller backup," on page 493.

Advantages of the Multiple Controller Feature

MCF offers several advantages where multiple store controllers are required. The advantages are:

- Improved system control as a result of having one store controller manage all the other store controllers on the network.
- The capability of backing up files and increasing system availability through the use of file redundancy. Image versions of files that are kept current by the DDA can be used when the prime version is unavailable.
- Increased computing power through the use of multiple store controllers that process and exchange data across the LAN (MCF Network).
- Increased system file capacity provided by multiple store controllers that have multiple hard disk drives.

Concepts and Elements of the Multiple Controller Feature

MCF introduces many new concepts and elements. Some of the main concepts that characterize MCF are:

- On a LAN (MCF Network), *distributed files* are assigned two *file distribution attributes*. The MCF uses the *file type* attribute to determine where the file should be distributed on the network. The MCF uses the *file mode* attribute to tell when a file is distributed on the network.
- Store controllers are assigned *roles*. These roles determine where redundant files are placed and which store controller assumes control for another in the event of store controller failure.

- Each store controller is assigned a *node ID*. The node ID is used when addressing store controllers on the network.
- *Logical file names* allow needed flexibility in locating files when a store controller is disabled. The system automatically changes the file destination to the store controller that is assuming control for the disabled store controller.

The following sections describe these concepts and other elements of MCF.

Distributed Files

A *distributed file* is a file that has multiple copies on the LAN (MCF Network). There is always a *prime version* and one or more *image versions*. Image versions are copies of the prime versions.

An application can change only the prime version of a file. An image version of a file can be changed only by the operating system code. If an application attempts to change an image version file, it always receives an error code.

If an application attempts to change a prime version of a file and that file is unavailable for any reason (for example, it is on another store controller that is not currently in operation), it will get a “node not found” error.

If the operating system attempts to update an image version file, because an application has updated a prime version file, and the image version is unavailable, the update request is not lost. Instead, it is stored in a file that is called the Exception Log, where it will be retrieved and updated when the disabled store controller is brought back into operation.

File Distribution Attributes

On a LAN (MCF Network), all distributed files have two file attributes: *file type* and *file mode*. The *file type* attribute tells *where* a file is distributed. The *file mode* attribute tells *when* a file is distributed. These file attributes are stored in each file's directory where the file name, length, and location are stored. The distribution attributes are not part of the file, but rather part of the store controller's file directory. A file's attributes can be copied to and from a diskette or hard disk drive by using the COPY command or the BACKUP and RESTORE commands when in Command Mode.

A file attribute can not be copied for those commands when using a DVD media. Use the Optical Drive Utility to copy to a DVD media and recover its distribution attribute from a hard disk. See “Disk identification and contents” on page 398.

Attention: Toshiba advises against using DOS to copy distributed files. This action can result in the loss of distribution attributes.

File Type Attribute

The file type attribute indicates which store controller a file is to be distributed to. Your files can be one of three file types: *local*, *mirrored*, or *compound*. The file type is defined when a file is created. A file type can be modified at any time by using the Distributed File Utility (DFU). See Chapter 28, “Using the Distributed File Utility,” on page 465 for more information.

Local File

A *local file* resides on only one store controller and is not distributed on the LAN (MCF Network) system. When a local file is updated, other store controllers on the network are not affected by the change.

A local file on one store controller can have the same file name as a file on another store controller, but the two files might be entirely different from each other. Even though the files have the same name, they are not updated from store controller to store controller when changes are made.

Mirrored File

A *mirrored file* is one that exists on only two store controllers.

Mirrored files can be either system-mirrored files or application-mirrored files. The file has two file versions on the system: a *prime* version and an *image* version. The application has read and write access to the prime version and read-only access to the image version. The image version is a copy of the prime version that is distributed by the operating system.

The prime version of a *system*-mirrored file resides on the master store controller while the image version resides on the alternate master store controller. See “Store Controller Roles” on page 448 for more information.

The prime version of an *application*-mirrored file resides on the file server controller while the image version is on the alternate file server controller.

Compound File

A *compound file* is distributed to all store controllers except ineligible subordinate store controllers. Compound files are more common than mirrored files.

There are *system* compound files, and *application* compound files. System compound files are distributed to all store controllers by the master store controller. Application compound files are also distributed by the master store controller.

System-compound files and application-compound files are distributed to all store controllers.

A compound file has a prime version, but can also have one or more image versions. A good example of an application compound file is an item record file.

The prime version of each compound file is stored on the master store controller. An image version exists on all other store controllers in the network.

If your system has only two store controllers, all distributed files appear on the same two store controllers so the distinction between mirrored and compound files might not seem very significant. However, if you should later add another store controller to your network, the proper choice of distribution can have an effect on overall system performance.

File Mode Attribute

Distributed files must be synchronized. That is, when a change is made to a prime version of a distributed file, all of its image versions on the network must also be changed. The *file mode* attribute tells when a file is to be distributed. Files are distributed according to one of the following methods:

- When a record in the prime version is written or deleted

- When a file is closed

Distributing a record every time a record is written or deleted is called per update (sometimes referred to as record mode). Distributing a file when the file is closed is called at close. The file mode attribute is specified when a file is created.

Per Update

With the per update file mode method, when a change is made to a file record, the record (not the whole file) is distributed to all eligible store controllers on the LAN (MCF Network) system immediately. Both mirrored and compound files can be distributed per update. The choice depends on whether there is a need to keep all image versions of a file current. A transaction summary log would normally be a mirrored file updated using the per update method. An item record file would normally be a compound file updated using the per update method.

At Close

With this method, image version files are updated across the network whenever the prime version file is closed, and the entire file is sent across the network to all store controllers that have a copy of it. An example of such a file is an accounting totals file.

At close also means an application file is distributed whenever a TCLOSE is issued. See the *4680 BASIC: Language Reference* for a discussion of the TCLOSE instruction.

Nested Subdirectories

The maximum length of the complete path name (extended path name) that DDA supports is 24 characters. Because of this, the number of levels of subdirectories that DDA supports when using nested subdirectories is also limited. If you issue a command that uses the extended path name, and the extended path name exceeds the 24 character limit, the function will be completed on the local node but will not be distributed by DDA to the other store controllers. Instead, DDA will log the W920 message.

Store Controller Roles

On a LAN (MCF Network) system, each store controller is assigned one or more *roles*. Each role indicates a type of responsibility that a store controller has on the network.

There are five possible store controller roles:

- Master
- Alternate master
- File server
- Alternate file server
- Subordinate

Store controller roles are assigned through the configuration process. A store controller can be assigned more than one role.

In some cases, a store controller is configured for one role, but it can assume another role. In that instance, the store controller is said to be acting the role. For example, when a store controller that is configured as the alternate master store controller assumes control for a disabled store controller that is configured as the

master store controller, the alternate master store controller is said to be the acting master store controller. This concept will become clearer as you learn more about store controller roles.

Figure 24 on page 450 helps make the roles clearer. Study this figure when you have finished reading about store controller roles. The following sections describe the store controller roles. Note that system files reside in ADX_?PGM, ADX_?MNT, ADX_?BUL, and ADX_SDT1. Files that reside in all other directories, including the root directory, are application files.

Master Store Controller

The master store controller controls the prime versions of system-mirrored files, and all (system and application) compound files. All configuration must be performed at this store controller. Most of the user interface with the system is performed at the master store controller.

The master store controller should be chosen for its physical location, its I/O options, and its ability to process data faster to improve store operations.

Alternate Master Store Controller

The alternate master store controller replaces the master store controller if it becomes disabled or deactivated. This process is not automatic, however. You must activate the alternate master store controller to replace the role of master store controller. When activated as master store controller, the alternate master store controller updates and distributes files as if it was the master store controller until you deactivate it and reactivate the configured master store controller. Image versions of all compound files and system-mirrored files reside on the alternate master store controller.

Subordinate Store Controller

All store controllers that are neither the master store controller nor the alternate master store controller are considered subordinates. This includes the file server store controller and the alternate file server store controller (unless they are also assigned to be the master store controller or alternate master store controller).

You do not specify what type of file the store controller should receive. All subordinate store controllers receive all compound files.

File Server

The file server store controller controls and updates the prime version of application-mirrored files. Normally, applications using the mirrored files would execute in this store controller.

Alternate File Server

The alternate file server store controller maintains image versions of all application-mirrored files that are distributed by the file server store controller. You can activate this controller to take over as file server store controller if the configured file server store controller is disabled or deactivated.

Combining Store Controller Roles

Some store controller roles can be performed by the same store controller. Most commonly, the master store controller and the file server store controller roles are assigned to the same store controller and the alternate master store controller, and

alternate file server store controller roles are assigned to another store controller. Toshiba recommends that the two store controller roles be assigned to the same store controller, and that the two alternate roles be assigned to the same store controller.

Some roles cannot be combined. For example, the master store controller and alternate master store controller roles cannot be assigned to the same store controller. Also, the file server store controller and alternate file server store controller roles cannot be assigned to the same store controller.

For the types of files stored on each store controller, see Figure 24.

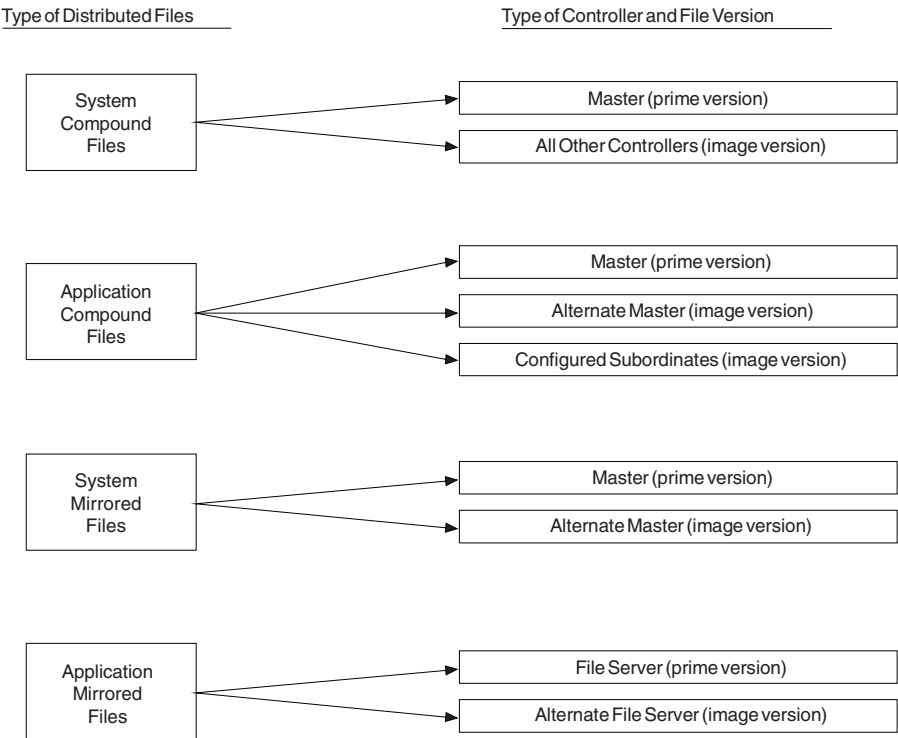


Figure 24. Types of Files Stored on Each Store Controller

Nodes

A *node* is a point of physical or logical connection to a network. Every store controller in a multiple-controller system is a node. Non-4690 store controllers or other PCs that are attached to the network are also considered nodes.

Your 4683, 4693, 4694, SurePOS 300/700 and TCxWave 6140 Series terminals are not considered nodes. Terminal applications can, however, access files and pipes on any store controller on the network.

Node IDs

Every store controller node must have a two-character identification by which it can be addressed by other nodes. This is its *node ID* (sometimes called the store controller ID). These IDs are specified during the installation process and assigned when the store controller is first IPLed. Each store controller is assigned alphabetic characters in the range of CC through ZZ, which are unique IDs. A and B cannot be used.

The following node IDs are special IDs that are reserved for specific store controllers. These IDs are assigned automatically by the system to the acting store controller and cannot be used when configuring the system.

REQTEXT	Reserved for
AA	Acting master
AB	Acting alternate master
AC	Acting file server
AD	Acting alternate file server

These node IDs readily identify the store controller that is performing a given role. For example, if the configured master store controller is disabled and the alternate master store controller becomes the acting master store controller, then using the AA node ID to refer to a file on the acting master store controller directs the access to the alternate master store controller that is the acting master store controller.

In other words, the special reserved node IDs relate to a controller *role* rather than to a specific store controller. In contrast, any of the IDs CC to ZZ relate to the specific store controller itself. If you assign CC to your master store controller, a reference to CC always tries to access that specific store controller even if it is disabled and another acting master store controller has been activated.

Node Names

To access a file on another store controller, your application must use a store controller *node name* when opening the file. For more information, see the *4690 OS: Programming Guide*.

Node names can be used for pipes, but pipe routing services should be used for extensive pipe use between store controllers for functions like Host Credit.

Every store controller node name has the form ADXLXxyN::, where xy is the store controller ID. Notice that node names end with a double colon. A single colon indicates physical devices on the same store controller, such as C:, D:, LPT1:, COM1:, and so on. The double colon indicates different nodes on the network.

Note: The NetBIOS protocol is not supported for the following drives: E:, F:, M:, P: and NFS drives. That is, ADXLXDDN::M: and ADXLXDDN::E: for example, are not supported in commands (where DD is the controller ID of the controller).

Logical File Names

A *logical file name* is a way to abbreviate a file name. The system expands this abbreviated file name to its full file name when accessing the file.

Logical file names are used extensively on a LAN (MCF Network) system. For more information on logical names, see the *4690 OS: Programming Guide*.

Logical Names Table

The operating system uses a single table of logical names for its files, store controller node names, subdirectories, and most I/O devices such as the printers and the displays. For a description of this table, see the *4690 OS: Programming Guide*.

Distribution Exception Logs

Two distribution exception logs exist on the LAN (MCF Network) system. Each log records unsuccessful attempts to distribute the prime version of file updates, deletes, and renames on the LAN to image version files.

How the Exception Logs are Used

On a LAN (MCF Network) system, if an attempt is made to distribute a change to a file on another store controller, and the change cannot be made for some reason, an entry is made to the appropriate distribution exception log. The logs record the type of exception, the error code that is causing the exception, the time and date, the store controller ID, and the complete file name.

Exception log tracking

Data distribution maintains data integrity on multi-controller systems by logging distribution errors in exception logs. One exception log exists for the master store controller (ADX_SDT1:ADXLNDMF.DAT), and another exists for the file server store controller (ADX_SDT1:ADXLNDSF.DAT).

Increased size of exception logs

Previous to 4690 OS V5R2, exception logs were limited to a maximum of 819 entries. 4690 OS V5R2 has increased the size of the exception log to four times the previous size, or approximately 3276 entries.

Two types of exception log entries are generated based on the missed operations: *delete* and *update*. The exception log entries are classified as either *full* or *partial* entries. A full entry is reconciled by distributing the entire file. A partial entry is reconciled by distributing only the part of the file that was not distributed previously (either a 32-KB block for a nonkeyed file or the missed record for a keyed file). The operating system keeps the data to be distributed for partial entries in files in the ADX_SDT1 subdirectory. The amount of disk space that is used for partial entries is dictated by the number of keyed files that are tracked, and the number of records or 32-KB block pointers that are placed in the partial exception logs in ADX_SDT1.

When an error occurs, an entry is listed in the exception log for each store controller that missed the update information. Full exception log entries are created for distribute at close files and for missed creates, deletes, and renames to distribute per update files. If an error occurs tracking a partial exception log entry, the partial exception log entry will be changed to a full exception log entry. Partial exception log entries are only for distribute per update files that have successfully been created on an image version store controller, but have missed an update.

Partial exception log tracking differs if the file is keyed or nonkeyed. For a keyed file, the file must be larger than 32 768 bytes for partial exception logging to occur. During partial exception tracking for a keyed file, the individual records for each node are stored in the partial exception log data files in ADX_SDT1. For example, if node DD misses records 10 and 11, but node EE receives record 10 and misses record 11, node DD reconciles records 10 and 11 during reconciliation, and EE reconciles only record 11. If a large number of updates to a keyed file make the procedure more efficient for the operating system to reconcile the entire file, the partial exception log entries are changed to full exception log entries, and the entire file is reconciled instead of individual records.

If the keyed file is 32 768 bytes or smaller, the file is tracked on a full exception log entry basis.

If the file is nonkeyed, partial exception log tracking occurs regardless of the file's size. Rather than maintaining a list of single operations that are missed, the file is mapped into 32-KB segments. For each error that occurs, the 32-KB segment in which the update lies is flagged as needing reconciliation in the partial exception log. As with partial reconciliation on keyed files, store controllers reconcile only the 32-KB segments that were missed.

Exception Log Reconciliation

Reconciliation occurs any time a new connection is established between an acting master store controller or an acting file server store controller and another node. Reconciliation occurs even if an IPL does not occur.

When a nonmaster store controller or nonfile server store controller recognizes a new master store controller or file server store controller, it requests an update. The acting master store controller or acting file server store controller scans the appropriate exception log and sends a message to the requesting node of what it needs to reconcile. The requesting node then gets its updates from the master store controller or file server store controller and updates itself.

Multiple nodes can be reconciling themselves at the same time. When a node requests an update while in its IPL path, the following messages are displayed:

```
W903 WAITING FOR AN UPDATE FROM THE ACTING MASTER
W904 WAITING FOR AN UPDATE FROM THE ACTING FILE SERVER
```

While reconciliation is occurring during the IPL, the TCC Network on the store controller that is receiving its update is not active. This is done to prevent the terminals from accessing files that might be previous versions. After reconciliation is complete, the files are at the most current level. To get the terminals online as quickly as possible, you do not have to wait for reconciliation to complete. When the W903 message appears on the store controller display, press **F1** to bypass waiting for reconciliation from the master store controller. When the W904 message appears, press **F2** to bypass waiting for reconciliation from the file server store controller. Reconciliation will continue, but the TCC Network becomes active. This option must be used with caution because terminals can read back-level data.

When the acting master store controller or acting file server store controller searches the exception log for a node by requesting an update, it is selective in the entries it chooses to report to the requesting node. If the exception log entry is for a distribute at close file and that file is currently open in write mode, the exception log entry is not reported to the requesting node and the entry is not reconciled.

If the logical name ADXLNCLB is defined on the acting file server and acting master store controller, a W940 message will appear to indicate that one or more distribute at close files have been bypassed during reconciliation. The logical name must be defined as follows: ADXLNCLB = W940

When the next TCLOSE or CLOSE operation is performed on the file, and distribution is successful, the entry is removed from the exception log. This selection gives you more control over when distribute at close files are distributed.

Exception log entries that cannot be reconciled are automatically removed. For example, if an UPDATE exception log entry is found and the prime file has since been deleted, the exception log entry is removed. This automatic deletion lessens

the need for you to delete exception log entries by using the Distributed File Utility. A deletion of exception logs will log a W664 message for success; and a W665 message for failure.

Chapter 26. Multiple Controller Feature (MCF) Networks

This chapter presents options to consider when designing your LAN hardware and software configuration. This chapter does not attempt to present optimum solutions to your particular detailed needs because that would be beyond the scope of this guide. For additional assistance with determining the optimum solutions to your particular needs, contact your Toshiba marketing representative and explain your requirements. Using performance planning tools available to your Toshiba marketing representative, they can help you plan an optimum system for your needs.

When planning to use the Network Bridge Program, see information about LANs and SNA in the *4690 OS: Communications Programming Reference*.

TCC Support and Background Applications

The operating system supports communication between controllers and terminals in an Ethernet Terminal Controller Communications (TCC) Network environment.

Terminals running in a normal sales application mode send transactions to the store controller where these transactions are eventually distributed and logged into a transaction log. Normally, a non-interactive program runs in the background that continually processes these transactions, preparing files for reporting purposes, and so on. (See to the installation and planning guide for your application.) Both the TCC Network and the background program can require considerable processing power. It could be beneficial to split these tasks between two store controllers, if the file server controller becomes overloaded. For a description of primary and backup support, see Chapter 31, "Store controller backup," on page 493.

For every transaction, the terminal application might be required to read a particular application file, which is located on the store controller with the background program. This activity could require heavy traffic over the LAN (MCF Network) to read this application file. (It could be several times for each transaction, for each active terminal on the TCC Network.) System analyses should determine the best requirements, as to whether the TCC Network support and background processing should be on separate store controllers to spread the processing load, or should they be on the same store controller to minimize the LAN message traffic.

Defining File Types and Modes

A file type (where it is distributed) and mode (when it is distributed) are both defined when the file is created.

There are two distributed file types: *mirrored* and *compound*. Also, there are two distributed file modes: *distribute per update* and *distribute at close*. The two most common combinations of file type and mode that are chosen are *mirrored - distribute at close* and *compound - distribute per update*.

Mirrored File - Distribute At Close

A mirrored file consists of a prime version that is used for processing and an image version that is used for data redundancy purposes.

An application should not read an image version of any file that is distribute at close because this action would prevent the prime version from being distributed. The only reason for choosing file attributes of *mirrored - distribute at close* would be for data redundancy.

Distribute at close would be the distribution method of choice when there is a small file with many updates or there is a file that does not need to be updated every time a transaction is made. Using *distribute at close* minimizes the LAN (MCF Network) traffic overhead.

Mirrored File - Distribute Per Update

Distributing a mirrored file per update would be the method of choice when there is a large file with few updates. Distributing only the changes to the file generally causes much less LAN (MCF Network) traffic than distributing the whole file. The transaction log is an example of a *mirrored - distribute per update* file.

Also, an image version of the mirrored file could be more current using *distribute per update* than if it was updated at close. This consideration is important. However, it can be an expensive option because every write to the file requires two disk accesses and at least one update across the LAN.

Toshiba does not recommend that applications read the image version of a *mirrored* file that is *distribute per update*. Instead, you should use a *compound* file.

Compound files can also be *distribute per update*. Using a *compound* file enables you to add store controllers to your two-controller LAN without changing the distribution attribute of your files.

Compound File - Distribute At Close

For data files, the *compound file - distribute at close* combination is seldom used. Because image versions of files *distributed at close* should not be read by an application, as there would be no reason for the combination of *compound file - distribute at close*.

This combination is used, however, for distributing system and application program files to other store controllers on the LAN (MCF Network).

Compound File - Distribute Per Update

A compound file is distributed per update to other store controllers so that applications can use each file as a local version and avoid having to read an application file over the LAN (MCF Network). An example of this is the item record file.

Use the *compound file - distribute per update* combination when applications in each store controller require fast response time when reading a file record because the local, image version can be read by the application. There might be an overhead on the LAN, if this file is frequently updated. But in the case of an item record file, updates tend to be either a few records at a time at infrequent intervals, or a large number during off-hours when application read requirements are at a low point.

The following table summarizes the preceding information:

File Type	Distribute At Close	Distribute Per Update
Mirrored File	For data redundancy only. Application should not read image version.	For data redundancy. Application can read image version.
Compound File	Not recommended for data files. Distributed to all eligible store controllers. Application cannot read image version. Used for all program modules.	For distributed processing. Application can read local (image) version.

Connecting to a Host

Consider the following when deciding where to put the host connection:

- Which location generates the least LAN (MCF Network) traffic
- Which location (or locations) provide the best host backup solution
- Which store controller can best handle the additional processing load

Host Support Requirements

Requirements for host support fit into two categories:

- File maintenance and system support
- Applications such as host credit to support the terminal application

The terminal application specifies the location of the host credit pipe.

Two-Controller Configuration

There are four possibilities for connecting the host on a two-controller LAN (MCF Network) system. The four possibilities are:

- Host to file server store controller, LAN to alternate file server store controller.

The host connection is on the master/file server store controller, and the terminals are using a host credit application. The terminals are operating on a TCC Network that is attached to the alternate file server store controller. The terminals must use a pipe to send host credit messages to the host credit application. The messages are routed over the LAN to the master/file server store controller that supports the host connection. The request responses are sent over the LAN (MCF Network) to the terminal's store controller that made the request.

The requirement for system maintenance is satisfied. Also, if the master/file server store controller goes down, the terminal application is left without credit support. In this case, the terminal application must have support for substituting for the host function, until the master/file server store controller and the host link can be brought online.

- Host multi-dropped to both file server store controller and alternate file server store controller.

Another possibility is to have two host adapters in a multi-drop configuration: one in the master/file server store controller and the other in the alternate master/file server store controller.

The master/file server store controller runs sales support and supports store controller backup. The alternate master/alternate file server store controller is the primary store controller and supports the host credit application. The credit

application is supported by the host drop on the alternate master/alternate file server store controller. The file maintenance and system requirements are supported in the master store controller with the other host connection.

If the host applications are running in the master/file server store controller in an idle mode, host credit support can continue automatically if the logical pipe name used by the terminal application points to the local controller.

- Host adapter cards in both file server store controller and alternate file server store controller but with a single host line.

The host link can physically moved from the disabled store controller to the acting store controller and start up the required host background programs. The terminal application also needs to switch to a different pipe routing service controller ID for the new host link location.

Multiple Controller Configuration

For the configuration that requires three or more store controllers, the possible solutions are similar to the two-controller LAN system.

The master controller could have host support. The other store controllers, each supporting primary and backup terminals for another store controller, would require its terminal applications to address the host credit application pipe in the master store controller, with the request and the response going over the LAN.

Each store controller could support a multi-drop host connection with separate host applications for each store controller connection.

Terminal backup considerations are also similar, except the subordinate store controllers would support backup instead of the master/file server store controller.

Selecting Store Controller Roles

One of the most important considerations for your LAN (MCF Network) system is selecting the role each store controller is to perform. Plan to select one of the store controllers as the master store controller and the others as subordinates. When selecting the master store controller consider the following items:

Location

Because most of the operations are performed at the master store controller, select a store controller that is in a convenient location for the person who will be working with the store system.

Capacity

The size of the components of the store controller (number of hard disk drives installed, number of TCC Networks, and memory size) need to be considered when selecting the master store controller.

Host Link

There are a number of options to consider when deciding where to connect the host link. There might be limitations imposed by where the host link can be placed.

Processing Ability

The master store controller performs most of the operations; therefore, the store controller's ability to process data quickly should be considered.

Chapter 27. Using MCF Networks

This chapter discusses some of the facilities available to help you use your LAN (MCF Network) system.

Displaying the Status of Store Controllers

When a LAN (MCF Network) system is configured, or a store controller is added, or some other change to the system occurs, verify that the store controller roles previously configured are actually the acting roles.

Configured and acting store controller roles can be verified on any store controller on the LAN by using the display LAN status facility.

To verify the configured and acting store controller roles, do the following steps:

1. Press the System Request (**SysReq**) key to display the SYSTEM FUNCTIONS menu.
2. Press **C** to display the STORE CONTROL FUNCTIONS menu.
3. Press **2** and **Enter** to display the controller functions.
4. Press **1** and **Enter** to display the STORE CONTROLLER STATUS panel.

The STORE CONTROLLER STATUS panel shows which store controller IDs are configured as the master store controller, the alternate master store controller, the file server store controller, the alternate file server store controller, and the subordinates, as well as the acting store controller IDs for the same roles.

The acting store controller roles might be different from the store controllers that are configured to perform those roles. This situation can indicate one of the following conditions:

- The store controller assigned to a role has not been activated.
- A file that the store controller needs to identify the LAN store controller roles is missing.
- A store controller has been disconnected from the LAN.
- The store controller assigned during configuration has been manually deactivated and the alternate store controller has been activated.

For additional details on these conditions, see Chapter 30, "MCF network problem analysis and recovery," on page 485.

Activating/Deactivating the Master Store Controller

A primary task of the alternate master store controller is to assume control when the configured master store controller becomes disabled. Similarly, the primary task of the alternate file server store controller is to assume control when the configured file server store controller becomes disabled.

The first time that the system is IPLed, the configured master store controller is the acting master store controller. The alternate master store controller must be powered on and connected to the LAN (MCF Network) before performing this procedure. These steps ensure the integrity of the data. It remains the acting master store controller until it becomes disabled or deactivated as follows:

1. Press **SysReq** to display the SYSTEM KEYS panel.
2. Press **C** to display the STORE CONTROL FUNCTIONS panel.

3. Press **4** and **Enter** to display the MULTIPLE CONTROLLER FUNCTIONS panel.
4. Press **3** and **Enter** to deactivate the master store controller. If there are entries in the LAN exception log (see Chapter 28, "Using the Distributed File Utility," on page 465) for the master store controller, the master store controller cannot be deactivated until the following steps are performed:
 - a. IPL the controllers with the files needing reconciliation.
 - b. Delete the entries in the master exception log.

Attention: Deleting the entries in the master store controller exception log can cause previous image versions of files to be used as prime versions after activating the alternate master store controller as the acting master store controller.

If the configured master store controller is deactivated or becomes disabled, there is no acting master store controller until the alternate master store controller is activated as the acting master store controller.

Note: When the configured master store controller needs to be disabled for only a short length of time, activating the alternate master store controller as acting master store controller might not be needed.

When the configured master store controller resumes normal operation (the controller is re-IPLed and connected to the LAN), it does not automatically resume its role as acting master store controller. If the alternate master store controller is activated as the acting master store controller, the configured master store controller assumes the role of the alternate master store controller.

Before returning control to the configured master store controller, perform the following steps in sequential order:

1. Ensure that the master store controller is powered on and connected to the LAN
2. Verify that the master exception log is empty
3. Deactivate the acting master store controller (the original alternate master store controller)
4. Reactivate the configured master store controller

The order of this procedure is important for maintaining data integrity.

If the alternate master store controller is not activated as the acting master store controller, the configured master store controller automatically resumes its role as the acting master store controller.

Activating/Deactivating the File Server Controller

A primary task of the alternate file server store controller is to assume control when the configured file server store controller becomes disabled.

The first time the system is IPLed, the configured file server store controller is the acting file server store controller. The configured file server can be manually deactivated and the alternate file server store controller can be activated as the active file server. The alternate file server store controller must be powered on and connected to the LAN *before* performing this procedure to maintain data integrity. The alternate file server store controller remains the acting file server store controller until it becomes disabled or deactivated as described in the following steps:

1. Press **SysReq** to display the SYSTEM KEYS panel.

2. Press **C** to display the STORE CONTROL FUNCTIONS panel.
3. Press **4** and **Enter** to display the MULTIPLE CONTROLLER FUNCTIONS panel.
4. Press **4** and **Enter** to deactivate the file server store controller. If there are entries in the LAN exception log (see Chapter 28, "Using the Distributed File Utility," on page 465) for the file server store controller, the file server store controller cannot be deactivated until one of the following activities is performed:
 - a. IPL the controller with the files needing reconciliation.
 - b. Delete the entries in the file server exception log.

Attention: Deleting the entries in the file server exception log can cause previous image versions of files to be used as prime versions after activating the alternate file server store controller as the acting file server store controller.

If the configured file server store controller is deactivated or becomes disabled, there is no acting file server store controller until the alternate file server store controller is activated.

Note: If the configured file server store controller is disabled for only a short length of time, you might not need to activate the alternate file server store controller as acting file server store controller.

When the configured file server store controller resumes normal operation (such as the file server store controller is re-IPLed and connected to the LAN), it does not automatically resume its role as acting file server store controller. If the alternate file server store controller is activated as an acting file server store controller, the configured file server store controller assumes the role as the alternate file server store controller.

Before returning control to the configured file server store controller, perform the following steps in sequential order:

1. Ensure that the file server store controller is powered on and connected to the LAN
2. Verify that the file server exception log is empty
3. Deactivate the acting file server store controller (the original alternate file server controller)
4. Reactivate the configured file server store controller

The order of this procedure is important for maintaining data integrity.

If the alternate file server store controller was not activated as the acting file server store controller, the configured file server store controller automatically resumes its role as acting file server store controller.

Switching to Alternate Store Controller Roles

Switching LAN (MCF Network) roles from one store controller to another can affect terminal sales for price lookups because the application program uses a locally-stored image version of the item record file. When switching the store controller roles, the operating system must close all files that were opened on the master store controller and file server store controller and then open the same files on the store controllers that assume the master store controller and file server store controller roles. This file closing and opening procedure can take a few minutes, and during this time sales can not be performed.

Because of this time delay, do not switch the store controller roles unless you know that there will be an extended length of time before the master store controller and file server store controller are brought back into service.

During the interim time, between when the file server store controller ceases operation and the alternate file server store controller is activated as the acting file server store controller, transactions are being recorded in a spool file on the store controller that is controlling the TCC Network. (See “Store Controller Roles” on page 448.)

There is no immediate need to activate the alternate file server store controller. The length of time before the hard disk drive fills up depends on the current sales rate and how much space exists on the hard disk drive. This situation should not be a problem because the system file space usage allows for recording of transactions on the spool files.

If you decide you must activate your alternate master store controller and alternate file server store controller as acting master store controller and file server store controller, Toshiba recommends the following procedure:

1. Configure background applications to start when the alternate file server store controller becomes the acting file server store controller.
2. Activate the alternate master store controller as the acting master store controller.
3. Activate the alternate file server store controller as the acting file server store controller.

Note: If background applications are configured to start when the master store controller is activated and they access files controlled by the file server store controller, activate the file server store controller before activating the master store controller to ensure that the background applications have access to all of the files that they require to run.

Perform this procedure when a few minutes of slowed processing does not affect store operations.

Chapter 28. Using the Distributed File Utility

Note: The Distributed File Utility does not support DVD-ROMs.

The Distributed File Utility (DFU) is a menu-driven facility that enables an authorized operator to do the following tasks:

- Display or modify the distribution exception log on the master store controller.
- Display or modify the distribution exception log on the file server store controller.
- Modify a file's distribution attributes (for example, file type and mode).
- Distribute a file according to its current distribution attributes.

Accessing the Distributed File Utility

To access the DFU, select the File Utilities option from the SYSTEM MAIN menu on any store controller on the LAN. When the FILE UTILITIES menu appears, select the Distributed File Utilities option.

When the DISTRIBUTED FILE UTILITIES menu appears, select the option you want to use as shown in the following list:

- Process the Master Store Controller Exception log
- Process the File Server Exception log
- Modify the File Distribution Type Attributes
- Distribute the file

Distribution Exception Logs

Two distribution exception logs exist on a LAN system; one log on the master store controller and the second log on the file server store controller. Both exception logs record failed attempts to distribute files to other store controllers. The master exception log is used for compound files and system-mirrored files; the file server exception log is used for application-mirrored files.

Renames are logged as an update of the new name and a delete of the existing name. The master store controller and file server store controller use the exception logs to update distributed files on other LAN store controllers. If a store controller is removed from the LAN for any reason and is subsequently returned to the LAN, the master store controller and file server store controller check their respective exception logs. Any distributed files whose prime versions have been modified, renamed, or deleted since the store controller stopped communicating on the LAN are updated on the store controller. Each exception log entry is removed from each exception log upon successful redistribution. This process is called *file reconciliation*.

The file names for the exception logs are ADXLNDMF.DAT for files that are controlled by the master and ADXLNDSF.DAT for files that are controlled by the file server store controller.

Note: A copy of these exception logs also exists on the alternate master store controller and alternate file server store controller for backup.

Processing the Master and File Server Exception Logs

Options 1 and 2 from the DISTRIBUTED FILE UTILITIES menu enable an authorized operator to process either the exception log on the master store controller or the exception log on the file server store controller. The exception logs from any store controller on the network can be processed.

