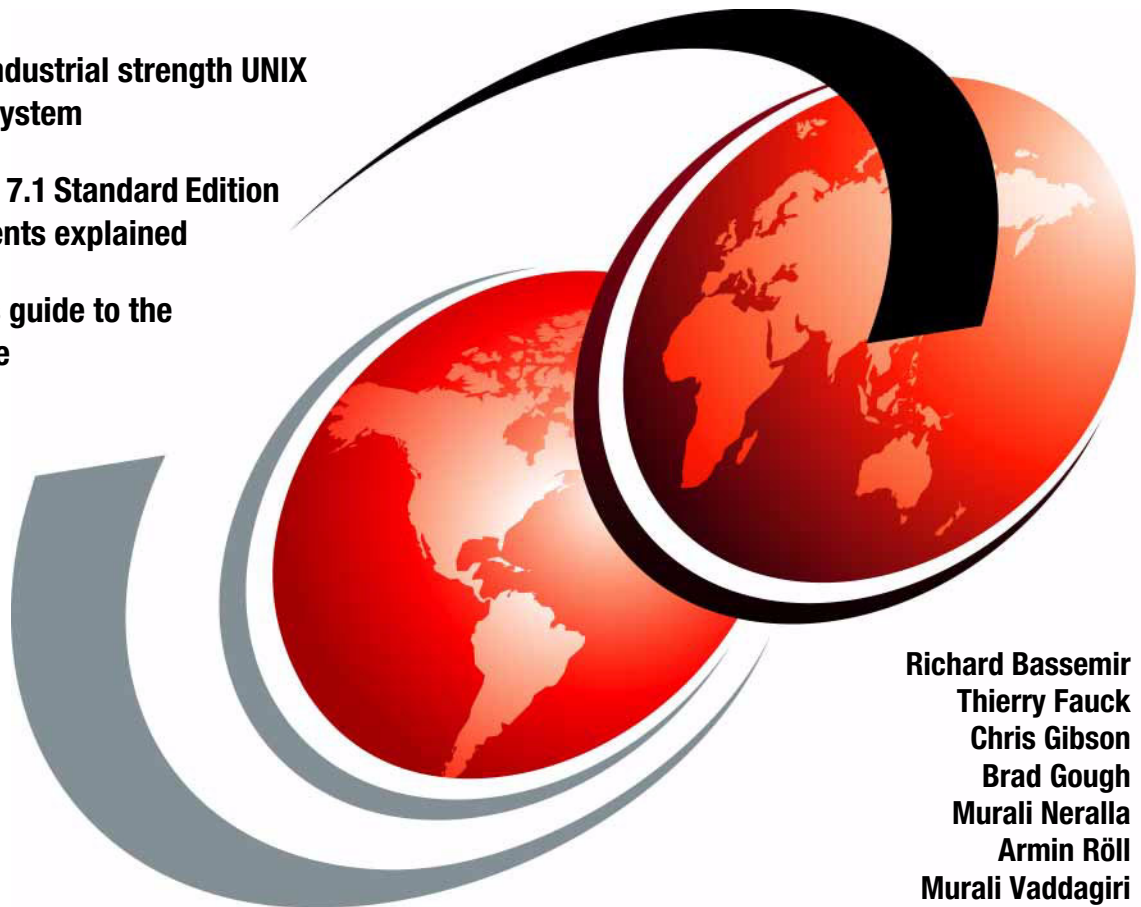


# IBM AIX Version 7.1 Differences Guide

**AIX - The industrial strength UNIX  
operating system**

**AIX Version 7.1 Standard Edition  
enhancements explained**

**An expert's guide to the  
new release**



**Richard Bassemir  
Thierry Fauck  
Chris Gibson  
Brad Gough  
Murali Neralla  
Armin Röhl  
Murali Vaddagiri**





International Technical Support Organization

**IBM AIX Version 7.1 Differences Guide**

December 2010

**Note:** Before using this information and the product it supports, read the information in “Notices” on page xxiii.

**First Edition (December 2010)**

This edition applies to AIX Version 7.1 Standard Edition, program number 5765-G98.

This document created or updated on September 17, 2010.

© Copyright International Business Machines Corporation 2010. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

# Contents

<b>Figures</b> .....	ix
<b>Tables</b> .....	xi
<b>Examples</b> .....	xiii
<b>Notices</b> .....	xxiii
Trademarks .....	xxiv
<b>Preface</b> .....	xxv
The team who wrote this book .....	xxv
Now you can become a published author, too! .....	xxvii
Comments welcome .....	xxviii
Stay connected to IBM Redbooks .....	xxviii
<b>Chapter 1. Application development and debugging</b> .....	1
1.1 AIX binary compatibility .....	2
1.2 Improved performance using 1 TB segments .....	2
1.3 Kernel sockets application programming interface .....	5
1.4 Unix08 Standard Conformance .....	6
1.4.1 stat structure changes .....	8
1.4.2 open system call changes .....	9
1.4.3 utimes system call changes .....	9
1.4.4 futimens and utimensat system calls .....	10
1.4.5 fexecve system call .....	10
1.5 AIX assembler enhancements .....	10
1.5.1 Thread Local Storage (TLS) support .....	10
1.5.2 TOCREL support .....	11
1.6 Malloc debug fill .....	11
1.7 Core dump enhancements .....	12
1.8 Disabled read write locks .....	13
1.9 DBX enhancements .....	16
1.9.1 Dump memory areas in pointer format .....	16
1.9.2 New dbx environment variable print_mangled .....	17
1.9.3 DBX malloc subcommand enhancements .....	18
<b>Chapter 2. File systems and storage</b> .....	21
2.1 LVM enhancements .....	22
2.1.1 LVM enhanced support for solid-state disks .....	22

2.2 Hot Files Detection in JFS2 . . . . .	27
<b>Chapter 3. Workload Partitions and resource management . . . . .</b>	<b>35</b>
3.1 Trusted Kernel Extension loading and configuration for WPARs . . . . .	36
3.2 WPAR list of features . . . . .	41
3.3 Versioned Workload Partitions (WPAR) . . . . .	41
3.3.1 Benefits of that feature . . . . .	42
3.3.2 Current requirements and restrictions . . . . .	42
3.3.3 Creation of a basic Versioned WPAR AIX 5.2 . . . . .	43
3.3.4 Creation of an AIX Version 5.2 rootvg WPAR . . . . .	51
3.3.5 Content of vwpar.52 package . . . . .	55
3.3.6 SMIT INTERFACE . . . . .	57
3.4 Devices support in WPAR . . . . .	57
3.4.1 Global device listing used as example . . . . .	58
3.4.2 Device command listing in a AIX 7.1 WPAR . . . . .	58
3.4.3 Dynamically adding a fiber channel adapter to a system WPAR . . . . .	61
3.4.4 Change in config file related to that device addition . . . . .	63
3.4.5 Isdev output from Global . . . . .	63
3.4.6 Removing of fiber channel adapter from Global . . . . .	63
3.4.7 Reboot of LPAR keeps fiber channel allocation . . . . .	63
3.4.8 Disk attached to fiber channel adapter . . . . .	66
3.4.9 Startwpar error if adapter busy on Global . . . . .	67
3.4.10 Startwpar with a fiber channel adapter defined . . . . .	68
3.4.11 Disk commands in the WPAR . . . . .	71
3.4.12 Access to the fiber disks from the Global . . . . .	72
3.4.13 Support of fiber channel devices in mkwpar command . . . . .	73
3.4.14 Config file created for the rootvg system WPAR . . . . .	80
3.4.15 Removing of a fiber disk in a running system WPAR . . . . .	81
3.4.16 Mobility restrictions . . . . .	81
3.4.17 Debugging log . . . . .	82
3.5 WPAR RAS enhancements . . . . .	83
3.5.1 Error logging mechanism aspect . . . . .	83
3.5.2 Goal for these messages . . . . .	84
3.5.3 Syntax of the messages . . . . .	84
3.6 WPAR Migration to AIX Version 7.1 . . . . .	86
<b>Chapter 4. Continuous availability . . . . .</b>	<b>101</b>
4.1 Firmware-assisted dump . . . . .	102
4.1.1 Default installation configuration . . . . .	102
4.1.2 Full memory dump options . . . . .	103
4.1.3 Changing the dump type on AIX V7.1 . . . . .	105
4.1.4 Firmware-assisted dump on POWER5 and earlier hardware . . . . .	108
4.1.5 Firmware-assisted dump support for non boot iSCSI device . . . . .	109

4.2	User keys enhancements	110
4.3	Cluster Data Aggregation Tool	111
4.4	Cluster Aware AIX	117
4.4.1	Cluster configuration	118
4.4.2	Cluster system architecture flow	130
4.4.3	Cluster event management	131
4.4.4	Cluster socket programming	132
4.4.5	Cluster storage communication configuration	135
4.5	SCTP component trace and RTEC adoption	137
4.6	Cluster aware perfstat library interfaces	139
	<b>Chapter 5. System management</b>	<b>145</b>
5.1	CPU interrupt disablement	146
5.2	Distributed System Management	147
5.2.1	dpasswd command	148
5.2.2	dkeyexch command	149
5.2.3	dgetmacs command	149
5.2.4	dconsole command	150
5.2.5	dcp command	152
5.2.6	dsh command	153
5.2.7	Using DSM and NIM	154
5.3	AIX System Configuration Structure Expansion	165
5.3.1	kgetsystemcfg kernel service	165
5.3.2	getsystemcfg Subroutine	165
5.4	AIX Runtime Expert	166
5.4.1	AIX Runtime Expert overview	167
5.4.2	Changing mkuser defaults example	171
5.4.3	Schedo and ioo profile merging example	174
5.4.4	Latest enhancement	176
5.5	Removal of CSM	176
5.6	Removal of IBM Text-to-Speech	179
5.7	AIX device renaming	180
5.8	1024 Hardware thread enablement	181
5.9	Kernel Memory Pinning	185
5.10	ksh93 enhancements	188
5.11	DWARF	188
5.12	AIX Event Infrastructure extension and RAS	189
5.12.1	Some advantages of AIX Event Infrastructure	189
5.12.2	Configuring the AIX Event Infrastructure	190
5.12.3	Use of monitoring sample	191
	<b>Chapter 6. Performance management</b>	<b>197</b>
6.1	Support for Active Memory Expansion	198

6.1.1	amepat command	198
6.1.2	Enhanced AIX performance monitoring tools for AME	221
6.2	Hot Files Detection and filemon	227
6.3	Memory affinity API enhancements	241
6.3.1	API enhancements	242
6.3.2	pthread attribute API	243
6.4	iostat command enhancement	244
<b>Chapter 7. Networking</b>		247
7.1	Enhancement to IEEE 802.3ad link aggregation	248
7.1.1	EtherChannel and Link Aggregation in AIX	248
7.1.2	IEEE 802.3ad Link Aggregation functionality	248
7.1.3	AIX v7.1 enhancement to IEEE 802.3ad Link Aggregation	249
7.2	Removal of BIND 8 application code	258
7.3	Network Time Protocol version 4	259
7.4	Reliable Datagram Sockets (RDS) v3 for RDMA support	263
<b>Chapter 8. Security, authentication, and authorization</b>		265
8.1	Domain Role Based Access Control	266
8.1.1	The traditional approach to AIX security	266
8.1.2	Enhanced and Legacy Role Based Access Control	267
8.1.3	Domain Role Based Access Control	269
8.1.4	Domain RBAC command structure	272
8.1.5	LDAP support in Domain RBAC	283
8.1.6	Scenarios	284
8.2	Auditing enhancements	321
8.2.1	Auditing with full pathnames	322
8.2.2	Auditing support for Trusted Execution	323
8.2.3	Recycling Audit trail files	324
8.2.4	Role based auditing	326
8.2.5	Object auditing for NFS mounted files	328
8.3	Propolice or Stack Smashing Protection	328
8.4	Security enhancements	329
8.4.1	ODM directory permissions	329
8.4.2	Configurable NGROUPS_MAX	330
8.4.3	Kerberos client kadmind_timeout option	330
8.4.4	KRB5A load module removal	331
8.4.5	Chpasswd support for LDAP	331
8.4.6	AIX password policy enhancements	332
8.5	Remote Statistic Interface (Rsi) client firewall support	336
8.6	AIX LDAP authentication enhancements	337
8.6.1	Case sensitive LDAP user names	337
8.6.2	LDAP alias support	337



8.6.3 LDAP caching enhancement . . . . .	337
8.6.4 Other LDAP enhancements . . . . .	338
8.7 RealSecure Server Sensor . . . . .	338
<b>Chapter 9. Installation, backup, and recovery . . . . .</b>	<b>339</b>
9.1 AIX V7.1 minimum system requirements . . . . .	340
9.1.1 Required hardware . . . . .	340
9.2 Loopback device support in NIM . . . . .	346
9.2.1 Support for loopback devices during the creation of lpp_source and spot resources . . . . .	346
9.2.2 Loopmount command . . . . .	347
9.3 Bootlist command path enhancement . . . . .	348
9.3.1 Bootlist device pathid specification . . . . .	348
9.3.2 Common new flag for pathid configuration commands . . . . .	349
9.4 NIM thin server 2.0 . . . . .	350
9.4.1 Functional enhancements . . . . .	351
9.4.2 Limitations . . . . .	352
9.4.3 NIM commands option for NFS setting on NIM master . . . . .	353
9.4.4 Simple Kerberos server setting on NIM master NFS server . . . . .	354
9.4.5 IPv6 Boot Firmware syntax . . . . .	354
9.4.6 /etc/export file syntax . . . . .	354
9.4.7 AIX Problem Determination Tools . . . . .	354
9.5 Activation Engine for VDI customization . . . . .	355
9.5.1 Step by step usage . . . . .	356
9.6 SUMA and Electronic Customer Care integration . . . . .	361
9.6.1 SUMA installation on AIX 7 . . . . .	362
9.6.2 AIX 7 SUMA functional and configuration differences . . . . .	363
9.7 Network Time Protocol version 4 . . . . .	367
<b>Chapter 10. National language support . . . . .</b>	<b>373</b>
10.1 Unicode 5.2 support . . . . .	374
10.2 Code set alias name support for iconv converters . . . . .	374
10.3 NEC selected characters support in IBM-eucJP . . . . .	375
<b>Chapter 11. Hardware and graphics support . . . . .</b>	<b>377</b>
11.1 X11 Font Updates . . . . .	378
11.2 AIX V7.1 storage device support . . . . .	387
11.3 Hardware support . . . . .	392
11.3.1 Hardware support . . . . .	392
<b>Abbreviations and acronyms . . . . .</b>	<b>395</b>
<b>Related publications . . . . .</b>	<b>401</b>
IBM Redbooks . . . . .	401

Other publications .....	402
Online resources .....	402
How to get Redbooks .....	402
Help from IBM .....	402
<b>Index</b> .....	<b>405</b>

# Figures

8-1 Illustration of Role based auditing . . . . .	326
11-1 The IBM System Storage Interoperation Centre . . . . .	388
11-2 The IBM System Storage Interoperation Centre - search example . . .	390
11-3 The IBM System Storage Interoperation Centre - the export to .xls option. 391	



# Tables

1-1	Kernel service socket API . . . . .	5
1-2	short list of new library functions and system calls . . . . .	7
1-3	new library functions to test character in a locale . . . . .	8
1-4	Malloc abc fill pattern . . . . .	12
1-5	Kernel and kernel extension services . . . . .	14
3-1	migwpar flags and options . . . . .	87
4-1	Full memory dump options available with the sysdumpdev -f command	104
4-2	Number of storage keys supported . . . . .	110
4-3	Cluster commands . . . . .	118
4-4	Cluster events . . . . .	131
5-1	DSM components . . . . .	147
5-2	Removed CSM fileset packages . . . . .	177
6-1	System Configuration details reported by amepat . . . . .	203
6-2	System resource statistics reported by amepat . . . . .	205
6-3	AME statistics reported using amepat . . . . .	205
6-4	AME modeled statistics . . . . .	206
6-5	Optional command line flags to amepat . . . . .	208
6-6	AIX performance tool enhancements for AME . . . . .	221
6-7	topas -C memory mode values for an LPAR . . . . .	224
6-8	Hot Files Report Description . . . . .	227
6-9	Hot Logical Volumes Report Description . . . . .	228
6-10	Hot Physical Volumes Report Description . . . . .	229
6-11	filemon -O hot flag options . . . . .	230
7-1	The LACP interval duration . . . . .	250
7-2	NTP binaries directory mapping on AIX . . . . .	261
8-1	Domain RBAC enhancements to existing commands . . . . .	277
8-2	audit event list . . . . .	323
8-3	Example scenario for Rule 1s . . . . .	334
8-4	Example scenario for Rule 2 . . . . .	335
8-5	The caseExactAccountName values . . . . .	337
8-6	TO_BE_CACHED valid attribute values . . . . .	338
9-1	Disk space requirements for AIX V7.1 . . . . .	341
9-2	AIX edition and features . . . . .	343
9-3	NFS available options . . . . .	351
9-4	New or modified NIM objects . . . . .	352
9-5	NTP binaries directory mapping on AIX . . . . .	369
10-1	Locales and code sets supporting NEC selected character . . . . .	375
11-1	TrueType Fonts original and new XLFD family and file names . . . . .	378

11-2 Updated TrueType Font file names, filesset packages, glyph list and CDE usage .....	379
11-3 Additional East Asian XLFD family and file names .....	380
11-4 East Asian subset font file names, filesset packages and CDE usage. .	380
11-5 Middle Eastern glyph subset XLFD family and file names.....	381
11-6 Middle Eastern font file names, filesset packages and CDE usage . . . .	381
11-7 Additional Hong Kong XLFD family and file names .....	381
11-8 Additional Hong Kong file names, filesset packages, and CDE usage. .	382
11-9 Removed WGL file names and filesset packages .....	382

# Examples

1-1	The shm_1tb_shared tunable . . . . .	3
1-2	The shm_1tb_unshared tunable . . . . .	3
1-3	The esid_allocator tunable . . . . .	4
1-4	A sample application call sequence . . . . .	8
1-5	proc_getattr(), proc_setattr() APIs. . . . .	13
1-6	Display 32-bit pointers. . . . .	16
1-7	Display 64-bit pointers. . . . .	17
1-8	print_mangled dbx environment variable . . . . .	17
1-9	unset print_mangled dbx environment variable . . . . .	18
1-10	malloc freespace dbx subcommand output. . . . .	18
1-11	address argument to malloc sub command . . . . .	19
2-1	Output from the lsdev command showing SSD disk . . . . .	23
2-2	Creating an SSD restricted VG . . . . .	24
2-3	The volume group PV RESTRICTION is set to SSD . . . . .	24
2-4	Changing the PV type restriction on a volume group . . . . .	25
2-5	Attempting to create an SSD restricted VG with a non-SSD disk . . . . .	25
2-6	Attempting to add a non-SSD disk to an SSD restricted volume group . . . . .	26
2-7	Creating a volume with both non-SSD and SSD disk . . . . .	27
2-8	/usr/include/sys/hfd.h header file. . . . .	28
2-9	Example HFD_QRY code . . . . .	33
3-1	Creation of a simple WPAR . . . . .	37
3-2	Trying to load a kernel extension in a simple WPAR . . . . .	37
3-3	Successful loading of kernel extension. . . . .	38
3-4	Parameter -X of lswpar command. . . . .	39
3-5	Loading kernel extension . . . . .	39
3-6	Changing type of kernel extension and impact to Global. . . . .	40
3-7	Missing vwpar packages installation message . . . . .	44
3-8	Simple Versioned WPAR creation . . . . .	45
3-9	.Lswpar queries. . . . .	47
3-10	Multiple lswpar queries over Versioned WPAR. . . . .	47
3-11	startwpar of a Versioned WPAR . . . . .	48
3-12	Commands within a Versioned WPAR . . . . .	49
3-13	Execution of a AIX 7.1 binary command in a Versioned WPAR. . . . .	50
3-14	rootvg Versioned WPAR creation . . . . .	51
3-15	Rootvg Versioned WPAR file system layout. . . . .	53
3-16	Startwpar of a rootvg Versioned WPAR . . . . .	53
3-17	Devices and file systems in a rootvg Versioned WPAR. . . . .	54
3-18	vwpar.52 lpp content. . . . .	55

3-19	Physical adapter available from Global. . . . .	58
3-20	Simple WPAR file system layout. . . . .	58
3-21	Start of the WPAR. . . . .	59
3-22	Iscfg display in a simple system WPAR . . . . .	60
3-23	Packages related to wio installed in WPAR . . . . .	61
3-24	Virtual device support abstraction layer . . . . .	61
3-25	Dynamically adding FC adapter to a running WPAR. . . . .	62
3-26	/etc/wpars/wpar1.cf entry update for device fcs0 . . . . .	63
3-27	Isdev -x output. . . . .	63
3-28	rmdev failure for a busy device . . . . .	63
3-29	Fiber channel adapter queries from Global after reboot . . . . .	64
3-30	Removal of the fiber channel adapter from the Global . . . . .	64
3-31	Removal of missing device allows WPAR start. . . . .	65
3-32	Devices commands issued on the Global . . . . .	66
3-33	WPAR1 start error message if disk busy . . . . .	67
3-34	Startwpar whit fiber channel device available for WPAR use. . . . .	68
3-35	Devices within the WPAR . . . . .	69
3-36	Disk no more visible from Global . . . . .	70
3-37	Creation of volume in a WPAR . . . . .	71
3-38	Stopping WPAR releases fiber channel allocation . . . . .	72
3-39	mkwpar -D option syntax. . . . .	73
3-40	Creation of a rootvg system WPAR. . . . .	74
3-41	Mkwpar failure if end-point device busy . . . . .	74
3-42	Listing of the rootvg system WPAR file systems from the Global . . . . .	75
3-43	Allocated devices to a WPAR not available to Global. . . . .	75
3-44	Startwpar of a rootvg WPAR on fiber channel disk . . . . .	76
3-45	File systems of the rootvg WPAR seen from inside the WPAR . . . . .	77
3-46	Isdev within a rootvg system WPAR . . . . .	77
3-47	Exclusive device allocation message . . . . .	78
3-48	New rootvg system WPAR creation . . . . .	78
3-49	Global vue of exported disks to rootvg WPARs . . . . .	80
3-50	/etc/wpars/wpar3.cf listing . . . . .	80
3-51	Rootvg disk of a rootvg WPAR can't be removed if WPAR active . . . . .	81
3-52	Listing of the environment flags of the system WPAR . . . . .	81
3-53	Checkpoint WPAR is not allowed with rootvg WPAR . . . . .	82
3-54	/var/adm/wpars/event.log example . . . . .	82
3-55	mkwpar user command error message. . . . .	84
3-56	Same command, other message number. . . . .	84
3-57	Same component, other command. . . . .	85
3-58	Multiple messages for a command . . . . .	85
3-59	WPAR mobility command error messages . . . . .	85
3-60	Few other informative messages . . . . .	85
3-61	Migrating a shared system WPAR to AIX V7.1 . . . . .	88



3-62	Migrating a detached WPAR to AIX V7.1	88
3-63	Migrating all shared WPARs to AIX V7.1	88
3-64	Migrating all detached WPARs to AIX V7.1	88
3-65	Confirming the WPAR state is active	89
3-66	Verifying global and WPAR AIX instances prior to migration	90
3-67	Clean shutdown of the WPAR	92
3-68	AIX version 7.1 after migration	93
3-69	WPAR not started after global instance migration to AIX V7.1	93
3-70	Starting the WPAR after global instance migration	94
3-71	Global instance migrated to version 7, WPAR still running version 6	94
3-72	WPAR migration to AIX V7.1 with migwpar	95
3-73	Verifying WPAR started successfully after migration	96
3-74	Migrating a detached WPAR to AIX V7.1	97
4-1	The sysdumpdev -l output in AIX V6.1	102
4-2	The sysdumpdev -l output in AIX V7.1	103
4-3	Setting the full memory dump option with the sysdumpdev -f command	104
4-4	Changing to the traditional dump on AIX V7.1	105
4-5	Reinstating firmware-assisted dump with the sysdumpdev -t command	106
4-6	Partition reboot to activate firmware-assisted dump	107
4-7	The sysdumpdev -l command after partition reboot	108
4-8	Attempting to enable firmware-assisted dump on a POWER5	108
4-9	Configuring storage keys	110
4-10	Configuring Cluster Data Aggregation Tool	112
4-11	Before creating cluster	118
4-12	Creating the cluster	119
4-13	Verifying the cluster from another node	121
4-14	Displaying basic cluster configuration	123
4-15	Displaying cluster storage interfaces	123
4-16	Displaying cluster network statistics	124
4-17	Displaying cluster configuration interfaces	125
4-18	Deletion of node from cluster	127
4-19	Deletion of cluster disk from cluster	128
4-20	Addition of a disk to the cluster	129
4-21	Removal of cluster	129
4-22	Usage of <code>c1cmd</code> command	130
4-23	Cluster messaging example	132
4-24	Cluster storage communication configuration	137
5-1	Disabling interrupts	146
5-2	Creating a password file	148
5-3	Key exchange between NIM and an HMC	149
5-4	Using the dsh method	150
5-5	Starting dconsole in text mode with logging	151
5-6	Contents of the node info file	152

5-7	Example use of the dcp command	152
5-8	Checking dsh environment variables	153
5-9	Sample node list	153
5-10	Example using the dsh command	153
5-11	Setting up the environment variables	153
5-12	Sample node list	154
5-13	Collecting the system type, model and serial number from HMC	155
5-14	Collecting the LPAR id information from the HMC	155
5-15	Configuring ssh access to the HMC from the NIM master	155
5-16	Entry in the nodeinfo file for the new host, Power System and HMC	156
5-17	Defining the HMC and CEC NIM objects	156
5-18	Obtaining the MAC address for the LPARs virtual network adapter	157
5-19	Defining new NIM machine object with HMC, LPAR and CEC options	157
5-20	Displaying LPAR state and enabling NIM bos_inst on the NIM client	158
5-21	Monitoring the NIM installation with the dconsole command	159
5-22	Monitoring the NIM client installation status from the NIM master	160
5-23	DSM network boot log file output	161
5-24	DSM dconsole log file	162
5-25	lpar_netboot log file	163
5-26	Verifying AIX installed successfully from a dconsole session	164
5-27	kgetsystemcfg man page header	165
5-28	getsystemcfg libc sub-routine man page header	166
5-29	AIX 7.1 AIX Runtime Expert filesets	166
5-30	AIX Runtime Expert profile template listing	167
5-31	AIX Runtime Expert catalog listing	168
5-32	Catalog file schedoParam.xml	169
5-33	Profiles file referenceing catalogs	169
5-34	Default mkuser profile	172
5-35	Default user creation	172
5-36	Building a new profile based on the	172
5-37	XLM output of new profile and running system differences	173
5-38	Text output of the new profile and the running system differences	173
5-39	Applying the new profile	173
5-40	Creating a new user with the new defaults	174
5-41	Creating a new merged profile	175
5-42	Renaming device	180
5-43	Power 795 system configuration	181
5-44	Configuring kernel memory locking	187
5-45	The lspp -f bos.ahafs package listing	190
5-46	Mounting the file system	191
5-47	The syntax output from the mon_1event C program	192
5-48	Creating a monitoring the event	193
5-49	Using dd command to increase /tmp filesystem utilization	193

5-50	Increase of /tmp filesystem utilization to 55%	194
5-51	THRESH_HI threshold is reached or exceeded	194
6-1	Running amepat without privileged access	199
6-2	CPU and memory utilization snapshot from amepat	200
6-3	List possible AME configurations for an LPAR with amepat	201
6-4	Running amepat within a WPAR	207
6-5	Displaying the amepat monitoring report	211
6-6	Monitoring the workload on a system with amepat for 10 minutes	212
6-7	Capping AME CPU usage to 30%	213
6-8	AME modeling memory gain of 1000MB	215
6-9	Modeling a minimum uncompressed pool size of 2000MB	216
6-10	Starting amepat in recording mode	217
6-11	Generating an amepat report using an existing recording file	217
6-12	Modeled expansion factor report from a recorded file	218
6-13	AME monitoring report from a recorded file	219
6-14	Disable workload planning and only monitor system utilization	220
6-15	Using vmstat to display AME statistics	221
6-16	Using lparstat to display AME statistics	222
6-17	Using lparstat to view AME configuration details	223
6-18	Additional topas sub-section for AME	223
6-19	topas CEC view with AME enabled LPARs	224
6-20	AME statistics displayed using the svmon command	225
6-21	Viewing AME summary usage information with svmon	226
6-22	filemon -O hot is not supported in real-time mode	231
6-23	Generating filemon hot file report in automated offline mode	231
6-24	Information and summary sections of the hot file report	232
6-25	Hot Files Report	232
6-26	Hot Logical Volume Report	233
6-27	Hot Physical Volume Report	233
6-28	Hot Files sorted by capacity accessed	233
6-29	Hot Logical Volumes	234
6-30	Hot Physical Volumes	234
6-31	Hot Files sorted by IOP	235
6-32	Hot Logical Volumes sorted by IOP	235
6-33	Hot Physical Volumes sorted by IOP	236
6-34	Hot Files sorted by #ROP	236
6-35	Hot Logical Volumes sorted by #ROP	236
6-36	Hot Physical Volumes sorted by #ROP	237
6-37	Hot Files sorted by #WOP	237
6-38	Hot Logical Volumes sorted by #WOP	238
6-39	Hot Physical Volumes sorted by #WOP	238
6-40	Hot Files sorted by RTIME	238
6-41	Hot Logical Volumes sorted by RTIME	239

6-42	Hot Physical Volumes sorted by RTIME . . . . .	239
6-43	Hot Files sorted by WTIME . . . . .	239
6-44	Hot Logical Volumes sorted by WTIME . . . . .	240
6-45	Hot Physical Volumes sorted by WTIME . . . . .	241
6-46	Example of the new iostat output . . . . .	244
6-47	Using the raso command to turn on statistic collection . . . . .	245
6-48	Using raso -L command to see if statistic collection is on. . . . .	245
7-1	The lsdev -Cc adapter command . . . . .	250
7-2	Displaying the logical Ethernet devices in the ent6 pseudo Ethernet device 251	
7-3	The lsslot and lscfg commands display the physical Ethernet adapters .	251
7-4	The entstat -d ent6 output - Link Aggregation operational . . . . .	252
7-5	The entstat -d ent6 output - Link Aggregation non-operational . . . . .	254
7-6	The entstat -d ent6 output - Link Aggregation recovered and operational	257
8-1	Using the lsattr command to display the enhanced_RBAC status . . . . .	270
8-2	Using the chdev command to enable the enhanced_RBAC attribute . . . . .	270
8-3	Using the ps command to identify the process editing /tmp/myfile . . . . .	272
8-4	Using the mkdom command to create the domain Network with a Domain ID of 22. . . . .	273
8-5	Using the lsdom command -f to display the DBA and HR domains in stanza format . . . . .	274
8-6	Using the chdom command to change the ID attribute of the Network domain . . . . .	275
8-7	Using the rmdom command to remove the Network domain . . . . .	276
8-8	The setsecattr -o command . . . . .	278
8-9	Using the lssecattr and ls -ltra command to display the file named /home/dba/privatefiles . . . . .	278
8-10	The lssecattr -o command. . . . .	280
8-11	The rmsecattr -o command . . . . .	280
8-12	The setkst -t command updating the domain into the KST . . . . .	281
8-13	Listing the kernel security tables with the lskst -t command . . . . .	281
8-14	The lsuser -a command - display a user domain access . . . . .	281
8-15	The lssec -f command - display a user domain access . . . . .	281
8-16	The chuser command - change a user domain association . . . . .	282
8-17	The chuser command - remove all domain association from a user. . . . .	282
8-18	The chsec command - adding DBA domain access to the dba user . . . . .	282
8-19	The /etc/nscontrol.conf file . . . . .	283
8-20	Using the lssecattr command to identify command authorizations . . . . .	286
8-21	Using the mkrole command - create the apps_fs_manage role . . . . .	287
8-22	Using the lsrole, the swrole and the lskst commands . . . . .	288
8-23	The setkst -t command - updating the role database into the KST . . . . .	289
8-24	The lsuser and chuser commands - assigning the apps_fs_manage role to the appuser account with the chuser command . . . . .	289

8-25	Using the rolist -a and rolist -e commands . . . . .	290
8-26	The appuser account using the swrole command to switch to the apps_fs_manage role. . . . .	291
8-27	The appuser account using the chfs command to add 1 GB to the /apps04 file system. . . . .	291
8-28	The appuser account using the umount command to unmount the /apps01 file system. . . . .	292
8-29	The appuser account using the chfs command to change the /backup file system . . . . .	293
8-30	The mkdom command - creating the applvDom and privlvDom domains. . . . .	294
8-31	The df -kP output - file systems on the AIX V7.1 LPAR . . . . .	295
8-32	Using the setsecattr command to define the four application file systems as domain RBAC objects . . . . .	295
8-33	Using the setsecattr command to define the remaining file systems as domain RBAC objects . . . . .	296
8-34	Using the setkst command to update the KST . . . . .	297
8-35	Using the chuser command to associate the appuser account with the applvDom domain. . . . .	298
8-36	The appuser account uses the swrole command to switch to the apps_fs_manage role. . . . .	298
8-37	The appuser account using the chfs command to add 1 GB to the /apps01 file system. . . . .	299
8-38	The appuser account attempting to use the chfs command to add 1 GB to the /backup file system. . . . .	300
8-39	The appuser account using the touch and whoami commands . . . . .	301
8-40	The netuser account - using the head -15 command to view the first 15 lines of the /etc/hosts file . . . . .	302
8-41	The appuser account - using the head -15 command to view the first 15 lines of the /etc/hosts file . . . . .	302
8-42	Using the mkdom command to create the privDom domain . . . . .	303
8-43	Using the setsecattr command to define the /etc/hosts file as a domain RBAC object . . . . .	303
8-44	Updating the KST with the setkst command . . . . .	304
8-45	The netuser account using the head -15 command to access the /etc/hosts file. . . . .	304
8-46	The appuser account using the head -15 command to access the /etc/hosts file. . . . .	305
8-47	Using the chuser command to grant the netuser account association to the privDom domain . . . . .	305
8-48	The netuser account using the head -15 command to access the /etc/hosts file. . . . .	306
8-49	The appuser account using the head -15 command to access the	

/etc/hosts file . . . . .	306
8-50 The appuser account - using the head -15 command to view the first 15 lines of the /etc/ssh/sshd_config file . . . . .	308
8-51 Using the mkdom command to create the lockDom domain. . . . .	309
8-52 Using the setsecattr command to define the /etc/ssh/sshd_config file as a domain RBAC object . . . . .	309
8-53 Using the setkst command to update the KST and the lskst command to list the KST . . . . .	309
8-54 Using the head, more, cat, pg and vi commands to attempt access to the /etc/ssh/sshd_config file . . . . .	310
8-55 Using the lssecatr command from the root user to list the access authorizations for the ifconfig command. . . . .	312
8-56 Using the authrpt command from the root user to determine role association with the aix.network.config.tcpip authorization . . . . .	312
8-57 Using the mkrole command from the root user to create the netifconf role and associate with the aix.network.config.tcpip authorization . . . . .	313
8-58 Using the chuser command from the root user to associate the netuser account with the netifconf role . . . . .	313
8-59 The root user using the authrpt and rolerpt commands . . . . .	314
8-60 The root user using the ifconfig -a command to display the network interface status . . . . .	314
8-61 The root user using the mkdom command to create the netDom and the privNetDom RBAC domains . . . . .	315
8-62 The setsecattr command being used by the root user to define the en0 and en2 domain RBAC objects . . . . .	315
8-63 Using the chuser command to associate the netuser account with the netDom domain . . . . .	316
8-64 The netuser account uses the swrole command to switch to the netifconf role . . . . .	317
8-65 The netuser account using the ifconfig command to deactivate the en2 Ethernet interface . . . . .	317
8-66 The netuser account using the ifconfig command to activate the en2 Ethernet interface . . . . .	318
8-67 The netuser account is unsuccessful in using the ifconfig command to inactivate the en0 Ethernet interface . . . . .	319
8-68 The chuser command used to add the privNetDom association to the netuser account . . . . .	320
8-69 The netuser account using the ifconfig command to deactivate the en0 interface - the conflict set does not allow access to the en0 domain RBAC object . . . . .	321
8-70 Configuring auditing with full pathnames. . . . .	322
8-71 Recycling of audit trail files. . . . .	325
8-72 Merging audit trail files . . . . .	326

8-73	.....	326
8-74	Propolice or Stack Smashing Protection.....	329
8-75	Modifying ngroups_allowed.....	330
8-76	Kerberos client kadmind_timeout option.....	331
8-77	Disallowing user names in passwords.....	332
8-78	Disallowing regular expressions in passwords.....	333
8-79	Usage of minloweralpha security attribute.....	335
9-1	Using prtconf to determine the processor type of a Power system.....	340
9-2	Selecting the AIX edition during as BOS installation.....	343
9-3	The chedition command flags and options.....	344
9-4	Modifying the AIX edition with the chedition command.....	344
9-5	Bootlist man page pathid concerns.....	348
9-6	bootlist -m normal -o command output.....	349
9-7	lspath, rmpath and mkpath command.....	349
9-8	Content of ae package.....	356
9-9	.Enabling activation engine.....	357
9-10	The Activation Engine command syntax.....	357
9-11	Grep of script in user created template file.....	358
9-12	Sample script /usr/samples/ae/templates/ae_template.xml.....	358
9-13	Successful Activation Engine template file structure check.....	360
9-14	SUMA default base configuration on AIX V6.1.....	361
9-15	SUMA default base configuration on AIX V7.1.....	365
9-16	eccBase.properties file after SUMA default base configuration.....	366
11-1	X11 lpp package content.....	384
11-2	prtconf command to determine the processor type of the system.....	392





# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described 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:  
*IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.*

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION 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. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM 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 IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.


## COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

## Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

Active Memory™	GPFS™	PowerPC®
AIX 5L™	HACMP™	PowerVM™
AIX®	IBM Systems Director Active	POWER®
BladeCenter®	Energy Manager™	pSeries®
Blue Gene®	IBM®	Redbooks®
DB2®	LoadLeveler®	Redbooks (logo)  ®
developerWorks®	Parallel Sysplex®	Solid®
Electronic Service Agent™	Power Systems™	System p5®
Enterprise Storage Server®	POWER3™	System p®
eServer™	POWER4™	System Storage®
Everyplace®	POWER5™	Systems Director VMControl™
GDPS®	POWER6®	Tivoli®
Geographically Dispersed	POWER7™	WebSphere®
Parallel Sysplex™	PowerHA™	Workload Partitions Manager™

The following terms are trademarks of other companies:

Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

# Preface

This IBM® Redbooks® publication focuses on the differences introduced in IBM AIX® Version 7.1 Standard Edition when compared to AIX Version 6.1. It is intended to help system administrators, developers, and users understand these enhancements and evaluate potential benefits in their own environments.

AIX Version 7.1 introduces many new features, including the following.

- ▶ Domain Role Based Access Control
- ▶ Support for up to 254 partitions on the Power 795
- ▶ Workload Partition Enhancements
- ▶ Topas performance tool enhancements
- ▶ Terabyte segment support
- ▶ Cluster Aware AIX functionality

There are many other new features available with AIX Version 7.1, and you can explore them all in this publication.

For clients who are not familiar with the enhancements of AIX through Version 5.3, a companion publication, AIX Version 6.1 Differences Guide, SG24-7559 is available.

## The team who wrote this book

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, Austin Center.

**Richard Bassemir** is a Senior Software Engineer in the ISV Business Strategy and Enablement organization within the Systems and Technology Group located in Austin, Texas. He has seven years of experience in IBM System p® technology. He has worked at IBM for 33 years. He started in mainframe design, design verification, and test, and moved to Austin to work in the Software Group on various integration and system test assignments before returning to the Systems and Technology Group to work with ISVs to enable and test their applications on System p hardware.

**Thierry Fauck** is a certified IT specialist working in Toulouse, France. He has 25 years of experience in Technical Support with major HPC providers. As system

administrator of the franch dev. lab his areas of expertise include AIX, VIOS, SAN and PowerVM™. He is currently leading a FVT development team for WPAR and WPAR mobility features. He wrote a white paper on WPARs and actively contributed to the WPAR RedBook. This is his second RedBook publication with AIX difference guide.

**Chris Gibson** is an AIX and PowerVM specialist. He works for Southern Cross Computer Systems, an IBM Business Partner located in Melbourne, Australia. He has 11 years of experience with AIX and is an IBM Certified Advanced Technical Expert - AIX. He is an active member of the AIX community and has written numerous technical articles on AIX and PowerVM for IBM developerWorks®. He also writes his own AIX blog on the IBM developerWorks website. Chris is also available online via Twitter @cgibbo. This is his second Redbooks publication having previously co-authored the NIM from A to Z in AIX 5L™ Redbook.

**Brad Gough** is a technical specialist working for IBM Global Services in Sydney, Australia. Brad has been with IBM since 1997. His areas of expertise include AIX, PowerHA™ and PowerVM. He is an IBM Certified Systems Expert - IBM System p5® Virtualization Technical Support and IBM eServer™ p5 and pSeries® Enterprise Technical Support AIX 5L V5.3. This is his third IBM Redbooks publication.

**Murali Neralla** is a Senior Software Engineer in the ISV Business Strategy and Enablement organization. He is also a certified consulting IT Specialist. He has over 15 years of experience working at IBM. Murali currently works with the Financial Services Sector solution providers to enable their applications on IBM Power Systems™ running AIX.

**Armin Röhl** works as a Power Systems IT specialist in Germany. He has 15 years of experience in Power Systems and AIX pre-sales technical support and, as a team leader, he fosters the AIX skills community. He holds a degree in experimental physics from the University of Hamburg, Germany. He co-authored the AIX Version 4.3.3, the AIX 5L Version 5.0, the AIX 5L Version 5.3 and the AIX 6.1 Differences Guide IBM Redbooks..

**Murali Vaddagiri** is a Senior Staff Software Engineer working for IBM Systems and Technology Group in India. He has over 7 years of experience in AIX operating system and PowerHA development. He holds Master of Science degree from BITS, Pilani, India. His areas of expertise include Security, Clustering, and Virtualization. He has filed 9 US patents so far and authored several disclosure publications in these areas.

The project that produced this publication was managed by:  
**Scott Vetter**, PMP.

Thanks to the following people for their contributions to this project:

Pramod Bhandiwad, Kavana N Bhat, Shajith Chandran, David Clissold, Zhi-wei Dai, Frank Feuerbacher, Madhusudanan Kandasamy, Michael Lyons, Su Liu, Dave Marquardt, Steven Molis, Roocha K Pandya, David R. Posh, Harinipriya Raghunathan, Sameer K Sinha, Masato Suzuki, Frank L Nichols, Lakshmi Yadlapati, Saravanan Devendra, Kiran Grover, Manoj Kumar, Jeff Palm, Derwin Gavin, Carl Bender, Dan McNichol, Manoj Kumar, Michael Schmidt, Julie Craft, Xiaohan Qin, Patrick T Vo, Duen-wen Hsiao, Pramod Bhandiwad, Alex Medvedev, Jim Allen, Brian Crosswell, Robin Hanrahan, Kari Karhi, Ann Wigginton, Bruce M Potter, Brian Crosswell, Robin Hanrahan, Andy Wong, David Navarro, Rae Yang, Sungjin Yook, Nathaniel S. Tomsic, Philippe Bergehead, Paul B Finley, Marty Fullam, Marc Stephenson, Marc McConaughy, Kim Tran, Khalid Filali-Adib, Kent Hofer, Jaime Contreras, Gary Ruzek, Eric S Haase, David Sheffield, Cheryl L Jennings, Bruce Mealey, Prasad V Potluri, Vi T Tran, Alan Jiang, Brian Veale, Vishal Aslot, Rae Yang, Shawn Mullen, Saurabh Desai, James Moody, Nikhil Hegde, Amit Agarwal, Jyoti B Tenginakai, Christian Caudrelier, Ravi Shankar, Deborah McLemore, Eric Fried, Lance Russell, Chris Schwendiman, David Hepkin, André L. Albot, Rosa Davidson, Jason J. Jaramillo, Kam Lee, Baltazar De Leon III, Binh Hua, Bi`nh T. Chu, Christian Karpp, Dave Marquardt, David A Hepkin, David Bennin, Deanna M Johnson, Diane Chung, Felipe Knop, Francoise Boudier, Frank Dea, George M Koikara, Gerald McBrearty, Guhaa Prasad Venkataraman, James P Allen, Jim Czenkusch, Jim Gallagher, Kim-Khanh V. (Kim) Tran, Kunal Katyayan, Mark Alana, Omar Cardona, Poornima Sripada Rao, R Vidya, Rae Yang, Ray Longhi, Richard M Conway, Saurabh Desai, Shaival J Chokshi, Shawn Mullen, Teerasit Tinnakul, Timothy Damron, Wojciech Stryjewski, Xiaohan Qin, Saravanan Devendra, Madhusudanan Kandasamy, Lakshmanan Velusamy, David Clissold, Jyoti B Tenginakai, Nikhil Hegde, Stephen B. Peckham, Tommy (T.U.) Hoffner, Kari Karhi, Subhash C. Bose

## **Now you can become a published author, too!**

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

[ibm.com/redbooks/residencies.html](http://ibm.com/redbooks/residencies.html)

## Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- ▶ Use the online **Contact us** review Redbooks form found at:

[ibm.com/redbooks](http://ibm.com/redbooks)

- ▶ Send your comments in an email to:

[redbooks@us.ibm.com](mailto:redbooks@us.ibm.com)

- ▶ Mail your comments to:

IBM Corporation, International Technical Support Organization  
Dept. HYTD Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400

## Stay connected to IBM Redbooks

- ▶ Find us on Facebook:

<http://www.facebook.com/IBMRedbooks>

- ▶ Follow us on Twitter:

<http://twitter.com/ibmredbooks>

- ▶ Look for us on LinkedIn:

<http://www.linkedin.com/groups?home=&gid=2130806>

- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>

- ▶ Stay current on recent Redbooks publications with RSS Feeds:

<http://www.redbooks.ibm.com/rss.html>



# Application development and debugging

This chapter contains the major AIX Version 7.1 enhancements that are part of the application development and system debug category, including:

- ▶ 1.1, “AIX binary compatibility” on page 2
- ▶ 1.2, “Improved performance using 1 TB segments” on page 2
- ▶ 1.3, “Kernel sockets application programming interface” on page 5
- ▶ 1.4, “Unix08 Standard Conformance” on page 6
- ▶ 1.5, “AIX assembler enhancements” on page 10
- ▶ 1.6, “Malloc debug fill” on page 11
- ▶ 1.7, “Core dump enhancements” on page 12
- ▶ 1.8, “Disabled read write locks” on page 13
- ▶ 1.9, “DBX enhancements” on page 16

## 1.1 AIX binary compatibility

IBM guarantees that applications, whether written in house or supplied by an application provider, will run on AIX 7.1 if they currently run on AIX 6.1 or AIX 5L – without recompilations or modification. Even well-behaved 32-bit applications from AIX V4.1, V4.2, and V4.3 will run without recompilation.

Refer to the below URL for further information in regards to binary compatibility:

<http://www.ibm.com/systems/power/software/aix/compatibility/>

## 1.2 Improved performance using 1 TB segments

AIX V7.1 1 TB segments are an autonomic operating system feature to improve performance of 64-bit large memory applications. This enhancement optimizes performance when using shared memory regions (shmat/mmap). New restricted **vmo** options are available to change the operating system policy. A new **VMM\_CNTRL** environment variable is available to alter per process behavior

**Note:** Restricted tunables should not be changed without direction from IBM service.

1 TB segment aliasing improves performance by using 1 TB segment translations on Shared Memory Regions with 256 MB segment size. This support is provided on all 64 bit applications that use Shared Memory Regions. Both directed and undirected shared memory attachments are eligible for 1 TB segment aliasing.

If an application qualifies to have its Shared Memory Regions to use 1 TB aliases, the AIX operating system uses 1 TB segments translations without changing the application. This requires using the **shm\_1tb\_shared vmo** tunable, **shm\_1tb\_unshared vmo** tunable, and **esid\_allocator vmo** tunable.

The **shm\_1tb\_shared vmo** tunable can be set on a per-process basis using the **SHM\_1TB\_SHARED= VMM\_CNTRL** environment variable. The default value is set dynamically at boot time based on the capabilities of the processor. If a single Shared Memory Region has the required number of ESIDs, it is automatically changed to a shared alias. The acceptable values are in the range of 0 to 4 KB (approximately require 256 MB ESIDs in a 1 TB range).

Example 1-1 on page 3 shows valid values for **shm\_1tb\_shared** tunable parameter



*Example 1-1 The shm\_1tb\_shared tunable*


---

```
#vmo -F -L shm_1tb_shared
NAME CUR DEF BOOT MIN MAX UNIT TYPE
DEPENDENCIES
-----
shm_1tb_shared 0 12 12 0 4K 256MB segments D
-----
#
```

---

The `shm_1tb_unshared vmo` tunable can be set on a per-process basis using the `SHM_1TB_UNSHARED=VMM_CNTRL` environment variable. The default value is set to 256. The acceptable values are in a range of 0 to 4 KB. The default value is set cautiously (requiring the population of up to 64 GB address space) before moving to an unshared 1 TB alias.

The threshold number is set to 256 MB segments at which a shared memory region is promoted to use a 1 TB alias. Lower values must cautiously use the shared memory regions to use a 1 TB alias. This can lower the segment look-aside buffer (SLB) misses but can also increase the page table entry (PTE) misses, if many shared memory regions that are not used across processes are aliased.

Example 1-2 shows valid values from `shm_1tb_unshared` tunable parameter

*Example 1-2 The shm\_1tb\_unshared tunable*


---

```
#vmo -F -L shm_1tb_unshared
NAME CUR DEF BOOT MIN MAX UNIT TYPE
DEPENDENCIES
-----
shm_1tb_unshared 256 256 256 0 4K 256MB segments D
-----
#
```

---

The `esid_allocator vmo` tunable can be set on a per-process basis using the `ESID_ALLOCATOR=VMM_CNTRL` environment variable. The default value is set to 0 for AIX Version 6.1 and 1 for AIX Version 7.1. Values can be either 0 or 1. When set to 0, the old allocator for undirected attachments is enabled. Otherwise, a new address space allocation policy is used for undirected attachments.

This new address space allocator attaches any undirected allocation (such as SHM, and MMAP) to a new address range of 0x0A00000000000000 - 0x0AFFFFFFFFFFFFFF in the address space of the application.

The allocator optimizes the allocations in order to provide the best possible chances of 1 TB alias promotion. Such optimization can result in address space holes, which are considered normal when using undirected attachments.

Directed attachments is done for 0x0700000000000000 - 0x07FFFFFFFFFFFFFFF range, thus preserving compatibility with earlier version. In certain cases where this new allocation policy creates a binary compatibility issue, the legacy allocator behavior can be restored by setting the tunable to 0.

Example 1-3 shows valid values for `esid_allocation` tunable parameter

*Example 1-3 The `esid_allocator` tunable*

# vmo -F -L esid_allocator							
NAME	CUR	DEF	BOOT	MIN	MAX	UNIT	TYPE
DEPENDENCIES							
esid_allocator	1	1	1	0	1	boolean	D
#							

Shared memory regions that were not qualified for shared alias promotion are grouped into 1 TB regions. In a group of shared memory regions in a 1 TB region of the application's address space, if the application exceeds the threshold value of 256 MB segments they are promoted to use an unshared 1 TB alias.

In applications, where numerous shared memory is attached and detached, lower values of this threshold can result in increased PTE misses. Applications which only detach shared memory regions at exit can benefit from lower values of this threshold.

To avoid polluting the environments name space, all environment tunables are used under the master tunable `VMM_CNTRL`. The master tunable is specified with the `@` symbol separating the commands.

An example for using `VMM_CNTRL` is:

```
VMM_CNTRL=SHM_1TB_UNSHARED=32@SHM_1TB_SHARED=5
```

**Note:** All 32-bit applications are not affected by either `vmo` or environment variable tunable changes.

All `vmo` tunables and environment variables have analogous `vm_patr` commands. The exception is `esid_allocator` tunable. This tunable is not present in the `vm_patr` options to avoid situations where portions of the shared memory address space are allocated before running the command.

If using AIX Runtime Expert, the `shm_1tb_shared`, `shm_1tb_unshared` and `esid_allocator` tunables are all in the `vmoProfile.xml` profile template.