Processing the distribution exception logs enables displaying and erasing entries from the exception logs. Records cannot be added to the distribution exception logs. The entire contents of the exception log or selected entries can be displayed or printed. Log entries are selected in the same manner as scanning the system error logs, for example, by start time and date to end time and date. Also, select entries for a specific store controller or a specific file can be selected.

Displaying the Exception Log

This option enables printing or displaying either the entire log or only selected log entries.

Displaying All Log Entries

When displaying the entire log, the entries are displayed in the order of occurrence from the beginning of the file.

Displaying Selected Log Entries

When displaying only selected log entries, a panel appears prompting you to specify the following information.

Enter the starting date (the date to start the search) in the format: *mmdyy*. Use the same format to enter the ending date.

Where:

mm = The month (from 01 to 12)
dd = The day (from 01 to 31)
yy = The year (from 01 to 99)

Enter the starting time (the time to start the search) in the format: *hhmm*. Use the same format to enter the ending time.

Where:

hh = The hour (from 00 to 23)
mm = The minutes (from 00 to 59)

Enter the node ID in the format: *xx*

Where:

xx = The node ID of the appropriate LAN store controller (an asterisk (*) can be used to display any node IDs)

Enter the file name in the format: *subdir:fname.ext*.

Where:

subdir = The subdirectory on the specific store controller (for example, ADX_IDT1)
fname.ext = The file name (an asterisk (*) can be used to display any file names)

When unable to locate a file by using its current file name, the file might have been renamed. Try entering the old file name in the same format using the * to display any file names.

Enter the appropriate action: **U** for update or **D** for delete. Use the * to display any action. Press **Enter** to print or display the partial exception log.

Note: The first time the exception log is either displayed or printed, a file named ADXCSURF.DAT is created in the root directory. The system uses this file to format the exception log for displaying or printing. The ADXCSURF.DAT file is not erased, but it is reused the next time the log is displayed or printed.

Modifying the Exception Log

When modifying either the master store controller exception log or the file server store controller exception log, a menu appears providing the option to either erase the entire exception log or erase selected log entries in the exception log. You are asked to enter any character to display the log before any deletions take place.

There is no default value for this panel. To erase the entire exception log without first displaying it, type **1** and press **Enter**.

To erase selected entries, type **2** and press **Enter**. The SELECT EXCEPTION LOG ENTRIES menu appears. Use the same formats as previously described for displaying selected entries. To erase a selected entry, the name must be entered exactly as it appears in the exception log.

To cancel the erasing of exception log entries at this time, press the **Quit** function key to exit the function and the exception log is unchanged.

When to Display and Modify the Distribution Exception Logs

Displaying and modifying the exception log can be useful in cases where DDA tries unsuccessfully to reconcile files logged in the exception log, and one of the following messages appears: W906 UPDATE FROM ACTING MASTER WAS UNSUCCESSFUL or W907 UPDATE FROM ACTING FILE SERVER WAS UNSUCCESSFUL. The exception log entries indicate why the distribution failed and, for partial exception log entries, the number of updates missed.

In each case, the decision to erase an exception log record should be made after looking at the exception log records, determining what files are causing the problem, and investigating why the exception was logged. Using the DFU, a file can be distributed to correct the problems that are highlighted by the exception log.

Log Error on Clearing of Exception Log

4690 OS V5R2 adds logging of message W664 Event 110 when an exception log is successfully cleared and W665 Event 111 or 112 when an unsuccessful attempt is made to modify all exception log entries. Additionally, the currently active user ID will be added to instances of the W664 messages for event 110, and for W665 with events 11, 111, and 112 in the System Log Scan Utility.

Displaying and Modifying a File's Distribution Type

This option enables displaying or changing the distribution attributes of a file.

Displaying a File's Distribution Attributes

If you choose the option to modify the distribution attributes, you are prompted to enter the subdirectory name and the file name. Enter this information and press **Enter**; the file's current distribution type appears.

For example, ADX_IDT1:EALTAX1S.DAT displays the attributes of the file, EALTAX1S.DAT on the ADX_IDT1 directory, on the store controller you are working on.

To display the attributes, press **F3** to exit the function, and no changes are made.

Changing a File's Distribution Attributes

The Modify File Distribution Type option enables the changing of the current distribution attributes of only the prime version of a file. (See Chapter 25, "Using the Multiple Controller Feature," on page 445 for details about where prime version files are located.)

When this option is chosen, you are prompted to enter the name of the file whose distribution attributes you want to change.

The file name format is: *nodename::subdir:fname.ext*.

Where:

- | | |
|------------------|--|
| nodename | = The logical name of the store controller having the prime version of the appropriate file. |
| subdir | = The subdirectory of the prime version of the file. |
| fname.ext | = The file name and extension of the prime version of the file. |

For example:

ADXLNAAN::C:\ADX_IDT1\EALITEMR.DAT

points to the EALITEMR.DAT file in the ADX_IDT1 subdirectory on the C: drive of the acting master store controller. Another way of writing this is:

ADXLNAAN::ADX_IDT1:EALITEMR.DAT

where ADX_IDT1: is the logical name for C:\ADX_IDT1\.

When the file name is entered, the file's current distribution type appears. The system prompts for selection one of the following distribution types:

- Local file
- Mirrored file, distribute at update
- Mirrored file, distribute at close
- Compound file, distribute at update
- Compound file, distribute at close

Choose the appropriate set of file attribute options and press **Enter**. The file's attributes are changed and the update is distributed to the appropriate store controllers.

It is important to enter the logical node name of the store controller that has the prime version of the file because DFU does not allow changing the attributes of any image file version.

For example, when using the DFU on a subordinate, enter only the subdirectory and the file name, such as:

ADX_IDT1:EALITEMR.DAT (or C:\ADX_IDT1\EALITEMR.DAT)

When trying to specify a new set of attributes for the file, an error message is sent explaining that you are trying to update an image version of a file.

Changing a Local File to a Distributed File

When changing a file's distribution attributes from local to mirrored, the file must be on the store controller that keeps the prime version of the resulting mirrored file.

For example, if a user local file is currently on a subordinate store controller, it must be copied to the file server store controller before changing its distribution attributes to mirrored.

When the file is on the file server store controller, DFU can be used on any store controller as long as the file name of the file version is entered on the file server store controller.

Changing a Distributed File to a Local File

If a file attribute is changed from distributed to local, DDA changes the file's prime version to local.

The file image versions on the other store controllers are erased so that there is only one local copy on the system. If the same file is needed on another store controller as local, the file must be copied to the appropriate store controller. The copy also has the local attribute.

Distributing a File

The Distribute File option allows distribution of a file to its assigned nodes on the network.

Note: A file is considered to be successfully distributed when it has been delivered to a node, or, in the case of a configured node that is not currently connected to the LAN (MCF Network), when an entry is made in the exception log so that the file will be delivered when the node is reconnected.

This option does not change a file's attributes, it merely distributes the current prime version of that file to all store controllers that should receive it based on the file's current attributes.

The file name entered must include the store controller logical node name of the prime version of the file.

An image version of a file cannot be distributed.

Using the Distributed File Utility in Command Mode

The DFU can be invoked in two ways: by using the menus or by running the ADXCSU0L command in Command Mode. This section describes the interface to ADXCSU0L in Command Mode.

Use ADXCSU0L in Command Mode to use DFU in a batch file so that several LAN (MCF Network) file changes can be accomplished by running one program. Also, the DFU in Command Mode can be used to process a whole subdirectory with one command.

Use ADXCSU0L in a batch file or enter it in Command Mode with one of the following formats:

Format 1

ADXCSU0L 3 *m name*

Format 1 indicates to set the file attributes to the attributes that are specified by *m* and distribute the files immediately as required by the new attributes.

m = File attributes

- 1 = Local
- 2 = Mirrored, distribute on update
- 3 = Mirrored, distribute on close
- 4 = Compound, distribute on update
- 5 = Compound, distribute on close

name =

The name of a file or a subdirectory. A subdirectory can be specified in one of the following ways:

\directory
\directory
\directory.**

Format 2

ADXCSU0L 4 *name*

Format 2 distributes the specified files.

name =

The name of a file or a subdirectory. A subdirectory can be specified in one of the following ways:

\directory
\directory
\directory.**

Example 1

```
ADXCSU0L 3 3 ADXLXAAN::C:\ADX_IDT1\EALAUTH.DAT >>EALINST >> *EALINST
```

This example modifies the EALAUTH.DAT file on the master store controller in the ADX_IDT1 subdirectory to mirrored, distribute on close, and puts all messages in the EALINST report file.

Example 2

```
ADXCSU0L 3 5 ADXLXAAN::C:\ADX_UPGM\ >>EALINST >>*EALINST
```

This example distributes the EALAUTH.DAT file on the file server store controller in the ADX_IDT1 subdirectory and puts all messages in the EALINST report file.

Example 3

```
ADXCSU0L 4 ADXLXACN::C:\ADX_IDT1\ >>EALINST >>EALINST
```

This example distributes all the files in the ADX_IDT1 subdirectory on the file server store controller and puts all messages in the EALINST report file.

Chapter 29. Connecting OS/2 or DOS to the MCF Network

This chapter explains how to connect the Operating System/2[®] (OS/2) and DOS operating systems to the MCF Network.

Note: Connecting the OS/2 or DOS operating system to the MCF Network is not supported for systems that are configured for Controller-to-Controller Communications over Internet Protocol.

Connecting OS/2 to the MCF Network

This section describes the functions available when connecting OS/2 to the MCF Network. The following items are explained:

- Configuring an OS/2 server to enable store controllers to access files on the OS/2 hard disks (see the OS/2 LAN Server publications)
- Configuring an OS/2 requester to access files and printers
- Describing system commands that are used over the LAN
- Describing application program examples that read from and write to files on the LAN

The OS/2 functions that are not supported on the store controller are MESSENGER, NETPOPUP, NETRUN, ALERTER services, and remote IPL.

See the *OS/2 LAN Server Planning, Installation, and Customization Manual* for additional information on the OS/2 LAN Server product.

Operating System Restrictions and Required Software

Program incompatibility between the two operating systems prevents 4690 programs from being initiated or invoked remotely on OS/2 nodes. Similarly, OS/2 programs cannot execute on store controllers.

The 4690 Data Distribution Application (DDA), which provides file synchronization during normal store controller operation and file reconciliation when a disabled store controller comes online, is only supported under the operating system. Therefore, OS/2 nodes on the LAN (MCF Network) cannot perform any of the store controller roles, such as master store controller or file server store controller.

The file attributes assigned to mirrored or compound files are not recognized by an OS/2 node. A file copied from an OS/2 node to a store controller across the LAN will be created as a local file if the file does not exist. Replacing an existing file using the COPY command results in the file being updated, but the file attributes are not changed.

See the *4690 OS: Communications Programming Reference* for more information on LU 6.2 communication to an OS/2 PC.

OS/2 LAN Requester must be installed for the OS/2 node to have access to resources provided by a store controller. OS/2 LAN Server Version 1.0 is required for the store controller to have access to OS/2 shared resources and network commands.

Using the Universal Naming Convention

OS/2 uses a universal naming convention (UNC) name to identify resources (disks, printers, and so on) on the LAN. As described in the OS/2 LAN Server publications, the UNC name consists of the name of the server, the netname, and either the path and file name for a file resource or the device name for a printer resource. The server name is preceded by two backslashes (\\), and the netname and path are preceded by one backslash (\). The UNC name appears in the following form:

```
\\servername\netname\path and filename
```

In the 4690 environment, each of the UNC fields has a slightly different meaning. The *servername* corresponds to the 4690 node name (for example, ADXLXCCN); the *netname* corresponds to the 4690 system logical name for a directory or subdirectory; and the UNC *path* and *file name* correspond to a 4690 file name. Therefore, the UNC name for the file TEST.DAT in the subdirectory ADX_SDT1 on node CC would be:

```
\\ADXLXCCN\ADX_SDT1\TEST.DAT
```

For the same file name that is located on the root directory of drive D on node EE, the UNC name would be:

```
\\ADXLXEEN\D\TEST.DAT
```

Internally, the operating system expands the D netname to the system logical name equivalent of H1:\. A listing of the defined logical names for a specific store controller can be found using the following 4690 command:

```
DEFINE -S -N
```

For example, if a store controller has a second hard disk drive or D drive installed, you can find the UNC netname for the ADX_SDT1 subdirectory on the D drive by looking through the list of system logical names, using the DEFINE command. Issuing the DEFINE command shows the name ADX_SDT3, which is defined as follows:

```
ADX_SDT3: = h1:\\ADX_SDT1\
```

The UNC netname for the ADX_SDT1 subdirectory on drive D (drive H1:) would be ADX_SDT3.

When configuring OS/2-based PCs attached to the LAN, it is important to understand the 4690 equivalent of the UNC name. Additional examples of the UNC naming convention are provided in the following sections.

Network Access Instructions

Before a store controller can communicate with a node that is running OS/2, a network communication link must be established. The instructions for establishing this link vary depending on the type of communication that is needed. The following sections describe the network commands to enable the OS/2 node to use controller resources (OS/2 requester to 4690 server) and to enable a store controller to use OS/2 resources (4690 requester to OS/2 server).

Before using the following network commands, the OS/2 server and the requester software must be started. From the OS/2 Presentation Manager* Program Start menu, select **LAN REQUESTER**, or start the program by using the NET START command.

Accessing Files from an OS/2 Requester on a 4690 Server

The OS/2 network command, NET USE, enables an OS/2 requester to access files that reside on store controllers. The NET USE command must be issued from each OS/2 requester that wants access to 4690 resources. The format of the command is:

```
NET USE device name:\\computername\netname
```

For example, to access the ADX_UPGM subdirectory of a 4690 store controller that is designated as node CC, enter the following command at the OS/2 requester:

```
NET USE E:\\ADXLCCN\\ADX_UPGM
```

Immediately after invoking this command, the ADX_UPGM subdirectory of the 4690 store controller would be accessible from the OS/2 machine as device E:.

The OS/2 netname corresponds to a 4690 logical directory or subdirectory name. To access the ADX_SDT1 subdirectory on the D drive (H1:) of the store controller GG, as device F: of the OS/2 requester, enter the following command at the OS/2 requester:

```
NET USE F:\\ADXLXGGN\\ADX_SDT3
```

The ADX_SDT3 is used in the netname position because ADX_SDT3 is defined as the system logical name for the ADX_SDT1 subdirectory on drive D. (See "Using the Universal Naming Convention" on page 474 for information on identifying resources.)

The root directory of a 4690 hard disk drive is accessible also from an OS/2 node. To access the root directory of drive C (H0:) on store controller GG, as device P: on the OS/2 node, enter the following command:

```
NET USE P:\\ADXLXGGN\\C
```

If a subdirectory named OS2_DATA was created at the store controller, the subdirectory cannot be accessed from the OS/2 PC unless a logical name has been defined for the subdirectory by using controller configuration. For example, using controller configuration, a user logical name can be defined for OS2: as H0:\\OS2_DATA\\. After activating the configuration and IPLing the store controller, use the following NET USE command to access this subdirectory:

```
NET USE H:\\ADXLCCN\\OS2
```

After invoking the NET USE command, the device name that is assigned in the command to access the 4690 directory can be used. In the case of the last example, use the OS/2 device name H: to access the 4690 user-defined subdirectory OS2_DATA. From the OS/2 PC, the following command creates a listing of the entries in this subdirectory:

```
DIR H:
```

The next command copies all of the files in the subdirectory to the OS/2 PC:

```
COPY H:*,*
```

The device name that is assigned in the NET USE command can be used with any of the supported system commands or in an application program statement.

Accessing files from a 4690 Requester to an OS/2 Server

The LAN Server component of OS/2 must be installed before an OS/2 node can share resources with a store controller. The store controller node name must also

be defined as a user ID by the OS/2 Domain Administrator. To define a user in the Domain in OS/2 release 1.2 or later, do the following:

1. Select **User Profile Management Services** from the Desktop Manager or from the WorkPlace Shell menu.
2. Select **User Profile Management** from the Group - User Profile Management menu or from the User Profile Management Services - Icon View.
3. Select **Manage** from the User Profile Management - User Profile action bar.
4. Select **Manage Users** from the Manage pull-down menu.
5. Select **New** from the main panel and then **Actions** from the action bar.
6. Select **Add a new user ID** from the Actions pull-down menu.
7. Complete the **Add a New User** panel with the information that describes the new user.
 - a. Insert the eight-character 4690 node name (for example, ADXLXCCN) in the space that is provided for the user ID
 - b. Select a **User Type** of **User**
 - c. Select a **Password Option** of **No Password**
 - d. Select an **Access Authority** of **Allowed**
 - e. Press **Enter** to complete the new user definition.

For convenience in managing multiple user IDs, a Group ID can be set up and member user IDs selected.

Only OS/2 file resources that have been designated as shared by the domain administrator can be accessed by the store controllers on the LAN (MCF Network). A name must be assigned first to an OS/2 subdirectory, and group or user access must be allowed before the resource is available to a 4690 store controller. See the *OS/2 Local Area Network Server - Network Administrator's Guide* for additional information on sharing file resources.

No configuration changes are required at the store controller to access the resources of an OS/2 PC. After a store controller is defined as a user on the OS/2 server, the store controller accesses the OS/2 node by using the OS/2 computer name where the 4690 node name would normally be used. For example, to view an OS/2 shared directory assigned the name, OS2DISK, on an OS/2 server with the computer name of OS2SERV, the following command would be entered at the store controller command prompt:

```
DIR OS2SERV::OS2DISK:
```

To copy a file from this same OS/2 shared directory to the store controller, enter the following command at the store controller command prompt:

```
COPY OS2SERV::OS2DISK:filename.ext
```

where *filename.ext* is the name of the file being copied. Other system commands can be used in this same manner.

Also, the DEFINE command can be used to substitute a logical name for the longer, more complex name. For example:

```
DEFINE 0:=OS2SERV::OS2DISK:
```

Then use the DIR and COPY commands in the following manner:

```
DIR 0:  
COPY 0:filename.ext
```

Sharing Printers

Printers attached to the OS/2 PC are not accessible by the store controllers on the LAN (MCF Network). An OS/2 PC can use a printer that is attached to a store controller, but only by using the COPY command. Before using the printer, the NET USE command must be issued to define a local printer device name for the remote printer. For example, to define a printer attached to the store controller, ADXLXCCN, as local device name, LPT2:, the format of the OS/2 NET USE command would be:

```
NET USE LPT2: \\ADXLXCCN\PRN1
```

Once the local device name is defined, an OS/2 file can be printed from OS/2 to the printer by using the COPY command, and redirected to the device named LPT2:. The following command would be entered at the OS/2 PC to print the file on the printer:

```
COPY filename.ext LPT2:
```

System Commands

The following system commands can be used to access files on an OS/2 server:

ASSIGN	DIR	RENAME
BATCH commands	DEL	RMDIR
CHDIR	ERASE	SORT
COMP	FIND	TREE
COPY	MKDIR	TYPE
DEFINE	MORE	VOL
PATH		

For a description of these commands, see Chapter 7, "Command summary," on page 153.

The following OS/2 system commands can be used to access data and files on a Toshiba store controller:

ATTRIB	DEL	RENAME
BACKUP	DIR	RMDIR
BATCH commands	ERASE	SORT
CHDIR	FIND	TREE
COMP	MKDIR	TYPE
COPY	MORE	VOL
NET COPY	NET MOVE	

For a description of these commands, see the OS/2 and OS/2 LAN Server publications.

Using the Application Program Interface

The application program interface (API) between a store controller and an OS/2 PC enables file operations to be performed across the LAN (MCF Network) by using common application program statements. Application programs can be written to create, open, read, and write to files between two nodes on the LAN; thus providing a means of automating repeated file tasks.

To access a file on an OS/2 remote node, a 4680 or 4690 application program must specify the remote node name. The complete file specification should follow this format:

nodename::path:filename.ext

The OS/2 computer name (the name designated during installation) should replace the *nodename*, and the shared file resource name should be used in the *path* position.

OS/2 application programs reference files on a store controller by using the device name from the NET USE command. Before invoking an OS/2 application that opens or creates files on a store controller, the NET USE command should be issued.

See the *4690 OS: Programming Guide* for examples of programs that are written to the OS/2 application programming interface.

Connecting DOS to the 4690 MCF Network

This section describes the software and access instructions that are needed to connect a DOS-based PC to the 4690 LAN (MCF Network). It explains how to configure the DOS server to enable store controllers to access files on the DOS disks, and how to configure the DOS requester to enable access to 4690 files. A list of the supported system commands and examples of application programs that are written to access files across the LAN are also included.

For additional information about DOS or the PC LAN Program, see the DOS reference guide or the *PC LAN Program User's Guide*.

DOS and LAN Program Requirements

The following software packages must be installed on the DOS node to use the network access commands:

- DOS
- PC LAN Support Program (Version 1.1 or later)
- PC LAN Program (Version 1.2 or 1.30)

Note: Installation instructions are provided with each package.

After installation, the PC LAN Program command, NET START, is used to start the program. To start the DOS node as a server with the computer name, DOSPC, enter the following command at the DOS-based PC:

```
NET START SRV DOSPC
```

To start the DOS node as a requester with the computer name, DOSPC, enter the following command:

```
NET START RDR DOSPC
```

The PC LAN Program documentation includes many examples of the NET START command and describes a set of parameters that can be used to change network performance characteristics and memory usage.

The PC LAN Program Naming Convention

The PC LAN Program uses a specific naming convention to identify resources (such as disks, printers, and so on) on the LAN. As described in the *PC LAN Program Reference Manual*, the name consists of a computer name, a shortname

for the hard disk drive, directory, or printer, and an optional file name. The computer name is preceded by two backslashes (\\), and the netname and path are preceded by one backslash (\). The name is listed in the following format:

```
\\computername\shortname\filename
```

In the 4690 environment, each of the fields in the PC LAN Program network name have a different meaning. The *computername* corresponds to the 4690 node name (for example, ADXLXCCN), the *shortname* corresponds to the 4690 system logical name for a directory or subdirectory, and the *filename* corresponds to a 4690 file name. Therefore, the PC LAN Program network name for the file TEST.DAT on the subdirectory ADX_SDT1 on node CC would be:

```
\\ADXLXCCN\ADX_SDT1\TEST.DAT
```

For the same file name that is located on the root directory of drive D on node EE, the PC LAN Program network name would be:

```
\\ADXLXEEN\D\TEST.DAT
```

Internally, the operating system expands the D netname to the system logical name equivalent of h1:\. Use the following 4690 command to see a listing of the defined logical names for a specific controller:

```
DEFINE -S -N
```

If a store controller has a second hard disk drive or D drive installed, you can find the PC LAN Program network name for the ADX_SDT1 subdirectory on the D drive by looking through the list of system logical names, using the DEFINE command as shown above. Issuing the DEFINE command shows that the name ADX_SDT3: is defined as follows:

```
ADX_SDT3: = h1:\\ADX_SDT1\
```

The PC LAN Program network name for the ADX_SDT1 subdirectory on drive D (drive H1:) would be ADX_SDT3.

Establishing network access instructions

Before a store controller can communicate with a node that is running DOS, a network communication link must be established. The instructions for establishing this link vary depending on the type of communication that is needed. The following sections describe the network commands that are required for the DOS node to use store controller resources (DOS requester to 4690 server), and for the store controller to use DOS resources (4690 requester to DOS server).

DOS requester to 4690 Server

The PC LAN Program network command, NET USE, enables a DOS requester to access files that reside on store controllers. The NET USE command must be issued from each DOS requester that wants access to 4690 resources. The format of the command is as follows:

```
NET USE device name: \\computername\netname
```

For example, to access the ADX_UPGM subdirectory of a store controller that is designated as node CC, enter the following command at the OS/2 requester:

```
NET USE E: \\ADXLXCCN\ADX_UPGM
```

Immediately after invoking this command, the ADX_UPGM subdirectory of the store controller would be accessible from the DOS machine as device E:.

The DOS short name corresponds to a 4690 logical directory or subdirectory name. To access the ADX_SDT1 subdirectory on the D drive (H1:) of the store controller GG, as drive F of the OS/2 machine, enter the following command at the DOS requester:

```
NET USE F: \\ADXLXGGN\ADX_SDT3
```

The ADX_SDT3 is used in the shortname position because ADX_SDT3 is defined as the system logical name for the ADX_SDT1 subdirectory on drive D. (See "The PC LAN Program Naming Convention" on page 478 for additional information.)

The root directory of a 4690 hard disk drive device is also accessible from a DOS node. To access the root directory of drive C (H0:) on store controller GG, as device P: on the DOS node, enter:

```
NET USE P: \\ADXLXGGN\C
```

If a subdirectory named DOS_DATA was created at the 4690 store controller, the subdirectory could not be accessed from the DOS PC unless a logical name has been defined for the subdirectory by using store controller configuration. For example, using configuration, you can define a user logical name of DOS: as H0:\DOS_DATA\. After activating the configuration and performing IPL on the store controller, use the following format of the PC LAN Program NET USE command to access this subdirectory:

```
NET USE H: \\ADXLXCCN\DOS
```

After invoking the NET USE command, use the device name that is assigned in the command to access the 4690 directory. In the case of the last example, use the DOS device name H: to access the 4690 user-defined subdirectory DOS_DATA. From the DOS PC, the following command generates a listing of the entries in this subdirectory:

```
DIR H:
```

Use the following command from the DOS PC to copy all of the files in the 4690 DOS_DATA subdirectory from the store controller:

```
COPY H:*.*
```

Use the device name that is assigned in the NET USE command with any of the supported system commands or in an application program statement.

4690 requester to DOS server

DOS file resources are made accessible to the store controllers on the LAN by the NET SHARE network command. This command must be issued from each DOS PC that will be sharing files with the store controllers. The format of the command is as follows:

```
NET SHARE shortname=path
```

The *shortname* is used by PCs on the network to identify the directory that is being shared and it must be unique.

For example, to share the root directory of the C drive on a DOS PC, use the following command:

```
NET SHARE CDISK=C:\
```

This command enables the store controllers to have read, write, create, and delete access to the DOS root directory and all subdirectories above the root directory.

After entering the NET SHARE command, a listing of the DOS root directory at a store controller can be displayed by entering the following command:

```
DIR DOSPC::CDISK:
```

The preceding example assumes that the DOS PC was assigned the name, DOSPC, in the NET START command. (See “DOS and LAN Program Requirements” on page 478 for examples of the NET START command.) See the *PC LAN Program Reference Manual* for additional examples of the NET SHARE command.

Sharing printers

Printing across the LAN (MCF Network) is possible from both a 4690 store controller and a DOS node. A file is printed on a remote printer by using the COPY command and redirecting the output to the remote printer device name.

Before you can print a file by using DOS from a store controller, the printer must be made accessible to other nodes on the LAN by using the PC LAN Program NET SHARE command. The short name used in the command must be PRN. The format of the command to share the DOS printer device LPT1 would be:

```
NET SHARE PRN=LPT1:
```

After issuing this command, the 4690 file, *filename.ext*, can be printed at the DOS printer by entering the following command at the store controller:

```
COPY filename.ext DOSPC::PRN:
```

The preceding example assumes that a computer name of DOSPC was assigned to the DOS node in the NET START command.

To print a file on the 4690 from a DOS node, a local printer device name must be assigned to the remote printer by using the PC LAN Program NET USE command. To access the 4690 print device named PRN1, the format of the command that is entered at the DOS node would be the following:

```
NET USE LPT2: \\ADXLXCCN\PRN1
```

Once the local device name is defined at the DOS-based computer, a file can be printed from the 4690 by using the COPY command, and redirecting the output to device name LPT2:. The following command would be issued at the DOS PC to print the file from the 4690:

```
COPY filename.ext LPT2:
```

System commands

The following 4690 system commands can be used to access files on a DOS server:

ASSIGN	DIR	RENAME
BATCH commands	DEL	RMDIR
CHDIR	ERASE	SORT
COMP	FIND	TREE
COPY	MKDIR	TYPE
DEFINE	MORE	VOL
PATH		

For a description of these commands, see Chapter 7, “Command summary,” on page 153.

The following DOS system commands can be used to access data and files on a store controller:

ATTRIB	DEL	RESTORE
BACKUP	DIR	RMDIR
BATCH commands	ERASE	SORT
CHDIR	FIND	TREE
COMP	MKDIR	TYPE
COPY	MORE	VOL
RENAME		

See the *DOS Reference* for a description of these commands.

Using the Application Program Interface

The application program interface between 4690 and DOS enables file operations to be performed across the LAN (MCF Network) by using common application program statements. Application programs can be written to create, open, read, and write to files between two nodes on the LAN to provide a way to automate repeated file tasks.

To access a file on a remote DOS node, a 4680 or 4690 application program must use a file specification that includes the node name of the remote computer. The complete file specification should follow this format:

nodename::path:filename.ext

The DOS computer name (the name that is designated by the NET START command) should be substituted for the *nodename*, and the shared resource short name should be used in the *path* position.

DOS application programs reference files on a store controller by using the device name from the NET USE command. The NET USE command must be issued before invoking a DOS application that opens or creates files on a 4690 store controller.

See the *4690 OS: Programming Guide* for example programs written to the DOS application programming interface.

Alternative procedures for OS/2 and DOS

This section describes the creation of 4690 logical names for LAN resources and the use of single store controllers with OS/2 or DOS.

Creating 4690 logical names for LAN (MCF network) resources

As an alternative to entering the full NODE_NAME::DEVICE_NAME for an OS/2 or DOS resource, logical names can be substituted for their external resources. A temporary logical name can be defined using the 4690 DEFINE command. To assign the logical name of H: to the OS/2 or DOS disk resource with the name of EXT_ISP::C_DRIVE, use the following format of the command:

```
DEFINE H:=EXT_ISP::C_DRIVE:
```

After entering this command, the logical name of H: can be substituted in application programs or system commands for the longer name of EXT_ISP::C_DRIVE:. This substitution is valid only for the current window and is lost if the store controller is IPLed or the window is eliminated.

A logical name can be defined by using configuration, which is valid in all windows and is maintained if the store controller is IPLed. The method for creating a logical name by configuration is illustrated in the following example.

Example:

Assign the logical name of H to the OS/2 or DOS disk resource with the name EXT_ISP::C_DRIVE:.

1. From the SYSTEM MAIN menu on the master store controller, select **Configuration** (option 4).
2. Select **Controller Configuration**.
3. Continue pressing **Enter** until a complete list of the configured store controllers appears.
4. Select the store controller ID used to define the logical name, and press **Enter**.
5. Type an **X** beside the User Logical File Names option in the Allowed Definitions list and press **Enter**.
6. Select option 1, Define Logical File Name, and press **Enter**.
7. Type the logical file name being processed (in this case, H:) in the space provided, and press **Enter**.
8. Type the complete, expanded name in the space provided, (EXT_ISP::C_DRIVE:) and press **Enter**.
9. The logical name should now be defined, but the configuration must be activated before the redefinition takes place. Exit from the current panel using the **F3** key until the Activate Configuration option appears.
10. Select **Activate Configuration** to activate the logical name that was just defined.
11. After IPLing the store controllers to load the new configuration, the logical name H: can be used in place of EXT_ISP::C_DRIVE:.

Using a single 4690 controller with OS/2 or DOS

In store environments where only one store controller is used, you might still want to connect an OS/2 or DOS in-store processor to their single-controller by a LAN (MCF Network). This type of connection is fully supported by the 4690 Multiple Controller Feature (MCF); the single-controller user should use the standard installation instructions to install this feature.

After installing the Multiple Controller Feature, you are instructed to configure the LAN, inserting the node names of at least two store controllers. Single-controller users should follow these same instructions, using a nonexistent node name for the alternate master store controller and alternate file server store controller.

Once the new LAN configuration is active in the single store controller, DDA error messages, (W948, W949, and so on) are logged, and entries are added to the master controller exception log each time a compound or mirrored file is updated or changed. To avoid having these messages logged, create a file that is named ADXLNDXF.DAT in the ADX_SDT1 subdirectory and IPL the store controller (ADXLNDXF.DAT must exist in the ADX_SDT1 subdirectory). During the IPL, the file signals DDA that a single store controller is being used and that file updates to a second store controller should not be attempted. Single-controller MCF users might notice an improvement in system performance after creating this file because processor time will not be wasted in attempting file updates and logging of error messages that are not valid.

Chapter 30. MCF network problem analysis and recovery

This chapter discusses some problem analysis and recovery procedures for a LAN (MCF Network) system.

Recovering the master store controller hard disk drive

This section gives step-by-step procedures for recovering the master store controller after replacing a damaged hard disk drive. An example of a two-controller system is used to demonstrate the procedures. The procedures can be expanded to work on larger LAN (MCF Network) configurations.

In this example, one store controller is the master store controller and file server store controller. The other store controller is the alternate master store controller and alternate file server store controller.

Before beginning the procedures that are described below, the following three activities must be completed:

1. Activate the alternate master store controller as the acting master store controller and acting file server store controller. See the *4690 OS: Planning, Installation, and Configuration Guide* for an explanation of how to activate them using screen options.
2. Disconnect the LAN cable from the master store controller.
3. Replace the damaged hard disk drive in the master store controller.

Perform the following steps to bring your system back to a state of normal operation.

Step 1. Install the software on the replacement hard disk drive

During this step, message 3015 might appear on the screen, which reports that the LAN (MCF Network) is not operating. Press **F1** to bypass this message. This message is resolved in a later step.

Install the operating system, and its patches on the master store controller's new hard disk drive. Installing the operating system also partitions and formats the new hard disk drive. Next, install the 4690 Multiple Controller Feature and its patches. Be sure to specify the ID as the same ID that you had on the original master (which is the master store controller).

During the Multiple Controller Feature installation process, you are asked (Yes or No) if this is the master store controller. Answer **N**, that this is not the master store controller. The steps that follow reassign the master store controller as a master store controller.

After this point, message W915 might appear on the screen, which again reports that the LAN is not operating. Press **F1** to bypass this message. This message is resolved in a later step.

Do not IPL the master store controller at this time.

Step 2. Check for inactive changes waiting to be activated

At the alternate master store controller, check to see if there are any inactive configuration changes that are activated by the next step. If there are, verify that

these changes are correct and are needed before proceeding. Otherwise, activate a partial change in configuration with unwanted results.

Verify this by using Print Configuration as follows:

1. From the SYSTEM MAIN menu, press **4** and **Enter**. The INSTALLATION AND UPDATE AIDS screen appears.
2. Press **2** and **Enter**. The REPORT LEGACY CONFIGURATION DATA screen appears.
3. Enter **3** for Controller Configuration, **2** for Inactive Configuration, and **1** for Printer. Press **Enter**.
4. Perform this step and step 5 for each controller on the screen: Highlight the controller ID and press **Enter**.
5. Place an **X** beside all the configuration entries and press **Enter**. If you receive a message that indicates no inactive configuration changes, you can proceed knowing that the configuration being distributed is the same configuration as the one that was running before.

If configuration data appears, ensure that the displayed changes are acceptable before proceeding to the next step.

Step 3. Activate the acting master's configuration

To initiate the distribution of the appropriate configuration files that are on the alternate master store controller to the master store controller (and all the other controllers on the LAN (MCF Network), if there are others), activate the controller configuration as follows:

1. From the SYSTEM MAIN menu, press **4** and **Enter**. The INSTALLATION AND UPDATE AIDS screen appears.
2. Press **1** and **Enter**. The CONFIGURATION screen appears.
3. Press **4** and **Enter**. The ACTIVATE CONFIGURATION screen appears.
4. Press **2** and **Enter**. This activation step starts a three-sequence process:

Note: This activation step does not depend on making any changes to configuration. In other words, activation occurs whether the configuration has been changed or not.