## 1.3 Kernel sockets application programming interface

To honor the increasing customer and ISV demand to code environment and solutions specific kernel extensions with socket level functionality AIX V7.1 and AIX V6.1 with TL 6100-06 provide a documented kernel sockets application programming interface (API). The kernel service sockets API is packaged with other previously existing networking APIs in the base operating system 64-bit multiprocessor runtime fileset `bos.mp64`.

The header file `/usr/include/sys/kern_socket.h` which defines the key data structures and function prototypes is delivered along with other existing header files in the `bos.adt.include` fileset. As shown by Table 1-1 the implementation of the new programming interface is comprised of 12 new kernel services for TCP protocol socket operations. The API support the address families of both, IPv4 (`AF_INET`) and IPv6 (`AF_INET6`).

Table 1-1 Kernel service socket API

TCP protocol socket operation	Kernel service name	Function
Socket creation	<code>kern_socreate</code>	Creates a socket based on the address family, type and protocol.
Socket binding	<code>kern_sobind</code>	Associates the local network address to the socket.
Socket connection	<code>kern_soconnect</code>	Establishes connection with a foreign address.
Socket listen	<code>kern_solisten</code>	Prepares to accept incoming connections on the socket.
Socket accept	<code>kern_soaccept</code>	Accepts the first queued connection by assigning it to the new socket.
Socket get option	<code>kern_sogetopt</code>	Obtains the option associated with the socket, either at the socket level or at the protocol level.

TCP protocol socket operation	Kernel service name	Function
Socket set option	kern_sosetopt	Sets the option associated with the socket, either at the socket level or at the protocol level.
Socket reserve operation to set send and receive buffer space	kern_soreserve	Enforces the limit for the send and receive buffer space for a socket.
Socket shutdown	kern_soshutdown	Closes the read-half, write-half or both read and write of a connection.
Socket close	kern_soclose	Aborts any connections and releases the data in the socket.
Socket receive	kern_soreceive	The routine processes one record per call and tries to return the number of bytes requested.
Socket send	kern_sosend	Pass data and control information to the protocol associated send routines.

For a detailed description of each individual kernel service refer to *Technical Reference: Kernel and Subsystems, Volume 1, SC23-6612* of the AIX product documentation.

<http://publib.boulder.ibm.com/infocenter/aix/v6r1/topic/com.ibm.aix.kerneltechref/doc/ktechrf1/ktechrf1.pdf>

## 1.4 Unix08 Standard Conformance

The POSIX UNIX® standard is periodically updated. Recently, a draft standard for issue 7 has been released. It is important from both an open standards and a customer perspective to implement these new changes to the standards.

AIX v7.1 has implemented IEEE POSIX.1-200x The Open Group Base Specifications, Issue 7 standards in conformance to these standards.

The Base Specifications volume contains general terms, concepts, and interfaces of this standard, including utility conventions and C-language header definitions. It also contains the definitions for system service APIs and

subroutines, language-specific system services for the C programming language, and API issues, including portability, error handling, and error recovery.

The Open Group Base Specifications, Issue 7 can be found at following website:

<http://www.unix.org/2008edition>

In adherence to IEEE POSIX.1-200x The Open Group Base Specifications, Issue 7 standards, several enhancements were made in AIX v7.1.

New system calls are added so that users can open a directory and then pass the returned file descriptor to a system call, together with a relative path from the directory. The names of the new system calls in general have been taken from the existing system calls with an *at* added at the end. For example, an `accessxat()` system call has been added, similar to `accessx()` and `openat()` for an `open()`.

There are several advantages when these using these enhancements . For example, users can implement a per-thread current working directory with the newly added system calls. Another example is, users can avoid race conditions where part of the path is being changed while the pathname parsing is ongoing.

Table 1-2 shows a subset of new library functions and system calls that are added.

*Table 1-2 short list of new library functions and system calls*

<b>System Calls</b>	<b>System Calls</b>
acessxat	mknodat
chownxat	openat
faccessat	openxat
fchmodat	readlinkat
fchownat	renameat
fexecve	stat64at
fstatat	statx64at
futimens	statxat
kopenat	symlinkat
linkat	ulinkat
mkdirat	utimensat

System Calls	System Calls
mkfifoat	

Example 1-4 shows how applications can make use of these calls. The overall effect is same as a user had done an open call to the path dir\_path/filename:

*Example 1-4 A sample application call sequence*

---

```

.....
dirfd = open(dir_path, ...);
.....
accessxat(dirfd, filename, ...);
.....
fd = openat(dirfd, filename, ...);
.....

```

---

Table 1-3 Shows a subset of added routines that are the same as isalpha, isupper, islower, isdigit, isxdigit, isalnum, isspace, ispunct, isprint, isgraph, iscntrl subroutines respectively, except that they test character C in the locale represented by Locale, instead of the current locale.

*Table 1-3 new library functions to test character in a locale*

Name	Name
isupper_l	ispunct_l
islower_l	isprint_l
isdigit_l	isgraph_l
isxdigit_l	iscntrl_l
isspace_l	isalnum_l

## 1.4.1 stat structure changes

The stat, stat64, and stat64x are changed. New st\_atim field, of type struct timespec, replaces the old st\_atime and st\_atime\_n fields:

```

struct timespec {
    time_t tv_sec; /* seconds */
    long tv_nsec; /* and nanoseconds */
};

```

The old fields are now macros defined in <sys/stat.h> file:  

```
#define st_atime      st_atim.tv_sec
```

```
#define st_mtime      st_mtim.tv_sec
#define st_ctime      st_ctim.tv_sec
#define st_atime_n    st_atim.tv_nsec
#define st_mtime_n    st_mtim.tv_nsec
#define st_ctime_n    st_ctim.tv_nsec
```

## 1.4.2 open system call changes

Two new open flags are added to the open() system call:

```
#include <fcntl.h>
```

```
int open(const char *path, int oflag, ...);
```

- ▶ O\_DIRECTORY:

If path field does not name a directory, open() fails and sets errno to ENOTDIR.

- ▶ O\_SEARCH:

Open a directory for search, open() returns an error EPERM if there is no search permissions.

**Note:** The O\_SEARCH flag value is the same as the O\_EXEC flag therefore, the result is unspecified if this flag is applied to a non-directory file.

## 1.4.3 utimes system call changes

The utimes() system call is changed as follows:

```
#include <sys/stat.h>
```

```
utimes(const char *fname, const struct timeval times[2]);
```

- ▶ If either of the times parameter timeval structure tv\_usec fields have the value UTIME\_OMIT, then this time value is ignored.
- ▶ If either of the times parameter timespec structure tv\_usec fields have the value UTIME\_NOW, then this time value is set to the current time.

This provides a way the access and modify times of a file can be better adjusted.

## 1.4.4 futimens and utimensat system calls

Two new system calls `futimens()` and `utimensat()` are added. Both provide nanosecond time accuracy, and include the `UTIME_OMIT` and `UTIME_NOW` functionality. The `utimensat()` is for pathnames and `futimens()` is for open file descriptors.

```
int utimensat(int dirfd, const char *fname, const struct timespec times[2], int flag);
```

```
int futimens(int fd, const struct timespec times[2]);
```

## 1.4.5 fexecve system call

The new `fexecve` system call is added as follows:

```
#include <unistd.h>
```

```
int fexecve(int fd, const char *argp[], const char *envp[]);
```

The `fexecve` works same as `execve()` system call, except it takes a file descriptor of an open file instead of a pathname of a file. The `fexecve` call may not be used with RBAC commands (the file must have DAC execution permission).

For a complete list of changes, refer to AIX v7.1 documentation at:

[http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.ntl/releasenotes\\_kickoff.htm](http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.ntl/releasenotes_kickoff.htm)

## 1.5 AIX assembler enhancements

The following section discusses the enhancements made to the assembler in AIX v7.1.

### 1.5.1 Thread Local Storage (TLS) support

Thread Local Storage (TLS) support has been present in the IBM XL C/C++ compiler for some time. The compiler's `-qtls` option enables recognition of the `__thread` storage class specifier, which designates variables that are allocated from threadlocal storage

When this option is in effect, any variables marked with the `__thread` storage class specifier are treated as local to each thread in a multi-threaded application.



At run time, an instance of each variable is created for each thread that accesses it, and destroyed when the thread terminates. Like other high-level constructs that you can use to parallelize your applications, thread-local storage prevents race conditions to global data, without the need for low-level synchronization of threads.

The TLS feature is extended to the assembler in AIX v7.1 to allow the assembler to generate object files with TLS functionality from an associated assembler source file.

## 1.5.2 TOCREL support

Recent versions of the IBM XL C/C++ compilers support compiler options (for example: `-qfuncsect`, `-qxflag=tocrel`) that can reduce the likelihood of TOC overflow. These compiler options enable the use of new storage-mapping classes and relocation types, allowing certain TOC symbols to be referenced without any possibility of TOC overflow.

The TOCREL functionality is extended to the assembler in AIX v7.1. This allows the assembler to generate object files with TOCREL functionality from an associated assembler source file.

## 1.6 Malloc debug fill

Malloc debug fill is a debugging option which allows a user to fill up the allocated memory with a certain pattern.

The advantage of using this feature for debugging purposes is that it allows memory to be *painted* with some user-decided initialized value. This way, it can then be examined to determine if the requested memory has subsequently been used as expected by the application. Alternatively, an application could fill in the memory itself in the application code after returning from malloc, but this requires re-compilation and does not allow the feature to be toggled on or off at runtime.

For example, a user might fill the spaces with a known string, and then look (during debug) to see what memory has been written to or not, based on what memory allocations are still filled with the original fill pattern. When debugging is complete, the user can simply unset the environment variable and rerun the application.

Syntax for enabling Malloc debug fill option is as follows:

```
#export MALLOCDEBUG=fill:pattern
```

where *pattern* can be an octal or hexadecimal numbers specified in the form of a string.

Following example shows, a user has enabled the Malloc debug fill option and set the fill *pattern* to string *abc*

```
#export MALLOCDEBUG=fill:"abc"
```

Table 1-4 shows the fill *pattern* for a user allocating eight bytes of memory with a fill pattern of *abc*

Table 1-4 Malloc abc fill pattern

1	2	3	4	5	6	7	8
a	b	c	a	b	c	a	b

**Note:** *pattern* can be a octal or hexadecimal numbers specified in the form of a string. A pattern `\101` is treated as the octal notation for character A. A pattern `\x41` is treated as the hexadecimal notation for character A.

The fill-pattern is parsed byte by byte, so the maximum that can be set for fill pattern is `"\xFF"` or `"\101"`. If the user sets fill pattern as `"\xFFA"`, then it will be taken as hex FF and char A. If user wants A also to be taken as hex, the valid way of giving is `"\xFFxA"`. The same holds good for octal, if the user sets fill pattern as `"\101102"`, then it will be taken as octal 101 and string "102".

If an invalid octal number is specified, for example `\777` that cannot be contained within 1 byte, it will be stored as `\377`, the maximum octal value that can be stored within 1 byte.

## 1.7 Core dump enhancements

AIX 6.1 TL6 and 7.1 provides Application Programming Interfaces (API) `proc_getattr`, `proc_setattr` to allow a process to dynamically change its core dump settings.

The API supports enabling, disabling, and querying the settings for the following core dump settings:

- ▶ `CORE_MMAP` - controls whether the contents of `mmap()` regions are written into the core file.
- ▶ `fullcore` - controls whether a full core or partial core file are generated by default for the current process.

- ▶ **CORE\_NOSHM** - controls whether the contents of system V shared memory regions are written into the core file.
- ▶ **CORE\_NAMING** - controls whether unique core files should be created with unique names.

Applications can use these interfaces to ensure adequate debug information is captured in the cases where they dump core.

Example 1-5 provides syntax of these two APIs:

*Example 1-5 `proc_getattr()`, `proc_setattr()` APIs*

---

```
#include <sys/proc.h>

int proc_getattr (pid, attr, size)
pid_t pid;
procattr_t *attr;
uint32_t size;
```

The **proc\_getattr** subroutines allows a user to retrieve the current state of certain process attributes. The information is returned in the structure `procattr_t` defined in `sys/proc.h`

```
int proc_setattr (pid, attr, size)
pid_t pid;
procattr_t *attr;
uint32_t size;
```

The **proc\_setattr** subroutines allows a user to set selected attributes of a process. The list of selected attributes is defined in structure `procattr_t` defined in `sys/proc.h`

---

## 1.8 Disabled read write locks

The existing complex locks used for serialization among threads works only in process context. Because of this, complex locks are not suitable for the interrupt environment.

When simple locks are used to serialize heavily used disabled critical sections which could be serialized with a shared read / write exclusive model, performance bottlenecks may result.

AIX 7.1 provides kernel services for shared read, write exclusive locks for use in interrupt environments. These services can be used in kernel or kernel extension components to get improved performance for locks where heavy shared read access is expected. Table 1-5 lists these services:

Table 1-5 Kernel and kernel extension services

Index	Kernel service
1	<p><b>drw_lock_init</b></p> <p><b>Purpose</b> Initialize a disabled read-write lock.</p> <p><b>Syntax</b> #include&lt;sys/lock_def.h&gt; void drw_lock_init(lock_addr) drw_lock_t lock_addr ;</p> <p><b>Parameters</b> lock_addr - Specifies the address of the lock word to initialize.</p>
2	<p><b>drw_lock_read</b></p> <p><b>Purpose</b> Lock a disabled read-write lock in read-shared mode.</p> <p><b>Syntax</b> #include&lt;sys/lock_def.h&gt; void drw_lock_read(lock_addr) drw_lock_t lock_addr ;</p> <p><b>Parameters</b> lock_addr - Specifies the address of the lock word to lock.</p>
3	<p><b>drw_lock_write</b></p> <p><b>Purpose</b> Lock a disabled read-write lock in write-exclusive mode.</p> <p><b>Syntax</b> #include&lt;sys/lock_def.h&gt; void drw_lock_write(lock_addr) drw_lock_t lock_addr ;</p> <p><b>Parameters</b> lock_addr - Specifies the address of the lock word to lock.</p>

Index	Kernel service
4	<p><b>drw_lock_done</b></p> <p><b>Purpose</b> Unlock a disabled read-write lock.</p> <p><b>Syntax</b> #include&lt;sys/lock_def.h&gt; void drw_lock_done(lock_addr) drw_lock_t lock_addr ;</p> <p><b>Parameters</b> lock_addr - Specifies the address of the lock word to unlock.</p>
5	<p><b>drw_lock_write_to_read</b></p> <p><b>Purpose</b> Downgrades a disabled read-write lock from write exclusive mode to read-shared mode.</p> <p><b>Syntax</b> #include&lt;sys/lock_def.h&gt; void drw_lock write_to_read(lock_addr) drw_lock_t lock_addr ;</p> <p><b>Parameter</b> lock_addr - Specifies the address of the lock word to lock.</p>
6	<p><b>drw_lock_read_to_write</b> <b>drw_lock_try_read_to_write</b></p> <p><b>Purpose</b> Upgrades a disabled read-write from read-shared to write exclusive mode.</p> <p><b>Syntax</b> #include&lt;sys/lock_def.h&gt; boolean_t drw_lock read_to_write(lock_addr) boolean_t drw_lock try_read_to_write(lock_addr) drw_lock_t lock_addr ;</p> <p><b>Parameters</b> lock_addr - Specifies the address of the lock word to lock.</p>

Index	Kernel service
7	<p><b>drw_lock_islocked</b></p> <p><b>Purpose</b> Determine whether a drw_lock is held in either read or write mode.</p> <p><b>Syntax</b> #include&lt;sys/lock_def.h&gt; boolean_t drw_lock_islocked(lock_addr) drw_lock_t lock_addr ;</p> <p><b>Parameters</b> lock_addr - Specifies the address of the lock word.</p>
8	<p><b>drw_lock_try_write</b></p> <p><b>Purpose</b> Immediately acquire a disabled read-write lock in write-exclusive mode if available.</p> <p><b>Syntax</b> #include&lt;sys/lock_def.h&gt; boolean_t drw_lock_try_write(lock_addr); drw_lock_t lock_addr ;</p> <p><b>Parameters</b> lock_addr - Specifies the address of the lock word to lock.</p>

## 1.9 DBX enhancements

The following dbx enhancements were first made available in AIX v7.1 and AIX v6.1 TL06.

### 1.9.1 Dump memory areas in pointer format

A new option ('p' to print a pointer/address in hexadecimal format) is added to dbx **display** sub-command to print memory areas in pointer format. Example 1-6 displays five pointers (32-bit) starting from address location 0x20000a90:

*Example 1-6 Display 32-bit pointers*

---

```
(dbx) 0x20000a90 /5p
0x20000a90: 0x20000bf8 0x20000bb8 0x00000000 0x20000b1c
```

```
0x20000aa0: 0x00000000
```

---

Example 1-7 displays five pointers (64-bit) starting from address location 0x0fffffffffa88

*Example 1-7 Display 64-bit pointers*

---

```
(dbx) 0x0ffffffffffa88/5p
0x0ffffffffffa88: 0x0000000110000644 0x0000000110000664
0x0ffffffffffa98: 0x000000011000064c 0x0000000110000654
0x0ffffffffffa08: 0x000000011000065c
(dbx)
```

---

## 1.9.2 New dbx environment variable `print_mangled`

A new dbx environment variable called `print_mangled` is added. This environment variable is used to determine whether to print the C++ functions in mangled form or demangled form. Default value of `print_mangled` is unset. If set, dbx prints mangled function names. This feature allows users to use both mangled and demangled C++ function names with dbx sub commands. This applies for binaries compiled in debug mode (-g compiled option) and for binaries compiled in non-debug mode.

Example 1-8 demonstrates exploiting `print_mangled` environment variable while setting a break point in function1() overloaded function:

*Example 1-8 print\_mangled dbx environment variable*

---

```
(dbx) st in function1
1. example1.function1(char**)
2. example1.function1(int)
3. example1.function1(int,int)
Select one or more of [1 - 3]: ^C
(dbx) set $print_mangled
(dbx) st in function1
1. example1.function1__FPPc
2. example1.function1__Fi
3. example1.function1__FiT1
Select one or more of [1 - 3]: ^C
```

---

Example 1-9 on page 18 demonstrates how to reset the `print_mangled` environment variable, by running `unset` command.

*Example 1-9 unset print\_mangled dbx environment variable*

---

```
(dbx) unset $print_mangled
(dbx) st in function1
1. example1.function1(char**)
2. example1.function1(int)
3. example1.function1(int,int)
Select one or more of [1 - 3]:
```

---

### 1.9.3 DBX malloc subcommand enhancements

The following dbx `malloc` subcommand enhancements are made in AIX 7.1

- ▶ **malloc allocation** sub command of dbx was allowed only, when AIX environment variable `MALLOCDEBUG=log` was set. This restriction is removed in AIX 7.1.
- ▶ Output of **malloc freespace** sub command of dbx is enhanced to display the memory allocation algorithms. Example 1-10 displays output of `malloc freespace` sub command

*Example 1-10 malloc freespace dbx subcommand output*

---

```
(dbx) malloc freespace
Freespace Held by the Malloc Subsystem:

    ADDRESS          SIZE HEAP    ALLOCATOR
0x20002d60          57120     0      YORKTOWN
(dbx)q
# export MALLOCATYPE=3.1
```

```
(dbx) malloc freespace
Freespace Held by the Malloc Subsystem:

    ADDRESS          SIZE HEAP    ALLOCATOR
0x20006028             16     0         3.1
0x20006048             16     0         3.1
.....
.....
(dbx)
```

---

- ▶ A new argument (address of a memory location) is added to `malloc` sub command. This dbx sub command will fetch and display the details of the node to which this address belongs to.



Example 1-11 displays address argument to `malloc` sub command

*Example 1-11 address argument to malloc sub command*

---

(dbx) `malloc 0x20001c00`

Address 0x20001c00 node details :

**Status : ALLOCATED**

ADDRESS	SIZE	HEAP	ALLOCATOR
0x20000c98	4104	0	YORKTOWN

(dbx)

(dbx) `malloc 0x20002d60`

Address 0x20002d60 node details :

**Status : FREE**

ADDRESS	SIZE	HEAP	ALLOCATOR
0x20002d60	57120	0	YORKTOWN

(dbx)

---





# 2

## File systems and storage

This chapter contains the major AIX Version 7.1 enhancements that are part of the file system and connected storage, including:

- ▶ 2.1, “LVM enhancements” on page 22
- ▶ 2.2, “Hot Files Detection in JFS2” on page 27

## 2.1 LVM enhancements

The following section discusses LVM enhancements in detail.

### 2.1.1 LVM enhanced support for solid-state disks

Solid<sup>®</sup>-state disks (SSDs) are a very popular option for enterprise storage requirements. SSDs are unique in that they do not have any moving parts and thus perform at electronic speeds without mechanical delays (moving heads or spinning platters) associated with traditional spinning Hard Disk Drives (HDDs). Compared to traditional HDDs, the characteristics of SSDs enable a higher level of I/O performance in terms of greater throughput and lower response times for random I/O. These devices are ideal for applications that require high IOPS/GB and/or low response times.

AIX V7.1 includes enhanced support in the AIX Logical Volume Manager (LVM) for SSD. This includes the capability for LVM to restrict a volume group (VG) to only contain SSDs and the ability to report that a VG only contains SSDs. This feature is also available in AIX V6.1 with the 6100-06 Technology Level.

Traditionally a volume group can consist of physical volumes (PVs) from a variety of storage devices, such as HDDs. There was no method to restrict the creation of a volume group to a specific type of storage device. The LVM has been enhanced to allow for the creation of a volume group on a specific storage type, in this case SSDs. The ability to restrict a volume group to a particular type of disk can assist in enforcing performance goals for the volume group.

For example, a DB2<sup>®</sup> database may be housed on a set of SSDs for best performance. Reads and writes in that VG will only perform as fast as the slowest disk. For this reason it is best to restrict this VG to SSDs only. To maximise performance, the mixing of SSD and HDD hdisks in the same volume group must be prohibited.

The creation, extension and maintenance of an SSD VG must ensure that the restrictions are enforced. The following LVM commands have been modified to support this enhancement and enforce the restriction:

- ▶ **lsvg**
- ▶ **mkvg**
- ▶ **chvg**
- ▶ **extendvg**
- ▶ **replacevg**

The LVM device driver has been updated to support this enhancement. The changes to the LVM device driver and commands rely upon the successful identification of an SSD device. To determine if a disk is an SSD, the IOCINFO operation is used on the disk's `ioctl()` function. Using the specified bits, the disk can be examined to determine if it is an SSD device. The structures, `devinfo` and `scdk64` are both defined in `/usr/include/sys/devinfo.h`. If `DF_IVAL` (0x20) is set in the `flags` field of the `devinfo` structure, then the `flags` field in the `scdk64` structure is valid. The flags can then be examined to see if `DF_SSD` (0x1) is set.

For information on configuring SSD disk on an AIX system please refer to the following websites:

<http://www.ibm.com/developerworks/wikis/display/WikiPtype/Solid+State+Drives>

<http://www.ibm.com/developerworks/wikis/display/wikiptype/movies>

To confirm the existence of the configured SSD disk on our lab system, we ran the `lsdev` command, as shown in Example 2-1.

*Example 2-1 Output from the `lsdev` command showing SSD disk*

---

```
# lsdev -Cc disk
hdisk0 Available 01-08-00 Other SAS Disk Drive
hdisk1 Available 01-08-00 Other SAS Disk Drive
hdisk2 Available 01-08-00 Other SAS Disk Drive
hdisk3 Available 01-08-00 Other SAS Disk Drive
hdisk4 Available 01-08-00 SAS Disk Drive
hdisk5 Available 01-08-00 Other SAS Disk Drive
hdisk6 Available 01-08-00 SAS Disk Drive
hdisk7 Available 01-08-00 SAS Disk Drive
hdisk8 Available 01-08-00 Other SAS Disk Drive
hdisk9 Available 01-08-00 SAS RAID 0 SSD Array
hdisk10 Available 01-08-00 SAS RAID 0 SSD Array
hdisk11 Available 01-08-00 SAS RAID 0 SSD Array
```

---

The `mkvg` command accepts an additional flag, `-X`, to indicate that a new VG must reside on a specific type of disk. This effectively restricts the VG to this type of disk while the restriction exists. The following list describes the options to the `-X` flag.

- X none** This is the default setting. This does not enforce any restriction. Volume group creation can use any disk type.
- X SSD** At the time of creation, the volume group is restricted to SSD devices only.

In Example 2-2 we create an SSD restricted volume, named dbvg, with an SSD disk.

*Example 2-2 Creating an SSD restricted VG*

---

```
# lsdev -Cc disk | grep hdisk9
hdisk9 Available 01-08-00 SAS RAID 0 SSD Array
# mkvg -X SSD -y dbvg hdisk9
dbvg
```

---

**Note:** Once a PV restriction is turned on, the VG can no longer be imported on a version of AIX that does not support PV type restrictions.

Even if a volume group PV restriction is enabled and then disabled, it will no longer be possible to import it on a version of AIX that does not recognise the PV type restriction.

The use of the **-I** flag on a PV restricted VG is prohibited.

Two examples of when this limitation should be considered are:

- ▶ When updating the AIX level of nodes in a cluster. There will be a period of time when not all nodes are running the same level of AIX.
- ▶ When reassigning a volume group (exportvg/importvg) from one instance of AIX to another instance of AIX that is running a previous level of the operating system.

The **lsvg** command will display an additional field, PV RESTRICTION, indicating whether a PV restriction is set for a VG. If the VG has no restriction, the field will display *none*. The **lsvg** command output shown in Example 2-3 is for a volume group with a PV restriction set to SSD.

*Example 2-3 The volume group PV RESTRICTION is set to SSD*

---

```
# lsvg dbvg
VOLUME GROUP:      dbvg                VG IDENTIFIER: 00c3e5bc00004c000000012b0d2be925
VG STATE:          active              PP SIZE:       128 megabyte(s)
VG PERMISSION:    read/write          TOTAL PPs:    519 (66432 megabytes)
MAX LVs:          256                  FREE PPs:     519 (66432 megabytes)
LVs:              0                    USED PPs:     0 (0 megabytes)
OPEN LVs:         0                    QUORUM:       2 (Enabled)
TOTAL PVs:        1                    VG DESCRIPTORS: 2
STALE PVs:        0                    STALE PPs:    0
ACTIVE PVs:       1                    AUTO ON:      yes
MAX PPs per VG:   32512
MAX PPs per PV:   1016                  MAX PVs:      32
LTG size (Dynamic): 256 kilobyte(s)    AUTO SYNC:    no
HOT SPARE:        no                    BB POLICY:    relocatable
MIRROR POOL STRICT: off
PV RESTRICTION:   SSD
```

---

The **chvg** command accepts an additional flag, **-X**, to set or change the device type restriction on a VG. The following list describes the options available.