- a. Configuration is verified (checked for errors)
- b. Configuration is distributed to the other store controllers on the LAN.
- c. Distributed configuration files are made active.

Note: Any store controllers that are defined but are not currently active on the LAN will have entries (for each configuration file) in the exception log.

Step 4. Stop all applications

Stop all applications on all of the LAN (MCF Network) store controllers at this point. This step avoids distribution errors.

Step 5. IPL the master store controller

Reconnect the LAN cable to the master store controller, and IPL. This message appears on the screen: No Acting Master Found. Do **not** press **F1** to bypass this message. The message is cleared automatically in the steps that follow.

Because the master store controller is still operating with the default configuration from installation, it does not have the information that indicates which store controller is the master store controller.

The following steps distribute the configuration files now in the alternate master store controller's exception log to the master store controller.

Step 6. Erase all file server exception log entries for the master store controller

Print the file server exception log located on the alternate master store controller. Identify and erase all log entries that have the master store controller address in them. These exceptions are resolved in Step 8.

Step 7. IPL all the other store controllers

On the alternate master store controller, select **Load Store Controllers** on the STORE CONTROL FUNCTIONS panel as follows:

1. Press **SysReq** and the SYSTEM KEYS panel appears.
2. Press **C** and the STORE CONTROL FUNCTIONS panel appears.
3. Press **2** and **Enter**. The CONTROLLER FUNCTIONS panel appears.
4. Press **4** and **Enter**. The LOAD CONTROLLER STORAGE panel appears.
5. Type an asterisk (*) in the highlighted area, and press **Enter**. This causes all the store controllers on the LAN to IPL.

W907 messages might be displayed indicating that attempted updates were unsuccessful. At each occurrence, press **F1** to bypass the message.

The subordinate controllers start IPLing first; the alternate master store controller starts last. The subordinate store controllers stop the IPL sequence with the message: No Acting Master Found. When the alternate master store controller reaches its "updating Alternate/Subordinate controller" step, the subordinate store controllers and master store controllers begin receiving image versions of all the active configuration files from the alternate master store controller. At completion, these store controllers automatically IPL again, so that the configuration files received can be read into memory.

The master store controller is now properly configured, but it is acting as the alternate master store controller and the alternate file server store controller, because the alternate master store controller is still the acting master store controller and the acting file server store controller.

Step 8. Distribute subdirectories

Now to put the application programs and data files on the master store controller, distribute the following subdirectories over the LAN (MCF Network) from the alternate master store controller:

```
ADX_IPGM
ADX_IBUL
ADX_IMNT
ADX_IDT4
ADX_IDT1
```

The easiest way to do this is by using the Distributed File Utility in Command Mode. See Chapter 28, "Using the Distributed File Utility," on page 465.

Note: If you get distribution errors during this step, display the exception log and resolve all entries. For information on resolving entries, see “Using the Distributed File Utility in Command Mode” on page 469.

The file updates should now be synchronized on all the store controllers on your network. The next step is to return the acting master store controller role to the master store controller.

Step 9. Reassign acting master and file server

The master store controller is now able to resume the role of master store controller. To do this, go to the CONTROLLER FUNCTIONS panel on the alternate master store controller. Using the options on this panel, deactivate the alternate master store controller as the acting master store controller and the acting file server store controller. Go to the CONTROLLER FUNCTIONS panel on the master store controller. Activate the master store controller as the master store controller and as the file server store controller.

Your system should now be working normally. If it is not and a message appears on any store controller panels, see the *4690 OS: Messages Guide*. If your system is not working normally and no messages appear, call your Toshiba.

Reconnecting a store controller to the MCF Network system

If the cable connecting a store controller to the MCF Network system is detached, the store controller is no longer a part of the MCF system. If this occurs, power off the store controller. Then, plug the Ethernet cable back in and power on the store controller. The store controller reconciles with the one currently on the network and ensures that data is not lost.

When the exception log or spool file fills up

Exception log

The exception log is allotted a specific amount of storage, which might fill up. If the storage fills up, a system message appears on the screen, and an entry is written to the system log. The record intended for the full exception log is lost, but the system message sent to the log contains the file name of the exception.

The sizes of the exception logs are determined during the IPL of the master store controller and file server. The space allocated for the files is computed by checking the number of logical names that define files and by adding an expansion buffer. The expansion buffer is 2% for the master LAN exception log and 7% for the File Server exception log.

When the W935 message is logged, the exception log is full. Check to ensure that all distributable files are logically defined. Exception log entries are cleared when the store controller is IPLed. To recover space on the hard disk drive, erase the exception log entry and use the Distributed File Utility to distribute the file. See Chapter 28, “Using the Distributed File Utility,” on page 465 for instructions on accessing this utility.

Previous to 4690 OS V5R2, exception logs were limited to a maximum of 819 entries. 4690 OS V5R2 has increased the size of the exception log to four times the previous size, or approximately 3276 entries.

Spool file

Unlike the exception log, the spool file is given more space as it expands. The spool file must grow to satisfy the system's spooling requirements. It can keep growing until the hard disk drive fills up. If the disk is full and another request for space is made, the system sends a message to the screen and writes an entry to the system log.

If the spool file has filled up the hard disk drive, the current application receives a write error. The application should provide alternate capabilities. Recovering and bringing up the file server allows the spool file to be de-spooled and the application to resume normal operation.

When the file server hard disk drive is full

The hard disk drive of the file server holds the prime version of the transaction log. If the hard disk drive fills up, it can no longer accept a WRITE MATRIX sent over the LAN (MCF Network) system. In such a case, the store controllers supporting terminals with applications that use WRITE MATRIX enter spool mode. All transactions are then held in the spool files of those store controllers until space on the file server's hard disk drive becomes available.

While the spooling store controllers wait, they try every 2 minutes to write to the transaction log. When space on the hard disk drive becomes available, the store controllers automatically send the spooled WRITE MATRIXes over the network to it.

Freeing space

The following items are ways to make space available on the file server's hard disk drive:

- Accept maintenance, which causes the ADX_xBUL subdirectory to be emptied.
- Erase unneeded files.
- Return to the host and erase files that are needed for host applications.

Another possibility is to recover lost clusters. Check for lost clusters by executing the CHKDSK command in Command Mode. If there are lost clusters and you want to recover them, execute this command, and use the -f option, like this:

```
CHKDSK C: -f
```

Use the Supplemental Option using the CD-ROM to use the -f option.

Any lost allocation clusters on the hard disk drive are converted to file names. The file names have the following form:

```
FILEnnnn.CHK
```

where *nnnn* are sequential numbers for each cluster chain found. These files can be erased to recover lost space.

When a store controller is listed as a non-MCF Network

When the CONTROLLER STATUS panel shows a non-MCF Network status for a LAN store controller, check the following items:

- **Check the subdirectory, ADX_SPGM, for the ADXLNDAF.DAT file..**

During configuration, assign an ID to each store controller. The system forms a name for each LAN store controller by using the ID you assigned. It stores these names in a file called the LAN Node List Entry file.

When bringing up the LAN, the system distributes the LAN Node List Entry file to all store controllers. If the distribution is not done correctly, non-LAN status for a store controller can result. Check the non-LAN status store controller for this LAN Node List Entry file. The LAN Node List Entry file should be in subdirectory named ADX_SPGM. If it is not listed in the ADX_SPGM subdirectory, the LAN Node List Entry file must be redistributed.

- **Verify that the keyboard file has the LAN flag set.**

Another possible cause of a non-MCF Network store controller status relates to the terminal keyboard definition file.

The system uses a file called ADXCCKF.DAT to find out if it is interfacing with a LAN system. The file is in the subdirectory named ADX_SPGM. The file contains a flag that, when set to on, indicates to the IPL process that a LAN system is configured.

If you think you have a keyboard file from a non-LAN store controller and your CONTROLLER STATUS panel indicates the store controller is not on the LAN, use the Display/Alter Utility to check the flag in the file named ADXCCKF.DAT. See Chapter 15, "Using the Display/Alter File Utility," on page 359 for a description of this utility. **Be sure to use ADXCCKF.DAT with the flag set to ON.** The flag is at offset X'3B'. The flag definitions are:

If byte = X'01', then the system is configured as LAN.

If byte = X'00', then the system is not configured as LAN.

If the flag is not set, use the Display/Alter Utility to set the flag.

Note: Before using the Display/Alter Utility to set the flag, IPL the system using the Supplemental Option using the CD-ROM.

Node list entry file

The node IDs are defined to the network by the LAN (MCF Network) Node List Entry file. This file, which contains a list of all the node IDs on the network, is found in the ADX_SPGM subdirectory in the file named ADXLNDAF.DAT.

Note: This file must exist on each LAN store controller for a system to come up as a LAN system.

The format of this file is:

ID-1,flags1,ID-2,flags2,ID-3,flags3, ----- ID-N,flagsN

where *ID-1*, *ID-2*, ..., *ID-N* are the store controller IDs (in ASCII format, one for each store controller) and the *flags1*, ..., *flagsN* are two-byte words indicating the roles of each store controller. They are defined (in hexadecimal) as:

00 00	- Subordinate not receiving application compound files
01 00	- Subordinate receiving application files
02 00	- Alternate file server
04 00	- File server
08 00	- Alternate master
10 00	- Master

These flags are added in hexadecimal to arrive at the combined store controller roles' flag value. For example, a "14 00" would be a store controller that is both the master store controller and the file server store controller (10 + 04). The designation "0A 00" would be a store controller that is both alternate master store controller and alternate file server store controller.

To display this file, use the Display/Alter File Utility (see Chapter 15, “Using the Display/Alter File Utility,” on page 359 for additional information).

Copying distributed files to and from diskettes

Do **not** use DOS to copy distributed files to and from a store controller hard disk drive. The reason is because DOS does not copy the file's distribution attributes (the file type and file mode) from the diskette. The result is the file is copied as a local file, even though it was originally defined as a distributed file. Because the store controller is operating under DOS when the copy is performed, nothing happens to the image versions on the other LAN store controllers except that they will not have a prime version to update them. This problem can be avoided if the COPY command in Command Mode is always used.

A problem can occur even when the file is copied from diskette in Command Mode, if the distributed file was originally copied to the diskette using DOS. In this case, the distribution attributes are not on the diskette; so when copying the file to the hard disk drive using Command Mode, the file is redefined as a local file. In addition, the LAN system erases the image versions on all the other store controllers. If this file is critical to the IPL process, it is imperative to repair the problem before the store controllers are re-IPLed so that the code, erased from the hard disk drive but still in memory, can be used.

Recovering when distributed files have been changed to local

If you mistakenly copy a file using Command Mode COPY that is defined as distributed on the hard disk drive, but is defined as local on the diskette, the LAN system erases the image versions on the other LAN (MCF Network) store controllers from their hard disk drives.

To recover from this situation, **do not IPL any store controller**. First, use the Distributed File Utility to change the prime version of the file's distribution attributes back to the correct (original) attributes. This step signals DDA to distribute the prime version to the required store controllers. When the distribution is complete, IPL all the store controllers in your system, which causes each controller to use the new version of the file that was copied from diskette.

See Chapter 28, “Using the Distributed File Utility,” on page 465 for instructions on accessing the Distributed File Utility.

Chapter 31. Store controller backup

A function of the operating system is store controller backup, which enables one store controller to automatically take over the TCC of another store controller. This function enables the terminals to remain online when the primary TCC controller is inoperative.

Store controller backup should not be confused with the Multiple Controller Feature. The Multiple Controller Feature is an optional feature of the operating system that provides the ability for interconnected store controllers to pass data between them.

How store controller backup works

LAN

Store Controller Backup

When a store controller leaves the LAN for any reason, every terminal with which it was communicating detects its absence. The terminals initiate the discovery mechanism of transmitting a FIND XID to a group address every 0.5 seconds. Upon receiving a FIND XID, a controller that finds the terminal number in the configured backup terminal list begins a count of the XIDs that it received from that terminal. When the count of the XIDs received from a terminal reaches three, the controller responds with a FOUND XID. This ensures that the primary controller is not attached to the LAN.

Store Controller Discovery

The discovery mechanism is used for both initial (primary) controller discovery and for establishment of a connection to a backup controller.

The controller discovery begins when the terminal issues a FIND XID to a broadcast address that is supported by all store controllers. The controllers, upon receiving the XID, verify the store number and check the terminal number against configured lists of terminals that are supported in primary and backup mode. The controllers then either respond with a FOUND XID, or discard the received FIND XID.

Note: The FOUND XID is only sent to the specific terminal that originated the FIND XID. Upon receipt of the FOUND XID, the terminal begins communications with the controller that responded to its FIND XID.

Store Controller Backup Operations

This section describes the store controller backup operations, including allowing and preventing backup, and recovering from store controller backup.

Allowing Store Controller Backup

To enable backup:

1. On the backup controller, press **SysRq**.
2. When the SYSTEM FUNCTIONS panel appears, press **C** to display the STORE CONTROL FUNCTIONS panel.
3. Press **3** and press **Enter** to display the TCC FUNCTIONS panel.
4. Press **1** and type the TCC Network to be allowed as backup.
5. Press **Enter** to enable store controller backup.

Preventing Store Controller Backup

Store controller backup should not be left enabled when turning off the store controller or detaching it from the TCC Network. Doing so could create a problem when the store controller is re-IPLed or it is re-attached (see “Recovering from Store Controller Backup”).

Disable (PREVENT) backup on the store controller that has the backup card every time the store controller is to be powered off or disconnected from the TCC Network.

To prevent backup:

1. On the backup controller, press **SysRq**.
2. When the SYSTEM FUNCTIONS panel appears, press **C** to display the STORE CONTROL FUNCTIONS panel.
3. Press **3** and press **Enter** to display the TCC FUNCTIONS panel.
4. Press **2** and type the TCC Network number to be disabled from backup.
5. Press **Enter** to disable (PREVENT) store controller backup.

Sometimes powering off the store controller supporting backup without disabling is unavoidable (for example, a power line disturbance) and when power is restored, either the backup store controller or both store controllers might start polling.

Another way to ensure that the store controller with the Ethernet network takes over the operation of the TCC Network when IPLing, even if backup is not disabled, is to be sure that the primary store controller completes its IPL before the backup store controller.

Recovering from Store Controller Backup

LAN

After the primary store controller is recovered, it must be IPLed. If the manual control mode is specified, it is recovered in secondary (monitor) mode.

To resume LAN control:

1. On the primary controller, press **SysRq**.
2. When the SYSTEM FUNCTIONS panel appears, press **C** to display the STORE CONTROL FUNCTIONS panel.
3. Press **3** and **Enter** to display the TCC FUNCTIONS panel.
4. Press **3** and **Enter** to resume control.

Appendix A. Loading Diagnostics for POS Systems over the Network

The operating system provides the ability to network load POS Diagnostics on newer model hardware in your 4690 store network. Using the PXE and DHCP capabilities provided in 4690 OS, the diagnostic application is automatically loaded over the network at the touch of a keystroke on a controller or terminal in the store network.

Supported Systems

- The following SurePOS system models support this capability:
 - SurePOS 700: 4900-7x5
 - SurePOS 300: 4810-350
 - TCxWave 6140: 100

Requirements

- 4690 OS V6R3 or later
- POS Diagnostic Remote Package Version 5.5.0 or later for the hardware models supported.
- A standard PC keyboard (PS/2 or USB) is required on the system running diagnostics to trigger the diagnostic load.
- A video display is also required for running the diagnostic application. Refer to the README included with the diagnostic package for additional requirements for using the diagnostic application.

To enable a 4690 store controller to load the diagnostic application, the user must download and install the diagnostic package on one or more 4690 controllers running as a DHCP server in the store network. The DHCP server on the controller is configured to load the diagnostic boot file as an alternate boot file. This controller becomes the source for loading the diagnostic application to other 4690 controllers or 4690 terminals on the store network. Details of the configuration process are provided in the “Configuring Remote Diagnostics on 4690” section below.

To run the diagnostic package on a system in the 4690 store network, power on the system and press the **F11** key during BIOS POST on the screen where it prompts the user to enter BIOS setup. Note that there is no specific prompt to press **F11**. A message (“Boot from Network now...” or similar message) appears at the bottom of the screen confirming an alternate boot method was selected. This sequence will trigger the appropriate DHCP request to load and run the diagnostic application.

Configuring Remote Diagnostics on 4690

To install the diagnostic package on the 4690 store controller:

1. Download the POS REMOTE Diagnostic package from the Toshiba support site
2. Unzip the diagnostic package and copy the following files (including subdirectories) from the remote/tftp directory in the diagnostic package to the c:/adx_boot directory on the 4690 store controller:
 - pxelinux.0
 - img2a
 - img3a.rd
 - pxelinux.cfg/default

3. No changes are required in the diagnostic default configuration file. Unattended mode diagnostics is not supported in this environment. Any messages regarding the use of unattended mode diagnostics should be ignored.

To configure the 4690 DHCP server to load the diagnostic package, edit the DHCP server configuration file (**adx_sdt1:adxipNnd.dat**) and add the following line:

altboot DIAG c:/adx_boot/pxelinux.0

where:

altboot is a new keyword indicating an alternate boot file will be used.

DIAG is a 4-character keyword indicating the boot type. This keyword is used to match a keyword sent by BIOS in the DHCP Discover and load the corresponding alternate boot file.

c:/adx_boot/pxelinux.0 is the alternate boot file to load. In this case, pxelinux.0 is the bootstrap downloaded from the diagnostic package.

After modifying the DHCP configuration file, the DHCP configuration is applied by issuing the command **dhcpcinfo -u** or by a controller IPL. It should be noted that other uses of **altboot** are not supported.

To verify the DHCP server configuration, run the **dhcpcinfo** utility to display the alternate boot configuration. Example output of a controller configured for the diagnostic alternate boot method is shown below. If the **dhcpcinfo** output does not show an “Alternate Boot Files” section as shown in the example output, then the alternate boot configuration is incorrect or has not been applied.

DHCP Status for Server 10.1.1.2:
RTU Server: none

Alternate Boot Files

DIAG c:/adx_boot/pxelinux.0

Address	Status	Lease	Client ID	Last Leased
-----	-----	-----	-----	-----
10.1.1.200	Leased	Infinite	5cf3fcd300a2	Feb 16 11:36:55 2012
10.1.1.201	Leased	Infinite	5cf3fcd30051	Dec 19 13:53:10 2011
10.1.1.202	Leased	Infinite	5cf3fcd30083	Dec 19 10:58:46 2011
10.1.1.203	Leased	Infinite	5cf3fcd30008	Dec 19 10:57:36 2011
10.1.1.204	Leased	Infinite	001a6440c704	Feb 16 11:36:43 2012
10.1.1.205	Leased	Infinite	5cf3fcd3003e	Jan 24 14:38:09 2012

Refer to the *4690 OS: Communications Programming Reference* for more information on the DHCP Server configuration and the **dhcpcinfo** utility.

Understanding DHCP Alternate Boot Behavior

If the alternate boot attempt succeeds, the client will obtain an IP address via DHCP and proceed to load “PXELINUX” followed by the diagnostic application. Progression messages of the load process will be displayed on the attached video display.

If the attempt to alternate boot the diagnostic application fails, the behavior of the client may vary depending on the configuration. In most cases, the client will

continue to boot normally as either a controller or a terminal instead of booting the diagnostic application. However, if the network supplemental boot is enabled on the DHCP server that services the alternate boot, the client will continue to network boot (PXE) to honor the network supplemental boot configuration, even if it normally boots as a controller. In the case of failure to boot the diagnostic application, verify that the alternate boot configuration is correct using the **dhcinfo** utility.

Appendix B. Systems Network Architecture 3270 Emulation

This appendix explains how to use 3270 emulation as a part of the operating system. Knowledge of the operating system, your host communication, and local area networks (LANs) is helpful.

Throughout this appendix, the term *store controller* refers to the SurePOS 700 Series, 4694, 4693-5x1, or 4693-7x1 Point-of-Sale Terminal, or any of the supported Personal Computers. The term *3270 emulator* refers to the operating system support for the 3270 emulation program.

3270 Emulator description

3270 emulation simulates many of the functions of the following hardware:

- 3278 Model 2 Display Station
- 3279 Model 2A Display Station
- 3287 Models 1 and 2 printers

In a Systems Network Architecture (SNA) network with sessions of LU types 1, 2, and 3, the emulator causes the store controller to appear as a 3274 Model 1C controller.

The 3270 emulator enables you to:

- Integrate your store controller into your existing 3270 host computer network.
- Use your store controller as a workstation (3270 console) and receive printer data from the host.
- Use a suitably configured Mod1 terminal as a workstation and receive printer data from the host.
- Support *programmed operators* by writing applications in 4680 BASIC to access 3270 console sessions.

The 3270 emulator is a set of applications that work together to provide the 3270 emulation function for the store controller and the Mod1 terminal. The emulator can run in the terminal component of the controller/terminal concurrent with execution in the controller component. Multiple 3270 emulators can run and provide support for multiple SNA sessions. The 3270 emulator does not preclude the running of other sessions for other applications, such as the host command processor (HCP), remote change management server (RCMS), or user-written LU 6.2 applications.

3270 Emulation on the store controller

On the store controller, the 3270 emulator is a single application. Multiple copies of it run, if multiple emulations are needed.

The 3270 emulator enables a controller console and a printer to function as a 3270 terminal device. When operating as a console interface, the 3270 emulator accepts keyboard data and processes this data into a 3270 data stream for transmission on the SNA communication link. 3270 data stream responses from the communication link control the keyboard and provide output to the display. You can toggle between the emulation and other applications that run other windows on the store controller.

The store controller display/keyboard can be replaced by a suitably written application accessing the emulator through an application program interface (API). In this case, the input provided by the application replaces human operator keystrokes, and no data appears on the display. The application can access the

3270 data stream responses from the SNA communication link. It is, also, possible for the application to share the session with a human operator or with other applications by giving up control of the 3270 emulator. If the emulation is solely controlled by applications with no human intervention at any time, it runs in the background and does not interact with the screen and keyboard.

The 3270 emulator enables the store controller to function as a 3270 printer, to receive printer output from the SNA communications link, and to format the data as output for the print spooler. The data prints on any IBM Proprinter or equivalent device that is attached to the controller and is supported by the print spooler.

In multiple-controller configurations, sessions can be established for consoles or printers on store controllers other than the store controller that has the physical communication link.

The 3270 emulator supports multiple 3270 SNA LU2 sessions while maintaining support for the existing capability of multiple LU0 sessions. LU2 and LU0 sessions can run concurrently.

Additional applications are not required to be written to take advantage of the 3270 emulator functions. Existing applications should operate concurrently with the 3270 emulator without making user programming source code changes.

3270 Emulation on the Point-of-Sale terminal

All of the 3270 functions that are supported on the store controller are also supported on suitably configured Toshiba point-of-sale terminals.

Each 4683, 4693, 4694, SurePOS 300/700 or TCxWave 6140 Series terminal can be configured to support one (or zero) host printer session and one (or zero) console session.

When a 3270 display is supported at the terminal, you can toggle between the emulation and the primary sales application. See “3270 Emulation on the Terminal” on page 516 for additional details about 3270 emulation on the terminal.

The physical communication link for 3270 sessions that are configured in the terminal is installed on a store controller, not on the terminal. In multiple store controller configurations, the physical link does not need to be installed on the store controller.

Equipment supported by the 3270 Emulator

Devices that can function as 3270 consoles are:

- All models of store controller:
 - Keyboard
 - Display
 - Printer
- 3161/3164 and 3151 American National Standard Code for Information Interchange (ASCII) terminals (or equivalent) attached through the multipoint communication adapter on the store controller

Note: The 3164 terminal displays data with black or dark-grey color attributes as green. Applications should avoid using color combinations like green on dark-grey or green on black (or conversely) for display on a 3164 terminal.

- Mod1 terminals
 - At least 2 MB of memory

- 9-in. or 12-in. full-screen displays (monochrome or color)
- Alphanumeric Point-of-Sale (ANPOS) keyboard
- 4693 ANPOS keyboard
- Mod2 terminals
 - Attached to one of the Mod1 terminals that are listed above
 - With full-screen displays and ANPOS keyboards

Note: For a 3270 console that is accessed only through the API (that is, from another application) the full-screen display and ANPOS keyboards are not required. Because of the differences between the store controller keyboards and 3270 keyboards, some adjustments are needed. Keyboard templates are provided to aid with these adjustments.

SNA communication can be over SDLC or X.25 links that are supported by the operating system. In a multiple-controller configuration, a store controller with a physical link that is installed is known as a *gateway*. A configuration can have multiple links and multiple gateways. Each controller gateway supports a maximum of 64 SNA communication sessions.

Devices that can function as 3270 printers are the:

- Proprinter attached to the store controller
- Proprinter serially-attached to the 4683, 4693, 4694, SurePOS 300/700 or TCxWave 6140 Series terminal (must be a 2-MB terminal, or a Mod2 terminal attached to a 2-MB Mod1 terminal)

3270 console sessions, also, have the ability (such as LOCAL COPY PRINT) to direct a copy of their screen image to a printer. The Proprinter, or equivalent device, that is attached to the store controller is used for this purpose.

Printers that are used for host print or local copy print from console sessions on the 4683, 4693, 4694, SurePOS 300/700 or TCxWave 6140 Series terminals are the:

- Proprinter, or equivalent device, that is serially-attached to the 4683, 4693, 4694, SurePOS 300/700 or TCxWave 6140 Series terminal
- Proprinter, or equivalent device, that is attached to the terminal's store controller

Keyboard templates

Keyboard templates in U.S. English in the operating system are in the following files:

- ADXPS2KF.DAT - for the enhanced PS/2 keyboards
- ADX316KF.DAT - for the 3161/3164 keyboard
- ADXANPKF.DAT - for the ANPOS keyboard
- ADXAN9KF.DAT - for the 4693 ANPOS or USB ANPOS keyboard

Note: These files are on the 4690 Optionals.

Print the files, and cut and join the ends as directed, to provide a reference for 3270 emulator key locations. The template for the ANPOS and 4693 ANPOS keyboards is for the default layout and is not applicable if the keyboard is reconfigured.

Modifying your system configuration

To use the 3270 emulator, your system configuration must be modified. See the *4690 OS: Planning, Installation, and Configuration Guide* for additional information.

You must configure a host link for 3270 emulation before using it.

Configure API sessions as console (screen) sessions. There are no differences between API sessions and screen sessions as far as the SNA configuration utility is concerned.

Changing your configuration using menus

To change your configuration:

1. On the SYSTEM MAIN menu, select **Installation and Update Aids**.
2. On the INSTALLATION AND UPDATE AIDS panel, select **Change Configuration Data**.
3. On the CONFIGURATION panel, select **Controller Configuration** or **Generic Terminal Configuration**.

Note: If you are configuring 3270 emulation for SurePOS 300/700 Series or TCxWave 6140 Series terminals, you must select to work with **Generic Terminal Configuration**. Then, select **Terminal Load Definition**.

4. From this point, you are typing information on panels. See the *4690 OS: Planning, Installation, and Configuration Guide* for this information.

Starting 3270 Emulator Sessions in the Controller

Start 3270 controller emulator sessions, when they are configured as a secondary application, from either the SYSTEM MAIN MENU panel or the command line. The command line entry is:

```
ADXHS30L  
[NODENAME], [LINKNAME], [PRINTNAME], [API/ID], [N], [L], [F]
```

NODENAME = Gateway Node Name:

This ASCII string is a maximum length of 8 characters plus a trailing null. It can either be the node name or a user logical file name that the operating system translates to the correct node name when it is used on an OPEN command. The 3270 emulator adds the double colon (::) node name delimiter, which must not be in the supplied string or its user logical file name.

LINKNAME = Link Configuration Name:

This ASCII string is a maximum length of 8 characters plus a trailing null. It is the name of the SNA link on which the 3270 sessions are configured.

If the command line parameters are omitted, the 3270 emulator uses two standard names, for which you must define user logical file names in the store controller configuration. The standard names are:

- SDX2NODE for the gateway node name. If no user logical file name is defined, the local node name is used.
- SDX2LINK for the link configuration name.

For example, SDX2NODE might be defined as ADXLCCN and SDX2LINK might be defined as 3270LINK.

PRINTNAME = Logical Printer Name:

This is the identifier of the logical printer assigned to the session if you are starting a 3270PRTR session or it is the identifier of the logical printer that receives local copy output from screen sessions. The identifier is specified as PRNx where: x is a value of 1 to 8.

An example of a command line entry is:

```
ADXHS30L ADXLXCCN, 3270LINK, PRN1:
```

API/PRT = API ID or Printer Disposition:

You must specify a single uppercase letter for a console session if it is to be accessed through the API. The letter should be the same as the session identifier in the CONNECT verb issued by the API program.

Specify **1**, to have the printer application end when the link terminates.

Specify **2**, to send untranslated control strings to the printer.

Specify **3**, to set both of the above options on.

Note: An untranslated control string is sent in the printer data stream that is preceded by X'35nn', where *nn* is the length of the control string.

N = Normal Character Density:

If the character N is in the parameter string, all 3270 printer sessions print in normal pitch (10 characters per inch). If the character N is omitted, the printer session defaults to condensed pitch (17 characters per inch).

L = Laser Printer Option:

If the character L is included as the sixth parameter in the parameter string, the 3270 emulator supports a 4029 laser printer in HP** compatibility mode.

F = Form Feed Passthrough:

If the character F is included as the seventh parameter in the parameter string, the 3270 emulator always sends form feed characters to the printer unchanged. If the character F is omitted, the form feed character sometimes changes to multiple line feed characters.

For servicing host-initiated printer output only, the 3270 emulator is started in background mode. It does not access the screen or keyboard. Host-initiated printer output can be handled only if the 3270 emulator is already running on the store controller with the target printer attached. An operator must ensure that the 3270 emulator has started on the target node before sending printer output. Background programs cannot be started from the command line; the background menu must be used. An example of a background menu entry, where SDX2NODE and SDX2LINK are defined in the logical names file, is:

```
NAME=ADX_SPGM:ADXHS30L.286 PARM=, ,PRN1:
```

Stopping the 3270 Emulator in the Controller

Press **Exit** to stop the 3270 emulator. See "Emulator-Specific Keys" on page 509. For printers, see the description of the printer disposition parameter on page 503. Emulation in the store controller can be restarted at any time.

When link errors are reported, the 3270 emulator logs the error and issues an immediate CLOSE for the session and link connections. The 3270 emulator's action varies, depending on the function that it is performing.

- For screen errors, the emulator displays COMM *nnn* (where *nnn* is a 3-digit diagnostic code) on the operator guidance line and waits until the **Exit** key is pressed.
- For printer errors, the 3270 emulator closes down and exits.

Using the 3270 Emulator with Switched Lines

If the 3270 emulator is used in a switched line (dial-up) system, the SNA driver needs to be installed before invoking the 3270 emulator. This step is required because the link between modems must be established before the 3270 emulation becomes active. If the modem link is not established when the 3270 emulator is invoked, the application immediately terminates.

Modems for use on switched lines normally do not produce the data carrier tone if the data terminal-ready (DTR) interface signal is not active. DTR normally is not active until the SNA driver is installed and running. Therefore, the typical procedure for originating a host connection on a switched line is to install the SNA driver, manually dial the telephone number, wait for the modem tones to exchange, and then hang up. Activate the 3270 emulation any time after the previously described procedure is complete.

Configuring a Printer Emulation at the Store Controller

Host printing at the store controller that is using the 3270 emulator, requires that an emulation run in the background. Configure the 3270 emulator as a background application in the same way as other applications. Use the application name, ADX_SPGM:ADXHS30L.286. The command line parameters to pass to the application when it starts are the node name, the link name, the printer name, the printer disposition, the print pitch, the laser print option, and form feed passthrough.

For example:

```
adx1xddn,3270link,prn1:,N,L,F.
```

If omitting the command line parameters, the store controller uses the defaults for console emulations. The store controller uses the system default printer as the physical print device (PRN0:).

Using 3270 Emulator Keyboard Functions at the Store Controller

This section provides a brief summary of the 3270 function keys, and a full description of the available 3270 emulator-unique key functions.

The keyboards supported by the 3270 emulator are the:

- PS/2 keyboard
- 3151, 3161, and 3164 keyboards
- Alphanumeric Point-of-Sale (ANPOS) keyboard
- 4693 ANPOS or USB ANPOS keyboard

PS/2 and 31xx keyboards have different layouts, but wherever possible, keys at the same physical location are used for each 3270 emulator function.

Graphic Keystrokes

Normal *graphic* keystrokes are treated so that, wherever possible, an EBCDIC character is generated by a key with a matching legend. Some common EBCDIC

characters, particularly some national usage variants, do not have matching keytops. These EBCDIC characters are generated as listed below:

- Use a [(left bracket) to produce a single back-quote on the U.K. English keyboard
- Use a [(left bracket) to produce a cent sign on the U.S. English keyboard
- Use a] (right bracket) to produce a solid vertical line.
- Use a ^ (caret) to produce the extended binary-coded decimal interchange code (EBCDIC) not sign.

Notes:

1. The 3151, 3161, and 3164 terminals have no | (split-vertical-bar) character on the keyboard or in the character set and cannot be generated on a 31xx console. The same is true for the # (pound sign) character (U.K. monetary symbol).
2. The macron that is normally provided on U.K. English 3270 keyboards is not provided on either the PS/2 or 31xx keyboards. Instead, use the ~ (tilde) character.

Unsupported Functions

The 3270 emulator function keys are discussed below. The alternative cursor styles (size) are selected locally on the 31xx keyboard; therefore, the emulator does not attempt to control cursor style.

Note: The convention used in the following tables list escape sequences in the format, Ctrl(x), where the relevant shift keys are held down while the target key is typed.

3270 Emulator Key Functions

3270 key functions are of the following types:

- Cursor movement
- Field cursor movement
- Program attention
- Delete
- Print control
- Miscellaneous
- 3270 emulator unique

Descriptions of the 3270 functions and their key sequences are described below.

Cursor Movement Keys

Cursor movement keys are **Up**, **Down**, **Right**, and **Left**. Screen and line wrap are under the control of the 3270 emulator.

Table 46. Cursor Movement Keys

Function	Store Controller Keyboards	31xx Keyboard	Either Keyboard
Cursor Up			Up Arrow
Cursor Down			Down Arrow
Cursor Right			Right Arrow
Cursor Left			Left Arrow

Field Cursor Movement Keys

Field-oriented cursor movement keys cause the cursor to be moved to screen positions that are determined by 3270 field attribute positions. If the display has no unprotected fields or no fields at all, the cursor is positioned at the home position (row 1, column 1).

Tab The Tab key moves the cursor to the first character location of the next unprotected field that is wrapping the screen, if necessary.

Back Tab

The Back Tab key moves the cursor to the first character location of the current field, if the current field is unprotected and the cursor is not at the first character location. Otherwise, the Back Tab key moves the cursor to the first character position of the previous unprotected field. Screen wrap occurs, if necessary.

Home The Home key moves the cursor to the first unprotected character position of the screen.

New Line

The New Line key moves the cursor to the next unprotected character position starting from the beginning of the next line and, if necessary, wraps the screen.