- X none** Removes any PV type restriction on a VG.
- X SSD** Places a PV type restriction on the VG if all the underlying disks are of type SSD. An error message is displayed if one or more of the existing PVs in the VG do not meet the restriction.

In Example 2-4 we first remove the PV type restriction from the volume group and then set the PV type restriction to SSD.

*Example 2-4 Changing the PV type restriction on a volume group*

---

```
# chvg -X none dbvg
# lsvg dbvg
VOLUME GROUP:      dbvg                VG IDENTIFIER: 00c3e5bc00004c000000012b0d2be925
VG STATE:          active              PP SIZE:       128 megabyte(s)
VG PERMISSION:    read/write          TOTAL PPs:    519 (66432 megabytes)
MAX LVs:          256                 FREE PPs:     519 (66432 megabytes)
LVs:              0                   USED PPs:     0 (0 megabytes)
OPEN LVs:         0                   QUORUM:       2 (Enabled)
TOTAL PVs:        1                   VG DESCRIPTORS: 2
STALE PVs:        0                   STALE PPs:    0
ACTIVE PVs:       1                   AUTO ON:      yes
MAX PPs per VG:   32512
MAX PPs per PV:   1016                MAX PVs:      32
LTG size (Dynamic): 256 kilobyte(s)  AUTO SYNC:    no
HOT SPARE:        no                  BB POLICY:    relocatable
MIRROR POOL STRICT: off
PV RESTRICTION: none
```

```
# chvg -X SSD dbvg
# lsvg dbvg
VOLUME GROUP:      dbvg                VG IDENTIFIER: 00c3e5bc00004c000000012b0d2be925
VG STATE:          active              PP SIZE:       128 megabyte(s)
VG PERMISSION:    read/write          TOTAL PPs:    519 (66432 megabytes)
MAX LVs:          256                 FREE PPs:     519 (66432 megabytes)
LVs:              0                   USED PPs:     0 (0 megabytes)
OPEN LVs:         0                   QUORUM:       2 (Enabled)
TOTAL PVs:        1                   VG DESCRIPTORS: 2
STALE PVs:        0                   STALE PPs:    0
ACTIVE PVs:       1                   AUTO ON:      yes
MAX PPs per VG:   32512
MAX PPs per PV:   1016                MAX PVs:      32
LTG size (Dynamic): 256 kilobyte(s)  AUTO SYNC:    no
HOT SPARE:        no                  BB POLICY:    relocatable
MIRROR POOL STRICT: off
PV RESTRICTION: SSD
```

---

If we attempt to create a volume group, using non-SSD disk with an SSD PV type restriction, the command will fail, as shown in Example 2-5.

*Example 2-5 Attempting to create an SSD restricted VG with a non-SSD disk*

---

```
# lsdev -Cc disk | grep hdisk1
hdisk1 Available 01-08-00 Other SAS Disk Drive
# mkvg -X SSD -y dbvg hdisk1
```

0516-1930 mkvg: **PV type not valid for VG restriction.**

**Unable to comply with requested PV type restriction.**

0516-1397 mkvg: The physical volume `hdisk1`, will not be added to the volume group.

0516-862 mkvg: Unable to create volume group.

---

Access to and control of this functionality is available via LVM commands only. At this time there are no SMIT panels for `mkvg` or `chvg` to set or change the restriction.

The `extendvg` and `replacepv` commands have been modified to honor any PV type restrictions on a volume group. For example, when adding a disk to an existing volume group with a PV restriction of SSD, the `extendvg` command will ensure that only SSD devices are allowed to be assigned, as shown in Example 2-6.

If you attempt to add a mix of non-SSD and SSD disks to an SSD restricted volume group, the command will fail. If any of the disks fail to meet the restriction, all of the specified disks are not added to the volume group, even if one of the disks is of the correct type. The disks in Example 2-6 are of type SAS (`hdisk7`) and SSD (`hdisk10`). So even though `hdisk10` is SSD, the volume group extension operation does not add it to the volume group as `hdisk7` prevents it from completing successfully.

*Example 2-6 Attempting to add a non-SSD disk to an SSD restricted volume group*

---

```
# lsdev -Cc disk | grep hdisk7
hdisk7 Available 01-08-00 SAS Disk Drive
# extendvg -f dbvg hdisk7
0516-1254 extendvg: Changing the PVID in the ODM.
0516-1930 extendvg: PV type not valid for VG restriction.
Unable to comply with requested PV type restriction.
0516-1397 extendvg: The physical volume hdisk7, will not be added to
the volume group.
0516-792 extendvg: Unable to extend volume group.
```

```
# lsdev -Cc disk | grep hdisk7
hdisk7 Available 01-08-00 SAS Disk Drive
# lsdev -Cc disk | grep hdisk10
hdisk10 Available 01-08-00 SAS RAID 0 SSD Array
# extendvg -f dbvg hdisk7 hdisk10
0516-1930 extendvg: PV type not valid for VG restriction.
Unable to comply with requested PV type restriction.
0516-1397 extendvg: The physical volume hdisk7, will not be added to
the volume group.
0516-1254 extendvg: Changing the PVID in the ODM.
0516-792 extendvg: Unable to extend volume group.
```

---



When using the `replacev` command to replace a disk, in a SSD restricted VG, the command will allow disks of that type only. If the destination PV is not the correct device type, the command will fail.

Currently, only the SSD PV type restriction will be recognized. In the future, additional strings may be added to the PV type definition, if required, to represent newly supported technologies.

Mixing both non-SSD and SSD disk in a volume group that does not have a PV type restriction is still possible, as shown in Example 2-7. In this example we created a volume group with a non-SSD disk (`hdisk7`) and an SSD disk (`hdisk9`). This will work as we did not specify a PV restriction with the `-X SSD` option with the `mkvg` command.

*Example 2-7 Creating a volume with both non-SSD and SSD disk*

---

```
# lsdev -Cc disk | grep hdisk7
hdisk7 Available 01-08-00 SAS Disk Drive
# lsdev -Cc disk | grep hdisk9
hdisk9 Available 01-08-00 SAS RAID 0 SSD Array
# mkvg -y dbvg hdisk7 hdisk9
dbvg
```

---

## 2.2 Hot Files Detection in JFS2

Solid-state disks (SSDs) offer a number of advantages over traditional hard disk drives (HDDs). With no seek time or rotational delays, SSDs can deliver substantially better I/O performance than HDDs. The following white paper, *Positioning Solid State Disk (SSD) in an AIX environment*, discusses these advantages in detail:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101560>

In order to maximize the benefit of SSDs it is important to only place data on them that requires high throughput and low response times. This data is referred to as *hot* data or *hot* files. Typically a *hot* file can be described as a file that is read from or written to frequently. It could also be a file that is read from or written to in large chunks of data.

Before making a decision to move suspected *hot* files to faster storage (for example SSDs), users of a file system need to determine which files are actually *hot*. The files must be monitored for a period of time in order to identify the best candidates.

AIX V7.1 includes enhanced support in the JFS2 file system for solid-state disks (SSDs). JFS2 has been enhanced with the capability to capture and report per-file statistics related to the detection of *hot* files that can be used to determine if a file should be placed on an SSD. These capabilities allow for applications to monitor and determine optimal file placement. This feature is also available in AIX V6.1 with the 6100-06 Technology Level.

JFS2 Hot File Detection (HFD) enables the collection of statistics relating to file usage on a file system. The user interface to HFD is through programming functions only. HFD is implemented as a set of ioctl function calls. The enhancement is designed specifically so that application vendors can integrate this function into their product code.

There is no AIX command line interface to the function or the statistics captured by HFD ioctl function calls.

These calls are implemented in the `j2_ioctl` function, where any of the `HFD_*` ioctl calls cause the `j2_fileStats` function to be called. This function handles the ioctl call and returns zero for success, or an error code on failure. When HFD is active in a file system, all reads and writes of a file in that file system cause HFD counters for that file to be incremented. When HFD is inactive, the counters are not incremented.

The HFD mechanism is implemented as several ioctl calls. The calls expect an open file descriptor to be passed to them. It does not matter which file in the file system is opened for this, as the system simply uses the file descriptor to identify the file system location and lists or modifies the HFD properties for the JFS2 file system.

The ioctl calls are defined in the `/usr/include/sys/hfd.h` header file. The contents of the header file are shown in Example 2-8.

*Example 2-8 /usr/include/sys/hfd.h header file*

---

```

/* IBM_PROLOG_BEGIN_TAG                               */
/* This is an automatically generated prolog.         */
/*                                                    */
/* $Source: aix710 bos/kernel/sys/hfd.h 1$ */
/*                                                    */
/* COPYRIGHT International Business Machines Corp. 2009,2009 */
/*                                                    */
/* Pvalue: p3 */
/* Licensed Materials - Property of IBM                */
/*                                                    */
/* US Government Users Restricted Rights - Use, duplication or
/* disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
/*                                                    */
/* Origin: 27 */
/*                                                    */
/* $Header: @(#) 1 bos/kernel/sys/hfd.h, sysj2, aix710, 0950A_710 2009-11-30T13:35:35-06:00$ */
/*                                                    */
/* IBM_PROLOG_END_TAG                               */

```

```

/* %Z%M%      %I% %W% %G% %U% */

/*
 * COMPONENT_NAME: (SYSJ2) JFS2 Physical File System
 *
 * FUNCTIONS: Hot Files Detection (HFD) subsystem header
 *
 * ORIGINS: 27
 *
 * (C) COPYRIGHT International Business Machines Corp. 2009
 * All Rights Reserved
 * Licensed Materials - Property of IBM
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
 */

#ifndef _H_HFD
#define _H_HFD

#include <sys/types.h>
#include <sys/ioctl.h>

#define HFD_GET      _IOR('f', 118, int)          /* get HFD flag */
#define HFD_SET      _IOW('f', 117, int)          /* set HFD flag */
#define HFD_END      _IOW('f', 116, int)          /* terminate HFD */
#define HFD_QRY      _IOR('f', 115, hfdstats_t)   /* get HFD stats */

/* Hot File Detection (HFD) ioctl specific structs and flags { */

typedef struct per_file_counters {
    ino64_t      c_inode;
    uint64_t     c_rbytes;
    uint64_t     c_wbytes;
    uint64_t     c_rops;
    uint64_t     c_wops;
    uint64_t     c_rtime;
    uint64_t     c_wtime;
    uint32_t     c_unique;
} fstats_t;

typedef struct hfd_stats_request {
    uint64_t     req_count;
    uint32_t     req_flags;
    uint32_t     req_resrvd;
    uint64_t     req_cookie;
    fstats_t     req_stats[1];
} hfdstats_t;

/* } Hot File Detection (HFD) ioctl specific structs and flags */

#endif /* _H_HFD */

```

The HFD ioctl calls are summarized as follows:

### **HFD\_GET**

A file descriptor argument is passed to this call, which contains an open file descriptor for a file in the desired file system. This ioctl call takes a pointer to an integer as its argument and returns the status of the HFD subsystem for the file system. If the returned integer is zero, then HFD is not active. Otherwise, HFD is active. All users can submit this ioctl call.

**HFD\_SET**

A file descriptor argument is passed to this call, which contains an open file descriptor for a file in the desired file system. This ioctl call takes a pointer to an integer as its argument. The integer needs to be initialized to zero before the call to disable HFD and to a non-zero to activate it. If the call would result in no change to the HFD state, no action is performed, and the call returns with success. If the user is not authorized, the call will return an EPERM error condition.

If HFD has not been active for the file system since it was mounted, it is initialized and memory is allocated for the HFD counters. Additional memory is allocated as required as files in the file system are read from, or written to. The HFD file counters are initialized to zeroes when they are allocated or reused (e.g. when a file is deleted). When the file system is unmounted, the HFD subsystem is terminated in the file system. The allocated memory is freed at this time. If HFD is deactivated, the counters are not incremented, but they are not reset either.

**HFD\_END**

This call causes the HFD subsystem to be terminated and memory allocated to it to be freed. Calling it while HFD is active in the file system causes an EBUSY error condition. If the user is not authorized, the call will return an EPERM error condition.

If the file system is activated again, the statistics counters will restart from zeroes. A file descriptor argument is passed to this call, which contains an open file descriptor for a file in the desired file system. This ioctl call takes only a NULL pointer as an argument. Passing any other value causes an EINVAL error condition.

**HFD\_QRY**

A file descriptor argument is passed to this call, which contains an open file descriptor for a file in the desired file system. This ioctl call takes a pointer to an hfdstats\_t structure as an argument. The structure must be initialized before the call, and it returns the current HFD statistics for active files in the file system.

If the argument is not a valid pointer, the call will return an EFAULT error condition. If the pointer is NULL, the call returns an EINVAL error condition. If HFD is not active, the call returns an ENOENT error condition. Depending on the passed-in values for the fields in the structure, the call returns different data in the same structure. If the user

is not authorized, the call will return an EPERM error condition.

The statistics counters for an active file are not reset. To find hot files, the HFD\_QRY ioctl call must be performed many times, over a set time interval. The statistics for each interval are calculated by subtracting the statistics values for each counter at the end and at the beginning of the interval.

The hfdstats\_t structure contains a one element long array of fstats\_t structures. Each structure contains the following fields: c\_inode, c\_unique, c\_rops, c\_wops, c\_rbytes, c\_wbytes, c\_rtime, and c\_wtime. These fields contain statistics of the file in question. The c\_rops and c\_wops fields contain the count of the read and write operations for the file. The c\_rbytes and c\_wbytes fields contain the number of bytes read from or written to the file. The c\_rtime and c\_wtime fields contain respectively the total amount of time spent in the read and write operations for the file. The c\_inode and c\_unique fields contain the inode and generation numbers of the file.

In addition, the mount and unmount functionality has been enhanced to allocate and free data structures required by the HFD subsystem. The j2\_rdw function has also been modified to increment HFD statistics counters. The file statistics collected for a file system are not saved when the file system is unmounted.

It is possible to activate, deactivate and terminate HFD for a file system. Per-file statistics are collected and can be retrieved via the programming interface. If HFD is activated for a file system there is minimal impact to the file systems performance and resource usage. After HFD is activated for a file system, its inodes will be write locked for the first read or write operation. A performance overhead associated with HFD would not be more than 2 % on a system with adequate memory, as measured by a standard file system test benchmark for read/write activity.

HFD uses memory to store the per-file statistics counters. This may cause a large increase in memory use while HFD is active. The extra memory is kept even when HFD is no longer active, until the file system is unmounted or HFD is terminated.

The memory requirement is about 64 bytes per active file. A file is considered active if it has had at least one read or write, while HFD has been active for the file system. However, the extra memory will not grow larger than the memory required by the number of files equal to the maximum number of inodes in the JFS2 inode cache (as specified by the j2\_inodeCacheSize ioo tuning parameter).

Since HFD is used only for statistics, its memory is not saved during a system dump, or live dump. The **kdb** and KDB utilities have been enhanced to print the values of the mount inode i\_j2fstats and the inode i\_fstats fields. There are no

additional trace hooks associated with HFD. The HFD memory heap can be inspected using **kdb** heap, pile, and slab commands.

Only authorized users may change the state of or retrieve statistics for a HFD enabled file system. HFD uses the PV\_KER\_EXTCONF privilege. To enable a program to modify the HFD state or to query active files, the program must have the appropriate privilege first. For example, the following set of commands would allow all users to run a program named `/tmp/test` to enable HFD on the `/testfs` file system:

```
# setsecattr -c secflags=FSF_EPS accessauths=ALLOW_ALL innateprivs=PV_KER_EXTCONF /tmp/test
# setkst
# su - guest
$ /tmp/test /testfs ON
HFD is now active
```

The following sample code demonstrates how the HFD\_QRY ioctl call can be used to find hot files in a file system, as shown in Example 2-9 on page 33.

The `print_stats` function would need to run `qsort` (or another sort function) to find hot files in the file system. The comparison function for the sort would need to have the selection criteria for a hot file built in, for example whether to use the number of bytes read or number of bytes written field. It also needs to check the `c_inode` and `c_unique` numbers and subtract the statistics counters of the two arrays to determine the count for the interval.

The `req_count` field allows the user to determine how large an array should be set in order to allocate data. The `req_stats` array contains entries for the statistics for each active file at the time of the HFD\_QRY call. Each entry has the inode number of the file in the `c_inode` field. If a file is deleted, its entry will become available for reuse by another file. For that reason, each entry also contains a `c_unique` field, which is updated each time the `c_inode` field changes.

The `ioctl` (`fd`, `HFD_QRY`, `&Query`) call returns per-file I/O statistics in the `Query` structure. There are three methods with which to use the HFD\_QRY `ioctl` call.

1. To query a single file, the passed-in value for `req_count` is zero. The `c_inode` field is also zero. This call will return file statistics for the file being referenced by the passed-in file descriptor. This method is useful for monitoring a single file.
2. To query all active files, the passed-in field for `req_count` is zero. This call returns with the `req_count` field set to the number of elements needed in the `req_stats` array. The size of the array will be set so that all of the data available at that point (that is the number of all active files) will be stored.
3. To query some active files in a file system, the passed-in field for `req_count` is set to a positive value. This call returns up to this many entries (`req_count`) in the `req_stats` array. If the passed-in value of `req_stats` array is large enough to

contain the number of active files, the req\_cookie field is set to zero on return. HFD\_QRY is called repeatedly until all entries are returned.

**Note:** The example code is for demonstration purposes only. It does not cater for any error handling, and does not take into account potential changes in the number of active files.

*Example 2-9 Example HFD\_QRY code*

---

```

int          fd, SetFlag, Count;
hfdstats_t  Query;
hfdstats_t  *QueryPtr1, *QueryPtr2;

fd = open("./filesystem.", O_RDONLY);    /* get a fd */
SetFlag = 1;
ioctl(fd, HFD_SET, &SetFlag);          /* turn on HFD */
Query.req_count = 0;
ioctl(fd, HFD_QRY, &Query);            /* find no of entries */
Count = Query.req_count + 1000; /* add some extras */
Size = sizeof(Query) + (Count . 1) * sizeof(fstats_t);
QueryPtr1 = malloc(Size);
QueryPtr2 = malloc(Size);
QueryPtr2->req_count = Count;
QueryPtr2->req_cookie = 0;
ioctl(fd, HFD_QRY, QueryPtr2);          /* get the data in 2 */
while (Monitor) {
    sleep(TimeInterval);
    QueryPtr1->req_count = Count;
    QueryPtr1->req_cookie = 0;
    ioctl(fd, HFD_QRY, QueryPtr1); /* get the data in 1 */
    print_stats(QueryPtr1, QueryPtr2); /* print stats 1 - 2 */
sleep(TimeInterval);
    QueryPtr2->req_count = Count;
    QueryPtr2->req_cookie = 0;
    ioctl(fd, HFD_QRY, QueryPtr2); /* get the data in 2 */
    print_stats(QueryPtr2, QueryPtr1); /* print stats 2 - 1 */
}
SetFlag = 0;
ioctl(fd, HFD_SET, &SetFlag);          /* turn off HFD */
ioctl(fd, HFD_END, NULL);              /* terminate HFD */

```

---







# Workload Partitions and resource management

This chapter discusses Workload Partitions (WPARs). WPARs are virtualized software-based partitions running within an instance of AIX. They are available in AIX V7.1 and AIX V6.1. This chapter contains the following sections:

- ▶ 3.1, “Trusted Kernel Extension loading and configuration for WPARs” on page 36
- ▶ 3.2, “WPAR list of features” on page 41
- ▶ 3.3, “Versioned Workload Partitions (WPAR)” on page 41
- ▶ 3.4, “Devices support in WPAR” on page 57
- ▶ 3.5, “WPAR RAS enhancements” on page 83
- ▶ 3.6, “WPAR Migration to AIX Version 7.1” on page 86

## 3.1 Trusted Kernel Extension loading and configuration for WPARs

This functionality allows the global administrator to select a set of kernel extensions that can then be loaded from within a system WPAR.

By default dynamic loading of a kernel extension in a WPAR returns a message:

```
sysconfig(SYS_KLOAD): Permission denied
```

In the following examples, Global> will refer to the prompt for a command issued in the Global instance of AIX. # will be the prompt inside the WPAR.

### Brief syntax overview

As user, functionality correspond to new flag -X for **mkwpar** and **chwp** commands. Multiple -X flags can be specified to load multiple kernel extensions.

The syntax described in man pages for the commands is as follows:

```
-X [exportfile=/path/to/file | [kext=[/path/to/extension|ALL]]
    [local=yes | no] [major=yes | no]
```

where specification can be direct (using kext=) or through a stanza (exportfile=)

And it can be private to WPAR or shared with Global.

To remove a explicit entry for an exported kernel extension you will use:

```
chwp -K -X [kext=/path/to/extension|ALL] wparname
```

**Note:** If the kernel extension is loaded inside a workload partition, the kernel extension will not be unloaded from the Global until the WPAR is stopped or rebooted. A restart of the workload partition will be required to completely unexport the kernel extension from the workload partition.

The kext path specification must match a value inside the workload partition's configuration file. This must either be a fully qualified path or ALL if the kext=ALL had previously been used.

### Simple example monitoring

When inquiring about kernel extension loading, the following reference, on the IBM developerWorks website, provides a good example. Please refer to *Writing AIX kernel extensions* at the following location:

<http://www.ibm.com/developerworks/aix/library/au-kernelext.html>

Going through that example with a default WPAR creation would result in the following output:

*Example 3-1 Creation of a simple WPAR*

---

```
Global> mkwpar -n testwpar
mkwpar: Creating file systems...
/
/home
/opt
/proc
/tmp
/usr
/var
.....
Mounting all workload partition file systems.
x ./usr
syncroot: Processing root part installation status.
syncroot: Installp root packages are currently synchronized.
syncroot: RPM root packages are currently synchronized.
syncroot: Root part is currently synchronized.
syncroot: Returns Status = SUCCESS
Workload partition testwpar created successfully.
mkwpar: 0960-390 To start the workload partition, execute the following
as root: startwpar [-v] testwpar
```

```
Global> startwpar testwpar
Starting workload partition testwpar.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.
Exporting workload partition kernel extensions.
Starting workload partition subsystem cor_test.
0513-059 The cor_test Subsystem has been started. Subsystem PID is
7340192.
Verifying workload partition startup.
```

---

When the WPAR is created we want to access to it and see if we can load a kernel extension.

*Example 3-2 Trying to load a kernel extension in a simple WPAR*

---

```
Global> clogin testwpar
*****
*
* Welcome to AIX Version 7.1!
```

```

* *
* Please see the README file in /usr/lpp/bos for information pertinent
to *
* this release of the AIX Operating System.
*
* *
*****
# ls
Makefile      hello_world.kex  loadkernext.o   sample.log
README        hello_world.o    main
hello_world.c loadkernext      main.c
hello_world.exp loadkernext.c    main.o
# ./loadkernext -q hello_world.kex
Kernel extensionKernel extension is not present on system.
# ./loadkernext -l hello_world.kex
sysconfig(SYS_KLOAD): Permission denied

```

---

As expected we are unable to load the kernel extension.

The aim is to create a new system WPAR with the kernel extension parameter as shown in Example 3-3 using the -X parameter of mkwpar. We verify the existence of the kernel extension in the Global instance.

---

*Example 3-3 Successful loading of kernel extension*

---

```

Global> mkwpar -X kext=/usr/src/kernext/hello_world.kex local=yes -n testwpar2
mkwpar: Creating file systems...
/
/home
/opt
/proc
....
syncroot: Processing root part installation status.
syncroot: Installp root packages are currently synchronized.
syncroot: RPM root packages are currently synchronized.
syncroot: Root part is currently synchronized.
syncroot: Returns Status = SUCCESS
Workload partition testwpar2 created successfully.
mkwpar: 0960-390 To start the workload partition, execute the following as
root: startwpar [-v] testwpar2

Global> cd /usr/src/kernext
Global> ./loadkernext -q hello_world.kex
Kernel extensionKernel extension is not present on system.
Global> ./loadkernext -l hello_world.kex
Kernel extension kmid is 0x50aa2000.
Global> genkex | grep hello

```

```
f1000000c0376000    2000 hello_world.kex
Global> ls
Makefile           hello_world.kex  loadkernext.o    sample.log
README            hello_world.o    main
hello_world.c      loadkernext      main.c
hello_world.exp    loadkernext.c    main.o
Global> cat sample.log
Hello AIX World!
```

---

The **loadkernext -q** command is querying state of the module. The **-l** option is loading the module and if it is successful, it returns the **kmid** value. The **genkex** command also confirms that the kernel extension is loaded on the system. The loaded module will write output to **sample.log** file in the current working directory.

### Enhancement of lswpar command

**lswpar** command has also been enhanced with the flag **X** to list detailed kernel extension information for each requested workload partition in turn.

*Example 3-4 Parameter -X of lswpar command*

---

```
Global> lswpar -X
lswpar: 0960-679 testwpar2 has no kernel extension configuration.
Name  EXTENSION NAME                               Local Major Status
-----
test2  /usr/src/kernext/hello_world.kex  yes  no  ALLOCATED
```

---

### mkwpar -X local=yesno parameter impact

Since we specified the parameter **local=yes** in the previous example (Example 3-3 on page 38), the **GLOBAL** instance does not see that kernel extension - it is private to the **WPAR** called **testwpar2**. The following query command in Example 3-5 shows it is not running on system.

*Example 3-5 Loading kernel extension*

---

```
Global> uname -a
AIX Global 1 7 00F61AA64C00
Global> cd /usr/src/kernext
Global> ./loadkernext -q hello_world.kex
Kernel extension is not present on system.
```

---

A change of that parameter to **local=no** will share the extension with the global as demonstrated in the following output Example 3-6 on page 40

*Example 3-6 Changing type of kernel extension and impact to Global.*

---

```
Global> chwpar -X local=no kext=/usr/src/kernext/hello_world.kex
testwpar2
Global> lswpar -X
lswpar: 0960-679 testwpar2 has no kernel extension configuration.
Name  EXTENSION NAME                               Local Major Status
-----
test2 /usr/src/kernext/hello_world.kex  no    no    ALLOCATED

Global> startwpar testwpar2
Starting workload partition testwpar2.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.
Exporting workload partition kernel extensions.
Starting workload partition subsystem cor_test2.
0513-059 The cor_test2 Subsystem has been started. Subsystem PID is
10879048.
Verifying workload partition startup.
Global> pwd
/usr/src/kernext
Global> ./loadkernext -q hello_world.kex
Kernel extension is not available on system.
```

---

The last command is verifying that it is allocated but not yet used.

But when we make use of it within the WPAR, it is available both in the WPAR and in the Global. Note that the kmid is coherent in both environments.

---

```
Global> clogin testwpar2
*****

* Welcome to AIX Version 7.1!

* Please see the README file in /usr/lpp/bos for information pertinent
to *
* this release of the AIX Operating System.
*
* *
*****

Last login: Wed Aug 25 18:38:28 EDT 2010 on /dev/Global from 7501lp01

# cd /usr/src/kernext
# ./loadkernext -q hello_world.kex
```

```

Kernel extension is not present on system.
# ./loadkernext -l hello_world.kex
Kernel extension kmid is 0x50aa3000.

# exit
Global> uname -a
AIX 75011p01 1 7 00F61AA64C00
Global> ./loadkernext -q hello_world.kex
Kernel extension is there with kmid 1353330688 (0x50aa3000).
Global> genkex | grep hello
f1000000c0378000      2000 hello_world.kex

```

---

**Note:** The mkwpar -X command has updated the config file called /etc/wpars/test2.cf with a new entry related to that kernel extension:

```

extension:
    checksum =
"4705b22f16437c92d9cd70babe8f6961e38a64dc222aaba33b8f5c9f4975981a:12
82772343"
    kext = "/usr/src/kernext/hello_world.kex"
    local = "no"
    major = "no"

```

Unload of kernel extension on one side would result as appearing to be unloaded from both side

## 3.2 WPAR list of features

With AIX 6.1 TL4 it was introduced the capability to create a WPAR with its root file systems on a storage device dedicated to that WPAR - that is called rootvg WPAR. With AIX 6.1 TL6 was introduced the capability to have VIOS based VSCSI disks in a WPAR. With AIX 7.1, the support of kernel extension load and VIOS disks and their management within a WPAR, allows to have a rootvg WPAR that supports VIOS disks.

## 3.3 Versioned Workload Partitions (WPAR)

A new product called Versioned Workload Partitions (WPAR) support the installation of a legacy version inside a system WPAR. Applications running a versioned WPAR will interact with the legacy AIX environment, that will be different from the Global environment.

All the features mentioned in 3.2, “WPAR list of features” on page 41 are supported within a Versioned WPAR.

This topic describes the support of that versioned WPAR support with a runtime environment of level AIX 5.2 in an AIX 7.1 WPAR. Runtime environment means commands, libraries and kernel interface semantics.

The example will refer to a Global> prompt when issued from the Global AIX instance. The # prompt is issued from within a Versioned WPAR.

### 3.3.1 Benefits of that feature

That capability to run an AIX 5.2 environment inside an AIX 7.1 WPAR is justified through the following needs and advantages:

- ▶ Ability to run a old version of AIX (currently AIX 5.2) on a new hardware (P7)
- ▶ Ability to extend service life for that old version of AIX
- ▶ Ability for users to run AIX 5.2 binary applications on new hardware without recompiling. Easy validation path.

### 3.3.2 Current requirements and restrictions

That Versioned WPAR product which is running an old OS on a new hardware in some transparent ways required some work with the compatibility of the operating systems. Here is a list of considerations.

**Note:** Versioned WPAR is an optional separate product (LPP) that runs on top of AIX 7.1

The requirements are as follows:

- ▶ The customer AIX 5.2 system to be integrated in the Versioned WPAR must run the final service pack called TL10 SP8 or 5200-10-08.

**Note:** The AIX 5.2 environment is not provided with the LPP.

- ▶ The product will only be supported on Power 7 hardware
- ▶ NFS server is not supported in a Versioned WPAR
- ▶ Device support within the Versioned WPAR is limited to devices directly supported in a AIX 7.1 WPAR
- ▶ No PowerHA support within a Versioned WPAR



- ▶ Versioned WPAR needs to be private meaning that /usr and /opt can't be shared with Global.
- ▶ Some commands and libraries from the AIX 5.2 environment that have extensive dependencies on data from the kernel extensions are replaced with the corresponding 7.1 command or library.
- ▶ Some additional software have to be installed into the Versioned WPAR.

Some other limitations have to be considered by user:

- ▶ When a Kernel extension is loaded in a WPAR 7.1 it is flagged as a private module <<see chapter WPAR Kernel Extension ...>>. On the Global side, user may see multiple instances of same module even if it is not used.
- ▶ These extensions can't be used to share data between WPARs.
- ▶ Versioned WPARs get support for /dev/[k]mem but it is limited to around 25 symbols only (the symbols being used in AIX 5.2). There is no access to other symbols.
- ▶ The device driver files, their associated configuration methods and ODM data must match the version of the corresponding files and data in the Global.

### 3.3.3 Creation of a basic Versioned WPAR AIX 5.2

Creation of a Versioned WPAR requires few steps being run in following example. These steps requires

- ▶ Creating a AIX 5.2 mksysb image
- ▶ Installing the support images for Versioned WPAR
- ▶ Creating of the Versioned WPAR
- ▶ Starting the WPAR and its management

#### **mksysb image**

From a running system AIX 5.2, user is responsible to create an mksysb image using the **mksysb** command. This can be available as a file, a disk, a CD or DVD or on tape.

As most of the AIX 5.2 systems used to have one root JFS file system, migration to current layout will be handled at time of WPAR creation. JFS file systems will also be restored as JFS2 file systems as rootvg WPAR do not support JFS.

In our example, we have an AIX 5.2 TL10 SP8 mksysb image file.

## Install the required LPP for Versioned WPAR support.

In order to instal the appropriate LPPs in a Versioned WPAR during the WPAR creation, you need to have the following packages available in `/usr/sys/inst.images`:

- ▶ `bos.wpars`
- ▶ `wio.common`
- ▶ `vwpar.52`

On the installation media DVD, the LPP packages to install with `installp` command are called `vwpar.images.52` and `vwpar.images.base`.

If you don't have the required packages installed you will receive a message stating that some software is missing, as shown in Example 3-7.

### *Example 3-7 Missing vwpar packages installation message*

---

```
Global> mkwpar -C -B mksysb52_TL10_SP8 -n vers_wpar1
mkwpar: 0960-669 Directory /usr/sys/inst.images does not contain the
software required to create a versioned workload partition.
```

---

**Note:** If you did a manual copy of the packages you need to execute the command `inutoc` to update the catalog file `.toc` to include the packages you just added.

## Creating a basic Versioned WPAR

The command to create a system WPAR is called `mkwpar`. It has been enhanced to support the creation of a versioned WPAR. The command flags relating to the creation of a versioned WPAR are:

```
/usr/sbin/mkwpar ... [-C] [-E directory] [-B wparbackupdevice] [-D ...
xfactor=n]
```

- ▶ `-C`: Specify Versioned WPAR creation. This option is valid only when additional versioned workload partition software has been installed
- ▶ `-B`: Specifies the 5.2 mksysb image to be used to populate the WPAR.
- ▶ `-D .... xfactor=n`
- ▶ `-E directory`: Directory which contains the filesets required to install the Versioned WPAR. The directory specification is optional as it is allowing an alternative location directory in place of `/usr/sys/inst.images` to be specified.

In order to provide the mandatory process of:

- ▶ Population of the WPAR file systems from the mksysb image.

- ▶ Since all JFS file systems will be restored as JFS2 file systems, and JFS2 doesn't support compression, the file system size may be no longer sufficient to hold the data. The new attribute *xfactor* of the -D option allows the administrator to control the expansion of the file system. The default value is 1 and the maximum value is 8.

### ***Other processes from mkwpar command***

For a Versioned WPAR, the `mkwpar` command will check that the `/usr` and `/opt` file systems from the global are in the mount list for the WPAR at `/nre/usr` and `/nre/opt` respectively.

### ***Simple Versioned WPAR creation output using a mksysb image file***

The initial command using a mksysb image file called `mksysb52_TL10_SP8` would be

```
mkwpar -C -B mksysb52_TL10_SP8 -n vers_wpar1
```

and the output

#### ***Example 3-8 Simple Versioned WPAR creation***

---

```
Global> /usr/sbin/mkwpar -C -B mksysb52_TL10_SP8 -n vers_wpar1
Extracting file system information from backup...
mkwpar: Creating file systems...
/
Creating file system '/' specified in image.data
/home
Creating file system '/home' specified in image.data
/opt
Creating file system '/opt' specified in image.data
/proc
/tmp
Creating file system '/tmp' specified in image.data
/usr
Creating file system '/usr' specified in image.data
/var
Creating file system '/var' specified in image.data
Mounting all workload partition file systems.
New volume on /var/tmp/mksysb52_TL10_SP8:
Cluster size is 51200 bytes (100 blocks).
The volume number is 1.
The backup date is: Tue Jun  8 12:57:43 EDT 2010
Files are backed up by name.
The user is root.
x      5473 ./bosinst.data
x      8189 ./image.data
x     133973 ./tmp/vgdata/rootvg/backup.data
x          0 ./home
```

```

x          0 ./home/lost+found
x          0 ./opt
x          0 ./opt/IBMinvscout
x          0 ./opt/IBMinvscout/bin
x          2428 ./opt/IBMinvscout/bin/invscoutClient_PartitionID
x          11781523 ./opt/IBMinvscout/bin/invscoutClient_VPD_Survey
x          0 ./opt/LicenseUseManagement

```

```

.....
The total size is 1168906634 bytes.
The number of restored files is 28807.

```

```

+-----+
                          Pre-installation Verification...
+-----+

```

```

Verifying selections...done
Verifying requisites...done
Results...

```

#### SUCSESSES

```
-----
```

Filesets listed in this section passed pre-installation verification and will be installed.

#### Selected Filesets

```
-----
```

<b>bos.wpars 7.1.0.1</b>	<b># AIX Workload Partitions</b>
<b>wvpar.52.rte 1.1.0.0</b>	<b># AIX 5.2 Versioned WPAR Runti...</b>
<b>wio.common 6.1.3.0</b>	<b># Common I/O Support for Workl...</b>

<< End of Success Section >>

#### FILESET STATISTICS

```
-----
```

```

  3 Selected to be installed, of which:
    3 Passed pre-installation verification

```

```
----
```

```

  3 Total to be installed

```

```

+-----+
                          Installing Software...
+-----+

```

```

installp: APPLYING software for:
          bos.wpars 7.1.0.1

```

```
.....
```

```

+-----+
                          Summaries:
+-----+

```

Installation Summary

```

-----
Name                               Level      Part      Event     Result
-----
bos.wpars                          7.1.0.1   USR       APPLY     SUCCESS
bos.wpars                          7.1.0.1   ROOT     APPLY     SUCCESS
wio.common                         6.1.3.0   USR       APPLY     SUCCESS
wio.common                         6.1.3.0   ROOT     APPLY     SUCCESS
vwpar.52.rte                       1.1.0.0   USR       APPLY     SUCCESS
vwpar.52.rte                       1.1.0.0   ROOT     APPLY     SUCCESS

```

**Workload partition vers\_wpar1 created successfully.**

mkwpar: 0960-390 To start the workload partition, execute the following as  
 root: startwpar [-v] vers\_wpar1

### ***Listing information about Versioned WPAR in the system***

A new parameter L has been added to **lswpar -t** option command to list Versioned WPARs.

The example Example 3-9 shows the difference between the simple **lswpar** and the **lswpar -t L** commands

#### *Example 3-9 .Lswpar queries*

```

Global> lswpar
Name      State  Type  Hostname  Directory      RootVG WPAR
-----
vers_wpar1 D      S     vers_wpar1 /wpars/vers_wpar1 no
wpar1     D      S     wpar1     /wpars/wpar1   no

Global> lswpar -t L
Name      State  Type  Hostname  Directory      RootVG WPAR
-----
vers_wpar1 D      S     vers_wpar1 /wpars/vers_wpar1 no

```

In the next **lswpar** queries (Example 3-10) we get the WPAR configuration and see that there is kernel extension (-X query) and no special device allocated to the WPAR (-D query). The last query with **lswpar -M** shows that the WPAR file systems have been allocated in the global system rootvg disk.

#### *Example 3-10 Multiple lswpar queries over Versioned WPAR*

```

Global> lswpar -X vers_wpar1
lswpar: 0960-679 vers_wpar1 has no kernel extension configuration.

Global> lswpar -D vers_wpar1
Name      Device Name  Type  Virtual Device  RootVG  Status
-----
vers_wpar1 /dev/null    pseudo
vers_wpar1 /dev/tty     pseudo

```

```

vers_wpar1 /dev/console    pseudo    ALLOCATED
vers_wpar1 /dev/zero                pseudo    ALLOCATED
vers_wpar1 /dev/clone        pseudo    ALLOCATED
vers_wpar1 /dev/sad          clone     ALLOCATED
vers_wpar1 /dev/xti/tcp      clone     ALLOCATED
vers_wpar1 /dev/xti/tcp6     clone     ALLOCATED
vers_wpar1 /dev/xti/udp      clone     ALLOCATED
vers_wpar1 /dev/xti/udp6     clone     ALLOCATED
vers_wpar1 /dev/xti/unixdg   clone     ALLOCATED
vers_wpar1 /dev/xti/unixst   clone     ALLOCATED
vers_wpar1 /dev/error        pseudo    ALLOCATED
vers_wpar1 /dev/errorctl     pseudo    ALLOCATED
vers_wpar1 /dev/audit        pseudo    ALLOCATED
vers_wpar1 /dev/nvram        pseudo    ALLOCATED
vers_wpar1 /dev/kmem         pseudo    ALLOCATED

```

```
Global> lswpar -M vers_wpar1
```

Name	MountPoint	Device	Vfs	Nodename	Options
vers_wpar1	/wpars/vers_wpar1	/dev/fs1v00	jfs2		
vers_wpar1	/wpars/vers_wpar1/home	/dev/lv01	jfs		
vers_wpar1	/wpars/vers_wpar1/nre/opt	/opt	namefs		ro
vers_wpar1	/wpars/vers_wpar1/nre/sbin	/sbin	namefs		ro
vers_wpar1	/wpars/vers_wpar1/nre/usr	/usr	namefs		ro
vers_wpar1	/wpars/vers_wpar1/opt	/dev/fs1v01	jfs2		
vers_wpar1	/wpars/vers_wpar1/proc	/proc	namefs		rw
vers_wpar1	/wpars/vers_wpar1/tmp	/dev/fs1v02	jfs2		
vers_wpar1	/wpars/vers_wpar1/usr	/dev/fs1v03	jfs2		
vers_wpar1	/wpars/vers_wpar1/var	/dev/fs1v05	jfs2		

```
Global> lsvg -l rootvg | grep vers
```

fs1v00	jfs2	1	1	1	closed/syncd	/wpars/vers_wpar1
lv01	jfs	1	1	1	closed/syncd	/wpars/vers_wpar1/home
fs1v01	jfs2	1	1	1	closed/syncd	/wpars/vers_wpar1/opt
fs1v02	jfs2	1	1	1	closed/syncd	/wpars/vers_wpar1/tmp
fs1v03	jfs2	18	18	1	closed/syncd	/wpars/vers_wpar1/usr
fs1v05	jfs2	1	1	1	closed/syncd	/wpars/vers_wpar1/var

## startwpar

The `startwpar` command gives a standard output, except that a message is displayed stating that the WPAR is not configured as checkpointable (file systems on Global root disk).

*Example 3-11 startwpar of a Versioned WPAR*

```
Global> startwpar vers_wpar1
```

```
Starting workload partition vers_wpar1.
```

```
Mounting all workload partition file systems.
```

```

Loading workload partition.
Exporting workload partition devices.
Exporting workload partition kernel extensions.
Starting workload partition subsystem cor_vers_wpar1.
0513-059 The cor_vers_wpar1 Subsystem has been started. Subsystem PID is
10289366.
startwpar: 0960-239 The workload partition vers_wpar1 is not configured to be
checkpointable.
Verifying workload partition startup.

```

---

## Accessing a Versioned WPAR

To access a WPAR we need to define the WPAR with an address and connect using `telnet` or `ssh` tools.

However for some administrative commands you can use the `cllogin` console access.

**Note:** The `cllogin` process is not part of the WPAR and prevents WPAR mobility.

Within the WPAR we can list the file systems mounted as well as list the drivers loaded in a Versioned WPAR.

### *Example 3-12 Commands within a Versioned WPAR*

---

```

Global> cllogin vers_wpar1
*****
*                                                                 *
*                                                                 *
* Welcome to AIX Version 5.2!                                     *
*                                                                 *
*                                                                 *
* Please see the README file in /usr/lpp/bos for information pertinent to *
* this release of the AIX Operating System.                       *
*                                                                 *
*                                                                 *
*****
Last unsuccessful login: Tue Apr 13 12:35:04 2010 on /dev/pts/1 from
p-eye.austin.ibm.com
Last login: Tue Jun  8 11:53:53 2010 on /dev/pts/0 from varnae.austin.ibm.com
# uname -a
AIX vers_wpar1 2 5 00F61AA64C00
# df
Filesystem      512-blocks      Free %Used    Iused %Iused Mounted on
Global          131072          106664   19%      1754   13% /
Global          131072          126872    4%        17    1% /home

```

<b>Global</b>	<b>786432</b>	<b>402872</b>	<b>49%</b>	<b>7044</b>	<b>14% /nre/opt</b>
<b>Global</b>	<b>1572864</b>	<b>1158728</b>	<b>27%</b>	<b>10137</b>	<b>8% /nre/sbin</b>
<b>Global</b>	<b>4849664</b>	<b>1184728</b>	<b>76%</b>	<b>41770</b>	<b>24% /nre/usr</b>
Global	131072	35136	74%	778	16% /opt
Global	-	-	-	-	- /proc
Global	131072	126520	4%	22	1% /tmp
Global	2359296	133624	95%	25300	59% /usr
Global	131072	111368	16%	350	3% /var

```

# lsdev
aio0      Available Asynchronous I/O (Legacy)
inet0     Defined   Internet Network Extension
posix_aio0 Available Posix Asynchronous I/O
pty0      Available Asynchronous Pseudo-Terminal
sys0      Available System Object
wio0      Available WPAR I/O Subsystem

```

---

WPAR reports it is running a AIX 5.2 system.

Its **hostname** has been modified to be the WPAR name.

AIX 7.1 binaries are found under /nre/opt, /nre/sbin, /nre/usr file systems.

The **lsdev** command reports the available devices in the Versioned WPAR. They are the ones expected to be in AIX 7.1 WPAR (see <<WPAR devices management chapter>>).

## Use of /nre commands in a Versioned WPAR

From the previous **df** display, some commands are available in the directory /nre/usr/bin. These are the AIX 7.1 binaries and the following listing displays the result of using them in a Versioned WPAR. The AIX 5.2 commands are located in /usr (for our example).

*Example 3-13 Execution of a AIX 7.1 binary command in a Versioned WPAR.*

---

```

# /nre/usr/bin/who
Could not load program /nre/usr/bin/who:
Symbol resolution failed for who because:
    Symbol __strcmp (number 3) is not exported from dependent
    module /usr/lib/libc.a(shr.o).
    Symbol __strcpy (number 5) is not exported from dependent
    module /usr/lib/libc.a(shr.o).
Examine .loader section symbols with the 'dump -Tv' command.

# /usr/bin/who
root      Global      Sep  2 15:48      (Global)

```

---



**Note:** You should not attempt to execute the AIX 7.1 commands under /nre directly.

### 3.3.4 Creation of an AIX Version 5.2 rootvg WPAR

As rootvg WPARs reside on a rootvg disk exported to the WPAR which is distinct from the global system rootvg, it must be specified in the mkwpar command under the option -D

The simplest mkwpar command to create a rootvg Versioned WPAR is:

```
mkwpar -D devname=hdisk? rootvg=yes [xfactor=[1-8]] [-O] -C -B
<mksysb_device] -n VersionedWPARname
```

- ▶ Multiple -D option can be specified if multiple disks have to be exported
- ▶ the rootvg=yes specification means these disks will be part of the WPAR rootvg disk. Other disks will be just exported to the WPAR.
- ▶ -O: overwrite the existing volume group data on the disk, or create one.
- ▶ xfactor parameter has been described in “Creating a basic Versioned WPAR” on page 44

**Note:** The storage devices exportable to a Version WPAR are devices which can be exported to a AIX 7.1 WPAR and that includes the devices not supported by standalone AIX 5.2

My example output using hdisk4 using the mksysb image called mksysb52\_TL10\_SP8 is listed below in Example 3-14. The device name being used called hdisk4 is a disk without any volume group. Therefore there is no need to specify the -O (override) option to the mkwpar command.:

#### *Example 3-14 rootvg Versioned WPAR creation*

---

```
Global> mkwpar -C -B mksysb52_TL10_SP8 -n vers_wpar2 -D devname=hdisk4
rootvg=yes <
Extracting file system information from backup...
Creating workload partition's rootvg. Please wait...
mkwpar: Creating file systems...
/
Creating file system '/' specified in image.data
/admin
/home
Converting JFS to JFS2
Creating file system '/home' specified in image.data
```

```

/opt
Creating file system '/opt' specified in image.data
/proc
/tmp
Creating file system '/tmp' specified in image.data
/usr
Creating file system '/usr' specified in image.data
/var
Creating file system '/var' specified in image.data
Mounting all workload partition file systems.
New volume on /var/tmp/mksysb52_TL10_SP8:
Cluster size is 51200 bytes (100 blocks).
The volume number is 1.
The backup date is: Tue Jun  8 12:57:43 EDT 2010
Files are backed up by name.
The user is root.
x          5473 ./bosinst.data
x          8189 ./image.data
x        133973 ./tmp/vgdata/rootvg/backup.data
x           0 ./home
x           0 ./home/lost+found
x           0 ./opt
x           0 ./opt/IBMinvsout

```

```

.....

```

```

+-----+
+-----+
Summaries:
+-----+
+-----+

```

#### Installation Summary

Name	Level	Part	Event	Result
bos.net.nis.client	7.1.0.0	ROOT	APPLY	SUCCESS
bos.perf.libperfstat	7.1.0.0	ROOT	APPLY	SUCCESS
bos.perf.perfstat	7.1.0.0	ROOT	APPLY	SUCCESS
bos.perf.tools	7.1.0.0	ROOT	APPLY	SUCCESS
bos.sysmgmt.trace	7.1.0.0	ROOT	APPLY	SUCCESS
clic.rte.kernext	4.7.0.0	ROOT	APPLY	SUCCESS
devices.chrp.base.rte	7.1.0.0	ROOT	APPLY	SUCCESS
devices.chrp.pci.rte	7.1.0.0	ROOT	APPLY	SUCCESS
devices.chrp.vdevice.rte	7.1.0.0	ROOT	APPLY	SUCCESS
devices.common.IBM.ethernet	7.1.0.0	ROOT	APPLY	SUCCESS
devices.common.IBM.fc.rte	7.1.0.0	ROOT	APPLY	SUCCESS
devices.common.IBM.mpio.rte	7.1.0.0	ROOT	APPLY	SUCCESS
devices.common.IBM.scsi.rte	7.1.0.0	ROOT	APPLY	SUCCESS
devices.fcp.disk.array.rte	7.1.0.0	ROOT	APPLY	SUCCESS
devices.fcp.disk.rte	7.1.0.0	ROOT	APPLY	SUCCESS
devices.fcp.tape.rte	7.1.0.0	ROOT	APPLY	SUCCESS
devices.scsi.disk.rte	7.1.0.0	ROOT	APPLY	SUCCESS

```

devices.tty.rte          7.1.0.0      ROOT      APPLY     SUCCESS
bos.mp64                 7.1.0.0      ROOT      APPLY     SUCCESS
bos.net.tcp.client      7.1.0.0      ROOT      APPLY     SUCCESS
bos.perf.tune           7.1.0.0      ROOT      APPLY     SUCCESS
perfagent.tools         7.1.0.0      ROOT      APPLY     SUCCESS
bos.net.nfs.client      7.1.0.0      ROOT      APPLY     SUCCESS
bos.wpars               7.1.0.0      ROOT      APPLY     SUCCESS
bos.net.ncs             7.1.0.0      ROOT      APPLY     SUCCESS
wio.common              7.1.0.0      ROOT      APPLY     SUCCESS

```

Finished populating scratch file systems.

**Workload partition vers\_wpar2 created successfully.**

mkwpar: 0960-390 To start the workload partition, execute the following as root: startwpar [-v] vers\_wpar2

When the Versioned WPAR is created, the hdisk4 is allocated to the WPAR and it is the rootvg disk for that WPAR. The Example 3-15 shows that file system layout of a rootvg Versioned WPAR is different from the layout of a non-rootvg Versioned WPAR described in Example 3-10 on page 47.

*Example 3-15 Rootvg Versioned WPAR file system layout.*

```

Global> lswpar -D | grep disk
vers_wpar2  hdisk4          disk          yes          ALLOCATED
Global> lswpar -M vers_wpar2
Name      MountPoint              Device      Vfs      Nodename  Options
-----
vers_wpar2 /wpars/vers_wpar2      /dev/fs1v10 jfs2
vers_wpar2 /wpars/vers_wpar2/etc/objrepos/wboot /dev/fs1v11 jfs2
vers_wpar2 /wpars/vers_wpar2/opt   /opt       namefs    ro
vers_wpar2 /wpars/vers_wpar2/usr   /usr       namefs    ro

```

For our rootvg Versioned WPAR, two file systems called /dev/fs1v010 and /dev/fs1v11 which will be used to bootstrap the WPAR have been created. They are located on the global rootvg disk.

## Startwpar of a rootvg Versioned WPAR

For a rootvg Versioned WPAR, a minimal file system set is created in the Global's rootvg and is used to bootstrap the WPAR and synchronize device information between the WPAR and the global. They are mounted as / and /etc/objrepos/wboot at start of WPAR. Then they are overmounted with the WPAR file systems.

*Example 3-16 Startwpar of a rootvg Versioned WPAR*

```

Global> startwpar vers_wpar2
Starting workload partition vers_wpar2.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.

```

**hdisk4** Defined

Exporting workload partition kernel extensions.