Table 47. Field Cursor Movement Keys

Function	Store Controller Keyboards	31xx Keyboard	Either Keyboard
Tab			Tab
Back tab		Back tab	Shift (tab)
Home			Home
New Line	Numeric Pad +		Ctrl (J)

Program Attention Keys

Use program attention keys to communicate with the host processor. Each key causes a different type of inbound transfer.

Enter The Enter key causes an inbound transmission if the display terminal is currently in session with the system services control point (SSCP) or the primary logical unit (PLU). The session governs the type of inbound transmission. The Enter key clears the screen of data and attributes. The cursor position is set to row 1 and column 1. If the display is in session with the PLU, an inbound transmission is initiated informing the host processor of the loss of presentation space integrity.

Program Access

There are three program access (PA) keys. Each of these PA keys causes an inbound transmission of one byte. Use the PA keys to communicate instructions rather than send data to the host processor. The PA keys are valid only for the PLU session.

Program Function

There are 24 programmed function (PF) keys. Each of these PF keys causes an inbound transmission with a unique attention identifier (AID) code and data from modified fields. The PF keys are valid only on the PLU session.

Cursor Select

The Cursor Select key provides a Selector Light Pen detection function by

keystroke rather than by the light pen. It is valid in detectable fields where the first character position contains a designator character. There are two types of detectable fields: selection and attention.

Attention

The Attention key is used to interrupt the host processor by causing an inbound status message to flow, which requests that direction is given to the secondary logical unit (SLU).

Table 48. Program Attention Keys

Function	Store Controller Keyboards	31xx Keyboard	Either Keyboard
Enter	Enter	Return	
Clear	PgUp	Clear	
PA1	Ctrl (End)	PA1	
PA2	Ctrl (Cursor Down)	PA2	
PA3	Ctrl (PgDn)	PA3	
PF1 -> PF10			F1 -> F10
PF11	Ctrl (F1)		F11
PF12	Ctrl (F2)		F12
PF13 -> PF22			Shift(F1 ->F10)
PF23	Ctrl+Shift (F1)		Shift (F11)
PF24	Ctrl+Shift (F2)		Shift (F12)
Cursor Select	Ctrl (G)		
Attention			Ctrl (A)

Delete Keys

Use delete keys to remove data and attributes from the screen.

Delete The Delete key deletes the character at the current cursor position.

Characters to the right of the cursor within the same field and on the same line are shifted one character position to the left. The cursor does not move.

Erase End of Field

The Erase End of Field key clears data and attributes from the current cursor position to the end of field or to the end of screen, if the display has no fields assigned.

Erase Input

The Erase Input key removes all unprotected data and attributes from the screen. The Erase Input key resets the Modified Data Tags in the field attribute byte for all unprotected fields and positions the cursor as for the Home key.

Clear The Clear key removes all data and attributes from the screen.

Table 49. Delete Keys

Function	Store Controller Keyboards	31xx Keyboard	Either Keyboard
Delete	Del	Delete	
Erase EOF	PgDn	Erase EOF	Ctrl (E)
Erase Input		Er Inp	Ctrl (N)
Clear	PgUp	Clear	

Print Control Keys

Print keys control the local printer functions that are associated with the display terminal.

Print The Print key initiates the local print function. Data is transferred from the screen image buffer to the associated local printer line-by-line. Attribute information is not shown on the printer in any way.

Identify

The Identify key enables the operator to assign an available printer as the local printer. When the operator guidance message, PRINTER__, appears, press **Identify** to place the cursor in the __ part of the operator guidance message. Complete the printer identifier by typing over the __ with a two-digit printer identifier. If the printer identity is valid, it becomes the identity of the local printer and is maintained in the operator guidance message. Substituting PRN_ for PRINTER__ maps the printer identity to logical printer names. For instance, PRINTER02 sends local copy to PRN2.

Dev Cncl

The Dev Cncl key cancels actions regarding the local copy device. This key stops a local copy print while in progress or an Identify sequence at any point.

Table 50. Print Control Keys

Function	Store Controller Keyboards	31xx Keyboard	Either Keyboard
Print			Ctrl (P)
Identify			Ctrl (L)
Dev Cncl			Ctrl (X)

Miscellaneous Keys

Miscellaneous keys are those 3270 function keys that do not naturally fall into any of the preceding groups.

System Request

The System Request key enables the terminal operator to toggle between either the SSCP and the PLU sessions, or between the SSCP session and the NO SESSION state. In either case, the interactive screen is cleared of data and attributes.

Insert The Insert key puts the keyboard into insert mode so that keystroke data can be inserted at the current cursor position. Characters at and to the right of the cursor within the same field are shifted one character position to the right. The INSERT operator guidance message appears while Insert mode is active. The Reset key returns the keyboard to replace mode.

Reset The Reset key recovers from all local and some remote keyboard locked conditions. The operator guidance prompt causing the inhibition is removed, and operator input is allowed. The purpose for this mechanism is to force the operator to physically acknowledge certain keyboard locked conditions.

Duplicate

The Duplicate key informs the host application program that a duplicate operation is performed for the rest of the field. In addition to the display of the duplicate code (an asterisk at the current cursor position), a Tab key function is performed.

Field Mark

The Field Mark key signals an end-of-field condition to the host application program. Field Mark appears as a semicolon (;).

Operation of the Duplicate key and the Field Mark key when the cursor is in a field attribute character location or within a protected data field disables the keyboard. These keys do not clear character locations, nor do they move the cursor.

Table 51. Miscellaneous Keys

Function	Store Controller Keyboards	31xx Keyboard	Either Keyboard
Sys Req			Ctrl (S)
Insert Mode	Ins & Insert	Insert	
Reset			Ctrl (R)
Dup			Ctrl (D)
Field Mark			Ctrl (F)

Emulator-Specific Keys

Emulator-specific keys provide unique 3270 emulator functions as described below:

Exit The Exit key stops 3270 emulation and returns the terminal to native mode.

OGL Toggle

The OGL Toggle key is for terminals with a limited display area. This key enables an operator to toggle where a line of the display is shared between interactive data and operator guidance messages. If a line of operator guidance is currently displayed, this key replaces it with an interactive line, and back again. The display of operator guidance messages does not, in itself, cause inhibition of keyed data.

Numovride

The Numovride key provides numeric checking override that acts as a toggle in numeric fields. If the cursor is located in a field defined by the outbound data stream as a numeric field, pressing this key causes inhibition of checking of the keyed numeric data in the field. Numovride works as a toggle; where pressing the key again, restores numeric checking. This key is ignored with non-numeric fields. An operator guidance message informs the operator of the current numeric checking state.

Color Mode

The Color Mode key corresponds to the Base Color switch on Toshiba color display terminals and applies only to terminals with four-color capability. The

Color Mode key toggles between two-color and four-color display support. An operator guidance message informs the operator of the current state of the Color Mode switch.

In the two-color (monochrome) state, field coloring depends on the Intensity indicator in the field attribute byte as shown below:

Normal field = Green
Intense field = White

In the four-color state, field coloring depends on the Protect and Intensity indicators in the field attribute byte as shown below:

Unprotected / Normal Intensity = Green
Unprotected / Intensified = Red
Protected / Normal Intensity = Blue
Protected / Intensified = White

Regardless of the state of the Color Mode switch, if the screen is unformatted (no fields defined), the screen is colored green.

Color attributes override the state of the Color Mode switch as defined in the outbound data stream. The display, in this case, is colored by data stream-extended field attributes and character color attributes in the current attribute buffer.

Temporary Exit

The Temporary Exit key suspends the keyboard/screen device so that an application can access the 3270 emulator. This key is used when the operator wants to allow a programmed operator to control the emulation. The emulator accepts no keystrokes until the Online key is pressed. (This does not affect the toggle key that is provided by the operating system.) A message appears on the operator guidance line: Waiting for Application or Waiting for Online.

Online

The Online key enables the operator to re-enable the screen and keyboard after a programmed operator has completed. The system accepts no other keystrokes until the Online is pressed. Also, when an application is using the emulator, it is indicated by a message on the operator guidance line and the Online key is not accepted.

Refresh

The Refresh key causes the emulator to re-draw the screen image.

Table 52. Emulator-Specific Keys

Function	Store Controller Keyboards	31xx Keyboard	Either Keyboard
Exit			Ctrl (Z,Z)
OGL Toggle			Ctrl (W)
Numovride			Ctrl (O)
Color Mode			Ctrl (B)
Temporary Exit			Ctrl (Q)
Online			Ctrl (K)
Refresh			Ctrl (T)

Using 3270 Emulator Printing Functions at the Store Controller

This section describes the 3270 emulator printing functions that can be performed at the store controller.

Local Printing

The 3270 local printing facility provides the ability to print the contents of the display screen to the designated local printer. Local printing is initiated in one of the following ways:

- Operator
- Host application program

Regardless of the initiating source, data is copied from the display and printed at the printer line-by-line. Any lines consisting entirely of nulls are omitted. Data in non-display fields is printed as spaces. Field and lighting attributes on the local copy device are not signified in any method. Field attribute positions appear as spaces.

Operator-Initiated Local Printing

The three printer control keys that control operator-initiated, local printing are **Print**, **Identify**, and **Dev Cncl**. (See “Print Control Keys” on page 508.)

Host Printing

When a 3270 printer emulation starts in the background on the store controller, operator control or intervention is not necessary. As the host starts print jobs, the emulator receives the data and formats it into print spooler output. This output automatically prints at the store controller printer. Host application-initiated, local printing is controlled by the start print bit in the write control character of the data stream. Messages describing the current status of the 3270 printer sessions appear on the background screen. These messages are:

3270 PRINTER SESSION ACTIVE	(LU active, no current print job)
3270 PRINTER PRINTING	(Actively receiving output)
3270 PRINTER SESSION ENDED	(LU not available)
3270 PRINTER SESSION STARTING	(Initializing)

Operator Guidance

All 3270 display screens support a line of operator guidance symbols that indicate the state of the host connection and other display-oriented information. The 3270 Emulator emulates this facility by providing clear, language-textual messages on a line of the display.

On the store controller, the operator guidance messages must share the 25th line of the display screen with interactive data. Under these circumstances, the display of operator guidance information is under the control of both the display terminal operator and the emulation software. The operator has the use of the OGL Toggle key to switch between the display of interactive data and operator guidance messages. Regardless of the current state of the message key, the 3270 emulator causes an operator guidance message to appear automatically whenever a keyboard lock condition arises from incorrect operator keying.

Operator guidance messages provide the operator with the following type of information:

- Outbound data stream validity
- Keyboard locked state
- Keying validity
- Host processor connection
- Printer states
- Field types

Operator Guidance Messages

The remainder of this section describes operator guidance messages of the 3270 emulator. The messages include any necessary operator action.

Application name

The display station is in session with the named host application. The name is extracted from the Bind information that is supplied by the host processor.

To log off, press **SysRq** for the SSCP session and enter the LOGOFF command.

BCOL The display terminal is currently in four-color state. The Color Mode key is used to toggle to the two-color state. This operator guidance is relevant only for display devices capable of color support.

COMM nnn

A hardware communication error has occurred.

Communication check messages refer to the status of the data link control (DLC) connection with the host processor. Comm check codes appear in the operator guidance message area as COMM *nnn* messages, where *nnn* is:

501 - Data Set Ready (DSR) signal from modem has dropped
 502 - Clear to Send (CTS) signal from modem is missing
 504 - Communication line is disconnected
 505 - DISC (disconnect) received from network
 510 - Physical unit (PU) is not active
 519 - Receive buffer overrun
 521 - Time out, idle line
 525 - Connection problem
 528 - Invalid command received
 529 - Abnormal response from modem
 530 - Write timeout

FIELD-FULL

An attempt was made to insert data into a full field.

Press **Reset** to continue.

FUNCTION UNAVAILABLE

A key has been pressed that is currently not valid (for example, a program function key has been pressed when in session with SSCP).

Press **Reset** to continue.

INSERT

Insert mode is active. Characters at and to the right of the cursor are moved to accommodate the inserted character. If there is no space in the field when keying graphic data, a FIELD FULL operator guidance appears.

Press **Reset** to return to replace mode.

KEYSTROKE LOST

A keystroke has been lost, usually by keying at a faster rate than can be serviced.

Press **Reset** to continue.

MONO

The display terminal is currently in two-color state. The Color Mode key can be used to toggle to four-color state. This operator guidance is only relevant for display devices capable of color support.

NO SESSION

The display station is in 3270 emulation, but is not in session with either the host SSCP or PLU. The NO SESSION message appears during:

- The terminal initialization, when no LU session has been established yet with the host system (ACTPU has been received but not ACTLU).
- Session activity, if it is detected that the communication link has been lost or if the session has been suddenly terminated (for example, session unbound). In this case, the NO SESSION message also contains a COMM *nnn* message indicating the source of the error.

If SysRq was pressed, press **SysRq** again and log on, if the SSCP session is active (ACTPU and ACTLU have been received).

NUMERIC

The cursor is in a numeric field. Any key, other than numeric, hyphen (minus), period, or Dup, results in a RANGE CHECK operator guidance message.

Press **Reset** to return to replace mode.

PRINTER BUSY

The operating system is opening a session with the local printer. This action could take some time if the printer is already in use by another device or a native mode application program.

Either wait for the printer to become available or press **Dev Cncl** to cancel the request.

PRINTER ERROR

The local printer is not working and the print request was cancelled.

Press **Dev Cncl** to continue.

PRINTER*nn*

The *nn* is the identifier for the local printing device.

When this operator guidance appears, press **Identify** to position the cursor on the ___ part of the operator guidance. Complete the printer name by overwriting ___ with a 2-digit printer identifier.

If the requested printer is not available or does not exist, PRINTER ERROR appears. Press **Reset** to acknowledge the operator guidance message. The local printer retains the previous printer identifier. Similarly, if **Dev Cncl** is used to terminate the Identify sequence prematurely, the previous printer identity is retained as the local printer.

PRINTING

Local printing is in progress.

PROG *nnn*

A program check was detected. The 3270 emulator can generate program checks.

Press **Reset** to continue or wait for the host processor to recover the terminal.

PROTECTED

An attempt to write data into a protected screen position (for example, field attribute positions and protected data fields) was detected.

Press **Reset** to continue.

RANGE CHECK

An incorrect character was entered for a numeric field.

Press **Reset** to continue.

SSCP The display station is in session with the SSCP.

Type the appropriate logon or logoff command.

SYSTEM *nnn*

The host processor has not explicitly unlocked the keyboard in the WCC.

Press **Reset** to continue.

WAIT The system is waiting for the host application or the 3270 emulator to complete processing.

Most operator actions are inhibited.

Using the Application Programming Interface

The 3270 application programming interface (API) provided by the 3270 emulator enables a programmed operator to carry out all of the functions that a real operator could perform. A typical session could involve the following steps:

- Start the 3270 emulator
- Log on to a host application and then repeatedly:
 - Receive a screen of data from the host application
 - On the basis of the text of that screen, type the appropriate responses (select options, enter values, and so on)
 - Send the responses to the host (by pressing **Enter**, or a program function key, and so on)
- Log off the host application
- Terminate the 3270 emulator

The 3270 API can be used in the store controller to automate all or part of this process with the following restriction. If the session is to be shared between 3270 API applications and users, then ***the emulator must be started in the foreground from the command line, and must be terminated in the foreground (by pressing Exit).***

In the 4683, the command tail, specified as part of the terminal load definition, is the equivalent of the command line in the store controller. Therefore, the equivalent of starting the emulator in the foreground is to specify the emulation type on the command tail as *console*.

For example, the store programmer could provide two 3270 API applications; one to LOGON and one to LOGOFF. The operator starts the 3270 emulator and then, starts the LOGON application. This application waits for the host's welcome screen at the start of a session, and types the LOGON commands. Once logged on, the application turns over control to the operator. After finishing the session activity, start the LOGOFF application that completes the session and logs the operator off

of the host application. The 3270 emulator continues to run in its operating system window until stopped by the operator (with the **Exit** key), if running in the store controller.

While the application is using the 3270 emulator, the operator cannot use it. Except for pressing **Exit** twice, keystrokes entered at the keyboard are ignored, and the screen is not updated. Conversely, while the operator has control of the 3270 emulator, it is not accessible to any 3270 API application.

The 3270 API can be accessed from within user exits in the sales applications. For example, a user exit invoked when a terminal operator signs on to the sales application can simultaneously log the user onto a 3270 session. In this example, the 3270 API application is Terminal Sales.

As another example, the 3270 API application can be written so that it needs no user intervention. A 3270 API application could then:

- Start the emulator
- Detect the welcome screen and log on to the host
- Search each screen of data that is received from the host to determine the required responses, and then type the responses
- Log off, stopping the 3270 emulator

In this example, the 3270 emulator runs in the background, and does not need access to the screen and keyboard of the store controller or to a 4683, 4693, 4694, SurePOS 300/700 or TCxWave 6140 Series terminal.

Operator Guidance from the Application Programming Interface

For a 3270 console session that is shared with an application (for example, sometimes used by a real operator; sometimes by a programmed operator), the operator guidance line (OGL) is used to indicate the status of the application. The APPLICATION RUNNING message replaces the text of the OGL while the session is being controlled by the programmed operator:

When toggled into the 3270 session window, the presence of this message on the OGL indicates that the session is in use by a program and that the keyboard and display are locked (that is, keystrokes cannot be entered). When the application gives up control of the session, the OGL text is replaced by the WAITING FOR APPLICATION OR ONLINE message:

The 3270 session can be used by pressing **Online, Ctrl (K)**.

Note: Use this key sequence only with 3270 emulation in the controller. The corresponding key on the terminal is the **Online** key. Alternatively, another application can access the session, in which case, the message is replaced by APPLICATION RUNNING again.

See the *4690 OS: Communications Programming Reference* for additional information about how to access the API from a 4680 BASIC program.

3270 Emulation on the Terminal

The emulation services available on the terminal mirror those available at the store controller, except for the printer functions that are identified. Users of the 4683, 4693, 4694, SurePOS 300/700 and the TCxWave 6140 Series can do the following items:

- Use a terminal with a full-screen monitor and ANPOS, 4693 ANPOS, or USB ANPOS keyboards as a 3278 display station
- Use a serially-attached, Proprinter to receive a 3270 printer data stream
- Run programmed operator applications using the 3270 API; if the session is not shared with an operator, the requirement to have a full-screen monitor/ANPOS keyboards is removed

These facilities are optional. For any given terminal or any store, the default configuration is with no 3270 emulation capability.

The emulation services are provided using a set of applications. ADXZE30L.286 runs on the 4683, 4693, 4694, SurePOS 300/700 and TCxWave 6140 Series terminals and provides the 3270 data stream management (formatting and display). ADXHST0L.286 and ADXHSK0L.286 are applications that run on the gateway controller. These two applications manage the transfer of data between the emulation on the terminal and the link on the gateway controller.

A single copy of ADXZE30L supports one 3270 session (either screen or printer); therefore, a fully-configured terminal runs two copies of this application — one to manage the printer session and one to manage the screen session.

There is one copy of ADXHSK0L on each gateway controller. If no terminals on a TCC Network are configured to support 3270 emulation using that gateway, then ADXHSK0L is not needed. ADXHSK0L starts as many copies of ADXHST0L as are needed, when they are needed, and halts them when they are no longer needed. Each copy of ADXHST0L can support up to 20 emulations. However, the users do not need to do anything to arrange for sufficient ADXHST0Ls to be running because ADXHSK0L manages this automatically.

ADXHSK0L runs in the store controller to support 3270 emulation in the terminal. It runs in the background and an operator can manually run the 3270 emulation. This is not a normal procedure and should be avoided. If it is done, all 3270 sessions in the terminals must be taken offline before ADXHSK0L can be successfully restarted.

3270 console sessions in the terminal can be taken offline by pressing **Offline** or **EXIT**. Use of the EXIT key stops the emulator and requires reloading the terminal to restart the emulator. Sessions in the terminal can also be taken offline from a user application by using the 3270 API DISCONNECT verb.

3270 printer sessions in the terminal are taken offline only by terminating the 3270 emulator. This requires that the terminal be reloaded to restart the emulator.

Configuration

The following list shows the configuration information needed for terminal emulation:

- Define the gateway controller configuration to have:
 - ADXHSK0L.286 as a background application
 - A controller RAM disk (T:) installed

- Define the SNA communications configuration on the gateway node store controller
- Define the terminal load to include the emulation and specify the command tail

See the *4690 OS: Planning, Installation, and Configuration Guide* for additional details about the command tail and configuring the terminal.

Using 3270 Emulation on the Terminal

This section describes some unique aspects of using 3270 emulation on the terminal.

Screen and Keyboard Functions

Bell: In addition to the unsupported functions that are listed for the store controller (such as alternate cursor and cursor blink), the Bell function is not supported on the terminal.

The 3270 data-stream includes a command (BEL) to sound a tone at the 3270 display station. This command is supported on the store controller emulation but not at the terminal. The emulator does not use the tone device, which allows that the sales application can exclusively access it. If an operator is currently in the 3270 emulation and a tone sounds, then an action is required in the sales application.

Toggle: The operating system initiates the terminal emulation as soon as the application completes loading. The terminal can run up to two foreground applications (such as sales or a 3270 screen emulation). To toggle between the two possible foreground applications, use the toggle sequence, **Alt+Esc**, which is provided by the operating system.

Starting a Session: Unlike the store controller emulation on the terminal, which contacts the SNA driver immediately, the emulator does not automatically initiate a SNA session. Store controller emulations are started only when needed by the operator or the applications. Terminal emulation is active as soon as the terminal completes loading. To save resources (LUs), the terminal screen emulation does not begin a SNA session until prompted by the operator or by the application.

To start a SNA session on the terminal, toggle to the emulation's window and press **Online**. An application issues the CONNECT verb.

Keying Sequences for 3270: When 3270 is in the foreground, keystrokes from the ANPOS, 4693 ANPOS, or USB ANPOS keyboards (except for the toggle sequence) are directed to the 3270 emulation. They are used in the same way as on the store controller, which is to provide the equivalent of 3270 graphic, function, and control keys.

For the 3270 emulator, the important data is the keycode. Except for the Clear, Insert, Delete, and cursor movement keys, the key positions can be reconfigured to choose which key to use for each 3270 function.

Note: The key locations for Insert, Delete, and cursor movement can be used for other functions. But the Insert, Delete, or cursor movement functions must be used at their default location. The Clear key is always the Esc key.

The terminal has one additional key called Offline. The Offline key ends the SNA session without ending the emulation itself. On the store controller, Exit ends the SNA session and stops the emulation. The emulation can be restarted at any time.

Also, on the terminal, Exit ends the session and stops the emulation, but the emulation cannot be restarted without reloading the terminal. Therefore, the Offline key is used to end the SNA session without stopping the terminal emulation. The session can be restarted using the Online key. The 3270 auto-offline function sends Offline to the 3270 application each time the toggle sequence is used to toggle away from the 3270 session. The ADX_SDT1:ADXZE3_F.DAT file activates this function. The only requirement for this file to activate the function is that the file can be opened at the time the 3270 application is invoked. This function does not reference the file's contents.

The 3270 functions assigned to the keys of the ANPOS, 4693 ANPOS, and USB ANPOS keyboards are the same as for emulation in the store controller. See "Using 3270 Emulator Keyboard Functions at the Store Controller" on page 504 for a description of these functions. The key sequence for obtaining these functions on the ANPOS, 4693 ANPOS, or USB ANPOS keyboards is not always the same as on the store controller. For example, press **Reset** on the ANPOS, 4693 ANPOS, or USB ANPOS keyboard rather than the **Ctrl+R** key sequence to do a reset. As indicated above, the location of the 3270 functions on the ANPOS, 4693 ANPOS, or USB ANPOS keyboards is configurable. The default layout is provided by the keyboard layout described previously in this appendix.

Unsupported Keys: The 3270 emulator does not support the double zero (00) and triple zero (000) keys on the ANPOS, 4693 ANPOS, or USB ANPOS keyboards. No response is obtained if they are pressed.

Reconfiguring the Keyboard

The operating system allows the terminal keyboards to be reconfigured. Using menus on the store controller, the keys on the keyboard can be rearranged to produce different keycodes.

Note: Ensure that 3270 emulation is not adversely affected when reconfiguring the keyboard.

Using Printing Functions at the Terminal

For host printer sessions, the terminal uses one of the serially-attached Proprinters. The operation of the printer session is entirely automatic as far as the terminal operator is concerned. When configured for a host printer session, the emulator on the terminal starts up as soon as it has been loaded and started by the terminal application loader. Print jobs are started and stopped under host control.

On the store controller, the status of the host printer sessions is in the system event log. The terminal printer sessions log messages to the system event log. See the *4690 OS: Messages Guide* for additional information about these messages.

Local copy printing on the terminal works in the same way as on the store controller. The 3270 emulator can be configured to use a printer at the store controller or at the terminal for local copy output. The Identify key can be used to redirect local copy printing to a different printer. However, it cannot alter the configured type (store controller or terminal) of printer. Hence, if a terminal screen emulation is configured to use CONTROLLER PRINTER 1, the Identify key can be used to redirect output to a different printer on the same controller. If the terminal is configured to use TERMINAL PRINTER 1, the Identify key can be used to redirect output to a different printer on the terminal. The Identify key cannot be used to switch output from the store controller to the terminal printers, or from the terminal printers to a store controller printer.

API

The terminal emulator offers the same API as the store controller emulator. The application program interface is fully described in the *4690 OS: Communications Programming Reference*.

Starting and Stopping the Terminal Emulator

The emulation is always started by the terminal application loader. It cannot be started by the application because the SHARED parameter of the S3.CONNECT is not relevant. It cannot be stopped by the application because the TERMINATE option of S3.DISCONNECT is not relevant. Emulation in the terminal cannot be restarted without reloading the terminal.

Appendix C. Using the Supplemental Option on 4690

This appendix describes the functions and use of Supplementals.

In this chapter, the term "Supplementals" is used to refer to the Supplemental Option.

Using the Supplementals

The Supplementals contain a subset of operating system functions and run alone without any dependency on the operating system that is stored on the fixed disk. Use the Supplementals for correcting problems with the fixed disk and for performing certain system functions and not for normal daily operation.

Note: Before the functions of the Supplementals can be used, you must specify the type of national language your keyboard uses. You should specify the national language as soon as you receive your system. See "Specifying a National Language for the Supplementals" on page 525 for instructions on specifying the national language.

Using the Supplemental Option on Enhanced 4690 Controllers

The following options are available for booting Supplementals on Enhanced 4690 controllers:

- A bootable CD to boot Enhanced Mode Supplementals can be generated using Enhanced Options Menu option **3 Generate supplemental CD**.
- A bootable memory key to boot Enhanced Mode Supplementals can be generated using Enhanced Options Menu option **4 Generate supplemental memory key**.
- Supplementals can be booted from the hard drive using Enhanced Options Menu option **2 Start 4690 supplemental OS from hard drive**.
- Supplementals can be booted over the network from a source controller. See "Booting Enhanced Supplementals over the network" on page 522 for details.

Supplementals fallback: To prevent an IPL loop, an Enhanced 4690 controller can be configured to automatically boot Supplementals from the hard drive when the controller dumps twice within a configured time. To modify this behavior, you can add a property to the user properties file (c:\adx_idt1\vx_ctlpr.dat) then IPL to activate the change. The property is `dump.loop.detect.seconds=number` where the number is either 0 or the number of seconds to use as an interval. This property must be present to use this feature. If it does not exist, or if you set it to 0, the function is completely disabled.

Using the Supplemental Option through the CD-ROM

The CD-ROM is an ISO 9660-compliant, data CD-ROM.

When using 4690 OS V3R1 and V3R2, a bootable diskette is provided along with the CD-ROM. You must **always** IPL or start the system with the **boot diskette** in diskette drive A: and the **CD-ROM** in the CD-ROM drive.

When using 4690 OS V3R3 or later, you can IPL or start the system with the **CD-ROM** in the CD-ROM drive. A boot diskette is not needed.

Notes:

1. If your controller is running in Classic Mode, you must use the Classic Supplemental CD.
2. If your controller is running in Enhanced Mode, you can use the normal Supplemental but in order to boot it, you must either power off/on the controller or use the Enhanced Options Menu to cold start the machine (requires more than using Ctrl-Alt-Del).

Booting Enhanced Supplementals over the network

Enhanced Supplemental boot over a network is accomplished by setting up a target controller to load the Supplemental boot image from a source controller, similar to a network-booted terminal. These are the steps to perform a network-enhanced Supplemental boot:

1. Ensure the BIOS of the target controller is configured to optionally allow a network boot using either the alternate boot sequence or the boot menu.
2. Enable Supplemental boot using the `dhcpcinfo` utility on all source DHCP configured controllers as follows:

```
dhcpcinfo -b supps
```

Notes:

- a. By default, network supplemental boot is enabled for a period of 5 minutes. If you want more or less time, use the `-v` parameter of `dhcpcinfo` to modify the time from 1-60 minutes. For example, **`dhcpcinfo -b supps -v 10`** extends the period of time the Supplemental boot is enabled to 10 minutes. You can enable the Supplemental boot without a timeout period by specifying 0 as the time out (**`dhcpcinfo -b supps -v 0`**).
- b. If there are multiple active controllers on the network running a DHCP server, any of these controllers might respond as the source controller for the network Supplemental boot. In this case, configure the Supplemental boot option on each controller or DHCP server on the network using **`dhcpcinfo -b supps [-s y.y.y.y]`** where `y.y.y.y` is the IP address of the DHCP server (the `-s` option is not required if the DHCP server is the local controller).

Note: Refer to the 4690 OS Communication Programming Guide for more information on the `dhcpcinfo` utility.

3. Power off/on the target controller and network boot the target controller by selecting the alternate boot sequence or the boot menu. This step forces a PXE boot of the target controller. The PXE Supplemental boot is very similar to an Enhanced terminal PXE boot and it displays the same messages.
4. When U005 is displayed, press the CTRL key on the target controller within 5 second to boot the Supplemental package instead of the normal terminal boot. The message 4690 Supplemental OS is displayed on the attached video display, indicating that the Supplemental image is loading.
5. The target controller loads the Enhanced Supplemental image from the source controller and boots to the Enhanced Supplemental Menu.
6. Network supplemental boot mode is automatically disabled in 5 minutes by default or in the time period specified with the `-v` parameter on the **`dhcpcinfo`** command. If the `-v` time period is specified as "0" (no time-out), network supplemental boot mode is manually disabled using the **`dhcpcinfo`** utility as follows:

```
dhcpcinfo -b 0
```

Note: The DHCP Supplemental boot option is active only in DHCP server memory and is not retained on a controller IPL. Therefore, IPLing the controller will also disable the Supplemental boot option.

Securing the Supplementals

The Supplementals provide the user access to important functions of the system; therefore, they are only for personnel who need to perform those functions. Once the Supplementals have been loaded, no special authorization, or ID, or password is required. The user of the Supplementals automatically has a user ID of zero, a group ID of zero, and can request any of the available file commands on the Supplementals.

Note: Service personnel might need access to the Supplementals for some corrective actions. Keep the Supplementals in a secure location where they are available only to designated personnel who need to use the functions on them. Restricting access to the Supplementals is your responsibility.

Functions on the Supplementals

Use the Supplementals for the following functions:

- To correct any problems with the operating system stored on the fixed disk
- To enable or disable the store controller RAM disks
- To provide a subset of the operating system commands needed for:
 - Formatting, copying, or comparing diskettes^{1, 2}
 - Backing up and restoring data on the fixed disk (using diskettes or streaming tape)²
 - Managing the subdirectories on the diskette and the fixed disk
 - Copying or comparing files
 - Enabling and disabling security on diskettes
- To enable changing or correcting the national language selection:
 - on the fixed disk
 - for the Alphanumeric Keyboard, which is attached to the point-of-sale terminal
 - for the code page used at the host
- To recover from a fixed disk failure using the Disk Rebuild Utility

Notes:

1. Formatting the hard drive (**CPREP** and **DPREP**) is not supported when Supplementals is running from the hard drive.
2. Use of the CD-ROM drive and of the Optical Disk Utility is not supported when Supplementals is running from the hard drive.

Some of the functions on the Supplementals are available from the STORE CONTROL FUNCTIONS panel, some are available from the SYSTEM MAIN MENU, and some are available from Command Mode. The following sections describe the available functions of the Supplementals and how to access them.

Using Store Control Functions

After you IPL from the Supplementals, the STORE CONTROL FUNCTIONS panel is accessible by pressing **SysRq**, and then pressing the letter **C**. Only a subset of the functions normally available from the STORE CONTROL FUNCTIONS panel is on the menu. The following outline shows the functions that are available. See “Using store control functions” on page 91 for more information on how to use these functions.

- Controller Functions
 - Enable the store controller RAM disk
 - Disable the store controller RAM disk
- System Functions
 - Set the system date and time
 - Set the System Message Display Level

Using the Menu Panels

The SYSTEM MAIN MENU is the first panel that is displayed after the system has completed the IPL process from the Supplementals. Because only a subset of functions is on the Supplementals, the SYSTEM MAIN MENU does not show all the options that are normally shown on this panel.

The following panel shows the SYSTEM MAIN MENU:

SYSTEM MAIN MENU

Select one of the following:

- 1
- 2
- 3 File Utilities
- 4 Installation and Update Aids
- 5 Command Mode

Type your selection number, then press ENTER _1_

F1 F2 F3 F4 F5 F6 F7 F8 F9Signoff

The options shown on the menu represent the tasks that can be performed using the Supplementals. To choose an option from the menu, type the number of the option you want to access, and then press **Enter**. The following sections describe the tasks that can be performed using each option on the menu.

Using File Utilities

The File Utilities option enables you to display or change file data, activate system file changes, and start the Streaming Tape Drive Utility. From the SYSTEM MAIN MENU, type **3** to select the File Utilities option and press **Enter**. The following menu appears:

FILE UTILITIES

Select one of the following:

- 1. Display/Alter File Data
- 2. Display/Alter in Physical Mode for Hard File
- 3. Reserved
- 4. Streaming Tape Drive Utilities
- 5. Optical Drive Utilities

Type your selection number, then press ENTER _1_

F1 F2 F3Quit F4 F5 F6 F7 F8 F9 F10

Displaying and Altering File and Disk Sectors: This option allows you to display or alter a sector of file or disk that is formatted for use on this system. For more information on these functions, see “Display/Alter File Data” on page 83.

Using the Streaming Tape Utilities: If you have an optional streaming-tape drive attached to your store controller, the Supplementals provide a utility to allow use of this device. See “Streaming Tape Drive Utilities” on page 83 for more information.

Using the Optical Drive Utilities: If you have an optional optical drive attached to your store controller, the Supplementals provide a utility to allow use of this device. See Chapter 18, “Using the optical drive utility,” on page 393 for more information.

Using Installation and Update Aids

To access the Installation and Update Aids from the SYSTEM MAIN MENU, type **4** and press **Enter**. The only function available from this selection is Apply Software Maintenance.

Apply Software Maintenance: The Apply Software Maintenance (ASM) Utility on the Supplementals can be used only to **cancel** maintenance for the operating system.

Note: If you are using the Supplementals, the **cancel** ASM step must be performed at each store controller that is on a multiple store controller system.

After selecting Apply Software Maintenance from the panel, the controller automatically IPLs. No other panels appear after the selection is made. When the store controller begins checking memory for the IPL, remove the Supplementals from the drive so that the ASM can run from the hard disk drive.

If the ASM does not run from the hard disk drive, IPL from the Supplementals. Inserting the Supplementals forces the IPL portion of ASM to run from the Supplementals.

Working in Command Mode

The Command Mode option lets you enter commands that help you create directories, copy, compare, or erase files, and run programs to change the national language selection on your Supplementals, hard disk drive, or Alphanumeric Keyboard. To access Command Mode from the SYSTEM MAIN MENU, type **5** and press **Enter**. The following sections describe the functions available from Command Mode.

Specifying a National Language for the Supplementals

When you installed your operating system on the hard disk drive, you had to specify a national language to be used on your keyboard. Because the Supplementals do not use the information that is stored on a hard disk drive, you must specify a national language as well.

Use the following procedure to specify a national language for use on the Supplementals. When entering data during the procedure, you must use only the numeric keypad section of your keyboard unless the instructions specify otherwise.

To specify a national language for the Supplementals:

1. At the system prompt, type **T:>222** and press **Enter**

Note: You are prompted to remove the write protection from the Supplemental Diskette 1 and insert it into the drive. Slide the tab on the diskette so that the square hole in the corner of the diskette is blocked. This enables you to write information on the diskette.

2. Insert the diskette and press **Enter**. A list of national languages appears.
3. Using the numeric keypad, type the number shown beside the language that your keyboard uses. Press **Enter** and your national language information is placed on the diskette.
4. Remove the diskette and slide the tab so that the square hole is not blocked.

Attention: Be sure to replace the write-protect tab over the notch on the diskette. If you do not replace the tab, you can accidentally write over the information on this diskette.

Note: If you make a mistake entering the selection number or if you decide to change the selection, use this program again.

When finished, make sure that the diskette drives are empty and then IPL your controller (by pressing **Ctrl**, **Alt**, and **Del** keys together) from the hard disk drive.

Changing National Language Selection/Hard Disk Drive and Keyboard

You can use a program on your Supplementals to change or correct the national language that you previously selected for your hard disk drive.

To change the previous language selection:

1. At the system prompt, type **T:>333** and press **Enter**. A list of national languages appears.

Note: Use the numeric keypad to enter your data if you have not already specified a national language for the Supplementals.

2. Type the number shown beside the language that your keyboard uses. Press **Enter** and the national language information is changed on the hard disk drive.

Note: This program changes the national language for any ANPOS or PS/2-attached keyboards on your terminal, controller, or controller/terminal.

This program also changes the text mode character set for these displays:

- Any video displays that are attached to your controller
- The controller side of a video display attached to your controller/terminal

To change the text mode character set for any video displays attached to your terminal or for the terminal side of a video display attached to your controller/terminal, you must perform the steps described in "Changing National Language Selection for Point-of-Sale Keyboard" on page 527.

This program can be used as many times as needed.

When you are finished using the Supplementals, make sure that the diskette and CD-ROM drives are empty and then IPL your controller from the hard disk drive.

After the IPL has completed, perform these steps in order to load the changed national language for your terminal's ANPOS or PS/2-attached keyboard:

1. Press **Alt+SysRq** to display the SYSTEM KEYS menu.
2. Select option **c** (Access the Store Control Functions Screen).

3. Select option **1** (Terminal Functions) from the STORE CONTROL FUNCTIONS menu and press **Enter**.
4. Select option **7** (Load Terminal Storage) from the TERMINAL FUNCTIONS menu and press **Enter**.
5. Enter the terminal number of the terminal where the changed keyboard information is to be loaded.

Changing National Language Selection for Point-of-Sale Keyboard

You can use a program on your Supplementals to change or correct the national language that you previously selected for the point-of-sale keyboard on your terminal.

To change the previous language selection:

1. At the system prompt, type **T:>444** and press **Enter**. A list of national languages appears.

Note: Use the numeric keypad to enter your data if you have not already specified a national language for the Supplementals.

2. Type the number shown beside the language that your terminal keyboard uses. Press **Enter** and the national language information is changed on the hard disk drive.

Note: This program also changes the text mode character set for these displays:

- Any video displays that are attached to your terminal
- The terminal side of a video display attached to your controller/terminal

To change the text mode character set for any video displays that are attached to your controller or for the controller side of a video display that is attached to your controller/terminal, you must perform the steps described in “Changing National Language Selection/Hard Disk Drive and Keyboard” on page 526.

This program can be used as many times as needed.

If you are finished using the Supplementals, make sure that the diskette and CD-ROM drives are empty and then IPL your controller from the hard disk drive.

After the IPL has completed, perform these steps:

- Press **Alt+SysRq** to display the SYSTEM KEYS menu.
- Select option **c** (Access the Store Control Functions Screen).
- Select option **1** (Terminal Functions) from the STORE CONTROL FUNCTIONS menu and press **Enter**.
- Select option **7** (Load Terminal Storage) from the TERMINAL FUNCTIONS menu and press **Enter**.
- Enter the terminal number of the terminal where the changed keyboard information is to be loaded.

Changing National Language Selection for Host Translation Table

You can use a program on the Supplementals to select, change, or correct the country that is needed or previously selected for the host translation table. The program also enables you to select whether to use the country-specific version or international version of the table.

To change the previous language selection:

1. At the system prompt, type **T:>555** and press **Enter**. A list of national languages appears.

Note: Use the numeric keypad to enter your data if you have not already specified a national language for the Supplementals.

2. Type the number shown beside the language that your host processor will use, and press **Enter**.
3. You must select either the international or country-specific code page. The translation table you select must be compatible with the translation table that is used at the host (partner) site. Contact your host site to determine which translation table to select.

This program can be used as many times as needed.

If you are finished using the Supplementals, make sure that the diskette and CD-ROM drives are empty and then IPL your controller from the hard disk drive.

Preparing Your Hard Disk Drive Using the Supplementals

The Supplementals also contain batch files that help to prepare your hard disk drive if you are replacing or adding a drive. These files define the hard disk drive as a single partition, format the partition, install the operating system loader for drive C, and create the appropriate subdirectories.

Attention: Using these batch files to prepare your hard disk drive destroys any data already on the hard disk drive.

To use the batch files to prepare your hard disk drive:

1. IPL from the Supplementals and select **Command Mode** from the menu.
2. When the system prompt appears, type the appropriate batch file name on the command line. For example, if you are preparing drive C, enter this command: **T:>CPREP**. If you are preparing drive D, enter this command: **T:>DPREP**.

Commands on the Supplementals

The Supplementals also contain a subset of the operating system commands. All of these commands are issued from Command Mode by typing the name of the command at the system prompt.

These are the commands on the Supplementals:

- ADXCSU1L
- ADXFSM3Z
- ADXFSM6Z
- ADXNSX0L
- BACKUP
- CHDIR
- CHKDSK
- CLS
- COMP
- COPY
- DIR
- DISKCOMP
- DISKCOPY
- DISKSET
- DVRLOAD*

- DVRUNIT*
- DVRUNLK*
- ERASE
- FORMAT
- FSET
- MKDIR
- RENAME
- RESTORE
- RMDIR
- SYS*
- VOL

Commands marked with an asterisk (*) are designed for use only on the Supplementals. These commands are described in the following section. For a description of other commands, see Chapter 7, “Command summary,” on page 153.

Command Format

The commands in this appendix use the following command format notation:

- [] Information shown inside the square brackets is optional. To include optional items, type only the information inside the brackets. Do not type the brackets.
- CAPS** Information shown in capital letters are called keywords. Keywords must be entered exactly as shown except the keywords can be entered in any combination of uppercase or lowercase letters. Your system automatically converts keywords to uppercase.
- italics* Information shown in italics represents items where you substitute or supply unique information. For example, *file name* in the command format means that you should substitute the name of your file in place of the word *file name*.
- | A vertical bar means that you can enter one of the items that are separated by the bar, but not both. For example, the format
ON|OFF
indicates that you can enter either ON or OFF. Do **not** type the vertical bar.
- ... An ellipsis indicates an item that can be repeated as many times as needed.

You must include all punctuation that is shown in the command format such as commas, slashes, backslashes, equal signs, and colons.

The commands in this appendix are listed in alphabetical order. Included with each command is a summary of the following information:

- Purpose of the command
- Format of the command
- Type of command
 - *Internal* — indicates that the command processing is resident in the operating system and can be used at any time.
 - *External* — indicates that the command is stored on a disk and must be read from the disk before it can be executed.
- Description of the command
- How to use the command

Unless otherwise noted, command names cannot be abbreviated.

Note: Many commands result in an immediate response from the system in the form of output to your display or printer (or other specified device). In some cases, the output contains names of disk drives and these disk drive names are different from the standard disk drive names (for example, A, B, C, or D).

The disk drive names and their equivalents are as follows:

f0	=	drive A	h0	=	drive C
f1	=	drive B	h1	=	drive D

ADXCSU1L Command

Purpose

To back up and restore the distribution attributes for all distributed files on both the C: and D: drives to the file named C:\ADX_SDT1\ADXCSU1F.DAT.

Formats

ADXCSU1L 1

ADXCSU1L 2

where 1 backs up the file, and 2 restores the files.

Type

External

Used normally from the Supplementals.

Description

The ADXCSU1L command backs up the distribution attributes for all distributed files on both the C: and D: drives to the file named C:\ADX_SDT1\ADXCSU1F.DAT. Also, the original attributes can be restored using this same program. The purpose of this command is to recover the distribution attributes after they have been reset, for example, by using a DOS backup program to restore a system.

This program is normally run from the Supplementals to ensure that there are no files open that would prevent setting the distribution attributes. The program is also used to ensure that there are no changes to the number of files that exist or to the distribution attributes of the files between the time that they are backed up and restored.

However, because the backup portion of the procedure does not need to open the files to read the distribution attributes, it could be run while running from the hard drive. Running the command from the hard disk drive is useful whenever a backup set of files on the system is needed while the controller is up. The command can be run from Command Mode or a BATCH file. See below for examples of backing up and restoring distribution attributes.

If ADXCSU1L failed with a fatal error and was not able to attempt the command on all files, the errorlevel is set to 2. If ADXCSU1L failed because one or more files failed but the program was able to continue attempting the command on all files, the errorlevel is set to 1. If ADXCSU1L completed successfully, the errorlevel is set to 0.

Example

To back up the current attributes to a file, boot the Supplementals and select **Command Mode** from the SYSTEM MAIN MENU. Type: **ADXCSU1L 1**. If an error occurs, see C:\ADX_SDT1\ADXCSUEF.DAT for error messages.

To back up the C: drive (and D: drive, if applicable), boot the bootable diskette that supports the backup device. Some devices have device drivers that must be installed on a bootable DOS diskette. Use the backup program on the bootable diskette to backup the C: drive (and D: drive, if applicable).

To create the system on the C: drive, boot the Supplementals. Select **Command Mode** from the SYSTEM MAIN MENU. Run CPREP (and DPREP, if applicable) to format a new hard disk drive.

To restore the C: drive (and D: drive, if applicable), boot the bootable diskette that supports the backup device. Use the backup program on the bootable diskette to restore the C: drive (and D: drive, if applicable).

To restore the original attributes, boot the Supplementals and select **Command Mode** from the SYSTEM MAIN MENU. At the prompt, type: **ADXCSU1L 2**. If an error occurs, see C:\ADX_SDT1\ADXCSUEF.DAT for error messages.

To restore the system to the other controllers on the LAN, boot the Supplementals on the store controllers that are to be rebuilt and select **Command Mode**. Run CPREP (and DPREP, if applicable) to format a new hard disk drive. Then, verify the presence of an acting master store controller and file server and start the Disk Rebuild Utility. Enter the LAN node ID of another store controller on the system and enter the name of the hard disk drive (C or D) to be rebuilt. Finally, enter the LAN node ID of the store controller to be rebuilt.

The following example shows how to use a BATCH file to run ADXCSU1L.

```
ECHO OFF
REM Backup the distribution attributes for all files on C:
REM (and D: if it exists).
ADXCSU1L 1
    IF ERRORLEVEL 2 GOTO FATAL
    IF ERRORLEVEL 1 GOTO BYPASSED
    IF ERRORLEVEL 0 GOTO SUCCESS
:FATAL
    ECHO ADXCSU1L failed with a fatal error and was not able to
    ECHO attempt the command on all files.
    GOTO END
:BYPASSED
    ECHO ADXCSU1L failed because one or more files failed,
    ECHO but the program was able to continue attempting the
    ECHO command on all files.
    GOTO END
:SUCCESS
    ECHO ADXCSU1L completed successfully.
:END
```

ADXF3M3Z Command

Purpose

To convert a hard drive with an existing legacy 4GB 4690 OS partition to the larger partition support available in 4690 OS V4R1.

Note: This command can only be used in Classic Mode.

Formats

ADXF3M3Z C: – converts the C: drive only

ADXF3M3Z D: – converts the D: drive only

ADXF3M3Z C: D: – converts both the C: drive and the D: drive

Type

External

Description

ADXF3M3Z can be used with the IPL Command Processor during time frame 1 or from the command line when running on the supplemental CD. The command should not be run from command mode when the system is booted off the hard drive.

This command performs the following steps:

1. Creates a new partition that begins after the existing legacy 4GB partition
2. Prepares the new partition for booting if the disk being converted is the C: disk
3. Copies the data from your existing 4GB partition to the new partition
4. Enables the new partition to be used by 4690 and disables the old partition

The legacy 4GB partition remains on the disk as a hidden partition that is not available to 4690 OS and that continues to use space on the hard drive.

Notes:

1. In order to reduce the quantity of data that must be copied from the old partition to the new partition, the store controller storage dump file (c:\adxcslcf.dat) and the terminal storage dump file (c:\adx_sdt1\adxcsltf.dat) are truncated prior to performing the conversion. The controller dump file is resized when the controller is reloaded, and the terminal dump file is resized the next time a terminal dump occurs to this store controller.
2. When running ADXF3M3Z in the IPL Command Processor, do not attempt to run any other commands. The store controller reboots as part of the procedure, and the IPL Command Processor file (ADX_SDT1:ADXILIOF.DAT) is erased to prevent running the command twice.

See also “ADXF3M6Z Command” on page 535. The ADXF3M6Z command can be used to switch back to the legacy 4GB partition in the event of catastrophic failure.

Refer to the *4690 OS: Programming Guide* for detailed information about using the IPL Command Processor.

Examples

This example shows the command to convert the C: drive when booted off the supplemental CD.

ADXFSM3Z C:

This example shows the contents of the IPL Command Processor command file to convert both the C: and D: drive on all controllers.

-1 COMMAND -C ADX_SPGM:ADXFSM3Z.BAT C: D:

This example shows the contents of the IPL Command Processor command file to convert only the D: drive of controller CC.

-nCC -1 COMMAND -C ADX_SPGM:ADXFSM3Z.BAT D:

ADXFMS6Z Command

Purpose

To go back to the legacy partition if a catastrophic failure occurs after running command ADXFMS3Z to convert a disk with an existing 4GB partition to use the larger hard disk partition.

Note: This command can only be used in Classic Mode.

Formats

ADXFMS6Z

No parameters are required. Both the C: and D: drives are checked to see if they should be backed off.

Type

External

Description

The ADXFMS6Z command reactivates the legacy 4GB partition while disabling the larger hard disk partition.

This command can be run at time frame 1 of the IPL Command Processor or from command mode when running on the supplemental CD. This command should not be run from command mode when the system is booted from the hard drive.

Note: This command returns the hard drive to exactly the state it was in when the ADXFMS3Z command was run. If you made any updates to files since the conversion, the updates are permanently lost.

Examples

This example shows running ADXFMS6Z from command mode when booted off the supplemental CD.

```
T:> ADXFMS6Z
```

This example shows the IPL Command Processor command file that backs off the changes on all store controllers in the store network.

```
-1 COMMAND -C ADX_SPM:ADXFMS6Z.BAT
```

ADXNSX0L Command

Purpose

To clear CMOS on a store controller.

Formats

ADXNSX0L

ADXNSX0L *ii*

Where *ii* is a valid 2-character Node ID.

Type

External

Used only when the store controller has been IPLed from the Supplementals.

Description

The ADXNSX0L command can be used to clear CMOS, if a problem is suspected in recovery from a power line disturbance (PLD), or if the Node ID in CMOS does not match the node-dependent, configuration files on the hard disk drive.

Example

```
T:>ADXNSX0L
```

In this example, the current controller Node ID is read, CMOS is cleared, and the original Node ID is restored.

```
T:>ADXNSX0L DD
```

In this example, CMOS is cleared and the Node ID is restored to DD.

Note: The node ID specified must be a controller that matches the existing configuration files or the controller does not complete the IPL. Node dependent configuration files are located in C:\ADX_SPGM and are of the form ADXD*xii*F.DAT

DVRLOAD Command

Purpose

To load a driver from a disk.

Formats

DVRLOAD *device_name file_name access_level*

Type

Internal

Used only when the store controller has been IPLed from the Supplementals.

Description

The instructions for the driver are contained in the specified file. DVRLOAD can be entered from the command line on the Supplementals.

Use DVRLOAD if the type of device you are installing is not installed in your system.

device_name

Is the logical name of the first unit that you are installing. The colon (:) is optional in this logical device name.

file_name

Is the name of the .286 file that you are installing.

access_level

Determines how the device driver can be used. You set the access level by including one or more parameters with the DVRLOAD command. The parameters are:

- P = Permanent Driver. This parameter means that once the driver is installed, it cannot be removed. You might not want some drivers to be permanent. For instance, you might want to install your printer drivers so that you can remove them at a later date by using the DVRUNLK command to remove a driver. Other drivers, such as the drivers for your hard disk drive, should be installed so that no one can remove them.
- R = Raw Read Access Allowed. This parameter means that you can read the data as it appears on the media. You do not see file names or the relationships of files within directories.
- W = Raw Write Access Allowed. This parameter means that you can write to the data as it appears on the media.
- S = Raw Set Access Allowed. This parameter means that you can change the attributes about the driver itself. For example, you can change the country code or the baud rate of a console.
- E = Exclusive Access Only. This parameter means that only one user at a time can work with the device. You would probably specify the E parameter when you load a printer driver.
- L = Lockable. This parameter gives you the option of exclusive control of the device. With the L parameter, you can set up a device for system-wide use and then, at a later time, lock it for use by a single user.
- M = Partition Mode. This parameter must be used to access files using file names.

- N = Shared Access. This parameter enables two or more processes to simultaneously access the same device.
- V = Verify Writes Allowed. This parameter causes all writes to a device to be verified by the system. You can specify more than one of the access level parameters when you use DVRLOAD.

If you do not specify any parameters for access level, the system assumes the parameters PRWSEL.

Example

```
REM This command installs the hard disk driver in
REM unpartitioned mode.
REM If the hard disk driver is already installed for a
REM drive, it must be removed first using the DVRUNLK
REM command.
REM
DVRLOAD h0: A:ATHD.286  LNWRS
REM
REM The file specification, A:ATHD.286
REM tells the system that the hard disk
REM driver is contained in the file ATHD.286
REM on the A: drive.
REM
```

DVRUNIT Command

Purpose

To add another unit of an existing device to the system.

Formats

DVRUNIT *old_device device_name access_level*

Type

Internal

Used only when the store controller has been IPLed from the Supplementals and is to be used only for the hard disk drive driver.

Description

The DVRUNIT command adds another unit of an existing device to the system. For example, if your system has one hard disk drive and you want to install another, you must use DVRUNIT.

old_device

The device name of a driver that has already been installed.

device_name

The logical name of the driver's new unit. The colon (:) is optional in this logical device name.

access_level

Parameters are the same for DVRUNIT as they are for DVRLOAD. If you do not specify any access-level parameters for DVRUNIT, the RWSL parameters are assumed.

Example

The following example installs another hard disk drive.

```
REM
DVRLOAD h0: A:ATHD.286 LNWRS
REM
REM The file specification, A:ATHD.286 tells
REM the system that the hard disk driver
REM is contained in the file ATHD.286 on drive A:
REM disk. h0: is the logical device name for
REM the hard disk drive.
REM
REM The DVRUNIT command shown below relates a new unit
REM to a driver that is already installed. h1: is
REM the logical device name of the second hard disk
REM drive.
REM
DVRUNIT h0: h1: LNWRS
REM
```

DVRUNLK Command

Purpose

To remove a driver that was installed previously with DVRLOAD.

Formats

DVRUNLK *device_name*

Type

Internal

Used only when the store controller has been IPLed from the Supplementals.

Description

The DVRUNLK command is used to remove a driver from your system that you have installed with DVRLOAD.

device_name

Is the name of a previously installed device.

Example

This example removes the hard disk drive driver h0:.

DVRUNLK h0:

SYS Command

Purpose

To prepare a hard disk drive for use as the IPL drive.

Formats

`SYS destination_drive filespec`

Type

External

Used only when the store controller has been IPLed from the Supplementals.

Description

Filespec specifies the name of a file that is present on the Supplementals (see “Preparing Your Hard Disk Drive Using the Supplementals” on page 528).

The SYS command is for use only on drive C. It places the operating system loader on the C drive. The first two cylinders must not contain any defective sectors and SYS must be executed after formatting.

Examples

The following command line places the operating system loader on drive C.

`T:>SYS C: ADXNSSDZ.SYS`

Disk Rebuild Utility

For a detailed description of the Disk Rebuild Utility, see “Introduction to the LAN disk rebuild utility” on page 409. Before using the Disk Rebuild Utility, the following items must have been completed:

- Installed the Multiple Controller Feature (MCF)
- Supplied the LAN node ID of the acting master store controller and the data path to be recovered
- Supplied the LAN node ID of any other store controller on the LAN to be installed or replaced

Note: The Disk Rebuild Utility can be run on only one store controller on the LAN at a time.

Appendix D. Notices

This information was developed for products and services offered in the U.S.A.

Toshiba Global Commerce Solutions may not offer the products, services, or features discussed in this document in other countries. Consult your local Toshiba Global Commerce Solutions representative for information on the products and services currently available in your area. Any reference to a Toshiba Global Commerce Solutions product, program, or service is not intended to state or imply that only that Toshiba Global Commerce Solutions product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Toshiba Global Commerce Solutions intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-Toshiba Global Commerce Solutions product, program, or service.

Toshiba Global Commerce Solutions may have patents or pending patent applications covering the subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Toshiba Global Commerce Solutions
Attn: General Counsel
3039 E. Cornwallis Rd
RTP, NC 27709

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: TOSHIBA GLOBAL COMMERCE SOLUTIONS PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Toshiba Global Commerce Solutions may make improvements and/or changes in the product(s) and/or program(s) described in this publication at any time without notice.

Toshiba Global Commerce Solutions may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this information to non-Toshiba Global Commerce Solutions Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Toshiba Global Commerce Solutions product and use of those Web sites is at your own risk.

Information concerning non-Toshiba Global Commerce Solutions products was obtained from the suppliers of those products, their published announcements or other publicly available sources. Toshiba Global Commerce Solutions has not tested those products and cannot confirm the accuracy of performance, compatibility or

any other claims related to non-Toshiba Global Commerce Solutions products. Questions on the capabilities of non-Toshiba Global Commerce Solutions products should be addressed to the suppliers of those products.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

Telecommunication regulatory statement

This product is not intended to be connected directly or indirectly by any means whatsoever to interfaces of public telecommunications networks, nor is it intended to be used in a public services network.

Electronic Emission Notices

When you attach a monitor to the equipment, you must use the designated monitor cable and any interference suppression devices that are supplied with the monitor.

Federal Communications Commission (FCC) Statement

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

Properly shielded and grounded cables and connectors must be used in order to meet FCC emission limits. Toshiba Global Commerce Solutions is not responsible for any radio or television interference caused by using other than recommended cables and connectors or by unauthorized changes or modifications to this equipment. Unauthorized changes or modifications could void the user's authority to operate the equipment.

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference, and
2. This device must accept any interference received, including interference that may cause undesired operation.

Industry Canada Class A Emission Compliance statement

This Class A digital apparatus complies with Canadian ICES-003.

Avis de conformité à la réglementation d'Industrie Canada

Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

Australia and New Zealand Class A Statement

Attention: This is a Class A product. In a domestic environment this product may cause radio interference, in which case the user may be required to take adequate measures.

European Union Electromagnetic Compatibility (EMC) Directive Conformance Statement

This product is in conformity with the protection requirements of EU Council Directive 2004/108/EC on the approximation of the laws of the Member States relating to electromagnetic compatibility. Toshiba Global Commerce Solutions

cannot accept responsibility for any failure to satisfy the protection requirements resulting from a non-recommended modification of the product, including the fitting of non-Toshiba Global Commerce Solutions option cards.

This product has been tested and found to comply with the limits for Class A Information Technology Equipment according to CISPR 22/European Standard EN 55022. The limits for Class A equipment were derived for commercial and industrial environments to provide reasonable protection against interference with licensed communication equipment.

Attention: This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Responsible manufacturer:

Toshiba Global Commerce Solutions
3039 Cornwallis Road
Building 307
Research Triangle Park, North Carolina 27709
United States of America

European Community contact:

Toshiba Global Commerce Solutions
Brand Manager - Europe, Middle East & Africa
3 NEW SQUARE, FELTHAM, TW14 8HB Great Britain
Building: 1 Floor: NA | Office: MOBILE
Tel: 44-7967-275819
e-mail: robin_lyon@uk.ibm.com

Germany Class A Statement

Deutschsprachiger EU Hinweis: Hinweis für Geräte der Klasse A EU-Richtlinie zur Elektromagnetischen Verträglichkeit

Dieses Produkt entspricht den Schutzanforderungen der EU-Richtlinie 2004/108/EG zur Angleichung der Rechtsvorschriften über die elektromagnetische Verträglichkeit in den EU-Mitgliedsstaaten und hält die Grenzwerte der EN 55022 Klasse A ein.

Um dieses sicherzustellen, sind die Geräte wie in den Handbüchern beschrieben zu installieren und zu betreiben. Des Weiteren dürfen auch nur von der Toshiba Global Commerce Solutions empfohlene Kabel angeschlossen werden. Toshiba Global Commerce Solutions übernimmt keine Verantwortung für die Einhaltung der Schutzanforderungen, wenn das Produkt ohne Zustimmung der Toshiba Global Commerce Solutions verändert bzw. wenn Erweiterungskomponenten von Fremdherstellern ohne Empfehlung der Toshiba Global Commerce Solutions gesteckt/eingebaut werden.

EN 55022 Klasse A Geräte müssen mit folgendem Warnhinweis versehen werden: "Warnung: Dieses ist eine Einrichtung der Klasse A. Diese Einrichtung kann im Wohnbereich Funk-Störungen verursachen; in diesem Fall kann vom Betreiber verlangt werden, angemessene Maßnahmen zu ergreifen und dafür aufzukommen."

Deutschland: Einhaltung des Gesetzes über die elektromagnetische Verträglichkeit von Geräten

Dieses Produkt entspricht dem "Gesetz über die elektromagnetische Verträglichkeit von Geräten (EMVG)". Dies ist die Umsetzung der EU-Richtlinie 2004/108/EG in der Bundesrepublik Deutschland.

Zulassungsbescheinigung laut dem Deutschen Gesetz über die elektromagnetische Verträglichkeit von Geräten (EMVG) (bzw. der EMC EG Richtlinie 2004/108/EG) für Geräte der Klasse A

Dieses Gerät ist berechtigt, in Übereinstimmung mit dem Deutschen EMVG das EG-Konformitätszeichen - CE - zu führen.

Verantwortlich für die Einhaltung der EMV Vorschriften ist der Hersteller:

Toshiba Global Commerce Solutions
3039 Cornwallis Road
Building 307
Research Triangle Park, North Carolina 27709
United States of America

Der verantwortliche Ansprechpartner des Herstellers in der EU ist:

Toshiba Global Commerce Solutions
Brand Manager - Europe, Middle East & Africa
3 NEW SQUARE, FELTHAM, TW14 8HB Great Britain
Building: 1 Floor: NA | Office: MOBILE
Tel: 44-7967-275819
e-mail: robin_lyon@uk.ibm.com

Generelle Informationen:

Das Gerät erfüllt die Schutzanforderungen nach EN 55024 und EN 55022 Klasse A.

Japan Voluntary Control Council for Interference Class A statement

Attention: This is a Class A product based on the standard of the Voluntary Control Council for Interference (VCCI). If this equipment is used in a domestic environment, radio interference may occur, in which case, the user may be required to take corrective actions.

この装置は、クラスA情報技術装置です。この装置を家庭環境で使用する
と電波妨害を引き起こすことがあります。この場合には使用者が適切な
対策を講ずるよう要求されることがあります。 VCCI-A

Japan Electronics and Information Technology Industries Association (JEITA) statement

高調波ガイドライン準用品

Japan Electronics and Information Technology Industries Association (JEITA)
Confirmed Harmonics Guidelines with Modifications (products greater than 20 A per phase)

Korean communications statement

Please note that this device has been approved for business purposes with regard to electromagnetic interference (Type A). If you find this is not suitable for your use, you may exchange it for a non-business purpose one.

이 기기는 업무용(A급)으로 전자파적합기기로서 판매자 또는 사용자는 이 점을 주의하시기 바라며, 가정외의 지역에서 사용하는 것을 목적으로 합니다.

Russian Electromagnetic Interference (EMI) Class A statement

ВНИМАНИЕ! Настоящее изделие относится к классу А. В жилых помещениях оно может создавать радиопомехи, снижения которых необходимы дополнительные меры

People's Republic of China Class A electronic emission Statement

Attention: This is a Class A product. In a domestic environment this product may cause radio interference, in which case the user may be required to take adequate measures.

中华人民共和国“A类”警告声明

声 明

此为 A 级产品，在生活环境中，该产品可能会造成无线电干扰。在这种情况下，可能需要用户对其干扰采取切实可行的措施。

Taiwan Class A compliance statement

警告使用者：
這是甲類的資訊產品，在居住的環境中使用時，可能會造成射頻干擾，在這種情況下，使用者會被要求採取某些適當的對策。

European Community (EC) Mark of Conformity Statement

This product is in conformity with the protection requirements of EC Council Directive 89/336/EEC on the approximation of the laws of the Member States relating to electromagnetic compatibility. Toshiba Global Commerce Solutions cannot accept responsibility for any failure to satisfy the protection requirements resulting from a non-recommended modification of the product, including the fitting of non-Toshiba Global Commerce Solutions option cards.

This product has been tested and found to comply with the limits for Class A Information Technology Equipment according to CISPR 22 / European Standard EN 55022. The limits for Class A equipment were derived for commercial and industrial environments to provide reasonable protection against interference with licensed communication equipment.

Warning: This is a Class A product. In a domestic environment this product may cause radio interference, in which case the user may be required to take adequate measures.

Electrostatic Discharge (ESD)

Attention: Electrostatic discharge (ESD) damage can occur when there is a difference in charge between the part, the product, and the service person. No damage will occur if the service person and the part being installed are at the same charge level.

ESD damage prevention

Anytime a service action involves physical contact with logic cards, modules, back-panel pins, or other ESD sensitive (ESDS) parts, the service person must be connected to an ESD common ground point on the product through the ESD wrist strap and cord.

The ESD ground clip can be attached to any frame ground, ground braid, green wire ground, or the round ground prong on the AC power plug. Coax or connector outside shells can also be used.

Handling removed cards

Logic cards removed from a product should be placed in ESD protective containers. No other object should be allowed inside the ESD container with the logic card. Attach tags or reports that must accompany the card to the outside of the container.

Japanese Electrical Appliance and Material Safety Law statement

本製品およびオプションに電源コードセットが付属する場合は、それぞれその装置専用のものでありますので他の機器には使用しないで下さい。

Japanese power line harmonics compliance statement

高調波ガイドライン適合品

高調波ガイドライン適合品

Cable ferrite requirement

All cable ferrites are required to suppress radiated EMI emissions and must not be removed.

Product recycling and disposal

This unit must be recycled or discarded according to applicable local and national regulations. Toshiba Global Commerce Solutions encourages owners of information technology (IT) equipment to responsibly recycle their equipment when it is no longer needed. Toshiba Global Commerce Solutions offers a variety of product return programs and services in several countries to assist equipment owners in recycling their IT products. Information on Toshiba Global Commerce Solutions product recycling offerings can be found on the Toshiba Global Commerce Solutions product recycling web site.

Español:

Esta unidad debe reciclarse o desecharse de acuerdo con lo establecido en la normativa nacional o local aplicable. Toshiba Global Commerce Solutions recomienda a los propietarios de equipos de tecnología de la información (TI) que reciclen responsablemente sus equipos cuando éstos ya no les sean útiles. Toshiba Global Commerce Solutions dispone de una serie de programas y servicios de devolución de productos en varios países, a fin de ayudar a los propietarios de equipos a reciclar sus productos de TI. Se puede encontrar información sobre las ofertas de reciclado de productos de Toshiba Global Commerce Solutions en el sitio web Toshiba Global Commerce Solutions product recycling.



Notice: This mark applies only to countries within the European Union (EU) and Norway.

Appliances are labeled in accordance with European Directive 2002/96/EC concerning waste electrical and electronic equipment (WEEE). The Directive determines the framework for the return and recycling of used appliances as applicable throughout the European Union. This label is applied to various products to indicate that the product is not to be thrown away, but rather reclaimed upon end of life per this Directive.

Remarque : Cette marque s'applique uniquement aux pays de l'Union Européenne et à la Norvège. L'étiquette du système respecte la Directive européenne 2002/96/EC en matière de Déchets des Equipements Electriques et Electroniques (DEEE), qui détermine les dispositions de retour et de recyclage applicables aux systèmes utilisés à travers l'Union européenne. Conformément à la directive, ladite étiquette précise que le produit sur lequel elle est apposée ne doit pas être jeté mais être récupéré en fin de vie.

注意: このマークは EU 諸国およびノルウェーにおいてのみ適用されます。

この機器には、EU 諸国に対する廃電気電子機器指令 2002/96/EC(WEEE) のラベルが貼られています。この指令は、EU 諸国に適用する使用済み機器の回収とリサイクルの骨子を定めています。このラベルは、使用済みになった時に指令に従って適正な処理をする必要があることを知らせるために種々の製品に貼られています。

In accordance with the European WEEE Directive, electrical and electronic equipment (EEE) is to be collected separately and to be reused, recycled, or recovered at end of life. Users of EEE with the WEEE marking per Annex IV of the WEEE Directive, as shown above, must not dispose of end of life EEE as unsorted municipal waste, but use the collection framework available to customers for the return, recycling, and recovery of WEEE. Customer participation is important to minimize any potential effects of EEE on the environment and human health due to the potential presence of hazardous substances in EEE. For proper collection and treatment, contact your local Toshiba Global Commerce Solutions representative.

Disposal of IT products should be in accordance with local ordinances and regulations.

Battery return program

This product may contain sealed lead acid, nickel cadmium, nickel metal hydride, lithium, or lithium ion battery. Consult your user manual or service manual for specific battery information. The battery must be recycled or disposed of properly. Recycling facilities may not be available in your area. For information on disposal of batteries go to the Battery disposal web site or contact your local waste disposal facility.

For Taiwan:



Please recycle batteries.

For the European Union:



Notice: This mark applies only to countries within the European Union (EU)

Batteries or packaging for batteries are labeled in accordance with European Directive 2006/66/EC concerning batteries and accumulators and waste batteries

and accumulators. The Directive determines the framework for the return and recycling of used batteries and accumulators as applicable throughout the European Union. This label is applied to various batteries to indicate that the battery is not to be thrown away, but rather reclaimed upon end of life per this Directive.