Starting workload partition subsystem cor\_vers\_wpar2.

0513-059 The cor\_vers\_wpar2 Subsystem has been started. Subsystem PID is 4456646.

startwpar: 0960-239 The workload partition vers\_wpar2 is not configured to be checkpointable.

Verifying workload partition startup.

**Device information queries from a rootvg Versioned WPAR**

The rootvg Versioned WPAR has all the standard file systems mounted from its own rootvg, plus read-only namefs mounts from the Global. These namefs are the native runtime environment file systems called `/nre/usr`, `/nre/opt` and `/nre/sbin`. There is also a root file system mounted from the Global to bootstrap the WPAR (see Example 3-15 on page 53) and a `/etc/objrepos/wboot` mount that is used to synchronize device information between the WPAR and the Global. The layout is displayed using the `df` command in Example 3-17

*Example 3-17 Devices and file systems in a rootvg Versioned WPAR.*

```
Global> clogin vers_wpar2
*****
*
*
* Welcome to AIX Version 5.2!
*
*
* Please see the README file in /usr/lpp/bos for information pertinent to
* this release of the AIX Operating System.
*
*
*****
Last unsuccessful login: Tue Apr 13 12:35:04 2010 on /dev/pts/1 from p-eye.austin.ibm.com
Last login: Tue Jun  8 11:53:53 2010 on /dev/pts/0 from varnae.austin.ibm.com

# uname -a
AIX vers_wpar2 2 5 00F61AA64C00
# df
Filesystem      512-blocks      Free %Used    Iused %Iused Mounted on
Global          131072          104472  21%      1795   14% /
/dev/hd4        131072          104472  21%      1795   14% /
Global         4849664         1184728  76%      41770  24% /nre/usr
Global          786432          402872  49%      7044   14% /nre/opt
Global         1572864         1158704  27%      10163   8% /nre/sbin
/dev/hd2        2359296         117536  96%      25300  62% /usr
/dev/hd10opt    131072          33088   75%      778    17% /opt
/dev/hd11admin  131072          128344   3%        4     1% /admin
/dev/hd1        131072          128344   3%        4     1% /home
/dev/hd3        131072          124472   6%       22     1% /tmp
/dev/hd9var     131072          109336  17%      350    3% /var
Global          131072          128336   3%        5     1% /etc/objrepos/wboot
Global          -                -         -         -     - /proc
# lsdev
fscsi0    Available 00-00-02 WPAR I/O Virtual Parent Device
hd1       Available          Logical volume
hd2       Available          Logical volume
hd3       Available          Logical volume
hd4       Available          Logical volume
```

hd10opt	Available	Logical volume
hd11admin	Available	Logical volume
hd9var	Available	Logical volume
<b>hdisk0</b>	<b>Available</b>	<b>00-00-02 MPIO Other DS4K Array Disk</b>
inet0	Defined	Internet Network Extension
pty0	Available	Asynchronous Pseudo-Terminal
rootvg	Available	Volume group
sys0	Available	System Object
wio0	Available	WPAR I/O Subsystem

---

### 3.3.5 Content of wpar.52 package

The wpar.52 package would install the following files in your WPAR.

*Example 3-18 wpar.52 lpp content*

---

```

Cluster size is 51200 bytes (100 blocks).
The volume number is 1.
The backup date is: Wed Aug 11 20:03:52 EDT 2010
Files are backed up by name.
The user is BUILD.
 0 ./
1063 ./lpp_name
 0 ./usr
 0 ./usr/lpp
 0 ./usr/lpp/wpar.52
189016 ./usr/lpp/wpar.52/liblpp.a
 0 ./usr/lpp/wpar.52/inst_root
1438 ./usr/lpp/wpar.52/inst_root/liblpp.a
 0 ./usr/aixnre
 0 ./usr/aixnre/5.2
 0 ./usr/aixnre/5.2/bin
8718 ./usr/aixnre/5.2/bin/timex
4446 ./usr/aixnre/5.2/bin/nrexec_wrapper
 0 ./usr/aixnre/5.2/ccs
 0 ./usr/aixnre/5.2/ccs/lib
 0 ./usr/aixnre/5.2/ccs/lib/perf
40848 ./usr/aixnre/5.2/ccs/lib/librtl.a
320949 ./usr/aixnre/5.2/ccs/lib/libwpardr.a
 0 ./usr/aixnre/5.2/lib
 0 ./usr/aixnre/5.2/lib/instl
186091 ./usr/aixnre/5.2/lib/instl/elib
 60279 ./usr/aixnre/5.2/lib/instl/instal
2008268 ./usr/aixnre/5.2/lib/liblvm.a
291727 ./usr/aixnre/5.2/lib/libperfstat.a
 1012 ./usr/aixnre/5.2/lib/perf/libperfstat_updt_dictionary
 0 ./usr/aixnre/bin

```

```

3524 ./usr/aixnre/bin/nre_exec
4430 ./usr/aixnre/bin/nrexec_wrapper
    0 ./usr/aixnre/diagnostics
    0 ./usr/aixnre/diagnostics/bin
939  ./usr/aixnre/diagnostics/bin/uspchrp
    0 ./usr/aixnre/lib
    0 ./usr/aixnre/lib/boot
    0 ./usr/aixnre/lib/boot/bin
1283 ./usr/aixnre/lib/boot/bin/bootinfo_chrp
1259 ./usr/aixnre/lib/boot/bin/lscfg_chrp
    0 ./usr/aixnre/lib/corrals
4446 ./usr/aixnre/lib/corrals/nrexec_wrapper
    0 ./usr/aixnre/lib/instl
4438 ./usr/aixnre/lib/instl/nrexec_wrapper
    0 ./usr/aixnre/lib/methods
4446 ./usr/aixnre/lib/methods/nrexec_wrapper
    0 ./usr/aixnre/lib/methods/wio
    0 ./usr/aixnre/lib/methods/wio/common
4470 ./usr/aixnre/lib/methods/wio/common/nrexec_wrapper
4430 ./usr/aixnre/lib/nrexec_wrapper
    0 ./usr/aixnre/lib/ras
4438 ./usr/aixnre/lib/ras/nrexec_wrapper
    0 ./usr/aixnre/lib/sa
4438 ./usr/aixnre/lib/sa/nrexec_wrapper
    0 ./usr/aixnre/objclass
3713 ./usr/aixnre/objclass/PCM.friend.vscsi.odmadd
    353 ./usr/aixnre/objclass/PCM.friend.vscsi.odmmdl
2084 ./usr/aixnre/objclass/adapter.vdevice.IBM.v-scsi.odmadd
    234 ./usr/aixnre/objclass/adapter.vdevice.IBM.v-scsi.odmmdl
6575 ./usr/aixnre/objclass/disk.vscsi.vdisk.odmadd
    207 ./usr/aixnre/objclass/disk.vscsi.vdisk.odmmdl
    0 ./usr/aixnre/pmapi
    0 ./usr/aixnre/pmapi/tools
4446 ./usr/aixnre/pmapi/tools/nrexec_wrapper
    0 ./usr/aixnre/sbin
4430 ./usr/aixnre/sbin/nrexec_wrapper
4508 ./usr/aixnre/sbin/nrexec_trace
4374 ./usr/aixnre/sbin/nrexec_no64
    0 ./usr/aixnre/sbin/helpers
4438 ./usr/aixnre/sbin/helpers/nrexec_wrapper
    0 ./usr/aixnre/sbin/helpers/jfs2
4446 ./usr/aixnre/sbin/helpers/jfs2/nrexec_wrapper
4544 ./usr/aixnre/sbin/helpers/jfs2/nrexec_mount
    0 ./usr/aixnre/sbin/perf
    0 ./usr/aixnre/sbin/perf/diag_tool

```

```
4462 ./usr/aixnre/sbin/perf/diag_tool/nrexec_wrapper
2526 ./usr/aixnre/sbin/stubout
6641 ./usr/ccs/lib/libcre.a
  0 ./usr/lib/corrals
37789 ./usr/lib/corrals/manage_overlays
4096 ./usr/lib/objrepos/overlay
4096 ./usr/lib/objrepos/overlay.vc
```

The total size is 3260358 bytes.  
The number of archived files is 78.

---

These are the files required to overlay 5.2 commands and libraries that have kernel data dependencies with a 7.1 version of the file.

### 3.3.6 SMIT INTERFACE

There is a new SMIT fastpath menus called `vwpar` for creating Versioned WPARs from `mksysb` images and from SPOTs. It is similar to the advance WPAR creation menu with new flags for the image to be loaded. it requires the `vwpar.sysmgt` package being installed.

## 3.4 Devices support in WPAR

In AIX 6.1 TL4 it was introduced the capability of creating a system WPAR with its root file systems on storage device(s) dedicated to the WPAR. Such a workload partition is referred to as a rootVG WPAR.

In AIX 6.1 TL 6 the support for VIOS-based VSCSI disks in a WPAR is being added.

SAN support for rootvg system WPAR released with AIX 6.1 TL 6 gave the support of individual devices (disk or tapes) in a WPAR.

The need to manage and support new devices like iSCSI and SAS required to have a common virtual device virtualization system within a WPAR. That is a new feature in AIX 7.1 called `wio` support in WPAR.

The result is that - without the action of a Global AIX instance system administrator - the administrator can manage the adapter as well as the storage devices attached to it. And there is no differences in syntax managing the device from the Global AIX instance or from the WPAR.

The controller example used will be the support of the fiber channel adapter introduced with AIX 7.1.

The following flow will detail user commands, behavior and outputs related to all these features. In the following, commands issued from the AIX Global instance are prefix with Global>. Commands issued from the WPAR are prefixed with the WPAR name (wpar2> for example). WPAR examples are named wpar1, wpar2 ...

**Note:** The fiber channel adapter can be either a physical or a virtual fiber channel adapter.

### 3.4.1 Global device listing used as example

Initially the test environment is running in a LPAR to which is attached a FC adapter with no disk.

From the Global we get the usual `lscfg` display in Example 3-19

*Example 3-19 Physical adapter available from Global*

---

```
Global> lscfg | grep fc
+fcs0          U5802.001.0086848-P1-C2-T1          8Gb PCI
Express Dual Port FC Adapter (df1000f114108a03)
* fcnet0      U5802.001.0086848-P1-C2-T1          Fibre
Channel Network Protocol Device
+ fcsio       U5802.001.0086848-P1-C2-T1          FC
SCSI I/O Controller Protocol Device
+ fcs1        U5802.001.0086848-P1-C2-T2          8Gb
PCI Express Dual Port FC Adapter (df1000f114108a03)
```

---

### 3.4.2 Device command listing in a AIX 7.1 WPAR

For our example, we created a single system WPAR using the `mkwpar -n wpar1` command which creates a WPAR with jfs2 file systems included in current Global rootvg volume. The Example 3-20 shows the output of the creation, the output of the `lswpar` queries for the file systems as well as a display of the global rootvg disk content.

*Example 3-20 Simple WPAR file system layout*

---

```
Global> mkwpar -n wpar1
.....
syncroot: Processing root part installation status.
syncroot: Installp root packages are currently synchronized.
syncroot: RPM root packages are currently synchronized.
```



```
syncroot: Root part is currently synchronized.
syncroot: Returns Status = SUCCESS
Workload partition wpar1 created successfully.
mkwpar: 0960-390 To start the workload partition, execute the following as
root: startwpar [-v] wpar1
```

```
Global> lswpar
```

Name	State	Type	Hostname	Directory	RootVG	WPAR
wpar1	D	S	wpar1	/wpars/wpar1	no	

```
Global> lswpar -M
```

Name	MountPoint	Device	Vfs	Nodename	Options
wpar1	/wpars/wpar1	/dev/fs1v00	<b>jfs2</b>		
wpar1	/wpars/wpar1/home	/dev/fs1v01	<b>jfs2</b>		
wpar1	/wpars/wpar1/opt	/opt	namefs		ro
wpar1	/wpars/wpar1/proc	/proc	namefs		rw
wpar1	/wpars/wpar1/tmp	/dev/fs1v02	<b>jfs2</b>		
wpar1	/wpars/wpar1/usr	/usr	namefs		ro
wpar1	/wpars/wpar1/var	/dev/fs1v03	<b>jfs2</b>		

```
Global> lsvg -l rootvg
```

```
rootvg:
LV NAME          TYPE      LPs    PPs    PVs  LV STATE  MOUNT POINT
hd5              boot      1      1      1    closed/syncd  N/A
hd6              paging    8      8      1    open/syncd   N/A
hd8              jfs2log   1      1      1    open/syncd   N/A
hd4              jfs2      4      4      1    open/syncd   /
hd2              jfs2      37     37     1    open/syncd   /usr
hd9var           jfs2      12     12     1    open/syncd   /var
hd3              jfs2      2      2      1    open/syncd   /tmp
hd1              jfs2      1      1      1    open/syncd   /home
hd10opt          jfs2      6      6      1    open/syncd   /opt
hd11admin        jfs2      2      2      1    open/syncd   /admin
lg_dump1v        sysdump   16     16     1    open/syncd   N/A
livedump         jfs2      4      4      1    open/syncd   /var/adm/ras/livedump
fs1v00           jfs2      2      2      1    closed/syncd /wpars/wpar1
fs1v01           jfs2      1      1      1    closed/syncd /wpars/wpar1/home
fs1v02           jfs2      2      2      1    closed/syncd /wpars/wpar1/tmp
fs1v03           jfs2      2      2      1    closed/syncd /wpars/wpar1/var
```

When we start the wpar (see Example 3-21) there is a mention of devices and kernel extensions loading.

#### *Example 3-21 Start of the WPAR*

```
Global> startwpar wpar1
Starting workload partition wpar1.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.
Exporting workload partition kernel extensions.
```

Starting workload partition subsystem cor\_wpar1.  
 0513-059 The cor\_wpar1 Subsystem has been started. Subsystem PID is  
 10158202.  
 Verifying workload partition startup.

---

In an AIX 7.1 system WPAR we can find a new entry in **lscfg** command output called WPAR I/O. This is the heart of the storage virtualization in a WPAR. This feature allows use of the usual AIX commands related to devices such as **lsdev**, **lscfg**, **cfgmgr**, **mkdev**, **rmdev**, **chdev**, **lsvpd**.

In the Example 3-22, we login in the system WPAR and issue that **lscfg** command which mentions the WPAR I/O subsystem entry.

*Example 3-22* lscfg display in a simple system WPAR

---

```
Global> clogin wpar1
*****
*
*
* Welcome to AIX Version 7.1!
*
*
* Please see the README file in /usr/lpp/bos for information pertinent to
* this release of the AIX Operating System.
*
*
*****
Last login: Tue Aug 31 15:27:43 EDT 2010 on /dev/Global from 75011p01

wpar1: /> lscfg
INSTALLED RESOURCE LIST

The following resources are installed on the machine.
+/- = Added or deleted from Resource List.
* = Diagnostic support not available.

Model Architecture: chrp
Model Implementation: Multiple Processor, PCI bus

+ sys0          System Object
* wio0         WPAR I/O Subsystem
```

---

The software packages being installed in the WPAR are as shown in Example 3-23 on page 61.

*Example 3-23 Packages related to wio installed in WPAR*


---

```
wpar1:/> ls|pp -L | grep wio
  wio.common          7.1.0.0   C    F    Common I/O Support for
  wio.fcp              7.1.0.0   C    F    FC I/O Support for Workload
  wio.vscsi            7.1.0.0   C    F    VSCSI I/O Support for
Workload
```

---

And when the specific package is installed, the subclass support is installed in /usr/lib/methods/wio. Support for fiber channel is called fcp and vscsi disk support is called vscsi as shown is the listing Example 3-4 on page 39.

*Example 3-24 Virtual device support abstraction layer*


---

```
wpar1:/> cd /usr/lib/methods/wio
wpar1:/> ls -R
common fcp      vscsi
./common:
cfg_wpar_vparent  cfgwio          defwio

./fcp:
configure      unconfigure

./vscsi:
configure      unconfigure
# file /usr/lib/methods/wio/common/defwio
/usr/lib/methods/wio/common/defwio: executable (RISC System/6000) or object
module
# /usr/lib/methods/wio/common/defwio
wio0
# lsdev | grep wio
wio0 Available WPAR I/O Subsystem
```

---

### 3.4.3 Dynamically adding a fiber channel adapter to a system WPAR

Following our environment example, dynamically adding a FC channel adapter to the WPAR will be done through the **chwp** **-D** option as shown in Example 3-25 on page 62. This **chwp** command is referred as an export processus, but is doesn't do the **cfgmgr** command to update the device listing.

The fiber channel adapter mentioned is the one found in Global as seen at Example 3-20 on page 58.

In that output Example 3-25 on page 62, we login in the WPAR and verify fiber channel information coherency comparing to Global.

*Example 3-25* Dynamically adding FC adapter to a running WPAR.

---

```

Global> chwpar -D devname=fcs0 wpar1
fcs0 Available
fscsi0 Available
fscsi0 Defined
line = 0
Global> lswpar -D wpar1
Name   Device Name      Type      Virtual Device  RootVG  Status
-----
wpar1  fcs0              adapter                   EXPORTED
wpar1  /dev/null          pseudo                 EXPORTED
....

Global> clogin wpar1
*****
*
* Welcome to AIX Version 7.1!
*
* Please see the README file in /usr/lpp/bos for information pertinent to   *
* this release of the AIX Operating System.                               *
*****
Last login: Thu Aug 26 14:33:49 EDT 2010 on /dev/Global from 75011p01

wpar1:/> lsdev
inet0 Defined      Internet Network Extension
pty0 Available    Asynchronous Pseudo-Terminal
sys0 Available    System Object
wio0 Available    WPAR I/O Subsystem

wpar1:/> fcstat fcs0
Error accessing ODM
Device not found
wpar1:/> lspath
wpar1:/> cfgmgr
wpar1:/> lspath
wpar1:/> fcstat fcs0
Error accessing ODM
VPD information not found

wpar1:/> lsdev
fcnet0 Defined    00-00-01 Fibre Channel Network Protocol Device
fcs0 Available 00-00 8Gb PCI Express Dual Port FC Adapter
fscsi0 Available 00-00-02 FC SCSI I/O Controller Protocol Device
inet0 Defined      Internet Network Extension
pty0 Available    Asynchronous Pseudo-Terminal
sys0 Available    System Object
wio0 Available    WPAR I/O Subsystem

```

---

**Note:** Dynamic allocation adapter to the WPAR requires a `cfgmgr` command update to update the ODM and make the new adapter and device available.

That dynamic allocation is referred as the export process to the WPAR

### 3.4.4 Change in config file related to that device addition

At that point the WPAR configuration file located in `/etc/wpars/wpar1.cf` has been updated with a new device entry listed in Example 3-26:

*Example 3-26 /etc/wpars/wpar1.cf entry update for device fcs0*

---

```
device:
    devname = "fcs0"
    devtype = "6"
```

---

### 3.4.5 lsdev output from Global

A new flag `-x` to the `lsdev` command allows printing of exported devices.

*Example 3-27 lsdev -x output*

---

```
Global> lsdev -x | grep -i export
fcs0      Exported 00-00-02    FC SCSI I/O Controller Protocol Device
```

---

### 3.4.6 Removing of fiber channel adapter from Global

When the fiber channel adapter is allocated to a running WPAR, that adapter is busy on the Global side and can't be removed.

*Example 3-28 rmdev failure for a busy device*

---

```
Global> rmdev -dl fcs0 -R
fcnet0 deleted
rmdev: 0514-552 Cannot perform the requested function because the
      fcs0 device is currently exported.
```

---

### 3.4.7 Reboot of LPAR keeps fiber channel allocation

From the previous state, reboot of the LPAR keeps the fiber channel allocation to the WPAR as shown in the Example 3-29 on page 64

*Example 3-29 Fiber channel adapter queries from Global after reboot*

```
Global> lscfg | grep fc
+ fcs0          U5802.001.0086848-P1-C2-T1          8Gb PCI
Express Dual Port FC Adapter (df1000f114108a03)
* fcnet0       U5802.001.0086848-P1-C2-T1          Fibre Channel
Network Protocol Device
+ fcsio        U5802.001.0086848-P1-C2-T1          FC SCSI I/O
Controller Protocol Device
+ fcs1         U5802.001.0086848-P1-C2-T2          8Gb PCI
Express Dual Port FC Adapter (df1000f114108a03)
* fcnet1       U5802.001.0086848-P1-C2-T2          Fibre Channel
Network Protocol Device
+ fcsi1        U5802.001.0086848-P1-C2-T2          FC SCSI I/O
Controller Protocol Device

Global> lswpar -Dq wpar1
wpar1 fcs0          adapter          ALLOCATED
wpar1 /dev/null      pseudo          ALLOCATED
....
Global> lswpar
Name State Type Hostname Directory RootVG WPAR
-----
wpar1 D S wpar1 /wpars/wpar1 no
```

Since the WPAR wpar1 is not started, we can now remove the fiber channel adapter from the Global. The result is seen in Example 3-30 and confirm that a WPAR can't start if it is missing some adapters.

*Example 3-30 Removal of the fiber channel adapter from the Global*

```
Global> rmdev -dl fcs0 -R
fcnet0 deleted
sfwcomm0 deleted
fcsio deleted
fcs0 deleted

Global> lswpar -D wpar1
Name Device Name Type Virtual Device RootVG Status
-----
wpar1 adapter MISSING
Global> startwpar wpar1
*****
ERROR
ckwpar: 0960-586 Attributes of fcs0 do not match those in ODM.

ERROR
ckwpar: 0960-587 fcs0 has un-supported subclass type.
```

\*\*\*\*\*

Removal of the adapter using the **chwp** command do clean the situation. **lswpar** command shows the device is not any more missing or allocated. And then WPAR can start.

*Example 3-31 Removal of missing device allows WPAR start*

---

```
Global> chwp -K -D devname=fcs0 wpar1
Global> lswpar -D wpar1
```

Name	Device Name	Type	Virtual Device	RootVG	Status
wpar1	/dev/null	pseudo			ALLOCATED
wpar1	/dev/tty	pseudo			ALLOCATED
wpar1	/dev/console	pseudo			ALLOCATED
wpar1	/dev/zero	pseudo			ALLOCATED
wpar1	/dev/clone	pseudo			ALLOCATED
wpar1	/dev/sad	clone			ALLOCATED
wpar1	/dev/xti/tcp	clone			ALLOCATED
wpar1	/dev/xti/tcp6	clone			ALLOCATED
wpar1	/dev/xti/udp	clone			ALLOCATED
wpar1	/dev/xti/udp6	clone			ALLOCATED
wpar1	/dev/xti/unixdg	clone			ALLOCATED
wpar1	/dev/xti/unixst	clone			ALLOCATED
wpar1	/dev/error	pseudo			ALLOCATED
wpar1	/dev/errorctl	pseudo			ALLOCATED
wpar1	/dev/audit	pseudo			ALLOCATED
wpar1	/dev/nvram	pseudo			ALLOCATED
wpar1	/dev/kmem	pseudo			ALLOCATED

```
Global> startwpar wpar1
Starting workload partition wpar1.
Mounting all workload partition file systems.
Replaying log for /dev/fslv04.
Replaying log for /dev/fslv05.
Replaying log for /dev/fslv06.
Replaying log for /dev/fslv07.
Loading workload partition.
Exporting workload partition devices.
Exporting workload partition kernel extensions.
Starting workload partition subsystem cor_wpar2.
0513-059 The cor_wpar2 Subsystem has been started. Subsystem PID is
7012438.
Verifying workload partition startup.
```

---







Exporting workload partition devices.

Method error (/usr/lib/methods/ucfgdevice):

0514-062 Cannot perform the requested function because the specified device is busy.

mkFCAdapExport:0: Error 0

Exporting workload partition kernel extensions.

Starting workload partition subsystem cor\_wpar2.

0513-059 The cor\_wpar2 Subsystem has been started. Subsystem PID is 9240666.

Verifying workload partition startup.

```
Global> clogin wpar1 lsdev
```

```
inet0    Defined    Internet Network Extension
pty0     Available  Asynchronous Pseudo-Terminal
sys0     Available  System Object
vg00     Available  Volume group
wio0     Available  WPAR I/O Subsystem
```

```
Global> lswpar -D
```

Name	Device Name	Type	Virtual Device	RootVG	Status
wpar1	fcs0	adapter			ALLOCATED

**Note:** Controller devices or End-point devices in AVAILABLE state are not exported to WPARs. They must be in DEFINED state.

### 3.4.10 Startwpar with a fiber channel adapter defined

To start the WPAR and have the fiber channel loaded you need to quiesce that adapter making the volume group not allocated on the Global side. A **varyoffvg** command as shown in Example 3-34 allows start of the wpar

*Example 3-34 Startwpar with fiber channel device available for WPAR use.*

```
Global> varyoffvg lpar1data
```

```
Global> lspv hdisk1
```

**0516-010 : Volume group must be varied on; use varyonvg command.**

```
PHYSICAL VOLUME:    hdisk1                VOLUME GROUP:    lpar1data
PV IDENTIFIER:      00f61aa6b48ad819  VG IDENTIFIER
00f61aa600004c000000012aba12d483
PV STATE:           ???????
STALE PARTITIONS:   ???????                ALLOCATABLE:     ???????
PP SIZE:            ???????                LOGICAL VOLUMES: ???????
TOTAL PPs:          ???????                VG DESCRIPTORS:  ???????
FREE PPs:           ???????                HOT SPARE:        ???????
USED PPs:           ???????                MAX REQUEST:     256 kilobytes
FREE DISTRIBUTION:  ???????
USED DISTRIBUTION:  ???????
```

```

MIRROR POOL:      ???????
Global> lspv
hdisk0            00f61aa68cf70a14          rootvg          active
hdisk1           00f61aa6b48ad819        lpar1data
hdisk2            00f61aa6b48b0139          None
hdisk3            00f61aa6b48ab27f          None
hdisk4            00f61aa6b48b3363          None

```

```

Global> startwpar wpar1
Starting workload partition wpar1.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.
hdisk1 Defined
hdisk2 Defined
hdisk3 Defined
hdisk4 Defined
sfwcomm0 Defined
fscsi0 Defined
line = 0
Exporting workload partition kernel extensions.
Starting workload partition subsystem cor_wpar2.
0513-059 The cor_wpar2 Subsystem has been started. Subsystem PID is 6029534.
Verifying workload partition startup.

```

---

So when WPAR is running, we can display the fiber channel and its associated devices from the WPAR side.