Les batteries ou emballages pour batteries sont étiquetés conformément aux directives européennes 2006/66/EC, norme relative aux batteries et accumulateurs en usage et aux batteries et accumulateurs usés. Les directives déterminent la marche à suivre en vigueur dans l'Union Européenne pour le retour et le recyclage des batteries et accumulateurs usés. Cette étiquette est appliquée sur diverses batteries pour indiquer que la batterie ne doit pas être mise au rebut mais plutôt récupérée en fin de cycle de vie selon cette norme.

バッテリーあるいはバッテリー用のパッケージには、EU 諸国に対する廃電気電子機器指令 2006/66/EC のラベルが貼られています。この指令は、バッテリーと蓄電池、および廃棄バッテリーと蓄電池に関するものです。この指令は、使用済みバッテリーと蓄電池の回収とリサイクルの骨子を定めているもので、EU 諸国にわたって適用されます。このラベルは、使用済みになったときに指令に従って適正な処理をする必要があることを知らせるために種々のバッテリーに貼られています。

In accordance with the European Directive 2006/66/EC, batteries and accumulators are labeled to indicate that they are to be collected separately and recycled at end of life. The label on the battery may also include a chemical symbol for the metal concerned in the battery (Pb for lead, Hg for mercury and Cd for cadmium). Users of batteries and accumulators must not dispose of batteries and accumulators as unsorted municipal waste, but use the collection framework available to customers for the return, recycling and treatment of batteries and accumulators. Customer participation is important to minimize any potential effects of batteries and accumulators on the environment and human health due to the potential presence of hazardous substances. For proper collection and treatment, contact your local Toshiba Global Commerce Solutions representative.

This notice is provided in accordance with Royal Decree 106/2008 of Spain: The retail price of batteries, accumulators and power cells includes the cost of the environmental management of their waste.

For California:

Perchlorate material – special handling may apply

Refer to California Department of Toxic Substances Control.

The foregoing notice is provided in accordance with *California Code of Regulations Title 22, Division 4.5, Chapter 33: Best Management Practices for Perchlorate Materials*. This product/part includes a lithium manganese dioxide battery which contains a perchlorate substance.

Flat panel displays

The fluorescent lamp in the liquid crystal display contains mercury. Dispose of it as required by local ordinances and regulations.

Monitors and workstations

Connecticut: Visit the website of the Department of Energy & Environmental Protection for information about recycling covered electronic devices in the State of Connecticut, or telephone the Connecticut Department of Environmental Protection at 1-860-424-3000.

Oregon: For information regarding recycling covered electronic devices in the state of Oregon, go to the Oregon Department of Environmental Quality site.

Washington: For information about recycling covered electronic devices in the State of Washington, go to the Department of Ecology Web site or telephone the Washington Department of Ecology at 1-800-Recycle.

Trademarks

The following are trademarks or registered trademarks of Toshiba, Inc. in the United States or other countries, or both:

Toshiba
The Toshiba logo

The following are trademarks of Toshiba Global Commerce Solutions in the United States or other countries, or both:

AnyPlace
SureMark
SurePoint
SurePOS
TCxWave

The following are trademarks of International Business Machines Corporation in the United States or other countries, or both:

DB2
DB2 Universal Database
IBM and the IBM logo
PS/2
Wake on LAN
WebSphere

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Magellan is a registered trademark of Datalogic Scanning, Inc.

SYMBOL a registered trademark of Symbol Technologies, Inc.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Celeron and Intel are trademarks of Intel corporation in the United States, or other countries.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, or other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Glossary

This glossary includes terms and definitions from the *IBM Dictionary of Computing* (New York; McGraw-Hill, Inc., 1994).

A

ABM. Asynchronous balanced mode.

access method. A software component in a processor for controlling the flow of information through a network.

ACF/VTAM. Advanced Communications Function for the Virtual Telecommunications Access Method.

active. (1) Able to communicate on the network. A token-ring network adapter is active if it is able to transmit and receive on the network. (2) Operational. (3) Pertaining to a node or device that is connected or is available for connection to another node or device. (4) Currently transmitting or receiving.

adapter. (1) In the point-of-sale terminal, a circuit card that, with its associated software, enables the terminal to use a function or feature. (2) In a LAN, within a communicating device, a circuit card that, with its associated software and/or microcode, enables the device to communicate over the network.

adapter address. Twelve hexadecimal digits that identify a LAN adapter.

ADCS. Advanced Data Communications for Stores

address. (1) In data communication, the IEEE-assigned unique code or the unique locally administered code assigned to each device or workstation connected to a network. (2) A character, group of characters, or a value that identifies a register, a particular part of storage, a data source, or a data link. The value is represented by one or more characters. (3) To refer to a device or an item of data by its address. (4) The location in the storage of a computer where data is stored.

Advanced Data Communications for Stores (ADCS). An IBM-licensed product that functions at the host processor to permit host-to-store communication.

alert. (1) An error message sent to the system services control point (SSCP) at the host system. (2) For Toshiba LAN management products, a notification indicating a possible security violation, a persistent error condition, or an interruption or potential interruption in the flow of data around the network. See also *network management vector transport*. (3) In SNA, a record sent to a system problem management focal point to communicate the existence of an alert condition. (4) In the NetView program, a high-priority event that warrants

immediate attention. This data base record is generated for certain event types that are designed by user-constructed filters.

alphanumeric. Pertaining to a character set containing letters, digits, and other special characters.

Alphanumeric point-of-sale keyboard (ANPOS keyboard). This keyboard consists of a section of alphanumeric keys, a programmable set of point-of-sale keys, a numeric keypad, and system function keys. If attached through the PS/2 port, this keyboard can optionally include a pointing device.

alternate adapter. In a personal computer that is used on a LAN and that supports installation of two network adapters, the adapter that uses alternate (not standard or default) mapping between adapter-shared RAM, adapter ROM, and designated computer memory segments. The alternate adapter is usually designated as adapter 1 in configuration parameters. Contrast with *primary adapter*.

Alternate File Server. A store controller that maintains image versions of all non-system mirrored files and that can assume control if the configured File Server becomes disabled.

Alternate Master Store Controller. The store controller that can take control of the LAN if the configured Master Store Controller becomes disabled. It maintains image versions of both system mirrored and system compound files.

American National Standard Code for Information Interchange (ASCII). The standard code, using a coded character set consisting of 7-bit coded characters (8 bits including parity check), used for information interchange among data processing systems, data communication systems, and associated equipment. The ASCII set consists of control characters and graphics characters.

ANPOS keyboard. Alphanumeric point-of-sale keyboard.

API. Application program interface.

application program. (1) A program written for or by a user that applies to the user's own work. (2) A program written for or by a user that applies to a particular application. (3) A program used to connect and communicate with stations in a network, enabling users to perform application-oriented activities.

application program interface (API). The formally defined programming language interface that is between a system control program or a licensed program and the user of the program.

architecture. A logical structure that encompasses operating principles including services, functions, and protocols. See *computer architecture*, *network architecture*, *Systems Application Architecture (SAA)*, *Systems Network Architecture (SNA)*.

ARTIC adapter. A family of communications coprocessor adapters that, with appropriate electrical interfaces, can support a wide range of communication devices. For the Toshiba Global Commerce Solutions Store System, an ARTIC adapter provides communications support for ASYNC, SDLC, and X.25 communications.

ASCII. American National Standard Code for Information Interchange.

async. asynchronous.

asynchronous (async). (1) Pertaining to two or more processes that do not depend upon the occurrence of specific events such as timing signals. (2) Without regular time relationship; unexpected or unpredictable with respect to the execution of program instructions.

asynchronous balanced mode (ABM). An operational mode of a balanced data link in which either combined station can send commands at any time and can initiate transmission of response frames without explicit permission from the other combined station.

attach. (1) To connect a device physically. (2) To make a device a part of a network logically. Compare with *connect*.

attaching device. Any device that is physically connected to a network and can communicate over the network.

B

background. On a color display, the part of the display screen that surrounds a character.

background application. A non-interactive program that can be selected from the background application screen or that can start automatically when the system is IPLed or when the controller is activated as the master or file server. Contrast with *foreground application*.

backup. Pertaining to a system, device, file, or facility that can be used in the event of a malfunction or the loss of data.

bar code. A code representing characters by sets of parallel bars of varying thickness and separation that are read optically by transverse scanning.

baseband. (1) A frequency band that uses the complete bandwidth of a transmission medium. Contrast with *broadband*, *carrierband*. (2) A method of data transmission that encodes, modulates, and impresses

information on the transmission medium without shifting or altering the frequency of the information signal.

base unit. The part of the 4683 Point-of-Sale terminal that contains the power supply and the interfaces.

BASIC. Beginner's All-purpose Symbolic Instruction Code. A programming language that uses common English words.

basic conversation. A conversation in which programs exchange data records in an SNA-defined format. This format is a stream of data containing 2-byte length prefixes that specify the amount of data to follow before the next prefix.

batch. Smaller subdivisions of price change records within an event. Each batch has a 12-character ID and a 30-character description field.

baud. The rate at which signal conditions are transmitted per second. Contrast with *bits per second (bps)*.

beacon. (1) A frame sent by an adapter on a ring network indicating a serious ring problem, such as a broken cable. It contains the addresses of the beaconing station and its nearest active upstream neighbor (NAUN). (2) To send beacon frames continuously. An adapter is *beaconing* if it is sending such a frame.

beaconing. An error-indicating function of token-ring adapters that assists in locating a problem causing a hard error on a token-ring network.

binary. (1) Pertaining to a system of numbers to the base two; the binary digits are 0 and 1. (2) Pertaining to a selection, choice, or condition that has two possible different values or states.

bind. In SNA products, a request to activate a session between two logical units.

BIND. See bind session.

bind session (BIND). In SNA products, a request to activate a session between two logical units (LUs).

bit. Either of the binary digits: a 0 or 1.

bits per second (bps). The rate at which bits are transmitted per second. Contrast with *baud*.

block size. (1) The minimum size that frames are grouped into for retransmission. (2) The number of data elements (such as bits, bytes, characters, or records) that are recorded or transmitted as a unit.

bootstrap. A sequence of instructions whose execution causes additional instructions to be loaded and executed until the complete computer program is in storage.

bps. Bits per second.

Bps. Bytes per second.

bridge. (1) An attaching device connected to two LAN segments to allow the transfer of information from one LAN segment to the other. A bridge may connect the LAN segments directly by network adapters and software in a single device, or may connect network adapters in two separate devices through software and use of a telecommunications link between the two adapters. (2) A functional unit that connects two LANs that use the same logical link control (LLC) procedures but may use the same or different medium access control (MAC) procedures. Contrast with *gateway* and *router*.

broadband. A frequency band divisible into several narrower bands so that different kinds of transmissions such as voice, video, and data transmission can occur at the same time. Synonymous with *wideband*. Contrast with *baseband*.

buffer. (1) A portion of storage used to hold input or output data temporarily. (2) A routine or storage used to compensate for a difference in data rate or time of occurrence of events, when transferring data from one device to another.

bus. (1) In a processor, a physical facility on which data is transferred to all destinations, but from which only addressed destinations may read in accordance with appropriate conventions. (2) A network configuration in which nodes are interconnected through a bidirectional transmission medium. (3) One or more conductors used for transmitting signals or power.

byte. A string consisting of 8 bits that is treated as a unit, and that represents a character. See *n-bit byte*.

C

C. A high-level programming language designed to optimize run time, size, and efficiency.

C & SM. Communications and systems management.

cable loss (optical). The loss in an optical cable equals the attenuation coefficient for the cables fiber times the cable length.

cable segment. A section of cable between components or devices on a network. A segment may consist of a single patch cable, multiple patch cables connected together, or a combination of building cable and patch cables connected together. See *LAN segment*, *ring segment*.

call. The action of bringing a function or subprogram into effect, usually by specifying the entry conditions and jumping to an entry point.

carrierband. A frequency band in which the modulated signal is superimposed on a carrier signal (as differentiated from baseband), but only one channel is present on the medium. Contrast with *baseband*, *broadband*.

cash drawer. A drawer at a point-of-sale terminal that can be programmed to open automatically. See *till*.

CCB. Command control block.

CCC/IP. Controller-to-Controller Communications over Internet Protocol.

CCITT. Comité Consultatif International Télégraphique et Téléphonique. The International Telegraph and Telephone Consultative Committee.

CD. Corrective diskette.

CD-ROM. Compact disc Read-only memory. High-capacity read-only memory in the form of an optically read compact disk.

chain. (1) Transfer of control from the currently executing program to another program or overlay. (2) Referencing a data record from a previous data record.

channel. (1) A functional unit, controlled by a host computer, that handles the transfer of data between processor storage and local peripheral equipment. (2) A path along which signals can be sent. (3) The portion of a storage medium that is accessible to a given reading or writing station.

CICS. Customer Information Control System.

circuit. (1) A logic device. (2) One or more conductors through which an electric current can flow.

class. (1) A template for creating objects; a class defines data and methods; a class is a unit of organization in a Java program. A class can pass on its public data and methods to its subclasses. (2) A collection of variables and methods that an object can have, or a template for building objects.

.class file. A file containing machine-independent Java bytecodes. The Java compiler generates *.class* files for the Java interpreter to read.

class method. A class method is a function that is defined as a part of a class.

classpath. An environment variable used to define all the directories where *.class* files are found.

.class variable. A variable allocated once per class. Class variables have global class scope and belong to the entire class instead of an instance.

clear. To delete data from a screen or from memory.

COBOL. Common business-oriented language. A high-level programming language, based on English, that is used primarily for business applications.

command. (1) A request for performance of an operation or execution of a program. (2) A character string from a source external to a system that represents a request for system action.

command control block (CCB). In the Token-Ring Network, a specifically formatted information provided from the application program to the adapter support software to request an operation.

Common Programming Interface-Communications (CPI-C). Provides languages, commands, and calls that allow the development of applications that are more easily integrated and moved across environments supported by Systems Applications Architecture (SAA).

communication adapter. A circuit card and its associated software that enable a device, such as a personal computer, to be connected to a network or another computer (examples include binary synchronous, asynchronous, modem, and LAN adapters).

communications and systems management (C & SM). A set of tools, programs, and network functions used to plan, operate, and control an SNA communications network. C & SM runs on the store controller and must also exist at the host site.

compact disc- read-only memory (CD-ROM). (1) A 4.75-inch optical memory storage medium, capable of storing approximately 650 megabytes of data. Data is read optically by means of a laser. (2) A disc with information stored in the form of pits along a spiral track. The information is decoded by a compact-disc player and interpreted as digital audio data, which most computers can process.

compile. (1) To translate all or part of a program expressed in a high-level language into a computer program expressed in an intermediate language, an assembly language, or a machine language. (2) To prepare a machine language program from a computer program written in another programming language by making use of the overall logic structure of the program, or generating more than one computer instruction for each symbolic statement, or both, as well as performing the function of an assembler. (3) To translate a source program into an executable program (an object program). (4) To translate a program written in a high-level programming language into a machine language program.

compound files. Files that are kept on all store controllers.

computer architecture. The organizational structure of a computer system, including hardware and software.

concurrent conversations. The ability of a transaction program (TP) to manage more than one LU 6.2 conversation at the same time. When this ability is written into a TP, the TP is said to be *managing concurrent conversations*.

configuration. The group of devices, options, and programs that make up a data processing system or network as defined by the nature, number, and chief characteristics of its functional units. More specifically, the term may refer to a hardware configuration or a software configuration. See also *system configuration*.

configuration parameters. Variables in a configuration definition, the values of which characterize the relationship of a product, such as a bridge, to other products in the same network.

connect. In a LAN, to physically join a cable from a station to an access unit or network connection point. Contrast with *attach*.

contention. In a LAN, a situation in which two or more data stations are allowed by the protocol to start transmitting concurrently and thus risk collision.

contention loser. In APPC, the LU that must request and receive permission from the session partner LU to allocate a session. Contrast with *contention winner*.

contention winner. The LU that can allocate a session without requesting permission from the session partner LU. Contrast with *contention loser*.

contiguous. Touching or joining at the edge or boundary; adjacent. For example, an unbroken consecutive series of memory locations.

controller. A unit that controls input/output operations for one or more devices.

conversation. A logical connection between two programs over an LU type 6.2 session that allows them to communicate with each other while processing a transaction. See also *basic conversation* and *mapped conversation*.

conversation partner. One of the two programs involved in a conversation.

conversation state. The condition of a conversation that reflects what the past action on that conversation has been and that determines what the next set of actions may be.

corrective diskette (CD). A set of diskettes that contain modules to replace the modules in the active program subdirectory. The first diskette of the set must contain a product control file that describes which product the modules are to be applied to and a list of all modules that are to be replaced.

CRC. Cyclic redundancy check.

cursor. A movable point of light (or a short line) that indicates where the next character is to be entered on the display screen.

Customer Information Control System (CICS). An IBM licensed program that enables transactions entered at remote terminals to be processed concurrently by user-written application programs. It includes facilities for building, using, and maintaining data bases.

customer receipt. An itemized list of merchandise purchased and paid for by the customer.

customize. To tailor a program or store system through option selection.

cyclic redundancy check (CRC). Synonym for *frame check sequence (FCS)*.

D

data. (1) A representation of facts, concepts, or instructions in a formalized manner suitable for communication, interpretation, or processing by human or automatic means. (2) Any representations such as characters or analog quantities to which meaning is or might be assigned.

data circuit-terminating equipment (DCE). In a data station, the equipment that provides the signal conversion and coding between the data terminal equipment (DTE) and the line.

data communication. (1) Transfer of information between functional units by means of data transmission according to a protocol. (2) The transmission, reception, and validation of data.

data file. A collection of related data records organized in a specific manner; for example, a payroll file (one record for each employee, showing such information as rate of pay and deductions) or an inventory file (one record for each inventory item, showing such information as cost, selling price, and number in stock.) See also *data set, file*.

data link. (1) Any physical link, such as a wire or a telephone circuit, that connects one or more remote terminals to a communication control unit, or connects one communication control unit with another. (2) The assembly of parts of two data terminal equipment (DTE) devices that are controlled by a link protocol, and the interconnecting data circuit, that enable data to be transferred from a data source to a data link. (3) In SNA, see also *link*. **Note:** A telecommunication line is only the physical medium of transmission. A data link includes the physical medium of transmission, the protocol, and associated devices and programs; it is both physical and logical.

data processing system. A network, including computer systems and associated personnel, that

accepts information, processes it according to a plan, and produces the appropriate results.

data rate. See *data transfer rate, line data rate*.

data set. Logically related records treated as a single unit. See also *file*.

data terminal equipment (DTE). (1) That part of a data station that serves as a data source, data receiver, or both. (2) Equipment that sends or receives data, or both.

data transfer. (1) The result of the transmission of data signals from any data source to a data receiver. (2) The movement, or copying, of data from one location and the storage of the data at another location.

data transfer rate. The average number of bits, characters, or blocks per unit of time passing between equipment in a data-transmission session. The rate is expressed in bits, characters, or blocks per second, minute, or hour.

data transmission. The conveying of data from one place for reception elsewhere by means of telecommunications.

dB. Decibel.

DBCS. Double-byte character set.

DCE. Data circuit-terminating equipment.

DDA. Data Distribution Application.

debug. To detect, diagnose, and eliminate errors in computer programs.

decibel (dB). (1) One tenth of a bel. (2) A unit that expresses the ratio of two power levels on a logarithmic scale. (3) A unit for measuring relative power. The number of decibels is 10 times the logarithm base (base 10) of the ratio of the measured power levels; if the measured levels are voltages (across the same or equal resistance), the number of decibels is twenty times the log of the ratio. See also *neper*.

default. Pertaining to an attribute, value, or option that is assumed when none is explicitly specified.

default value. The value the system supplies when the user does not specify a value.

delayed data maintenance. A function that allows the item record, the operator and the check authorization files to be maintained from the host on an immediate or a delayed basis.

destination. Any point or location, such as a node, station, or particular terminal, to which information is to be sent.

device. (1) A mechanical, electrical, or electronic contrivance with a specific purpose. (2) An input/output unit such as a terminal, display, or printer. See also *attaching device*.

device channel. In Toshiba Point-of-Sale terminals, a path along which signals for serial input/output devices can be sent. For these terminals, the device channel controller or adapter is contained on the system board.

diagnostic diskette. A diskette containing diagnostic modules or tests used by computer users and service personnel to diagnose hardware problems.

diagnostics. Modules or tests used by computer users and service personnel to diagnose hardware problems.

dialing. Using a dial or pushbutton telephone to initiate a telephone call. In telecommunication, attempting to establish a connection between a terminal and a telecommunication device over a switched line.

direct memory access (DMA). A procedure or method designed to transfer data between main storage and I/O units without intervention of the processing unit.

directory. (1) A table of identifiers and references that correspond to items of data. (2) An index that a control program uses to locate one or more blocks of data that are stored in separate areas of a data set in direct access storage.

disabled. (1) Pertaining to a state of a processing unit that prevents the occurrence of certain types of interruptions. (2) Pertaining to the state in which a transmission control unit or audio response unit cannot accept incoming calls on a line.

DISC. Disconnect character.

disk. A round, flat plate coated with a magnetic substance on which computer data is stored. See also *integrated disk*, *fixed disk*.

diskette. A thin, flexible magnetic disk permanently enclosed in a protective jacket. A diskette is used to store information for processing.

Disk Operating System (DOS). An operating system for computer systems that use disks and diskettes for auxiliary storage of programs and data.

display. (1) A visual presentation of data. (2) A device that presents visual information to the point-of-sale terminal operator and to the customer, or to the display station operator.

distributed. Physically separate but connected by cables.

Distributed Systems Executive (DSX). An IBM licensed program available for host systems that allows

the host system to get, send, and remove files, programs, formats and procedures in a network of computers.

DMA. Direct memory access

domain. An SSCP and the resources that it can control.

DOS. Disk Operating System.

double-byte character set (DBCS). A set of characters in which each character is represented by 2 bytes. Languages such as Japanese, Chinese, and Korean, which contain more symbols than can be represented by 256 code points, require double-byte character sets. Because each character requires 2 bytes, the typing, display, and printing of DBCS characters requires hardware and programs that support DBCS. Contrast with single-byte character set.

driver. Software component that controls a device.

drop. A cable that leads from a faceplate to the distribution panel in a wiring closet. When the Cabling System is used with the Token-Ring Network, a drop may form part of a lobe. See also *lobe*.

DSX. Distributed Systems Executive.

DTE. Data terminal equipment.

dump. (1) To write at a particular instant the contents of storage, or part of storage, onto another data medium for the purpose of safeguarding or debugging the data. (2) Data that has been dumped.

E

EAN. European article number.

EBCDIC. Extended binary-coded decimal interchange code.

EIA. Electronic Industries Association. See *EIA interface*.

EIA interface. An industry-accepted interface for connecting devices having voltage-related limits.

emulation. (1) The imitation of all or part of one computer system by another, primarily by hardware, so that the imitating system accepts the same data, executes the same programs, and achieves the same results as the imitated computer system. (2) The use of programming techniques and special machine features to permit a computing system to execute programs written for another system.

enabled. (1) On a LAN, pertaining to an adapter or device that is active, operational, and able to receive frames from the network. (2) Pertaining to a state of a processing unit that allows the occurrence of certain

types of interruptions. (3) Pertaining to the state in which a transmission control unit or an audio response unit can accept incoming calls on a line.

envelope. (1) Information added to a frame or other message unit to allow it to be transmitted using a protocol other than the protocol in which the message unit originated. (2) To surround or enclose a message unit in information to allow the message unit to be transmitted using a protocol other than the protocol in which the message originated.

error condition. The condition that results from an attempt to use instructions or data that are invalid.

error message. A message that is issued because an error has been detected.

Ethernet. A 10-megabit baseband local area network that allows multiple stations to access the transmission medium at will without prior coordination, avoids contention by using carrier sense and deference, and resolves contention by using collision detection and transmission. Ethernet uses carrier sense multiple access with collision detection (CSMA/CD).

European article number (EAN). A number that is assigned to and encoded on an article of merchandise for scanning in some countries.

evaluation. Reduction of an expression to a single value.

exchange identification (XID). The ID that is exchanged with the remote physical unit when an attachment is first established.

execute. To perform the actions specified by a program or a portion of a program.

execution. The process of carrying out an instruction or instructions of a computer program by a computer.

exit. To execute an instruction or statement within a portion of a program in order to terminate the execution of that portion. **Note:** Such portions of programs include loops, routines, subroutines, and modules.

expansion board. In a personal computer, a panel containing microchips that a user can install in an expansion slot to add memory or special features. Synonymous with *expansion card*, *extender card*.

expansion card. Synonym for *expansion board*.

expansion slot. In a personal computer, one of several receptacles in the system board of the system unit or expansion unit into which a user can install an expansion board such as a memory expansion option.

extended binary-coded decimal interchange code (EBCDIC). A coded character set consisting of 8-bit coded characters.

extender card. Synonym for *expansion board*.

F

fault. An accidental condition that causes a functional unit to fail to perform its required function.

feature. A part of a product that may be ordered separately by the customer.

Feature Expansion. A card that plugs into a 4683 Point-of-Sale Terminal and allows additional devices to be used.

field. On a data medium or a storage medium, a specified area used for a particular category of data; for example, a group of character positions used to enter or display wage rates on a panel.

file. A named set of records stored or processed as a unit. For example, an invoice may form a record and the complete set of such records may form a file. See also *data file* and *data set*.

file name. (1) A name assigned or declared for a file. (2) The name used by a program to identify a file.

file server. (1) A store controller that maintains prime versions of all non-system mirrored files. (2) A high-capacity disk storage device or a computer that each computer on a network can access to retrieve files that can be shared among the attached computers.

file type. The attribute of a file that specifies to which store controllers it is distributed.

fixed disk (drive). In a personal computer system unit, a disk storage device that reads and writes on rigid magnetic disks. It is faster and has a larger storage capacity than a diskette and is permanently installed.

foreground. On a color display, the part of the display area that is the character itself.

foreground application. An interactive program that can be selected by system menus or started in command mode. Contrast with *background application*.

formatted diskette. A diskette on which track and sector control information has been written and that can be used by the computer to store data. **Note:** A diskette must be formatted before it can receive data.

frame. (1) The unit of transmission in some LANs, including the Token-Ring Network. It includes delimiters, control characters, information, and checking characters. On a token-ring network, a frame is created from a token when the token has data appended to it. On a token-bus network, all frames including the token frame contain a preamble, start delimiter, control address, optional data and checking characters, end delimiter, and are followed by a minimum silence period. (2) A housing for machine elements. (3) In synchronous

data link control (SDLC), the vehicle for every command, every response, and all information that is transmitted using SDLC procedures. Each frame begins and ends with a flag.

frame check sequence (FCS). (1) A system of error checking performed at both the sending and receiving station after a block-check character has been accumulated. (2) A numeric value derived from the bits in a message that is used to check for any bit errors in transmission. (3) A redundancy check in which the check key is generated by a cyclic algorithm. Synonymous with *cyclic redundancy check (CRC)*.

franking. Printing an indication on a document that the document has been processed. This franking may be a store header line, a "total" line, or a transaction number that is printed when a check, a discount coupon, or a gift certificate is inserted in the document insert station of the point-of-sale terminal during certain types of transactions.

frequency. The rate of signal oscillation, expressed in hertz (cycles per second).

function. (1) A specific purpose of an entity, or its characteristic action. (2) A subroutine that returns the value of a single variable. (3) In data communications, a machine action such as a carriage return or line feed.

G

gateway. A device and its associated software that interconnect networks of systems of different architectures. The connection is usually made above the Reference Model network layer. For example, a gateway allows LANs access to System/370 host computers. Contrast with *bridge* and *router*.

group. (1) A set of related records that have the same value for a particular field in all records. (2) A collection of users who can share access authorities for protected resources. (3) A list of names that are known together by a single name.

H

hardware. Physical equipment as opposed to programs, procedures, rules, and associated documentation.

HCP. Host command processor for advanced data communications.

HCP. Host command processor.

header. The portion of a message that contains control information for the message such as one or more destination fields, name of the originating station, input sequence number, character string indicating the type of message, and priority level for the message.

host application program. An application program that the host processor executes.

host command processor (HCP). The SNA logical unit of the programmable Store System store controller.

host computer. (1) The primary or controlling computer in a multi-computer installation or network. (2) In a network, a processing unit in which resides a network access method. Synonymous with *host processor*.

host processor. (1) In a network, a computer that primarily provides services such as computation, data base access, or special programs or programming languages. (2) Synonym for *host computer*.

host processor. (1) A processor that controls all or part of a user application network. (2) In a network, the processing unit in which resides the access method for the network. (3) In an SNA network, the processing unit that contains a system services control point (SSCP). (4) A processing unit that executes the access method for attached communication controllers. (5) The processing unit required to create and maintain PSS. Synonymous with *host computer*.

I

Disk Operating System (DOS). A disk operating system based on MS-DOS**.

identifier. String of characters used to name elements of a program, such as variable names, reserved words, and user-defined function names.

idles. Signals sent along a ring network when neither frames nor tokens are being transmitted.

image version. Copy of a prime version of a file. See *prime version*.

inactive. (1) Not operational. (2) Pertaining to a node or device not connected or not available for connection to another node or device.

information (I) frame. A frame in I format used for numbered information transfer. See also *supervisory frame*, *unnumbered frame*.

initialize. In a LAN, to prepare the adapter (and adapter support code, if used) for use by an application program.

initial program load (IPL). The initialization procedure that causes an operating system to begin operation.

input device. Synonym for *input unit*.

input field. An unprotected display field that the terminal operator can add to, modify, or erase by using the keyboard. Contrast with *protected field*.

input/output (I/O). (1) Pertaining to a device whose parts can perform an input process and an output process at the same time. (2) Pertaining to a functional unit or channel involved in an input process, output process, or both, concurrently or not, and to the data involved in such a process.

input unit. A device in a data processing system by means of which data can be entered into the system. Synonymous with *input device*.

insert. To make an attaching device an active part of a LAN.

integrated. Arranged together as one unit.

integrated disk. An integral part of the processor that is used for magnetically storing files, application programs, and diagnostics. Synonymous with *disk*.

interactive. Pertaining to an application or program in which each entry calls forth a response from a system or program. An interactive program may also be conversational, implying a continuous dialog between the user and the system.

interface. (1) A shared boundary between two functional units, defined by functional characteristics, common physical interconnection characteristics, signal characteristics, and other characteristics as appropriate. (2) A shared boundary. An interface may be a hardware component to link two devices or a portion of storage or registers accessed by two or more computer programs. (3) Hardware, software, or both, that links systems, programs, or devices.

interference. (1) The prevention of clear reception of broadcast signals. (2) The distorted portion of a received signal.

interleave. To insert segments of one program into another program so that the two programs can, in effect, be executed at the same time.

interrupt. (1) A suspension of a process, such as execution of a computer program, caused by an external event and performed in such a way that the process can be resumed. (2) To stop a process in such a way that it can be resumed. (3) In data communication, to take an action at a receiving station that causes the sending station to end a transmission. (4) A means of passing processing control from one software or microcode module or routine to another, or of requesting a particular software, microcode, or hardware function.

interrupt level. The means of identifying the source of an interrupt, the function requested by an interrupt, or the code or feature that provides a function or service.

I/O. Input/output.

I/O device. Equipment for entering and receiving data from the system.

IP. Internet Protocol.

IPL. Initial program load.

isochronous. Time-dependent. Refers to processes in which data must be delivered within certain time constraints.

item. (1) One member of a group. (2) In a store, one unit of a commodity, such as one box, one bag, or one can. Usually an item is the smallest unit of a commodity to be sold.

J

Java. An object-oriented programming language designed to be platform independent.

Java application. A Java Virtual Machine (JVM) combined with its class and parameters.

Java Virtual Machine (JVM). Java interpreter that runs the class.

jumper. A connector between two pins on a network adapter that enables or disables an adapter option, feature, or parameter value.

JUCC. Japan Unified Cash Card.

JVM. See Java Virtual Machine.

K

K. When referring to storage capacity, a symbol that represents two to the tenth power, or 1024.

Kb. Kilobit.

KB. Kilobyte.

keyboard. A group of numeric keys, alphabetic keys, special character keys, or function keys used for entering information into the terminal and into the system.

keyed file. Type of file composed of keyed records. Each keyed record has two parts: a key and data. A key is used to identify and access each record in the file.

kilobit (Kb). 1024 binary digits.

kilobyte (KB). 1024 bytes for processor and data storage (memory) size.

L

label. Constant, either numeric or literal, that references a statement or function.

LAN. Local area network.

LAN segment. (1) Any portion of a LAN (for example, a single bus or ring) that can operate independently but is connected to other parts of the establishment network by bridges. (2) An entire ring or bus network without bridges. See *cable segment*, *ring segment*.

LCD. Liquid crystal display.

leased line. Synonym for *nonswitched line*.

LED. Light-emitting diode.

light-emitting diode (LED). A semiconductor chip that gives off visible or infrared light when activated.

line connection. In the Store System, the physical connection (or equipment) between nodes that provides two-way communication and error correction and detection between one link station and one or more other link stations. **Note:** In SNA, this physical connection is called a *link connection*. In the Store System, it is called a *line connection*.

line data rate. The rate of data transmission over a telecommunications link.

link. (1) In the Store System, the logical connection between nodes including the end-to-end link control procedures. (2) The combination of physical media, protocols, and programming that connects devices on a network. (3) In computer programming, the part of a program, in some cases a single instruction or an address, that passes control and parameters between separate portions of the computer program. (4) To interconnect items of data or portions of one or more computer programs. (5) In SNA, the combination of the link connection and link stations joining network nodes. See also *link connection*. **Note:** A link connection is the physical medium of transmission; for example, a telephone wire or a microwave beam. A link includes the physical medium of transmission, the protocol, and associated devices and programming; it is both logical and physical.

link connection. (1) All physical components and protocol machines that lie between the communicating link stations of a link. The link connection may include a switched or leased physical data circuit, a LAN, or an X.25 virtual circuit. (2) In SNA, the physical equipment providing two-way communication and error correction and detection between one link station and one or more other link stations. (3) In the Store System, the logical link providing two-way communication of data from one network node to one or more other network nodes.

listing. A printout of source code.

load. In computer programming, to enter data into memory or working registers.

lobe. In the Token-Ring Network, the section of cable (which may consist of several segments) that connects an attaching device to an access unit.

local area network (LAN). A computer network located on a user's premises within a limited geographical area. **Note:** Communication within a LAN is not subject to external regulations; however, communication across the LAN boundary may be subject to some form of regulation.

local program. The program being discussed within a particular context. Contrast with *remote program*.

logical file name (LFN). An abbreviated file name used to represent either an entire file name or the drive and subdirectory path part of the file name.

logical link. In an MVS/VS multisystem environment, the means by which a physical link is related to the transactions and terminals that can use the physical link.

logical unit (LU). (1) In SNA, a port through which an end user accesses the SNA network in order to communicate with another end user and through which the end user accesses the functions provided by system services control points (SSCPs). An LU can support at least two sessions, one with an SSCP and one with another LU, and may be capable of supporting many sessions with other logical units. (2) A type of network addressable unit that enables end users to communicate with each other and gain access to network resources.

logon (n). The procedure for starting up a point-of-sale terminal or store controller for normal sales operations by sequentially entering the correct security number and transaction number. Synonymous with *sign-on*.

log on (v). (1) To initiate a session. (2) In SNA products, to initiate a session between an application program and a logical unit (LU). Synonymous with *sign-on*.

loop. (1) A set of instructions that may be executed repeatedly while a certain condition prevails. See also *store loop*. (2) A closed unidirectional signal path connecting input/output devices to a network.