### *Example 3-35* Devices within the WPAR

---

```

Global> clogin wpar1
*****
* *
* Welcome to AIX Version 7.1! *
*
* Please see the README file in /usr/lpp/bos for information pertinent
* to this release of the AIX Operating System.
* *
*****
Last login: Sat Aug 28 15:33:14 EDT 2010 on /dev/Global from 75011p01

wpar1:/> lsdev
fcnet0 Defined 00-00-01 Fibre Channel Network Protocol Device
fcs0 Available 00-00 8Gb PCI Express Dual Port FC Adapter
(df1000f114108a03)
fscsi0 Available 00-00-02 FC SCSI I/O Controller Protocol Device
hdisk0 Available 00-00-02 MPIIO Other DS4K Array Disk
hdisk1 Available 00-00-02 MPIIO Other DS4K Array Disk
hdisk2 Available 00-00-02 MPIIO Other DS4K Array Disk

```

```

hdisk3 Available 00-00-02 MPIO Other DS4K Array Disk
inet0 Defined Internet Network Extension
pty0 Available Asynchronous Pseudo-Terminal
sys0 Available System Object
wio0 Available WPAR I/O Subsystem
wpar1:/> lspath
Enabled hdisk0 fscsi0
Enabled hdisk1 fscsi0
Enabled hdisk2 fscsi0
Enabled hdisk3 fscsi0

```

```

wpar1:/> lscfg
INSTALLED RESOURCE LIST

```

The following resources are installed on the machine.  
 +/- = Added or deleted from Resource List.  
 \* = Diagnostic support not available.

```

Model Architecture: chrp
Model Implementation: Multiple Processor, PCI bus

```

```

+ sys0 System Object
* wio0 WPAR I/O Subsystem
+ fcs0 8Gb PCI Express Dual Port FC Adapter (df1000f114108a03)
* fcnet0 Fibre Channel Network Protocol Device
+ fscsi0 FC SCSI I/O Controller Protocol Device
* hdisk0 MPIO Other DS4K Array Disk
* hdisk1 MPIO Other DS4K Array Disk
* hdisk2 MPIO Other DS4K Array Disk
* hdisk3 MPIO Other DS4K Array Disk

```

```

wpar1:/> lspv
hdisk0 00f61aa6b48ad819 None
hdisk1 00f61aa6b48b0139 None
hdisk2 00f61aa6b48ab27f None
hdisk3 00f61aa6b48b3363 None

```

Since the fiber channel adapter is being in use by the WPAR, it also means that all its child devices are allocated to the WPAR. The disks are then not anymore visible

#### *Example 3-36* Disk no more visible from Global

```

Global> lspv
hdisk0 00f61aa68cf70a14 rootvg active
Global> lsvg
rootvg
lpar1data

```

```
Global> lsvg lpar1data
0516-010 : Volume group must be varied on; use varyonvg command.
Global> varyonvg lpar1data
0516-013 varyonvg: The volume group cannot be varied on because
there are no good copies of the descriptor area.
```

**Note:** The `lssdev -x` command will give you the list of exported devices to WPAR.

When a device is exported, the `mkdev`, `rmdev`, `mkpath`, `chgpath` commands will fail. The `cfgmgr` command will take no action.

### 3.4.11 Disk commands in the WPAR

At that point disks commands are available as usual

*Example 3-37 Creation of volume in a WPAR*

```
wpar1:/> mkvg -y wpar1data hdisk1
wpar1data
wpar1:/> lspv
hdisk0          00f61aa6b48ad819          None
hdisk1          00f61aa6b48b0139          wpar1data      active
hdisk2          00f61aa6b48ab27f          None
hdisk3          00f61aa6b48b3363          None
wpar1:/> importvg hdisk0
syncldvdm: No logical volumes in volume group vg00.
vg00
wpar1:/> lspv
hdisk0          00f61aa6b48ad819          vg00           active
hdisk1          00f61aa6b48b0139          wpar1data      active
hdisk2          00f61aa6b48ab27f          None
hdisk3          00f61aa6b48b3363          None
wpar1:/> mk1v vg00 10
lv00
wpar1:/> lsvg vg00
VOLUME GROUP:      vg00          VG IDENTIFIER:
00f61aa600004c000000012aba12d483
VG STATE:          active          PP SIZE:        64 megabyte(s)
VG PERMISSION:     read/write      TOTAL PPs:      799 (51136
megabytes)
MAX LVs:           256            FREE PPs:       789 (50496
megabytes)
LVs:               1              USED PPs:       10 (640 megabytes)
OPEN LVs:          0              QUORUM:         2 (Enabled)
TOTAL PVs:         1              VG DESCRIPTORS: 2
STALE PVs:         0              STALE PPs:      0
```

```

ACTIVE PVs:          1                AUTO ON:          yes
MAX PPs per VG:     32512
MAX PPs per PV:     1016
LTG size (Dynamic): 256 kilobyte(s)  MAX PVs:         32
HOT SPARE:          no                AUTO SYNC:        no
PV RESTRICTION:     none              BB POLICY:        relocatable
wpar1:/> lsvg -l vg00
vg00:
LV NAME             TYPE      LPs    PPs    PVs  LV STATE    MOUNT POINT
lv00                 jfs       10     10     1    closed/syncd N/A

```

---

### 3.4.12 Access to the fiber disks from the Global

As seen previously in Example 3-36 on page 70, when the fiber channel is exported to the WPAR the disks are not any more visible from the Global.

To gain access to the disks from the Global, one simple solution is to stop the WPAR as demonstrated in Example 3-38.

#### *Example 3-38 Stopping WPAR releases fiber channel allocation*

---

```

Global> stopwpar wpar1
Stopping workload partition wpar1.
Stopping workload partition subsystem cor_wpar2.
0513-044 The cor_wpar2 Subsystem was requested to stop.
stopwpar: 0960-261 Waiting up to 600 seconds for workload partition to halt.
Shutting down all workload partition processes.
fcnet0 deleted
hdisk0 deleted
hdisk1 deleted
hdisk2 deleted
hdisk3 deleted
fscsi0 deleted
0518-307 odmdelete: 1 objects deleted.
wio0 Defined
Unmounting all workload partition file systems.
Global> lspv
hdisk0          00f61aa68cf70a14          rootvg          active
Global> cfgmgr
lspv
Method error (/usr/lib/methods/cfgefscsi -l fscsi1 ):
0514-061 Cannot find a child device.
Global> lspv
hdisk0          00f61aa68cf70a14          rootvg          active
hdisk1          00f61aa6b48ad819          lpar1data
hdisk2          00f61aa6b48b0139          None
hdisk3          00f61aa6b48ab27f          None

```

```

hdisk4          00f61aa6b48b3363          None
Global>
Global> lsvg -l lpar1data
0516-010 : Volume group must be varied on; use varyonvg command.
Global> varyonvg lpar1data
Global> lsvg -l lpar1data
lpar1data:
LV_NAME          TYPE          LPs          PPs          PVs  LV STATE      MOUNT POINT
lv00             ???          10           10           1    closed/syncd  N/A

```

**Note:** When the WPAR is removed or stopped, all devices instances will be removed from the WPAR allocation so they should be available from the Global.

### 3.4.13 Support of fiber channel devices in mkwpar command

The adapter specification is handled through the `-D` parameter in the `mkwpar` command.

```
mkwpar -n wpar2 -D devname=fcs0
```

The `mkwpar -D` option in the man page says :

*Example 3-39 mkwpar -D option syntax*

---

```

-D [devname=device name | devid=device identifier] [rootvg=yes | no]
  [devtype=[clone | pseudo | disk | adapter | cdrom | tape]] [xfactor=n]
  Configures exporting or virtualization of a Global device into the
  workload partition every time the system starts. You can specify
  more than one -D flag to allocate multiple devices. Separate the
  attribute=value by blank spaces. You can specify the following
  attributes for the -D flag:

```

---

The `devname` specification can be a controller name (see previous examples) or a end-point device name like a disk name. If not specified, the `devtype` will be queried from the Global ODM databases.

The `devname` or `devid` specification will result in an allocation phase modifying the WPAR definition to include the adapter or device. This phase is to compare with the export phase provided with the `startwpar` or `chwp` on active WPAR.

#### Creation of a rootvg system WPAR

If the `rootvg` flag is set to `yes`, the root file system of the WPAR will exist on the specified disk device (see example Example 3-40 on page 74) .

*Example 3-40 Creation of a rootvg system WPAR.*

```

Global> mkwpar -n wpar2 -D devname=hdisk3 rootvg=yes -0
Creating workload partition's rootvg. Please wait...
mkwpar: Creating file systems...
/
/admin
...
wio.common          7.1.0.0          ROOT          APPLY          SUCCESS
Finished populating scratch file systems.
Workload partition wpar2 created successfully.
mkwpar: 0960-390 To start the workload partition, execute the following as
root: startwpar [-v] wpar2

Global> lswpar -M wpar2
Name      MountPoint                Device      Vfs      Nodename  Options
-----
wpar2    /wpars/wpar2              /dev/fslv05 jfs2
wpar2    /wpars/wpar2/etc/objrepos/wboot /dev/fslv06 jfs2
wpar2    /wpars/wpar2/opt          /opt       namefs   ro
wpar2    /wpars/wpar2/usr          /usr       namefs   ro
Global> lswpar -D wpar2
Name      Device Name      Type      Virtual Device  RootVG  Status
-----
wpar2    /dev/null        pseudo
....
wpar2    hdisk3           disk          yes           ALLOCATED

```

**Note:** In the preceding examples, /dev/fslv05 and /dev/fslv06 are the file systems used to start the rootvg WPAR and contain the bare minimum elements to configure the WPAR storage devices.

**Rootvg system WPAR creation failure when device busy**

If wpar1 wpar was still active, meaning that the fiber channel device being busy, the mkwpar command won't process.

*Example 3-41 Mkwpar failure if end-point device busy*

```

Global> mkwpar -n wpar2 -D devname=hdisk3 rootvg=yes
Creating workload partition's rootvg. Please wait...
mkwpar: 0960-621 Failed to create a workload partition's rootvg. Please
use -0 flag to overwrite hdisk3.
If restoring a workload partition, target disks should be in
available state.
Global> mkwpar -n wpar2 -D devname=hdisk3 rootvg=yes -0
mkwpar: 0960-619 Failed to make specified disk, hdisk3, available.

```



**Note:** mkwpar -O flag may be used to force the overwrite of an existing volume group on the given set of devices specified with the -D rootvg=yes flag directive

## Rootvg system WPAR overview

When the system WPAR has been created (see Example 3-40 on page 74), two devices has been created in the Global rootvg disk for management and startup purpose: One for the root mount point and the other for the ODM customizing to be made during export phase.

*Example 3-42 Listing of the rootvg system WPAR file systems from the Global*

```
Global> lswpar -M wpar2
Name MountPoint Device Vfs Nodename Options
-----
wpar2 /wpars/wpar2 /dev/fs1v05 jfs2
wpar2 /wpars/wpar2/etc/objrepos/wboot /dev/fs1v06 jfs2
wpar2 /wpars/wpar2/opt /opt namefs ro
wpar2 /wpars/wpar2/usr /usr namefs ro

Global> lspv -l hdisk0 | grep wpar2
fs1v05 2 2 00..02..00..00..00 /wpars/wpar2
fs1v06 1 1 00..01..00..00..00 /wpars/wpar2/etc/objrepos/wboot
```

And what is important to understand is that all devices are known from the Global side even if they are exported to the WPAR. Of course they may not be accessible.

*Example 3-43 Allocated devices to a WPAR not available to Global*

```
Global> lswpar -D wpar2 | grep disk
wpar2 hdisk3 disk yes ALLOCATED
Global>
Global> lsdev -x
L2cache0 Available L2 Cache
...
fcnet0 Defined 00-00-01 Fibre Channel Network Protocol Device
fcnet1 Defined 00-01-01 Fibre Channel Network Protocol Device
fcs0 Available 00-00 8Gb PCI Express Dual Port FC Adapter
(df1000f114108a03)
fcs1 Available 00-01 8Gb PCI Express Dual Port FC Adapter
(df1000f114108a03)
fscsi0 Available 00-00-02 FC SCSI I/O Controller Protocol Device
fscsi1 Available 00-01-02 FC SCSI I/O Controller Protocol Device
fs1v00 Available Logical volume
fs1v01 Available Logical volume
fs1v02 Available Logical volume
fs1v03 Available Logical volume
fs1v04 Available Logical volume
```

```

fslv05    Available          Logical volume
fslv06    Available          Logical volume
hd1       Defined             Logical volume
hd2       Defined             Logical volume
hd3       Defined             Logical volume
hd4       Defined             Logical volume
hd5       Defined             Logical volume
hd6       Defined             Logical volume
hd8       Defined             Logical volume
hd10opt   Defined             Logical volume
hd11admin Defined             Logical volume
hd9var    Defined             Logical volume
hdisk0    Available          Virtual SCSI Disk Drive
hdisk1   Defined    00-00-02   MPIO Other DS4K Array Disk
hdisk2   Defined    00-00-02   MPIO Other DS4K Array Disk
hdisk3   Available 00-00-02   MPIO Other DS4K Array Disk
hdisk4   Defined    00-00-02   MPIO Other DS4K Array Disk
...
Global> lspv
hdisk0          00f61aa68cf70a14          rootvg          active
hdisk3          00f61aa6b48ab27f          None
Global> lspv -l hdisk3
0516-320 : Physical volume 00f61aa6b48ab27f00000000000000000 is not assigned to a volume group.

```

---

## Startwpar of the rootvg system WPAR

The startwpar command will effectively process the export phase and associate the devices to the WPAR. In case of the rootvg specification, the disk name appears in the listing. It also mentions that the kernel extension dynamic loading is being used to load the fiber channel and the wio driver (see genkex output for example).

### *Example 3-44 Startwpar of a rootvg WPAR on fiber channel disk*

```

Global> startwpar wpar2
Starting workload partition wpar2.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.
hdisk3 Defined
Exporting workload partition kernel extensions.
Starting workload partition subsystem cor_wpar3.
0513-059 The cor_wpar3 Subsystem has been started. Subsystem PID is
8650994.
Verifying workload partition startup.

```

---

**Note:** An FC controller would not be exported explicitly but would be implicitly exported when the `cfgmgr` command is being launched by `/etc/rc.boot` script.

Within the rootvg WPAR the file system structure is referencing internal devices (`/dev/...`) from the rootvg disk as well as file system mounted from Global since we didn't create private file systems. We can also see that the root mount point mounted from the Global is over-mounted with the local device.

*Example 3-45 File systems of the rootvg WPAR seen from inside the WPAR*

---

```
Global> clogin wpar2 df
Filesystem      512-blocks      Free %Used    Iused %Iused Mounted on
Global          262144      200840 24%    2005   9% /
/dev/hd4         262144         200840   24%     2005    9% /
Global          4063232        448200   89%     41657   44% /usr
Global          786432         427656   46%     7008    13% /opt
/dev/hd11admin   131072         128312   3%       5        1% /admin
/dev/hd1         131072         128312   3%       5        1% /home
/dev/hd3         262144         256864   3%       9        1% /tmp
/dev/hd9var      262144         220368   16%     349     2% /var
Global          131072      128336 3%     5      1% /etc/objrepos/wboot
Global          -              -        -        -        - /proc
Global> clogin wpar2 lspv
hdisk0          00f61aa6b48ab27f                rootvg      active
```

---

And the device listing is also as expected with disks and drivers `wio` and `fscsi0`:

*Example 3-46 lsdev within a rootvg system WPAR*

---

```
Global> clogin wpar2 lsdev
fscsi0    Available 00-00-02 WPAR I/O Virtual Parent Device
hd1       Available                Logical volume
hd3       Available                Logical volume
hd4       Available                Logical volume
hd11admin Available                Logical volume
hd9var    Available                Logical volume
hdisk0   Available 00-00-02 MPI0 Other DS4K Array Disk
inet0     Defined                Internet Network Extension
pty0     Available                Asynchronous Pseudo-Terminal
rootvg    Available                Volume group
sys0     Available                System Object
wio0    Available                WPAR I/O Subsystem
```

---

## Fiber channel controller can't be shared

As we started wpar2 rootvg system WPAR, the fiber channel controller can be exported to wpar1 system WPAR since one of its child is busy. As such, wpar1 WPAR start won't load the fcs0 controller and some warning messages appear on console.

### *Example 3-47 Exclusive device allocation message*

```
Global> startwpar wpar1
Starting workload partition wpar1.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.
rmdev: 0514-552 Cannot perform the requested function because the
      hdisk3 device is currently exported.
mkFCAdapExport:0: Error 0
Exporting workload partition kernel extensions.
Starting workload partition subsystem cor_wpar2.
5113-059 The cor_wpar2 Subsystem has been started. Subsystem PID is 8585362.
Verifying workload partition startup.
```

```
Global> lswpar
Name   State  Type  Hostname  Directory      RootVG WPAR
-----
wpar1  A      S     wpar1    /wpars/wpar1  no
wpar2  A      S     wpar2    /wpars/wpar2  yes
Global> lswpar -D
Name   Device Name      Type      Virtual Device  RootVG  Status
-----
wpar1  fcs0              adapter                    ALLOCATED
.....
wpar2  hdisk3            disk       hdisk0          yes     EXPORTED
```

## End-point devices are separated

However, the other disks (end-point devices) can be allocated to another WPAR if the fiber channel controller hasn't been explicitly exported.

We can now create a new rootvg system WPAR on disk called hdisk4. A summary of the console messages issued from the mkwpar command is listed in Example 3-48. Startwpar console messages are also included.

### *Example 3-48 New rootvg system WPAR creation*

```
Global> mkwpar -0 -D devname=hdisk4 rootvg=yes -n wpar3
.....
syncroot: Processing root part installation status.
syncroot: Installp root packages are currently synchronized.
syncroot: RPM root packages are currently synchronized.
syncroot: Root part is currently synchronized.
```

```

syncroot: Returns Status = SUCCESS
.....
Exporting a workload partition's rootvg. Please wait...
Cleaning up the trace of a workload partition's rootvg population...
mkwpar: Removing file system /wpars/wpar3/usr.
mkwpar: Removing file system /wpars/wpar3/proc.
mkwpar: Removing file system /wpars/wpar3/opt.
Creating scratch file system...
Populating scratch file systems for rootvg workload partition...
Mounting all workload partition file systems.
x ./usr
x ./lib
....
Installation Summary
-----
Name                                Level      Part      Event     Result
-----
bos.net.nis.client                   7.1.0.0   ROOT      APPLY     SUCCESS
bos.perf.libperfstat                 7.1.0.0   ROOT      APPLY     SUCCESS
bos.perf.perfstat                    7.1.0.0   ROOT      APPLY     SUCCESS
bos.perf.tools                        7.1.0.0   ROOT      APPLY     SUCCESS
bos.sysmgmt.trace                    7.1.0.0   ROOT      APPLY     SUCCESS
clic.rte.kernext                     4.7.0.0   ROOT      APPLY     SUCCESS
devices.chrp.base.rte                7.1.0.0   ROOT      APPLY     SUCCESS
devices.chrp.pci.rte                 7.1.0.0   ROOT      APPLY     SUCCESS
devices.chrp.vdevice.rte             7.1.0.0   ROOT      APPLY     SUCCESS
devices.common.IBM.ethernet          7.1.0.0   ROOT      APPLY     SUCCESS
devices.common.IBM.fc.rte            7.1.0.0   ROOT      APPLY     SUCCESS
devices.common.IBM.mpio.rte          7.1.0.0   ROOT      APPLY     SUCCESS
devices.common.IBM.scsi.rte          7.1.0.0   ROOT      APPLY     SUCCESS
devices.fcp.disk.array.rte           7.1.0.0   ROOT      APPLY     SUCCESS
devices.fcp.disk.rte                 7.1.0.0   ROOT      APPLY     SUCCESS
devices.fcp.tape.rte                 7.1.0.0   ROOT      APPLY     SUCCESS
devices.scsi.disk.rte                7.1.0.0   ROOT      APPLY     SUCCESS
devices.tty.rte                      7.1.0.0   ROOT      APPLY     SUCCESS
bos.mp64                             7.1.0.0   ROOT      APPLY     SUCCESS
bos.net.tcp.client                   7.1.0.0   ROOT      APPLY     SUCCESS
bos.perf.tune                         7.1.0.0   ROOT      APPLY     SUCCESS
perfagent.tools                      7.1.0.0   ROOT      APPLY     SUCCESS
bos.net.nfs.client                   7.1.0.0   ROOT      APPLY     SUCCESS
bos.wpars                            7.1.0.0   ROOT      APPLY     SUCCESS
bos.net.ncs                          7.1.0.0   ROOT      APPLY     SUCCESS
wio.common                           7.1.0.0   ROOT      APPLY     SUCCESS
Finished populating scratch file systems.
Workload partition wpar3 created successfully.
mkwpar: 0960-390 To start the workload partition, execute the following as
root: startwpar [-v] wpar3
Global>

```

```
Global> startwpar wpar3
Starting workload partition wpar3.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.
hdisk4 Defined
Exporting workload partition kernel extensions.
Starting workload partition subsystem cor_wpar4.
0513-059 The cor_wpar4 Subsystem has been started. Subsystem PID is 7405614.
Verifying workload partition startup.
```

---

And from the global instance we can check that both disks are exported.

*Example 3-49 Global view of exported disks to rootvg WPARs*

```
Global> lswpar -D
```

Name	Device Name	Type	Virtual Device	RootVG	Status
wpar1	fcs0	adapter			ALLOCATED
...					
wpar2	hdisk3	disk	hdisk0	yes	<b>EXPORTED</b>
...					
wpar3	hdisk4	disk	hdisk0	yes	<b>EXPORTED</b>

```
Global> lsdev -x | grep -i export
hdisk3      Exported 00-00-02  MPIIO Other DS4K Array Disk
hdisk4      Exported 00-00-02  MPIIO Other DS4K Array Disk
```

---

### 3.4.14 Config file created for the rootvg system WPAR

When system WPAR is being created, a config file is also created in /etc/wpars and includes the rootvg device specification as well as the rootvg WPAR type.

*Example 3-50 /etc/wpars/wpar3.cf listing*

```
Global> cat /etc/wpars/wpar3.cf
general:
    name = "wpar3"
    checkpointable = "no"
    hostname = "wpar3"
    privateusr = "no"
    directory = "/wpars/wpar3"
    ostype = "0"
    auto = "no"
    rootvgwpar = "yes"
    routing = "no"
```

```
resources:
    active = "yes"
.....
device:
    devid = "3E213600A0B8000291B080000E299059A3F460F1815
FASTT03IBMfcp"
    devtype = "2"
    rootvg = "yes"
```

---

### 3.4.15 Removing of a fiber disk in a running system WPAR

It is not possible to remove the rootvg disk of the system WPAR when it is active since it is busy.

*Example 3-51 Rootvg disk of a rootvg WPAR can't be removed if WPAR active*

---

```
Global> chwpar -K -D devname=hdisk4 wpar3
chwpar: 0960-604 the device with devname, hdisk4, is still being used
in the WPAR.
chwpar: 0960-018 1 errors refreshing devices.
```

---

### 3.4.16 Mobility restrictions

Use of rootvg devices and fiber channel in system WPAR currently prevent mobility.

#### **Mobility of a fiber channel adapter**

Use of fiber channel adapter in a system WPAR prevent mobility.

```
Global> chwpar -c wpar1
chwpar: 0960-693 Cannot checkpoint a WPAR that has adapter(s).
```

#### **Mobility of a rootvg system WPAR**

In order to change the checkpointable flag of a system WPAR, it must be stopped. Then - providing you get the required optional package mcr.rte being installed on your system, you can change the checkpoint flag of the wpar using the command **chwpar -c wpar2**.

A listing of the system WPAR wpar2 states it is checkpointable.

*Example 3-52 Listing of the environment flags of the system WPAR*

---

```
Global> lswpar -G wpar2
=====
wpar2 - Defined
```

```

=====
Type:                               S
RootVG WPAR:                         yes
Owner:                               root
Hostname:                            wpar2
WPAR-Specific Routing:              no
Directory:                           /wpars/wpar2
Start/Stop Script:
Auto:                                no
Private /usr:                        no
Checkpointable:                  yes
Application:
OStype:                              0
=====

```

But the rootvg system WPAR can be checkpointed.

*Example 3-53 Checkpoint WPAR is not allowed with rootvg WPAR*

```

/opt/mcr/bin/chkptwpar -d /wpars/wpar2/tmp/chpnt -o
/wpars/wpar2/tmp/ckplog -l debug wpar2
1020-235 chkptwpar is not allowed on rootvg (SAN) WPAR [02.291.0168]
[8650894 29:8:2010 12:23:7]
1020-187 chkptwpar command failed.
=====

```

### 3.4.17 Debugging log

As usual all events related to WPAR commands are added to the file `/var/adm/wpars/event.log`.

For example, last commands being issue such as **stopwpar** on wpar2 and **chwp** on wpar3 get appropriate error messages to ease debugging.

*Example 3-54 /var/adm/wpars/event.log example*

```

Global> tail /var/adm/wpars//event.log
I 2010-08-29 12:22:04 7929932 runwpar wpar2 Removing work directory
/tmp/.workdir.7077910.7929932_1
V 2010-08-29 12:22:05 7929932 startwpar - COMMAND START, ARGS: -I wpar2
I 2010-08-29 12:22:05 7929932 startwpar wpar2 Removing work directory
/tmp/.workdir.8454242.7929932_1
I 2010-08-29 12:22:05 10289288 startwpar wpar2 Lock released.
I 2010-08-29 12:22:05 10289288 startwpar wpar2 Removing work directory
/tmp/.workdir.8781954.10289288_1
V 2010-08-29 12:22:05 10289288 startwpar wpar2 Return Status = SUCCESS.

```



```
E 2010-08-29 12:25:28 7209076 corralinstcmd wpar3
/usr/lib/corrals/corralinstcmd: 0960-231 ATTENTION:
'/usr/lib/corrals/wpardevstop hdisk0' failed with return code 1.
E 2010-08-29 12:25:28 8126600 chwpar wpar3 chwpar: 0960-604 the device
with devname, hdisk4, is still being used in the WPAR.
W 2010-08-29 12:25:28 8126600 chwpar wpar3 chwpar: 0960-018 1 errors
refreshing devices.
W 2010-08-29 12:26:10 8126606 chwpar wpar3 chwpar: 0960-070 Cannot find
a device stanza to remove from /etc/wpars/wpar3.cf where devname=fcs0.
```

---

## 3.5 WPAR RAS enhancements

This section will discuss how the enhancement introduced with RAS error logging mechanism have been propagated to WPARs with AIX 7.1.