LU. Logical unit.

M

magnetic stripe. The magnetic material (similar to recording tape) on merchandise tickets, credit cards, and employee badges. Information is recorded on the stripe for later "reading" by the magnetic stripe reader (MSR) or magnetic wand reader attached to the point-of-sale terminal.

magnetic stripe reader (MSR). A device that reads coded information from a magnetic stripe on a card, such as a credit card, as it passes through a slot in the reader.

maintenance analysis procedure (MAP). Deprecated term for *procedure*. See *procedure*.

maintenance diskette. See *corrective diskette*.

Manufacturing Automated Protocol (MAP). A broadband LAN with a bus topology that passes tokens from adapter to adapter on a coaxial cable.

MAP. (1) Maintenance analysis procedure. (2) Manufacturing Automated Protocol.

mapped conversation. A conversation in which programs exchange data records with arbitrary data formats agreed upon by the applications programmers.

mapping. Establishing a correspondence between the elements of one set and the elements of another set.

master store controller. The store controller that maintains prime versions of system mirrored files and all compound files.

Mb. Megabit.

MB. Megabyte.

MCF Network. Multiple store controllers communicating on a network using DDA. This provides data redundancy among the store controllers.

media. Plural form of *medium*.

medialess. Not fitted with a direct access storage device, such as a diskette drive or fixed disk drive, as in some models of Toshiba Point of Sale Terminals.

medium. (1) A physical carrier of electrical or optical energy. (2) A physical material in or on which data may be represented.

megabit (Mb). A unit of measure for throughput. 1 megabit = 1,048,576 bits.

megabyte (MB). A unit of measure for data. 1 megabyte = 1,048,576 bytes.

megahertz (MHz). A unit of measure of frequency. 1 megahertz = 1,000,000 hertz.

memory. Program-addressable storage from which instructions and other data can be loaded directly into registers for subsequent execution or processing.

message. (1) An arbitrary amount of information whose beginning and end are defined or implied. (2) A group of characters and control bit sequences transferred as an entity. (3) In telecommunication, a combination of characters and symbols transmitted from

one point to another. (4) A logical partition of the user device's data stream to and from the adapter. See also *error message*, *operator message*.

MHz. Megahertz.

Micro Channel. The architecture used by Personal System/2 computers, Models 50 and above. This term is used to distinguish these computers from personal computers using a PC I/O channel, such as an PC, XT, or a Personal System/2 computer, Model 25 or 30.

migration. Upgrade of a program to a newer version or release.

mirrored files. Files that are kept on both the Master Store Controller and the Alternate Master Store Controller or on both the File Server and Alternate File Server. System mirrored files are kept on the Master Store Controller and Alternate Store Controller and non-system mirrored files are kept on the File Server and Alternate File Server.

Mod1. A generic name used to refer to a point-of-sale terminal in the 4690 Store System that loads and executes programs. A Mod1 can be any of the following models: 4683-001, 4683-A01, 4683-P11, 4683-P21, 4683-P41, 4683-421, 4693-xx1, and 4694-xx4 (terminal part if a controller/terminal).

Mod2. A generic name used to refer to a point-of-sale terminal in the 4690 Store System that does not load and execute programs, but attaches to a terminal that does. A Mod2 can be one of the following models: 4683-002, 4683-A02, or 4693-2x2.

modem (MOdulator/DEModulator). A device that converts digital data from a computer to an analog signal that can be transmitted in a telecommunication line, and converts the analog signal received to data for the computer.

module. A program unit that is discrete and identifiable with respect to compiling, combining with other units, and load; for example, the input to, or output from, an assembler, compiler, linkage editor, or executive routine.

modulo check. A function designed to detect most common input errors by performing a calculation on values entered into a system by an operator or scanning device.

monitor. (1) A functional unit that observes and records selected activities for analysis within a data processing system. Possible uses are to show significant departures from the norm, or to determine levels of utilization of particular functional units. (2) Software or hardware that observes, supervises, controls, or verifies operations of a system.

monochrome display. A display device that presents display images in only one color.

MSR. Magnetic stripe reader.

multiple controller system. Synonym for *MCF Network*.

multipoint. Pertaining to communication among more than two stations over a single telecommunication line.

multipoint line. A telecommunication line or circuit connecting two or more stations. Contrast with *point-to-point line*.

N

name. An alphanumeric term that identifies a data set, statement, program, or cataloged procedure.

n-bit byte. A string that consists of n bits.

NCP. Network control program.

neper. A unit for measuring power. The number of nepers is the logarithm (base e) of the ratio of the measured power level.

NetBIOS. Network Basic Input/Output System.

NetView. A host-based network management licensed program that provides communication network management (CNM) or communications and systems management (C & SM) services.

NetView Distribution Manager (NetView DM). A component of the NetView family supporting resource distribution within *Change Management*, and providing central control of software and microcode distribution and installation, to processors in a distributed/departmental (SNA) network system. It allows a similar control of user data objects across the network, and provides the facilities to support the remote initiation of command lists.

network. (1) A configuration of data processing devices and software connected for information interchange. (2) An arrangement of nodes and connecting branches. Connections are made between data stations.

network administrator. A person who manages the use and maintenance of a network.

network architecture. The logical structure and operating principles of a computer network. See also *systems network architecture (SNA)* and *Open Systems Interconnect (OSI) architecture*. **Note:** The operating principles of a network include those of services, functions, and protocols.

Network Basic Input/Output System (NetBIOS). A message interface used on LANs to provide message, print server, and file server functions. The NetBIOS application program interface (API) provides a programming interface to the LAN so that an application

program can have LAN communication without knowledge and responsibility of the data link control (DLC) interface.

network control program (NCP). A control program for the 3704 or 3705 Communications Controller, generated by the user from a library of supplied modules.

network file system (NFS). A system that allows you to mount remote file systems across homogeneous and heterogeneous systems.

network management vector transport (NMVT). The portion of an alert transport frame that contains the alert message.

NFS. network file system

node. (1) Any device, attached to a network, that transmits and/or receives data. (2) An end point of a link, or a junction common to two or more links in a network. Nodes can be processors, controllers, or workstations. Nodes can vary in routing and other functional capabilities. (3) In a network, a point where one or more functional units interconnect transmission lines.

node address. The address of an adapter on a LAN.

nonswitched line. (1) A connection between systems or devices that does not have to be made by dialing. Contrast with *switched line*. (2) A telecommunication line on which connection does not have to be established by dialing. Synonymous with *leased line*.

nonvolatile random access memory (NVRAM). Random access memory that retains its contents after electrical power is shut off.

NRZI. (1) Non-return-to-zero inverted transmission. (2) Non-return-to-reference transmission in which the zeros are represented by a bit cell boundary transition in the information signal, and ones are represented by the absence of a bit cell boundary transition.

NVRAM. nonvolatile random access memory

O

OCR. Optical character recognition.

offline. Operation of a functional unit without the control of a computer or control unit.

online. Operation of a functional unit that is under the continual control of a computer or control unit. The term also describes a user's access to a computer using a terminal.

open. (1) To make an adapter ready for use. (2) A break in an electrical circuit. (3) To make a file ready for use.

Open Systems Interconnect (OSI). (1) The interconnection of open systems in accordance with specific ISO standards. (2) The use of standardized procedures to enable the interconnection of data processing systems. **Note:** OSI architecture establishes a framework for coordinating the development of current and future standards for the interconnection of computer systems. Network functions are divided into seven layers. Each layer represents a group of related data processing and communication functions that can be carried out in a standard way to support different applications.

Open Systems Interconnect (OSI) architecture. Network architecture that adheres to a particular set of ISO standards that relates to Open Systems Interconnect (OSI).

Open Systems Interconnect (OSI) Reference Model. A model that represents the hierarchical arrangement of the seven layers described by the Open Systems Interconnect (OSI) architecture.

operating system. Software that controls the execution of programs. An operating system may provide services such as resource allocation, scheduling, input/output control, and data management. Examples are DOS and OS/2.

Operating System/2 (OS/2). A set of programs that control the operation of high-speed large-memory Personal Computers (such as the Personal System/2 computer, Models 50 and above), providing multitasking and the ability to address up to 16 MB of memory. Contrast with *Disk Operating System (DOS)*.

operation. (1) A defined action, namely, the act of obtaining a result from one or more operands in accordance with a rule that completely specifies the result for any permissible combination of operands. (2) A program step undertaken or executed by a computer. (3) An action performed on one or more data items, such as adding, multiplying, comparing, or moving.

operational environment. (1) A summation of all of the supplied basic functions and the user programs that can be executed by the store controller to enable the devices in the system to perform specific operations. (2) The collection of supplied controller data and user programs, plus lists, tables, control blocks, and files that reside in a controller and control its operation. (3) The physical environment (for example: temperature, humidity, layout, or power requirements) that is needed for proper machine performance.

operator. (1) A symbol that represents the action being performed in a mathematical operation. (2) A person who operates a machine.

operator message. A message from the operating system or a program telling the operator to perform a specific function or informing the operator of a specific condition within the system, such as an error condition.

optical character recognition (OCR). The machine identification of printed characters through the use of light-sensitive devices.

option. (1) A specification in a statement, a selection from a menu, or a setting of a switch, that may be used to influence the execution of a program. (2) A hardware or software function that may be selected or enabled as part of a configuration process. (3) A piece of hardware (such as a network adapter) that can be installed in a device to modify or enhance device function.

OS. Operating system.

OS/2. Operating System/2.

OSI. Open Systems Interconnect.

OS/VS. Operating System/Virtual Storage.

owner. In relation to files, an owner is the user that creates the file and therefore has complete access to the file.

P

pacing. A technique by which a receiving component controls the rate of transmission by a sending component to prevent overrun or congestion.

packet. (1) In data communication, a sequence of binary digits, including data and control signals, that is transmitted and switched as a composite whole. (2) Synonymous with *data frame*. Contrast with *frame*.

packet assembler/disassembler (PAD). A functional unit that enables data terminal equipments (DTEs) not equipped for packet switching to access a packet switched network.

packing. Method of conserving disk storage space by stripping the high-order nibbles from ASCII numerals and storing the remaining low-order nibbles two to a byte.

PAD. Packet assembler/disassembler.

page. (1) The portion of a panel that is shown on a display surface at one time. (2) To move back and forth among the pages of a multiple-page panel. See also *scroll*. (3) In a virtual storage system, a fixed-length block that has a virtual address and is transferred as a unit between main storage and auxiliary storage.

panel. The complete set of formatted information that appears in a single display on a visual display unit.

parallel port. (1) A port that transmits the bits of a byte in parallel along the lines of the bus, one byte at a time, to an I/O device. (2) On a personal computer, it is used to connect a device that uses a parallel interface, such as a dot matrix printer, to the computer. Contrast with *serial port*.

parameter. (1) A name in a procedure that is used to refer to an argument passed to that procedure. (2) A variable that is given a constant value for a specified application and that may denote the application. (3) An item in a menu or for which the user specifies a value or for which the system provides a value when the menu is interpreted. (4) Data passed between programs or procedures.

parity (even). A condition when the sum of all of the digits in an array of binary digits is even.

parity (odd). A condition when the sum of all of the digits in an array of binary digits is odd.

partner. See *conversation partner*.

partner terminal. The term used to describe the relationship of a Mod 1 terminal and Mod 2 terminal when they are attached to each other.

password. In computer security, a string of characters known to the computer system and a user, who must specify it to gain full or limited access to a system and to the data stored within it.

path. (1) Reference that specifies the location of a particular file within the various directories and subdirectories of a hierarchical file system. (2) In a network, any route between any two nodes. (3) The route traversed by the information exchanged between two attaching devices in a network. (4) A command in DOS and OS/2 that specifies directories to be searched for commands or batch files that are not found by a search of the current directory.

PCI DSS. Payment Card Industry Data Security Standards.

peer node. Any *other* SNA type (2.1) node (another 4680/4690 store controller, AS/400, or others).

permanent virtual circuit (PVC). A virtual circuit that has a logical channel permanently assigned to it at each data terminal equipment (DTE). A call establishment protocol is not required.

personal computer (PC). A desk-top, free-standing, or portable microcomputer that usually consists of a system unit, a display, a keyboard, one or more diskette drives, internal fixed-disk storage, and an optional printer. PCs are designed primarily to give independent computing power to a single user and are inexpensively priced for purchase by individuals or small businesses. Examples include the various models of the Personal Computers, and the Personal System/2 computer.

personal identification number (PIN). A numeric identification code assigned to a customer to protect funds and data from unauthorized users.

physical link. In an MVS/VS multisystem environment, the actual hardware connection between two systems. Contrast with *logical link*.

physical unit (PU). In SNA, the component that manages and monitors the resources of a node, such as attached links and adjacent link stations, as requested by a system services control point (SSCP) through an SSCP-SSCP session.

pipe. A sequential file in a memory buffer that is used to pass messages from one program to another.

PLD. Power line disturbance.

plug. (1) A connector for attaching wires from a device to a cable, such as a store loop. A plug is inserted into a receptacle or plug. (2) To insert a connector into a receptacle or socket.

point-of-sale terminal. A unit that provides point-of-sale transaction, data collection, credit authorization, price look-up, and other inquiry and data entry functions.

point-to-point line. A switched or nonswitched telecommunication line that connects a single remote station to a computer. Contrast with *multipoint line*.

polling. (1) Interrogation of devices for purposes such as to avoid contention, to determine operational status, or to determine readiness to send or receive data. (2) In data communication, the process of inviting data stations to transmit, one at a time. The polling process usually involves the sequential interrogation of several data stations.

polling characters (address). A set of characters specific to a terminal and the polling operation; response to these characters indicates to the computer whether the terminal has a message to enter.

port. (1) An access point for data entry or exit. (2) A connector on a device to which cables for other devices such as display stations and printers are attached. Synonymous with *socket*.

post. (1) To affix to a usual place. (2) To provide items such as return code at the end of a command or function. (3) To define an appendage routine. (4) To note the occurrence of an event.

POST. Power-On Self Test.

power line disturbance (PLD). Interruption or reduction of electrical power.

Power-On Self Test (POST). A series of diagnostic tests that are run automatically each time the computer's power is switched on.

presentation space (PS). In 3270 emulation, the image of the 3270 screen data that is held in random access memory. This screen appears on the store

controller or the terminal display when 3270 emulation is used in operator console mode; it is the virtual screen for applications using the 3270 emulator API. The presentation space is fixed as 24 lines of 80 characters on the display.

primary adapter. In a personal computer that is used on a LAN and that supports installation of two network adapters, the adapter that uses standard (or default) mapping between adapter shared RAM, adapter ROM, and designated computer memory segments. The primary adapter is usually designated as adapter 0 in configuration parameters. Contrast with *alternate adapter*.

primary application. A program that controls the normal operating environment of your store (for example, programs that provide sales support).

primary store controller. The store controller designated to control the store loop under normal conditions.

prime version. The version of a file to which updates are made. The prime version of a file may be maintained on either the Master Store Controller or the File Server. Copies of the prime version, called image versions, are distributed to other store controllers.

printout. Any printed document produced by a point-of-sale terminal printer or by some other printer.

problem determination. The process of determining the source of a problem as being a program component, a machine failure, a change in the environment, a common-carrier link, a user-supplied device, or a user error.

procedure. (1) A set of related control statements that cause one or more programs to be performed. (2) In a programming language, a block, with or without formal parameters, whose execution is invoked by means of a procedure call. (3) A set of instructions that gives a service representative a step-by-step procedure for tracing a symptom to the cause of failure.

processor. In a computer, a functional unit that interprets and executes instructions.

Programmable Store System (PSS). A store system, such as the Store System, that can be programmed to perform user-determined functions.

prompt. A character or word displayed by the operating system to indicate that it is ready to accept input.

protected field. A display field that the terminal operator cannot add to, modify, or erase using the keyboard. Contrast with *input field* and *unprotected field*.

protocol. (1) A set of semantic and syntactic rules that determines the behavior of functional units in achieving

communication. (2) In SNA, the meanings of and the sequencing rules for requests and responses used for managing the network, transferring data, and synchronizing the states of network components. (3) A specification for the format and relative timing of information exchanged between communicating parties.

PS. Presentation space.

PSS. Programmable Store System.

PU. Physical unit.

public switched (telephone) network (PSN). A telephone network that provides lines and exchanges to the public. It is operated by the communication common carriers in the USA and Canada, and by the PTT Administrations in other countries.

PVC. Permanent virtual circuit.

Q

queue. A line or list formed by items in a system waiting for service; for example, tasks to be performed or messages to be transmitted in a message routing system.

R

RAM. Random access memory.

RAM disk. Synonym for *virtual drive*.

RAM paging. A technique that allows the computer software to access all of the RAM on adapters that contain 64 KB of RAM, without having to map the entire shared RAM into the computer's memory map. The shared RAM on the adapter is paged into the computer's memory map one 16 KB page at a time.

random access. An access mode in which specific logical records are obtained from or placed into a mass storage file in a nonsequential manner.

random access memory (RAM). A computer's or adapter's volatile storage area into which data may be entered and retrieved in a nonsequential manner.

RCMS. Remote change management server.

read. To acquire or to interpret data from a storage device, from a data medium, or from another source.

read-only memory (ROM). A computer's or adapter's storage area whose contents cannot be modified by the user except under special circumstances.

real storage. The main storage in an virtual storage system. Contrast with *virtual storage (VS)*.

receive. To obtain and store information transmitted from a device.

record. A collection of related items of data, treated as a unit; for example, in stock control, each invoice could constitute one record. A complete set of such records may form a file.

reference diskette. A diskette shipped with the point-of-sale equipment. The diskette contains code and files used for configuration of options and for hardware diagnostic testing.

remote change management server (RCMS). The Store System function that interfaces with the host DSX program for file transmission.

remote program. The program at the other end of a conversation with respect to the reference program. Contrast with *local program*.

remote program load (RPL). A function provided by adapter hardware components and software that enables one computer to load programs and operating systems into the memory of another computer, without requiring the use of a diskette or fixed disk at the receiving computer.

remove. (1) To take an attaching device off a network. (2) To stop an adapter from participating in data passing on a network.

response. The information the network control program sends to the access method, usually in answer to a request received from the access method. (Some responses, however, result from conditions occurring within the network control program, such as accumulation of error statistics.)

retry. In data communication, sending the current block of data a prescribed number of times or until it is entered correctly and accepted.

return code. (1) A value (usually hexadecimal) provided by an adapter or a program to indicate the result of an action, command, or operation. (2) A code used to influence the execution of succeeding instructions. (3) A value established by the programmer to be used to influence subsequent program action. This value can be printed as output or loaded in a register.

ring network. A network configuration in which a series of attaching devices is connected by unidirectional transmission links to form a closed path.

ring segment. Any section of a ring that can be isolated (by unplugging connectors) from the rest of the ring. A segment can consist of a single lobe, the cable between access units, or a combination of cables, lobes, and/or access units. See *cable segment*, *LAN segment*.

ring station. A station that supports the functions necessary for connecting to the LAN and for operating with the token-ring protocols. These include token handling, transferring copied frames from the ring to the

using node's storage, maintaining error counters, observing medium access control (MAC) sublayer protocols (for address acquisition, error reporting, or other duties), and (in the full-function native mode) directing frames to the correct data link control (DLC) link station.

ring status. The condition of the ring.

RIPL. Remote IPL.

RMA. Remote Management Agent.

ROM. Read-only memory.

root directory. Highest or base level directory in a hierarchical file system. Subdirectories branch off of the root directory.

router. An attaching device that connects two LAN segments, which use similar or different architectures, at the Reference Model network layer. Contrast with *bridge* and *gateway*.

routing. (1) The assignment of the path by which a message will reach its destination. (2) The forwarding of a message unit along a particular path through a network, as determined by the parameters carried in the message unit, such as the destination network address in a transmission header.

RPL. Remote program load.

S

SAA. Systems Application Architecture.

SABM. Set asynchronous balanced mode.

satellite. (1) A computer that is under the control of another computer and performs subsidiary operations. (2) An offline auxiliary computer. (3) A Toshiba point-of-sale terminal under the control of a master terminal.

SBCS. Single-byte character set.

scan. To pass an item over or through the scanner so that the encoded information is read. See also *wandering*.

scanner. A device that examines the bar code on merchandise tickets, credit cards, and employee badges and generates analog or digital signals corresponding to the bar code.

scroll. To move all or part of the display image vertically or horizontally to display data that cannot be observed within a single display image. See also *page (2)*.

SDLC. Synchronous Data Link Control.

SDLC link. A data link over which communications are conducted using the Synchronous Data Link Control (SDLC) discipline.

secondary application. A user-written program that is designed to operate with operator intervention.

sector. A 512-byte area of the control unit diskette, the amount of data that is transferred at one time to or from the diskette.

segment. See *cable segment*, *LAN segment*, *ring segment*.

serial port. On personal computers, a port used to attach devices such as display devices, letter-quality printers, modems, plotters, and pointing devices such as light pens and mice; it transmits data one bit at a time. Contrast with *parallel port*.

server. (1) A device, program, or code module on a network dedicated to providing a specific service to a network. (2) On a LAN, a data station that provides facilities to other data stations. Examples are a file server, print server, and mail server.

session. (1) A connection between two application programs that allows them to communicate. (2) In SNA, a logical connection between two network addressable units that can be activated, tailored to provide various protocols, and deactivated as requested. (3) The data transport connection resulting from a call or link between two devices. (4) The period of time during which a user of a node can communicate with an interactive system, usually the elapsed time between log on and log off. (5) In network architecture, an association of facilities necessary for establishing, maintaining, and releasing connections for communication between stations.

session group. In System/36 advanced program-to-program communication, a number of sessions managed as a unit.

set asynchronous balanced mode (SABM). In communications, a data link control command used to establish a data link connection with the destination in asynchronous balanced mode. See also *asynchronous balanced mode (ABM)*.

shared RAM. Random access memory on an adapter that is shared by the computer in which the adapter is installed.

signal. (1) A time-dependent value attached to a physical phenomenon for conveying data. (2) A variation of a physical quantity, used to convey data.

sign-on. (1) A procedure to be followed at a terminal or workstation to establish a link to a computer. (2) To begin a session at a workstation.

single-byte character set (SBCS). A character set in which each character is represented by a one-byte code. Contrast with double-byte character set.

SNA. Systems Network Architecture.

socket. Synonym for *port* (2).

source. The origin of any data involved in a data transfer.

SSCP. System services control point.

state. See *conversation state*.

station. (1) A point-of-sale terminal that consists of a processing unit, a keyboard, and a display. It can also have input/output devices, such as a printer, a magnetic stripe reader or cash drawers. (2) A communication device attached to a network. The term used most often in LANs is an *attaching device* or *workstation*. (3) An input or output point of a system that uses telecommunication facilities; for example, one or more systems, computers, terminals, devices, and associated programs at a particular location that can send or receive data over a telecommunication line. See also *attaching device*, *workstation*.

store controller. A programmable unit in a network used to collect data, to direct inquiries, and to control communication within a point-of-sale system.

store loop. In the Store System, a cable over which data is transmitted between the store controller and the point-of-sale terminals.

Store Loop Adapter. A hardware component used to connect the loop to a store controller.

subarea node. An SNA type 5 node (a host processor), which will control all communications with the store controller.

subdirectory. Any level of file directory lower than the root directory within a hierarchical file system.

subordinate store controller. A store controller that receives copies of all system compound files and may also receive all application compound files.

supervisory (S) frame. A frame in supervisory format used to transfer supervisory control functions. See also *information frame*, *unnumbered frame*.

SVC. Switched virtual circuit.

switch. On an adapter, a mechanism used to select a value for, enable, or disable a configurable option or feature.

switched line. A telecommunication line in which the connection is established by dialing. Contrast with *nonswitched line*.

switched virtual circuit (SVC). A virtual circuit that is requested by a virtual call. It is released when the virtual circuit is cleared.

symbolic destination name. Variable corresponding to an entry in the side information.

synchronous. (1) Pertaining to two or more processes that depend upon the occurrence of a specific event such as a common timing signal. (2) Occurring with a regular or predictable timing relationship.

Synchronous Data Link Control (SDLC). A discipline conforming to subsets of the Advanced Data Communication Control Procedures (ADCCP) of the American National Standards Institute (ANSI) and High-level Data Link Control (HDLC) of the International Organization for Standardization, for managing synchronous, code-transparent, serial-by-bit information transfer over a link connection. Transmission exchanges may be duplex or half-duplex over switched or nonswitched links. The configuration of the link connection may be point-to-point, multipoint, or loop.

system. In data processing, a collection of people, machines, and methods organized to accomplish a set of specific functions. See also *data processing system* and *operating system*.

system board. In a system unit, the main circuit board that supports a variety of basic system devices, such as a keyboard or a mouse, and provides other basic system functions.

system configuration. A process that specifies the devices and programs that form a particular data processing system.

Systems Application Architecture (SAA). An architecture developed by IBM that consists of a set of selected software interfaces, conventions, and protocols, and that serves as a common framework for application development, portability, and use across different hardware systems.

system services control point (SSCP). In SNA, the focal point within an SNA network for managing the configuration, coordinating network operator and problem determination requests, and providing directory support and other session services for end users of the network. Multiple SSCPs, cooperating as peers, can divide the network into domains of control, with each SSCP having a hierarchical control relationship to the physical units (PUs) and logical units (LUs) within its domain.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through, and controlling the configuration and operation of, networks. **Note:** The layered structure of SNA allows the ultimate origins and destinations of information, that is, the end users, to be independent of,

and unaffected by, the specific SNA network services and facilities used for information exchange.

T

task. A basic unit of work.

TCC Network. A system in which the terminals and controllers communicate using either a store loop, a token-ring or an Ethernet.

telephone twisted pair. One or more twisted pairs of copper wire in the unshielded voice-grade cable commonly used to connect a telephone to its wall jack. Also referred to as “unshielded twisted pair” (UTP).

tender. Money, checks, coupons, or trading stamps used as payment for merchandise or service.

terminal. In data communication, a device, usually equipped with a keyboard and a display, capable of sending and receiving information over a communication channel.

terminal number. A number assigned to a terminal to identify it for addressing purposes.

threshold. (1) A level, point, or value above which something is true or will take place and below which it is not true or will not take place. (2) In Toshiba bridge programs, a value set for the maximum number of frames that are not forwarded across a bridge due to errors, before a “threshold exceeded” occurrence is counted and indicated to network management programs. (3) An initial value from which a counter is decremented from an initial value. When the counter reaches zero or the threshold value, a decision is made and/or an event occurs.

till. A tray in the cash drawer of the point-of-sale terminal, used to keep the different denominations of bills and coins separated and easily accessible.

token. A sequence of bits passed from one device to another on the token-ring network that signifies permission to transmit over the network. It consists of a starting delimiter, an access control field, and an end delimiter. The frame control field contains a token bit that indicates to a receiving device that the token is ready to accept information. If a device has data to send along the network, it appends the data to the token. When data is appended, the token then becomes a frame. See *frame*.

token-ring. A network with a ring topology that passes tokens from one attaching device (node) to another. A node that is ready to send can capture a token and insert data for transmission.

token-ring network. (1) A ring network that allows unidirectional data transmission between data stations by a token-passing procedure over one transmission

medium so that the transmitted data returns to and is removed by the transmitting station. The Token-Ring Network is a baseband LAN with a star-wired ring topology that passes tokens from network adapter to network adapter. (2) A network that uses a ring topology, in which tokens are passed in a circuit from node to node. A node that is ready to send can capture the token and insert data for transmission. (3) A group of interconnected token-rings.

TP. Transaction program.

trace. (1) A record of the execution of a computer program. It exhibits the sequences in which the instructions were executed. (2) A record of the frames and bytes transmitted on a network.

transaction. (1) The process of recording item sales, processing refunds, recording coupons, handling voids, verifying checks before tendering, and arriving at the amount to be paid by or to a customer. The receiving of payment for merchandise or service is also included in a transaction. (2) In an SNA network, an exchange between two programs that usually involves a specific set of initial input data that causes the execution of a specific task or job. Examples of transactions include the entry of a customer's deposit that results in the updating of the customer's balance, and the transfer of a message to one or more destination points.

transaction program (TP). A program that processes transactions in or through a logical unit (LU) type 6.2 in an SNA network. Application transaction programs are end users in an SNA network; they process transactions for service transaction programs and for other end users. Service transaction programs are Toshiba-supplied programs that typically provide utility services to application transaction programs.

transmission. The sending of data from one place for reception elsewhere.

transmit. To send information from one place for reception elsewhere.

twisted pair. A transmission medium that consists of two insulated conductors twisted together to reduce noise.

typematic. A keyboard button that will continue to enter characters or repeat its function as long as the button is held down.

U

uninterruptible power supply. A buffer between utility power or other power source and a load that requires uninterrupted, precise power.

universal product code (UPC). An encoded number that can be assigned to and printed on or attached to an article of merchandise for scanning.

universal serial bus. An industry standard that makes it easy to expand PC functionality. The USB is a 12-Mbps serial bus designed to replace almost all low-to-medium speed connections to peripheral devices such as keyboards, mice, and printers.

unnumbered acknowledgment. A data link control (DLC) command used in establishing a link and in answering receipt of logical link control (LLC) frames.

unnumbered (U) frame. A frame in unnumbered format, used to transfer unnumbered control functions. See also *information frame*, *supervisory frame*.

unprotected field. A display field that the terminal operator can add to, modify, or erase using the keyboard. Contrast with *protected field*.

UPC. Universal product code.

UPS. Uninterruptible power supply.

USB. universal serial bus

user. (1) Category of identification defined for file access protection. (2) A person using a program or system.

user exit. A point in a supplied program at which a user-written program may be given control.

utility program. (1) A computer program in general support of the processes of a computer; for instance, a diagnostic program, a trace program, a sort program. (2) A program designed to perform an everyday task such as copying data from one storage device to another.

V

variable. (1) A named entity that is used to refer to data and to which values can be assigned. Its attributes remain constant, but it can refer to different values at different times. (2) In computer programming, a character or group of characters that refers to a value and, in the execution of a computer program, corresponds to an address. (3) A quantity that can assume any of a given set of values.

version. A separate Toshiba-licensed program, based on an existing Toshiba-licensed program, that usually has significant new code or new function.

VFD. Vacuum fluorescent display.

VFS. virtual file system.

video display. (1) An electronic transaction display that presents visual information to the point-of-sale terminal operator and to the customer. (2) An electronic display screen that presents visual information to the display operator.

virtual circuit. Synonym for *virtual connection*.

virtual connection. (1) A connection between two nodes on the network that is established using the transport layer and provides reliable data between nodes. (2) A logical connection established between two data terminal equipment (DTE) devices. Synonymous with *virtual circuit*.

virtual drive. Computer memory used as if it were a direct access storage device. Synonym for *RAM disk*.

virtual file system (VFS). Within 4690 OS V2, the virtual file system is used to provide support for long file names by creating two virtual drives that support file names greater than eight characters in length.

virtual machine (VM). A functional simulation of a computer and its associated devices. Each virtual machine is controlled by a suitable operating system, for example, a conversational monitor system. VM controls concurrent execution of multiple virtual machines on one host computer.

virtual storage (VS). (1) The storage space that may be regarded as addressable main storage by the user of a computer system in which virtual addresses are mapped into real addresses. The size of virtual storage is limited by the addressing scheme of the computer system and by the amount of auxiliary storage available, not by the actual number of main storage locations. (2) Addressable space that is apparent to the user as the processor storage space, from which the instructions and the data are mapped into the processor storage locations. Contrast with *real storage*.

VM. Virtual machine.

VS. Virtual storage.

W

wand. A commercially available device used to read information encoded on merchandise tickets, credit cards, and employee badges.

wanding. Passing the tip of the wand reader over information encoded on a merchandise ticket, credit card, or employee badge.

wideband. Synonym for *broadband*.

work file. A file that is both created and deleted in the same job.

workstation. (1) An I/O device that allows either transmission of data or the reception of data (or both) from a host system, as needed to perform a job: for example, a display station or printer. (2) A configuration of I/O equipment at which an operator works. (3) A terminal or microcomputer, usually one connected to a mainframe or network, at which a user can perform tasks.

X

XID. Exchange identification.

X.21. In data communication, a recommendation of the CCITT that defines the interface between data terminal equipment (DTE) and public data networks for digital leased and circuit switched synchronous services.

X.21 bis. In data communication, an interim specification of the CCITT that defines the connection of data terminal equipment (DTE) to an X.21 (public data) network using V-series interchange circuits such as those defined by CCITT V.24 and CCITT V.35.

X.25. A CCITT Recommendation that defines the physical level (physical layer), link level (data link layer), and packet level (network layer), of the OSI Reference Model. An X.25 network is an interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) operating in the packet mode, and connected to public data networks by dedicated circuits. X.25 networks use the connection-mode network service.

Index

Special characters

- ? global file name character 142
- * global file name character 142
- @# keys 136
- > redirect output character 144
- >> append file character 144
- < input source character 144
- ^ global file name character 143
- | 145

Numerics

- 3270 emulation
 - See SNA 3270 emulation
- 3270 emulator standard names 502
- 4680 BASIC statements and SNA
 - connecting OS/2 PC to the LAN 473
- 4690 Directory Services 341
- 4690 operating system, DVD support for the 10
- 4693/4694 ANPOS keyboard 117
- 4693/4694 modifiable layout keyboard 117
- 4693/4694 POS keyboard 117
- 50-key keyboard 117
- 6157 streaming tape drive 83

A

- accessing
 - display/alter utility 359
 - file compress/decompress utility 405
 - files on another controller 451
 - streaming tape drive utility 386
- activate
 - configuration 75
 - controller configuration 77
 - file server 102
 - logical name changes 75
 - master store controller 102
 - system configuration 78
 - system file size 75
 - terminal configuration 76
- activating
 - alternate store controller roles 463
 - file server store controller 462
 - master store controller 423, 461
 - when to 463
- active terminal configuration data, loading 98
- ADX_SDT1:ADXNOWRV.DAT 14
- ADXCLLOG.DAT, master log file 31
- ADXCS30L commands 406
- ADXCSJ0L, logging activity 359
- ADXCSU0L, using in command mode 470
- ADXCSU1L command 531
- ADXEPMWF.DAT 21
- ADXFMS3Z command
 - format 533

- ADXFMS6Z command
 - format 535
- ADXHSNLL 106
- ADXHSNSL 107
- ADXIPCFG.286 412
- ADXNDSKL command
 - format 261
- ADXNSL0L 414
- ADXNSX0L command 536
- ADXSDEL command 163
- aids, installation and update 84
- alarm, audible 86
- allowing store controller backup 100
- alphanumeric display character set 53, 59
- alphanumeric keyboard 114
 - system function keys 114
- alphanumeric point-of-sale keyboard 117
- alternate file server 449
- alternate master store controller
 - as backup 9
 - definition 9
 - file server 9
 - general discussion 449
- ANPOS keyboard 117
- API 477
- application
 - background 19
- application modes
 - multiple 16
 - single 16
- application program interface
 - using 477
 - using with 3270 emulation 514
- applications
 - mirrored file 447
 - where file resides 447
- Apply Software Maintenance 365
 - combining a delta with maintenance 373
 - preparing for 371
 - procedure for applying changes 365
 - using product-level files 369
- applying software maintenance 85
- areas
 - command line 286
 - message 271, 287
 - prompt 270
 - status 271, 287
 - text 270, 286
- ASM 373
- ASSIGN command
 - format 163, 165
 - LAN format 166
- at close method 448
- attributes
 - compound 447
 - file mode 446
 - file type 446
 - local file extension 447
 - mirrored file extension 447
- audible alarm
 - message file 86

- audible alarm (*continued*)
 - setting 86
- audible alarm file
 - building 86
- authorization
 - file 18
 - for menu options 19
- authorized operators 18
- auto sign off 50
- auxiliary console
 - control 15
 - introduction to 15
 - key functions 38
 - back tab 506
 - clear 506, 507
 - color mode 509
 - cursor select 506
 - delete 507
 - dev cncl 508
 - down arrow 505
 - duplicate 508
 - EBCDIC character 504
 - Enter 506
 - erase EOF 507
 - erase input 507
 - exit 509
 - field mark 508
 - home 506
 - identify 508
 - insert 508
 - left arrow 505
 - new Line 506
 - numovride 509
 - OGL toggle 509
 - PA1 506
 - PA2 506
 - PA3 506
 - PF1 -> PF10 506
 - PF11 506
 - PF12 506
 - PF13 -> PF22 506
 - PF23 506
 - PF24 506
 - print 508
 - program access 506
 - program function 506
 - reset 508
 - right arrow 505
 - system request 508
 - tab 506
 - up arrow 505
 - keyboards 15

B

- background
 - application 19
 - control operator 19
- backspace key 135
- backup
 - allowing store controller 100

- backup (*continued*)
 - LAN 493
 - preventing store controller 101
 - providing store controller 9
 - store controller 493
 - using optical diskettes 393
- BACKUP command
 - format 167
 - LAN format 158
- backup files
 - using tape streaming 385
- BATCH command
 - LAN format 170
- batch commands
 - BATCH 170
 - CHDIR 171
 - ECHO 203
 - FOR 213
 - GOTO 221
 - IF 223
 - PAUSE 234
 - REM 240
 - SHIFT 249
- batch files
 - creating 157
 - general purpose 158
- battery return program 551
- build software maintenance control file
 - installing a user product 381
 - using product control files 379
- building a software library 381
 - naming conventions for library 381
 - using 381
- burning ISO image
 - CD or memory key 262

C

- cable ferrite requirement 550
- CD
 - burning ISO image 262
- changes
 - configuration data 84
 - current directory 140
 - input sequence table 85
 - keyboard tone volume 120
 - Mod1 (or A01) configuration 129
 - Mod2 configuration 129
 - terminal configuration 128
- changing
 - current directory using display/alter utility 359
 - distributed file to local 469
 - file distribution attributes 468
 - local file to distributed on LAN system 469
- changing national language selection
 - alphanumeric keyboard 527
 - country-specific code page 527
 - hard disk drive 526
- character set
 - printer, model 1 or 2 53
 - printer, model 3 or 4 53

- character set *(continued)*
 - worksheet for 53, 59
- CHDIR (change directory) command 171
- CHKDSK (check disk) command 173
- choosing options from menu 81
- Class A compliance statement
 - Australia and New Zealand 545
 - China 548
 - European Union 545, 549
 - FCC (USA) 545
 - Germany 546
 - Industry Canada 545
 - Japan 549
 - Russia 548
 - Taiwan 548
- clearing message display 119
- CLS (clear screen) command 178
- code page
 - country-specific 527
- collecting problem analysis data 100
- combined keyboard/display 117
- combining controller roles 449
- command
 - ADXSDDEL 163
- COMMAND command 179
- command line activity, logging 134
- command line logging 31
- Command Line Logging and the XE Editor 289
- command mode
 - exiting 133
 - selecting 133
 - using 91
 - using distributed file utility 469
 - working in 133
- command tail 514
- commands
 - ADXCUSU1L 531
 - ADXFMS3Z 533
 - ADXFMS6Z 535
 - ADXNDSKL 261
 - ADXNSCOL 536
 - ADXSDDEL 163
 - ASSIGN 165
 - BACKUP 167
 - batch 157, 170, 250
 - CHDIR 171
 - CHKDSK 173
 - CLS 178
 - COMMAND 179
 - COMP 181
 - COPY 184
 - DEFINE 190
 - DIR 193
 - DISKCOMP 198
 - DISKCOPY 200
 - DVRLOAD 537
 - DVRUNIT 539
 - DVRUNLK 540
 - ECHO 203
 - ELSE 205
 - entering several 136

commands (*continued*)

- ERASE 206
- EXIT 208
- external 153
- FCOPY 209
- FIND 210
- FOR 213
- FORMAT 214
- FSET 218
- GOTO 221
- IF 223
- internal 153
- LAN commands 156
- LIST 226
- MKDIR 227
- MORE 229
- ORDER 231
- PATH 232
- PAUSE 234
- PRINT 235
- PROMPT 237
- QUERY 239
- REM 240
- RENAME 241
- RESTORE 243
- RMDIR 247
- SECURITY 248
- SHIFT 249
- SORT 251
- SYS 541
- system 147
- TREE 255
- TYPE 257
- types of 153
- using on LAN 153
- VER 259
- VERIFY 260
- VOL 261
- commands on the supplementals 528
- communications functions 105
- COMP command 181
- compound file
 - example of 447
 - from host processor 406
 - from panels 405
 - where file resides 447
 - with diskettes 406
- compressing files using the utility 405
- compression, file 83
- configuration
 - activating 75
 - changing Mod1 (or A01) 129
 - changing Mod2 129
 - changing terminal 128
 - controller 71
 - description 8
 - general information 47
 - terminal load 52
 - terminal, non-USB 52
 - terminals, SurePOS 4694 700 and TCxWave 6140 Series
 - TCxWave 6140 Series 58, 59

- configuration (*continued*)
 - using worksheets for 47
- configuration data
 - changing 84
 - reporting generic 84
 - reporting legacy 84
- configuration data, loading into terminal 98
- connecting
 - DOS to the 4690 MCF 478
 - OS/2 PC to LAN 473
- console
 - auxiliary 15
 - main 15
- console ID security 19
- console session through API 503
- console, auxiliary
 - See auxiliary console
- controller functions menu 99
- controller ID
 - node ID 450
- controller NFS configuration 74
- controller, store
 - store controller backup 493
- COPY command
 - format 184
 - LAN format 187
- copying diskettes 148, 149
- country-specific code page 527
- CPREP 411
- critical sector repair 363
- Ctrl+R keys 135
- Ctrl-C keys 136
- Ctrl-Q keys 135
- Ctrl+F keys 136
- Ctrl+S keys 135
- Ctrl+V keys 136
- Ctrl+X keys 135
- current directory
 - changing using display/alter utility 359
 - making a 139
- current directory, changing 140

D

- damage from electrostatic discharge 549
- data distribution application (DDA) 445
- date and time, setting system 104
- DDA system 72
- deactivate
 - file server 102
 - file server store controller 462
 - master store controller 102, 423, 461
- decompressing files
 - from host processor 406
 - from panels 405
 - with diskettes 406
- decompression, file 83
- default configuration
 - reconfiguring 48
 - saving 48
- default ID 48

- default password 48
- default value 34
- defaults
 - operator identification 48
 - password 48
- DEFINE command 190
- definitions 19
 - command-driven option 5
 - default value 34
 - function keys line 35
 - local area network system 9
 - menu-driven screens 5
 - message line 35
 - status line 35
 - store controller backup 9
 - windows 16
- delete key 36, 135
- deleting
 - files securely 33
- Delta diskette
 - combining with maintenance 373
- destination diskette 148
- determining controller IP information, Disk Rebuild Utility 412
- device characteristics 58
- device groups 52
- DFU (distributed file utility) 465
- DIR command
 - format 193
 - LAN format 197
- directory
 - current 139
 - path 139
 - root 138
- Directory
 - Services 341
- disable LAN for LU 6.2 107
- disable link (force) 107
- disable link (no force) 106
- disable terminal storage retention 97
- disabling RAM disk 99
- disconnect 40
- discovery, store controller 493
- disk rebuild utility 409, 541
- disk rebuild utility over IP 409
- Disk Rebuild Utility, determining controller IP information 412
- disk space, compress/decompress utility 408
- DISKCOMP command 198
- DISKCOPY command 200
- diskette 9
 - backup copy 13
 - copying 148, 149
 - destination 148
 - drives 9
 - formatting 13
 - handling 12
 - IPLing using CD-ROM 521
 - source 148
 - supplemental 45, 521
 - write-protecting 12
- diskette drive
 - selecting using display/alter utility 359

- display
 - clearing terminal message 119
- display LAN status for LU 6.2 108
- display link 107
- display/alter file utility
 - accessing 359
 - changing current directory using 359
 - functions in file mode 360
 - modes of use 359
 - physical sector mode 361
 - replacing current sector 360
 - requesting file scan function 360
 - saving the current sector 360
 - scanning a file 360, 361
 - selecting a disk drive 359
 - selecting a file using 360
 - selecting file mode using 359
 - selecting new relative sector 360
 - starting 359
 - from command mode 359
 - from main menu 359
 - using 359
 - writing changes to a file function 360
- displaying
 - displaying offline messages 118
 - displaying pending system messages 118
 - displaying terminal configuration data 126
 - displaying wait messages 118
 - file distribution attributes 468
 - log entries 466
 - status of store controllers 461
- disposal of equipment 550
- distributed file
 - at close 455
 - changing local file to a 469
 - changing to local file 469
 - compound file 455
 - mirrored file 455
 - per update 455
- distributed file utility
 - changing file distribution attributes 468
 - displaying file distribution attributes 468
 - displaying log entries 466
 - distributing files 469
 - modifying exception log 467
 - modifying file distribution attributes 468
 - processing log 466
 - using 465
 - using in command mode 469
- distributing files
 - at close method 448
 - distributing a file 469
 - per update method 448
- Distribution Exception Log
 - handling distribution exceptions on a LAN (MCF Network) system 371
 - reconciliation 453
 - tracking 452
 - using 452
 - using the Distributed File Utility 465
 - when to display and modify 467
- down arrow key 36

- DPREP 411
- DR EDIX
 - commands 278
 - editing commands 267
 - entering commands 266
 - error messages 280
 - messages 271
 - SYSTEM MAIN MENU 265
- DR EDIX panel 269
- drives
 - identifying 14
 - names of 157
- dual touch 112
- dual-track magnetic stripe reader 111
- dump terminal storage 98
- dumping store controller storage 100
- dumping terminal storage 121
- DVD support for the 4690 operating system 10
- DVRLOAD command 537
- DVRUNIT command 539
- DVRUNLK command 540

E

- ECHO command 203
- editing keys
 - @# 136
 - backspace 135
 - Ctrl-C 136
 - Ctrl-Q 135
 - Ctrl+F 136
 - Ctrl+R 135
 - Ctrl+S 135
 - Ctrl+V 136
 - Ctrl+X 135
 - delete 135
- electromagnetic Interference statement
 - Russia 548
- electronic emissions notices 545
 - Australia and New Zealand 545
 - China 548
 - European Union 545, 549
 - FCC (USA) 545
 - Germany 546
 - Industry Canada 545
 - Japan 549
 - Korea 548
 - Taiwan 548
- electrostatic discharge (ESD) 549
- ELSE command 205
- enable terminal storage retention 96
- enabling MBrowser 327
- enabling RAM disk 99
- end key 36
- end of life disposal 550
- Enhanced Security
 - configuring 23
 - operator authorization records 24
 - password settings 23
 - description 21
 - disabling 22

- Enhanced Security *(continued)*
 - enabling 22
 - operator authorizations records 24
 - password rules 22
- Enter key 36
- entering several commands 136
- entering terminal numbers 122
- equipment disposal 550
- ERASE command
 - format 206
 - LAN format 207
- erasing 507
- Esc key 36
- Ethernet 5
- European Union battery recycling statement 551
- exception log
 - displaying entries 466
 - modifying 467
 - warning 462, 463
 - when it fills up 488
 - when to display and modify 467
- EXIT command 208
- exiting command mode 133
- external commands 153
 - table of 156

F

- FCOPY command 209
- ferrite requirement 550
- file ADXCLLOG.DAT, master log 31
- file attributes
 - file extension 446
- file compression 83
- file compression/decompression utility
 - accessing through panels 405
 - error logging 408
 - from host processor 406
 - LAN considerations 407
 - using 405
 - with diskettes 406
- file decompression 83
- file distribution attribute
 - changing 468
 - displaying 468
 - modifying 468
- file extension
 - attribute 446
 - compound file 447
 - valid characters 138, 141
- file extension attribute
 - local 447
 - mirrored 447
- file mode
 - attribute 446, 447
 - functions for display/alter utility 360
 - selecting using display/alter utility 359
- file name on a FAT system 137
- file names on VFS systems 140
- file scan function for display/alter utility 360

- file selection
 - using display/alter utility 360
- file server
 - activating 462
 - deactivating 462
- file utilities
 - display/alter disk sectors 83
 - display/alter file sectors 83
 - distributing files 83
 - keyed file 83
 - using 82
- file version
 - image, of a mirrored file 447
 - prime, of a mirrored file 447
- file, audible alarm 86
 - See audible alarm file
- files
 - authorization 18
 - batch 157
 - compressing/decompressing 405
 - deleting securely 33
 - finding on a FAT system 137
 - finding on VFS systems 140
 - naming on a VFS system 141
 - naming on FAT systems 137
- filespec 137, 140
- FIND command
 - format 210
 - LAN format 212
- finding a file on a FAT system 137
- finding a file on VFS systems 140
- fixed disk
 - disk rebuild utility 409
 - disk rebuild utility over IP 409
- flat panel displays 552
- font/logo utility 54
 - logo conversion 54
- FOR command 213
- FORMAT command 214
- formatting
 - a hard disk drive 411
 - diskettes 13
- FSET command
 - format 218
- function keys line 35
- function keys, using 40
- functions
 - file mode for display/alter utility 360
 - file scan for display/alter utility 360
 - replacing current sector for display/alter utility 360
 - saving current sector for display/alter utility 360
 - scanning a file for display/alter utility 360, 361
 - selecting new relative sector for display/alter utility 360
 - store control, using 91
 - writing changes to a file for display/alter utility 360

G

- getting started 3
- global file name characters 142
- GOTO command 221

graphical extensions screen
switching to 122

H

hard disk drive, restoring data 411
home key 36

I

identification
operator 18
identifying drives 14
IF command 223
indicators
message pending 118
offline 118
wait 118
input sequence table
changing data in 85
insert key 36
installation and update aids 84
internal commands 153
IPLing using CD-ROM 521

J

Japan Electronics and Information Technology Industries Association statement 547
Japanese Electrical Appliance and Material Safety Law statement 549
Japanese power line harmonics compliance statement 549
Japanese VCCI Council Class A statement 547
Java
accessing Java console 110
configuring as a background application 293
configuring as a primary application 293
configuring as a secondary application 293
configuring your system 293
displaying Java application status 97
mouse support 301
starting applications 98
stopping applications 98
system configuration 75
terminal functions 96
terminal load definition 52, 59
understanding with the operating system 291
using a keyboard with applications 110
using application functions 97
using functions for all terminals 97
using graphics with the operating system 301
using on monochrome monitors 301
using with 4690 5
using with the operating system 291

K

keyboard and display, using 110
keyboard layouts 52, 59
keyboard, changing tone volume 120
keyed file, creating 83

- keys
 - editing 135
 - function 40
 - special 36
 - system request 42
 - system reset 44
- Korean communications statement 548

L

- LAN
 - commands 156
 - using commands on 153
- LAN attributes, compress/decompress utility 407
- LAN disk rebuild utility 541
- LAN Disk Rebuild Utility, determining controller IP information 412
- LAN functions 101
- LAN system
 - ASM on 371
 - considerations 371
 - distribution exception logs 452
- LAN, enabling and disabling 107
- LDAP server, secure logon through 30
- left arrow key 36
- line logging, command 31
- links, enabling and disabling 105
- LIST command 226
- load terminal configuration data 98
- load terminal storage
 - after applying software maintenance 98
 - for all terminals 98
 - for terminals on TCC network 98
- Loading Diagnostics for POS Systems over the Network 495
- loading store controller storage 100
- loading terminals 122
- local area network 9
- local area network (LAN)
 - alternate master controller 449
 - application-mirrored file on 447
 - at close 448
 - changing distributed file to local 469
 - changing file distribution attributes 468
 - changing local file to distributed on 469
 - combining controller roles 449
 - compound file on 447
 - connecting DOS to the 4690 MCF 478
 - connecting OS/2 PC to the LAN 473
 - disk rebuild utility 409
 - disk rebuild utility over IP 409
 - displaying file distribution attributes 468
 - displaying log entries 466
 - distribution exception log 465
 - file attributes for files on 446
 - file mode attribute on 447
 - image version of mirrored file on 447
 - local file extension on 447
 - logical file name 451
 - master controller 449
 - mirrored file extension on 447
 - modifying file distribution attributes 467, 468
 - node ID 450

- local area network (LAN) *(continued)*
 - node names 451
 - nodes on 450
 - per update 448
 - pipes and node names on 451
 - prime version of mirrored file on 447
 - processing exception log 466
 - recovering master controller 485
 - store controller backup 445
 - store controller roles 448
 - subordinate store controller 449
 - system problem analysis 485
 - system-mirrored file on 447
 - verifying nodes 490
 - when to display exception logs 467
 - when to distribute file 447
- local file
 - changing to distributed file 469
 - definition of 447
- local file extension defined 447
- local printing 511
- locally administered LAN address
 - changing prefix 80
 - disabling 80
 - setting 79
- log
 - distribution exception 465
- log file ADXCLLOG.DAT, master 31
- logging
 - ADXCSJ0L activity 359
 - command line activity 134
- logging, command line 31
- logical file names
 - definition of 451
- logical names
 - activating 75

M

- maintaining programs 382
 - build software maintenance control file utility 382
 - procedures for developing diskettes 382
- maintenance, preparing for and applying 371
- master log file ADXCLLOG.DAT 31
- master store controller
 - activating 423, 461
 - deactivating 423, 461
 - functions of 449
 - recovering after hard disk drive damaged 485
- matrix keyboard 117
- MBrowser
 - enabling 327
- MCF
 - See multiple controller feature
- measuring store controller performance data 89
- memory key
 - burning ISO image 262
- menu
 - authorization for 19
 - function keys line 35
 - message line 35

- menu (*continued*)
 - screen information 34
 - screens, using 33
 - status line 35
 - system main 33
- mercury-added statement 552
- message
 - displaying 117, 118
 - offline 118
 - pending system 118
- message display
 - clearing 119
- message line 35
- message pending light 119
- messages 35
- mirrored file
 - application 447
 - image 447
 - image version of mirrored file defined 447
 - master 447
 - mirrored file defined 447
 - system 447
 - type defined 447
- MKDIR (make directory) command
 - format 227
 - LAN format 228
- modes
 - application 16
 - operating 15
 - user 16
- modes for display/alter utility 359
- modifying
 - exception log 467
 - file distribution attributes 468
- module level, reporting 85
- MORE command
 - format 229
 - LAN format 230
- multiple controller feature
 - role in operating system 5
- multiple controller feature (MCF)
 - data distribution application (DDA) 445
 - defined 445
 - description of 445
 - where to distribute files 446
- multiple controller functions 101
- multiple controller functions menu 101
- multiple-application mode 16
- multiple-user mode 16

N

- names, files on a FAT system 137
- names, files on VFS systems 140
- naming files on a VFS system 141
- naming files on FAT systems 137
- naming subdirectories 139
- nested subdirectories 448
- NetBIOS security 20
- network file system data
 - 4690 NFS client 8

- network file system data *(continued)*
 - 4690 NFS server 8
 - controller NFS configuration 74
 - description 7
 - terminal NFS configuration, 4683, 4693, 4694 53
 - terminal NFS configuration, SurePOS 700 Series 59
- NFS
 - 4690 NFS client 8
 - 4690 NFS server 8
 - controller NFS configuration 74
 - description 7
 - terminal NFS configuration, 4683, 4693, 4694 53
 - terminal NFS configuration, SurePOS 700 Series 59
- NFS client 8
- NFS security 20
- NFS server 8
- node ID 450
- Node List Entry File 490
- node name
 - and pipes 451
- nodes
 - accessing 141
 - accessing files on other nodes 141
 - defined 450
 - naming 141
 - types of 141
 - verifying 490
- non-DDA system 72
- notices 543
 - battery recycling 551
 - cable ferrites 550
 - electronic emissions 545
 - electrostatic discharge (ESD) 549
 - end of life disposal 550
 - Toshiba 543

O

- offline light 118
- operating system
 - general information 5
- operating system panels 3
- operating system, DVD support for the 4690 10
- operator
 - authorized 18
 - background control 19
 - privileged window 19
 - system control 19
- operator identification
 - default 48
- Optical Drive Utility
 - accessing 395
 - advantages of using 394
 - alternative drives 404
 - backing up files 398
 - backing up files in use 403
 - command interface 404
 - comparing to streaming tape drive utilities 394
 - contents 398
 - disadvantages of using 394
 - entire hard disks 401

- Optical Drive Utility *(continued)*
 - erasing files 400
 - file data accessing considerations 397
 - formatting 394, 400
 - functions 398
 - identification 398
 - introduction 393
 - LAN system considerations 400
 - listing contents 399
 - logical drive name 396
 - recommended usage 401
 - restoring files 399
 - selected files 402
 - selection list file creation 400
 - status file definition 397
 - supplemental option considerations 401
 - using 84
 - using from an application program 396
 - using from the host 395
 - using the optical drive from command mode 396
 - write recovery 403
- optical media formatted with the UDF file system, using 11
- Optional RCP 372
- options
 - primary application 81
 - secondary application 82
- ORDER command 231
- overview 3

P

- page down key 36
- page up key 36
- panel, understanding 34
- password
 - changing 18
 - default 48
- path
 - screens 3
- PATH command
 - format 232
 - LAN format 233
- PAUSE command 234
- PCF (product control files) 367
- PCF (Product Control Files) 379
- perchlorate 552
- performance
 - measuring store controller data 89
 - measuring system 86
- performance monitor, running 89
- performance report utility 87
- physical sector mode
 - displaying and modifying a sector 361
 - scanning a sector 361
- pipe
 - and node names 451
- power management 93
- preventing store controller backup 101
- PRINT command 235
- printer character set, models 1 and 2 53
- printer character set, models 3 and 4 53

- printing, local 511
- privileged window operator 19
- problem analysis data 86
- processing exception log 466
- product control file (PCF)
 - building a software library 381
 - example of 367
 - installing a user product 381
 - module integrity value 383
 - naming the 379
 - user checksums 383
 - using other subdirectories 383
- product-level files 369
- PROMPT command 237
- prompt to change expired password 18
- prt sc key 36

Q

- QUERY command 239

R

- RAM disks
 - configuration 15
 - enabling 99
- RCP, optional no-IPL parameter 372
- reconfiguring default configuration 48
- record mode
 - per update method 448
- recovering master controller after hard disk drive is damaged 485
- redirecting input and output 143
- REM command 240
- RENAME command
 - format 241
 - LAN format 242
- replacing current sector function for display/alter utility 360
- reporting generic configuration data 84
- reporting legacy configuration data 84
- reporting module level 85
- requesting a system function 113
- requesting STC
 - displaying configuration data for 126
- restarting your system 44
- RESTORE command 243
- restoring data 411
- restrictions, operating system 473
- resuming store TCC control 101
- right arrow key 36
- RMDIR (remove directory) command 247
- roles
 - store controller 448
 - verifying 461
 - when to activate alternate store controller 463
- root directory
 - symbol for 138
- running other store applications 82
- running the primary application 81

S

- safety information xxvii
- saving current sector function for display/alter utility 360
- saving default configuration 48
- scanning a file using display/alter utility 360, 361
- screen titles 3
- sector
 - display/alter file data 83
 - using Physical Sector Mode
 - displaying and modifying a sector 361
 - general discussion 361
 - scanning a sector 362
 - selecting a disk drive 361
 - selecting a sector 361
 - using the Critical Sector Repair function 363
- security
 - system 17
- SECURITY command 248
- selecting
 - a disk drive using display/alter utility 359
 - a file using display/alter utility 360
 - file mode using display/alter utility 359
 - new relative section for display/alter utility 360
- selecting command mode 133
- selecting options 81
- set system date and time 104
- set terminal characteristics program, requesting 122
- setting system message audible alarm 86
- shell 179
- SHIFT command 249
- sign off, auto 50
- signing on 48
- SIGNOFF key 41
- single-application mode 16
- single-user mode 16
- SNA 3270 emulation
 - 3270 data stream 499
 - 4693 ANPOS keyboard 504, 516
 - ADXHSK0L.286 516
 - ADXHST0L.286 516
 - ADXZE30L.286 516
 - ANPOS keyboard 504, 516
 - API in the 4683 terminal 519
 - application program interface 514, 515
 - bel command 517
 - configuring a printer emulation 504
 - console interface 499
 - cursor movement keys 505
 - down arrow 505
 - left arrow 505
 - right arrow 505
 - up arrow 505
 - delete key 507
 - clear 507
 - delete 507
 - erase EOF 507
 - erase input 507
 - description 499
 - emulator-specific key 509
 - color mode 509
 - exit 509

- SNA 3270 emulation *(continued)*
 - emulator-specific key *(continued)*
 - numovride 509
 - OGL Toggle 509
 - equipment supported 500
 - 4683 Point-of-Sale Terminal 500
 - console devices 500
 - key location 500
 - keyboard difference 500
 - multiprotocol communication adapter 500
 - physical host link 500
 - Proprinter use 500
 - SNA communication 500
 - standard SDLC adapter 500
 - field cursor movement key 506
 - back tab 506
 - home 506
 - new line 506
 - tab 506
 - function key 504
 - gateway 500
 - graphic key 504
 - [character 504
 -] character 504
 - ¬ character 504
 - broken vertical bar 504
 - caret 504
 - tilde 504
 - hot key 499, 500, 517
 - keyboard 504
 - 3151 504
 - 3161 504
 - 3164 504
 - PS/2 504
 - keyboard template 501
 - 3161/3164 501
 - PS/2 501
 - knowledge required 499
 - message, operator guidance 511
 - BCOL 512
 - COMM nnn 512
 - FIELD FULL 512
 - FUNCTION UNAVAILABLE 512
 - INSERT 512
 - KEYSTROKE LOST 513
 - MONO 513
 - NO SESSION 513
 - NUMERIC 513
 - PRINTER BUSY 513
 - PRINTER ERROR 513
 - PRINTERnn 513
 - PRINTING 513
 - PROG nnn 513
 - PROTECTED 514
 - RANGE CHECK 514
 - SSCP 514
 - SYSTEM nnn 514
 - WAIT 514
 - migration 500
 - existing application 500
 - miscellaneous key 508

- SNA 3270 emulation *(continued)*
 - duplicate 508
 - field mark 508
 - insert 508
 - reset 508
 - system request 508
 - multiple controller configuration 499
 - on the 4683 terminal 500
 - on the store controller 499
 - on the terminal 516
 - print control key 508
 - dev cncl 508
 - identify 508
 - print 508
 - print file 501
 - ADX316KF.DAT 501
 - ADXATOKF.DAT 501
 - ADXPS2KF.DAT 501
 - printer interface 499
 - printing function 511
 - host application-initiated local printing 511
 - local printing 511
 - operator initiated local printing 511
 - program attention key 506
 - clear 506
 - cursor select 506
 - Enter 506
 - PA1 506
 - PA2 506
 - PA3 506
 - PF1 -> PF10 506
 - PF11 506
 - PF12 506
 - PF13 -> PF22 506
 - PF23 506
 - PF24 506
 - program access 506
 - program function 506
 - starting SNA 3270 emulation sessions 502
 - command 502
 - example of a command 502
 - gateway node name 502
 - link configuration name 502
 - logical printer name 502
 - stopping emulation sessions 503
 - exit key 503
 - link error 503
 - printer error 503
 - screen error 503
 - store controller 499
 - switched line 504
 - ADXHSNLL 504
 - application to install SNA driver 504
 - data terminal ready (DTR) 504
 - host connection procedure 504
 - originating host connection 504
 - SNA driver 504
 - system configuration 501
 - configuring a host link 501
 - modification 501
 - unique key function 504

- SNA 3270 emulation (*continued*)
 - unsupported function 505
 - USB ANPOS keyboard 504, 516
- SNA 3270 Emulation 499
- SNA 3270 emulator standard names 502
- software changes
 - applying to terminal 128
- software maintenance
 - applying 85
- software maintenance control file 85
- software maintenance utilities
 - apply software maintenance 365
 - build software maintenance control file utility 365
- SORT command
 - format 251
 - LAN format 254
- source diskette 148
- special characters
 - ? 142
 - * 142
 - > redirect output character 144
 - >> append file character 144
 - >>* 144
 - < input source character 144
 - ^ 143
 - | global file name character 145
- special feature 5
- special keys
 - delete 36
 - down arrow 36
 - end 36
 - enter 36
 - esc 36
 - for screen information 36
 - home 36
 - insert 36
 - left arrow 36
 - page down 36
 - page up 36
 - prt sc 36
 - right arrow 36
 - screen information keys 36
 - tab 36
 - up arrow 36
- specifying national language
 - on supplementals 525
- spool file
 - when it fills up 488
- standard names for 3270 emulator 502
- start terminal application
 - application name 97
 - terminal number 97
- starting 3
- status information 35
- status line 35
- status, store controller 461
- stop key 41
- stop terminal application 97
- storage retention function 120
 - disabling 121
 - enabling 120

- storage, terminal, dumping 121
- store control functions
 - communications 105
 - LAN functions 101
 - multiple controller functions 101
 - store controller functions 99
 - system functions 103
 - TCC functions 100
 - terminal functions 96
 - using 91
- store controller
 - backup 9
 - combining roles on 449
 - displaying 461
 - displaying status of 461
 - name formation 141
 - of store controllers 461
 - status 461
 - verifying roles 461
 - when to activate alternate roles 463
- store controller backup
 - allowing 493
 - functions of 493
 - preventing 494
 - resuming control 494
 - using LAN with 445
- store controller backup recovery 494
- store controller configuration 71
- store controller functions 99
- store controller operations 81
- store controller roles
 - alternate file server 449
 - alternate master 449
 - file server 449
 - master 449
 - selecting 458
 - subordinate 449
- store controller status 99
- store system configuration 8
- streaming tape 385
- streaming tape drive
 - using 83
- subdirectories, nested 448
- subdirectory
 - creating 138
 - maximum depth on VFS system 141
 - maximum length on VFS system 141
 - naming 139
- subordinate store controller 449
- supplemental option using the CD-ROM
 - changing national language 526, 527
 - specifying national language 525
- support for the 4690 operating system, DVD 10
- swapping
 - terminal video screens 121
- SYS command 541
- system
 - commands 147
 - security 17
- system audible alarm message file 86
- system configuration 8, 75

- system control operator 19
- system date and time, setting 104
- system display default 123
- system file size
 - activating 75
- system functions
 - at the terminal 109
 - requesting 113
 - requesting on alphanumeric 114
 - requesting using touch screen 115
- system functions menu 103
- system functions table 109
- system keys 42
- system message display levels 104
- system modes
 - of operation 15
- system panel paths, diagram 3
- system performance, measuring 86
- system request keys
 - moving to next 42
 - moving to preceding 42
- system reset keys 44
- system security
 - console ID security 19
 - IP security 30
 - IPsec 30
 - logon through LDAP server 30
 - NetBIOS security 20
 - NFS security 20
 - Secure Shell 30
 - SSH 30
- System Services Process Utility 423
- system-mirrored file 447
 - image version 447
 - prime version 447
 - where file resides 447

T

- tab key 36
- Taiwanese battery recycling statement 551
- tape streamer 385
- TCC functions
 - selecting TCC functions 100
 - store controller backup 455
- TCC functions menu 100
- TCC Network
 - controlling 9
 - resuming control of 101
 - store controller backup 493
 - store controller discovery 493
- terminal
 - applying software changes 128
 - changing configuration 128
 - video screens 121
- terminal application, starting 97
- terminal application, stopping 97
- terminal configuration 52, 58
- terminal configuration data, loading 98
- terminal device characteristics 58
- terminal device groups 52

- terminal functions 96
- terminal functions menu 96
- terminal hard disk
 - formatting 130
 - on Mod1 terminals 131
 - on SurePOS 700 Series 130
- terminal load 122
- terminal load definition 52
- terminal load definition, SurePOS 4694 and 700 Series 59
- terminal NFS configuration, 4683, 4693, 4694 53
- terminal NFS configuration, SurePOS 700 Series 59
- terminal number
 - entering 122
 - entering, 4694 123
 - entering, Mod1 terminal 124
 - entering, Mod2 terminal 125
 - entering, SurePOS 700 Series 123
- terminal RAM disk preload utilities 419
 - using 419
- terminal screen saver 117
- terminal storage retention
 - disabling
 - from the store controller 97
 - on a TCC network 97
 - enabling 96
- terminal storage, dumping 121
- terminal storage, loading 98
- terminals, Mod1, Mod2 52
- terminals, SurePOS 4694 700 and TCxWave 6140 Series 58
- titles, panels 3
- tone, changing volume 120
- touch screen, using 111
- trademarks 554
- TREE command
 - format 255
 - LAN format 256
- TYPE command
 - format 257
 - LAN format 257

U

- UDF file system, using optical media formatted with the 11
- uninterruptible power supply 116
- unlock id
 - console ID security 19
- up arrow key 36
- updating operating system 85
- USB
 - configuring your system 324
- USB ANPOS keyboard 117
- USB modifiable layout keyboard 117
- USB Retail POS keyboard 117
- user modes
 - multiple 16
- user modifications
 - applying to programs 382
 - maintaining programs 382
- using 8
- using command mode 91
- using file utilities 82

- using function keys 40
- using LAN system 461
- using optical media formatted with the UDF file system 11
- using Set Terminal Characteristics program 122
- using store control functions 91
- using the CD-ROM 521
- using the SSP utility 423
- using USB
 - USB devices with 4690 OS 324
- utilities
 - apply software maintenance 365
 - build software maintenance control file utility 365, 379, 382
 - compress/decompress utility 405
 - disk rebuild utility 409
 - disk rebuild utility over IP 409
 - disk space, compress/decompress utility 408
 - display/alter 359
 - font/logo 54
 - logo conversion 54
 - LAN attributes, compress/decompress utility 407
 - performance report 87
 - streaming tape drive utility 385
 - terminal RAM disk preload utilities 419

V

- value, default 34
- VER command 259
- VERIFY command 260
- verifying nodes 490
- verifying store controller roles 461
- video screens
 - swapping 121
- virtual file system
 - description 8
 - using in 4690 OS 8
- VOL command
 - format 261
 - LAN format 261

W

- wait indicator 118
- wait light 118
- when to display and modify exception logs 467
- windows
 - figure 16
 - starting 17
 - stopping 17
- working in command mode 133
- write verify function for hard disk drive, disabling 14
- write-protect
 - covering 13
 - tab 13
- write-protecting diskettes 12
- writing changes to a file function for display/alter utility 360

X

- X.25 5

- XE editor
 - editing commands 284
 - entering commands 282
 - messages 287
 - SYSTEM MAIN MENU 282
- XE Editor panel 285



Product Number: 5639-P70

G362-0572-02