This feature first became available in AIX 7.1 and is included in AIX 6.1 TL 06.

### 3.5.1 Error logging mechanism aspect

The Reliability, Availability, and Serviceability (RAS) kernel services are used to record the occurrence of hardware or software failures and to capture data about these failures. The recorded information can be examined using the **errpt** or **trcrpt** commands.

WPAR mobility commands are integrating AIX messages as well as kernel services error messages when possible. When an error occurs these messages were considered as not descriptive enough for end user.

Since AIX 7.1 is integrating a common error logging and reporting mechanism, the goal was to propagate that mechanism to WPAR commands as well as for WPAR mobility commands.

Mobility commands error messages are available in the IBM Director WPAR plug-in or WPAR manager logs.

This section describe the message format of the WPAR commands error or informative messages.

## 3.5.2 Goal for these messages

This new messages structure tends to address the following need:

- ▶ Have end-user messages explicit with a resolution statement as possible
- ▶ The messages include errno values when a failure without direct resolution statement occurs.
- ▶ When a failure occurs, the message will give information on the cause and the location of that failure to the support team to ease debugging.
- ▶ Use of formatted messages with components names, component id and message number allow easy scripting.

## 3.5.3 Syntax of the messages

The message structure is:

*<component name> <component number>-<message number within the component> <message >*

For example the user command **mkwpar** would issue a syntax error if the parameter is invalid, knowing that the following field are fixed for that command:

- ▶ The component is the command name: **mkwpar**:
- ▶ The component id: **0960**
- ▶ The message number: **077**

*Example 3-55 mkwpar user command error message*

---

```
Global> mkwpar wpar1
mkwpar: 0960-077 Extra arguments found on the command line.
Usage: mkwpar [-a] [-A] [-b devexportsFile] [-B wparBackupDevice] [-c] [-C]...
```

---

For the same command like in Example 3-56, the error type being different, the message number is 299 when the component name and id remains the same.

*Example 3-56 Same command, other message number*

---

```
Global> mkwpar -c -n test
mkwpar: 0960-299 Workload partition name test already exists in
/etc/filesystems. Specify another name.
```

---

For another WPAR command like **rmwpar** the component will remain 0960, but other fields would change.

*Example 3-57 Same component, other command*


---

```
Global> rmwpar wpar2
rmwpar: 0960-419 Could not find a workload partition called wpar2.
```

---

In some cases, two messages with different numbers can be displayed for an error. One usually giving resolution advice and one specifying the main error.

*Example 3-58 Multiple messages for a command*


---

```
Global> rmwpar wpar1
rmwpar: 0960-438 Workload partition wpar1 is running.
rmwpar: 0960-440 Specify -s or -F to stop the workload partition before
removing

Global> lswpar -I
lswpar: 0960-568 wpar1 has no user-specified routes.
lswpar: 0960-559 Use the following command to see the
full routing table for this workload partition:
netstat -r -@ wpar1
```

---

As mentioned, WPAR mobility commands follow these rules as shown in that command line output:

*Example 3-59 WPAR mobility command error messages*


---

```
Global> /opt/mcr/bin/chkptwpar
1020-169 Usage:
To checkpoint an active WPAR:
    chkptwpar [-k | -p] -d /path/to/statefile [-o /path/to/logfile
[-l <debug|error>]] wparName

Global> /opt/mcr/bin/chkptwpar wpar1
1020-054 WPAR wpar1 is not checkpointable [09.211.0449]
1020-187 chkptwpar command failed.
```

---

These message structure may also apply to informative messages

*Example 3-60 Few other informative messages*


---

```
Global> mkwpar -c -n test2 -F
....
syncroot: Processing root part installation status.
syncroot: Installp root packages are currently synchronized.
syncroot: RPM root packages are currently synchronized.
syncroot: Root part is currently synchronized.
syncroot: Returns Status = SUCCESS
```

Workload partition test2 created successfully.  
mkwpar: **0960-390** To start the workload partition, execute the following as  
root: startwpar [-v] test2

```
Global> /opt/mcr/bin/chkptwpar -l debug -o /test2/tmp/L -d /wpars/test2/tmp/D test2
1020-052 WPAR test2 is not active [09.211.0352]
1020-187 chkptwpar command failed.
```

```
Global> startwpar test2
Starting workload partition test2.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.
Exporting workload partition kernel extensions.
Starting workload partition subsystem cor_test2.
0513-059 The cor_test2 Subsystem has been started. Subsystem PID is 4456462.
Verifying workload partition startup.
```

```
Global> /opt/mcr/bin/chkptwpar -l debug -o /wpars/test2/tmp/L -d /wpars/test2/tmp/D test2
1020-191 WPAR test2 was checkpointed in /wpars/test2/tmp/D.
1020-186 chkptwpar command succeeded
```

---

## 3.6 WPAR Migration to AIX Version 7.1

After successfully migrating a global instance running AIX V6.1 to AIX V7.1, all associated Workload Partitions (WPARs) also need to be migrated to the newer version of the operating system. The WPAR shares the same kernel as the global system. System software must be kept at the same level as the global environment in order to avoid unexpected results. There may be unexpected behavior if system calls, functions or libraries are called from a WPAR that has not been migrated.

Prior to the migration to AIX V7.1, the global instance level of AIX is V6.1. WPARs were created with AIX V6.1. In order for the WPARs to function correctly after the migration to AIX V7.1, they must also be migrated. This is accomplished with the **migwpar** command.

A global instance of AIX is migrated with a normal AIX migration from one release of AIX to another. Refer to the AIX Installation and Migration Guide, SC23-6722 for information on migrating AIX.

[http://publib.boulder.ibm.com/infocenter/aix/v7r1/topic/com.ibm.aix.install/doc/insgdrf/insgdrf\\_pdf.pdf](http://publib.boulder.ibm.com/infocenter/aix/v7r1/topic/com.ibm.aix.install/doc/insgdrf/insgdrf_pdf.pdf)

WPAR migration is separate from a global instance migration. WPARs are not migrated automatically during an AIX migration. Once the global instance has been successfully migrated from AIX V6.1 to AIX V7.1, any associated WPARs must also be migrated to AIX V7.1.

Currently, only system WPARs are supported for migration. Both shared and detached system WPARs are supported. Shared system WPARs are those that do not have their own private /usr and /opt file systems. They share the /usr and /opt file systems from the global system.

A detached system WPAR (or non-shared system WPAR) has a private /usr and /opt file system, which is copied from the global environment. In order to migrate a WPAR of this type, the administrator must specify install media as the software source for the migration.

WPAR types that are not supported for migration are:

- ▶ Application WPARs.
- ▶ Versioned WPARs.

The **migwpar** command migrates a WPAR, that was created in an AIX V6.1 global instance, to AIX V7.1. Before attempting to use the **migwpar** command, you must ensure that the global system has migrated successfully first. The `pre_migration` and `post_migration` scripts can be run in the global instance before and after the migration to determine what software will be removed during the migration, to verify that the migration completed successfully and identify software that did not migrate.

The `pre_migration` script is available on the AIX V7.1 media in the following location, `/usr/lpp/bos/pre_migration`. It can also be found in an AIX V7.1 NIM SPOT, for example, `/export/spot/spotaix7100/usr/lpp/bos/pre_migration`. The `post_migration` script is available in the following location, `/usr/lpp/bos/post_migration` on an AIX V7.1 system. Please refer to the following website for further information relating to these scripts:

[http://publib.boulder.ibm.com/infocenter/aix/v7r1/topic/com.ibm.aix.install/doc/insgdrf/migration\\_scripts.htm](http://publib.boulder.ibm.com/infocenter/aix/v7r1/topic/com.ibm.aix.install/doc/insgdrf/migration_scripts.htm)

Table 3-1 describes the available flags and options to the **migwpar** command.

*Table 3-1 migwpar flags and options*

Flag	Description
-A	Migrates all migratable WPARs.

Flag	Description
-f <i>wparNameFile</i>	Migrates the list of WPARs contained in the file <i>wparNamesFile</i> , one per line.
-d <i>software_source</i>	Installation location used for the detached (private) system WPAR migration.

Only the root user can run the **migwpar** command.

To migrate a single shared system WPAR from AIX V6.1 to AIX V7.1 you would execute the **migwpar** command shown in Example 3-61.

*Example 3-61 Migrating a shared system WPAR to AIX V7.1*

---

```
# migwpar wpar1
```

---

A detached system WPAR can be migrated using the command shown in Example 3-62. The */images* file system is used as the install source. This file system contains AIX V7.1 packages, copied from the install media.

*Example 3-62 Migrating a detached WPAR to AIX V7.1*

---

```
# migwpar -d /images wpar1
```

---

To migrate all shared system WPARs to AIX V7.1, enter the command shown in Example 3-63.

*Example 3-63 Migrating all shared WPARs to AIX V7.1.*

---

```
# migwpar -A
```

---

To migrate all detached WPARs, using */images* as the software source, you would enter the command shown in Example 3-64.

*Example 3-64 Migrating all detached WPARs to AIX V7.1*

---

```
# migwpar -A -d /images
```

---

WPAR migration information is logged to the */var/adm/ras/migwpar.log* file in the global environment. Additional software installation information is logged to the */wpars/wparname/var/adm/ras/devinst.log* file for the WPAR, for example, */wpars/wpar1/var/adm/ras/devinst.log* for *wpar1*.

**Note:** If you attempt to run the **syncroot** command after a global instance migration and you have not run the **migwpar** command against the WPAR(s), you will receive the following error message:

```
syncroot: Processing root part installation status.
Your global system is at a higher version than the WPAR.
Please log out of the WPAR and execute the migwpar command.
syncroot: Returns Status = FAILURE
```

If you run the **syncwpar** command to sync a version 6 WPAR, on a version 7 global system, the **syncwpar** command will call the **migwpar** command and will migrate the WPAR. If the SMIT interface to **syncwpar** is used (**smit syncwpar\_sys**) the **migwpar** command will be called as required.

In the example that follows, we migrated a global instance of AIX V6.1 to AIX V7.1. We then verified that the migration was successful, before migrating a single shared system WPAR to AIX V7.1.

We performed the following steps to migrate the WPAR:

1. It is recommended that the **syncroot** and **syncwpar** commands be run prior to migrating the global instance. This is to verify the system software package integrity of the WPAR(s) before the migration. The **oslevel**, **lspp**, and **lppchk** commands can also assist in confirming the AIX level and fileset consistency.
2. Stop the WPAR prior to migrating the global instance.
3. Migrate the global instance of AIX V6.1 to AIX V7.1. The WPAR is not migrated and remains at AIX V6.1. Verify that the global system migrates successfully first.
4. Start the WPAR and verify that the WPAR is functioning as expected, after the global instance migration.
5. Migrate the WPAR to AIX V7.1 with the **migwpar** command.
6. Verify that the WPAR migrated successfully and is functioning as expected.

We confirmed that the WPAR was in an active state (A) prior to the migration, as shown in Example 3-65.

*Example 3-65 Confirming the WPAR state is active*

---

```
# lswpar
Name   State Type  Hostname  Directory      RootVG WPAR
-----
wpar1  A     S     wpar1    /wpars/wpar1  no
```

---

Prior to migrating the global instance we first verified the current AIX version and level in both the global system and the WPAR, as shown in Example 3-66.

*Example 3-66 Verifying global and WPAR AIX instances prior to migration*

---

```
# uname -W
0
# syncwpar wpar1
*****
Synchronizing workload partition wpar1 (1 of 1).
*****
Executing /usr/sbin/syncroot in workload partition wpar1.
syncroot: Processing root part installation status.
syncroot: Installp root packages are currently synchronized.
syncroot: RPM root packages are currently synchronized.
syncroot: Root part is currently synchronized.
syncroot: Returns Status = SUCCESS
Workload partition wpar1 synchronized successfully.

Return Status = SUCCESS.

# clogin wpar1
*****
*
*
* Welcome to AIX Version 6.1!
*
*
*
* Please see the README file in /usr/lpp/bos for information pertinent to
* this release of the AIX Operating System.
*
*
*****
# uname -W
1
# syncroot
syncroot: Processing root part installation status.
syncroot: Installp root packages are currently synchronized.
syncroot: RPM root packages are currently synchronized.
syncroot: Root part is currently synchronized.
syncroot: Returns Status = SUCCESS
# exit

AIX Version 6
Copyright IBM Corporation, 1982, 2010.
login: root
root's Password:
*****
*
*
```



```

*
* Welcome to AIX Version 6.1!
*
*
* Please see the README file in /usr/lpp/bos for information pertinent to
* this release of the AIX Operating System.
*
*
*****
Last login: Fri Aug 27 17:14:27 CDT 2010 on /dev/vty0

# uname -W
0
# oslevel -s
6100-05-01-1016
# lppchk -m3 -v
#

# clogin wpar1
*****
*
* Welcome to AIX Version 6.1!
*
*
* Please see the README file in /usr/lpp/bos for information pertinent to
* this release of the AIX Operating System.
*
*
*****
Last login: Fri Aug 27 17:06:56 CDT 2010 on /dev/Global from r2r2m31

# uname -W
1
# oslevel -s
6100-05-01-1016
# lppchk -m3 -v
#

```

---

Before migrating the global system, we stopped the WPAR cleanly, as shown in Example 3-67 on page 92.

**Note:** the -F flag has been specified with the **stopwpar** command to force the WPAR to stop quickly. It is recommended that this only be performed after all applications within a WPAR have been stopped first.

The -v flag has been specified with the **stopwpar** command to produce verbose output. This has been done in order to verify that the WPAR has in fact been stopped successfully. This is confirmed by the Return Status = SUCCESS message.

Messages relating to the removal of Inter-process communication (IPC) segments and semaphores are also shown, for example ID=2097153 KEY=0x4107001c UID=0 GID=9 RT=-1 . These messages are generated by the **/usr/lib/corral/removeipc** utility which is called by the **stopwpar** command when stopping a WPAR.

---

*Example 3-67 Clean shutdown of the WPAR*

---

```
# stopwpar -Fv wpar1
Stopping workload partition wpar1.
Stopping workload partition subsystem cor_wpar1.
0513-044 The cor_wpar1 Subsystem was requested to stop.
Shutting down all workload partition processes.
WPAR='wpar1' CID=1
ID=2097153 KEY=0x4107001c UID=0 GID=9 RT=-1
ID=5242897 KEY=0x0100075e UID=0 GID=0 RT=-1
ID=5242898 KEY=0x620002de UID=0 GID=0 RT=-1
ID=9437203 KEY=0xffffffff UID=0 GID=0 RT=-1
wio0 Defined
Unmounting all workload partition file systems.
Unmounting /wpars/wpar1/var.
Unmounting /wpars/wpar1/usr.
Unmounting /wpars/wpar1/tmp.
Unmounting /wpars/wpar1/proc.
Unmounting /wpars/wpar1/opt.
Unmounting /wpars/wpar1/home.
Unmounting /wpars/wpar1.
Return Status = SUCCESS.
```

---

We then migrated the global system from AIX V6.1 to AIX V7.1. This was accomplished with a normal AIX migration, using a virtual SCSI CD drive. Once the migration completed successfully, we verified that the correct version of AIX was now available in the global environment, as shown in Example 3-68 on page 93.

**Note:** It is recommended that AIX V7.1 Technology Level 0, Service Pack 1 be installed in the global instance prior to running the **migwpar** command.

*Example 3-68 AIX version 7.1 after migration.*

---

```

AIX Version 7
Copyright IBM Corporation, 1982, 2010.
login: root
root's Password:
*****
*
*
* Welcome to AIX Version 7.1!
*
*
* Please see the README file in /usr/lpp/bos for information pertinent to
* this release of the AIX Operating System.
*
*
*****
1 unsuccessful login attempt since last login.
Last unsuccessful login: Tue Aug 31 17:21:56 CDT 2010 on /dev/pts/0 from
10.1.1.99
Last login: Tue Aug 31 17:21:20 CDT 2010 on /dev/vty0

# oslevel
7.1.0.0
# oslevel -s
7100-00-01-1037
# lppchk -m3 -v
#

```

---

The WPAR was not started and was in a defined (D) state, as shown in Example 3-69.

*Example 3-69 WPAR not started after global instance migration to AIX V7.1*

---

```

# lswpar
Name State Type Hostname Directory RootVG WPAR
-----
wpar1 D S wpar1 /wpars/wpar1 no

```

---

The WPAR was then started successfully, as shown in Example 3-70 on page 94.

**Note:** The -v flag has been specified with the **startwpar** command to produce verbose output. This has been done in order to verify that the WPAR has in fact been started successfully. This is confirmed by the Return Status = SUCCESS message.

*Example 3-70 Starting the WPAR after global instance migration*

---

```
# startwpar -v wpar1
Starting workload partition wpar1.
Mounting all workload partition file systems.
Mounting /wpars/wpar1
Mounting /wpars/wpar1/home
Mounting /wpars/wpar1/opt
Mounting /wpars/wpar1/proc
Mounting /wpars/wpar1/tmp
Mounting /wpars/wpar1/usr
Mounting /wpars/wpar1/var
Loading workload partition.
Exporting workload partition devices.
Exporting workload partition kernel extensions.
Starting workload partition subsystem cor_wpar1.
0513-059 The cor_wpar1 Subsystem has been started. Subsystem PID is 6619348.
Verifying workload partition startup.
Return Status = SUCCESS.
```

---

Although the global system was now running AIX V7.1, the WPAR was still running AIX V6.1, as shown in Example 3-71.

*Example 3-71 Global instance migrated to version 7, WPAR still running version 6*

---

```
# uname -W
0
# lslpp -l -0 r bos.rte
Fileset                                Level  State      Description
-----
Path: /etc/objrepos
bos.rte                                7.1.0.0  COMMITTED  Base Operating System Runtime
#
# clogin wpar1 lslpp -l -0 r bos.rte
Fileset                                Level  State      Description
-----
Path: /etc/objrepos
bos.rte                                6.1.5.0  COMMITTED  Base Operating System Runtime
```

---

The **migwpar** command was run against the WPAR to migrate it to AIX V7.1, as shown in Example 3-72. Only partial output is shown as the actual migration log is extremely verbose.

*Example 3-72 WPAR migration to AIX V7.1 with migwpar*

---

```
# migwpar wpar1

Shared /usr WPAR list:
wpar1
WPAR wpar1 mount point:
/wpars/wpar1
WPAR wpar1 active
MIGWPAR: Saving configuration files for wpar1
MIGWPAR: Removing old bos files for wpar1
MIGWPAR: Replacing bos files for wpar1
MIGWPAR: Merging configuration files for wpar1
0518-307 odmdelete: 1 objects deleted.
0518-307 odmdelete: 0 objects deleted.
0518-307 odmdelete: 2 objects deleted.
....
x ./lib
x ./audit
x ./dev
x ./etc
x ./etc/check_config.files
x ./etc/consdef
x ./etc/cronlog.conf
x ./etc/csh.cshrc
x ./etc/csh.login
x ./etc/dlpi.conf
x ./etc/dumpdates
x ./etc/environment
x ./etc/ewlm
x ./etc/ewlm/limits
x ./etc/ewlm/trc
x ./etc/ewlm/trc/config_schema.xsd
x ./etc/ewlm/trc/output_schema.xsd
x ./etc/filesystems
x ./etc/group
x ./etc/inittab
...
MIGWPAR: Merging configuration files for wpar1
0518-307 odmdelete: 1 objects deleted.
MIGWPAR: Running syncroot for wpar1
```

```

syncroot: Processing root part installation status.
syncroot: Synchronizing installp software.
syncroot: Processing root part installation status.
syncroot: Installp root packages are currently synchronized.
syncroot: RPM root packages are currently synchronized.
syncroot: Root part is currently synchronized.
syncroot: Returns Status = SUCCESS
Cleaning up ...

```

---

We logged into the WPAR using the **clogin** command after the migration to verify the WPAR was functioning as expected, as shown in Example 3-73.

*Example 3-73 Verifying WPAR started successfully after migration*

---

```

# clogin wpar1
*****
*
*
* Welcome to AIX Version 7.1!
*
*
* Please see the README file in /usr/lpp/bos for information pertinent to
* this release of the AIX Operating System.
*
*
*****
Last login: Tue Aug 31 17:32:48 CDT 2010 on /dev/Global from r2r2m31

# oslevel
7.1.0.0
# oslevel -s
7100-00-01-1037
# lppchk -m3 -v
#
# ls1pp -l -0 u bos.rte
Fileset                                Level  State      Description
-----
Path: /usr/lib/objrepos
  bos.rte                               7.1.0.1 COMMITTED Base Operating System Runtime

# uname -w
1
# df
Filesystem    512-blocks    Free %Used    Iused %Iused Mounted on
Global        262144        205616  22%      1842    8% /
Global        262144        257320   2%         5    1% /home
Global        786432        377888  52%      8696   18% /opt
Global         -              -      -         -     - /proc

```

Global	262144	252456	4%	15	1% /tmp
Global	3932160	321192	92%	39631	51% /usr
Global	262144	94672	64%	4419	29% /var

---

Both the global system and the shared system WPAR have been successfully migrated to AIX V7.1.

In Example 3-74, a detached WPAR is migrated to AIX V7.1. Prior to migrating the WPAR, the global instance was migrated from AIX V6.1 to AIX V7.1.

**Note:** After the global instance migration to AIX V7.1, the detached version 6 WPAR (wpar0) is unable to start as it must be migrated first.

The **migwpar** command is called with **-d /images** flag and option. The **/images** directory is an NFS mounted file system that resides on a NIM master. The file system contains an AIX V7.1 LPP source on the NIM master.

Once the **migwpar** command has completed successfully, we start the WPAR and confirm that it has migrated to AIX V7.1. Only partial output from the **migwpar** command is shown as the actual migration log is extremely verbose.

*Example 3-74 Migrating a detached WPAR to AIX V7.1*

---

```
# uname -W
0
# oslevel -s
7100-00-01-1037
# lswpar
Name   State  Type  Hostname  Directory      RootVG WPAR
-----
wpar0  D      S     wpar0    /wpars/wpar0  no

# startwpar -v wpar0
Starting workload partition wpar0.
Mounting all workload partition file systems.
Mounting /wpars/wpar0
Mounting /wpars/wpar0/home
Mounting /wpars/wpar0/opt
Mounting /wpars/wpar0/proc
Mounting /wpars/wpar0/tmp
Mounting /wpars/wpar0/usr
Mounting /wpars/wpar0/var
startwpar: 0960-667 The operating system level within the workload partition is not supported.
Unmounting all workload partition file systems.
Unmounting /wpars/wpar0/var.
Unmounting /wpars/wpar0/usr.
```

```

Unmounting /wpars/wpar0/tmp.
Unmounting /wpars/wpar0/proc.
Unmounting /wpars/wpar0/opt.
Unmounting /wpars/wpar0/home.
Unmounting /wpars/wpar0.
Return Status = FAILURE.
#
# mount 75021p01:/export/lppsrc/aix7101 /images
# df /images
Filesystem      512-blocks      Free %Used    Iused %Iused Mounted on
75021p01:/export/lppsrc/aix7101 29425664    4204400    86%    3384    1%
/images

# ls -ltr /images
total 0
drwxr-xr-x   3 root      system      256 Sep 09 09:31 RPMS
drwxr-xr-x   3 root      system      256 Sep 09 09:31 usr
drwxr-xr-x   3 root      system      256 Sep 09 09:31 installp

# migwpar -d /images wpar0

Detached WPAR list:
wpar0
WPAR wpar0 mount point:
/wpars/wpar0
Mounting all workload partition file systems.
Loading workload partition.
Saving system configuration files.

Checking for initial required migration space.
Setting up for base operating system restore.
/

Restoring base operating system.
Merging system configuration files.
.....
Installing and migrating software.
Updating install utilities.
.....
FILESET STATISTICS
-----
725 Selected to be installed, of which:
    720 Passed pre-installation verification
      5 Already installed (directly or via superseding filesets)
      2 Additional requisites to be automatically installed
-----
722 Total to be installed

+-----+

```



## Installing Software...

+-----+

```
installp: APPLYING software for:
          x1C.aix61.rte 11.1.0.1
```

```
. . . . . << Copyright notice for x1C.aix61 >> . . . . .
Licensed Materials - Property of IBM
```

```
5724X1301
```

```
Copyright IBM Corp. 1991, 2010.
```

```
Copyright AT&T 1984, 1985, 1986, 1987, 1988, 1989.
```

```
Copyright Unix System Labs, Inc., a subsidiary of Novell, Inc. 1993.
```

```
All Rights Reserved.
```

```
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.
```

```
. . . . . << End of copyright notice for x1C.aix61 >>. . . . .
```

```
Filesets processed: 1 of 722 (Total time: 4 secs).
```

```
installp: APPLYING software for:
          wio.vscsi 7.1.0.0
```

```
.....
```

```
Restoring device ODM database.
```

```
Shutting down all workload partition processes.
```

```
Unloading workload partition.
```

```
Unmounting all workload partition file systems.
```

```
Cleaning up ...
```

```
# startwpar -v wpar0
```

```
Starting workload partition wpar0.
```

```
Mounting all workload partition file systems.
```

```
Mounting /wpars/wpar0
```

```
Mounting /wpars/wpar0/home
```

```
Mounting /wpars/wpar0/opt
```

```
Mounting /wpars/wpar0/proc
```

```
Mounting /wpars/wpar0/tmp
```

```
Mounting /wpars/wpar0/usr
```

```
Mounting /wpars/wpar0/var
```

```
Loading workload partition.
```

```
Exporting workload partition devices.
```

```
Exporting workload partition kernel extensions.
```

```
Starting workload partition subsystem cor_wpar0.
```

```
0513-059 The cor_wpar0 Subsystem has been started. Subsystem PID is 7995618.
```

```
Verifying workload partition startup.
```

```
Return Status = SUCCESS.
```

```
#
```

```
# clogin wpar0
*****
*
*
* Welcome to AIX Version 7.1!
*
*
* Please see the README file in /usr/lpp/bos for information pertinent to
* this release of the AIX Operating System.
*
*
*****
Last login: Mon Sep 13 22:19:20 CDT 2010 on /dev/Global from 75021p03

# oslevel -s
7100-00-01-1037
```

---



# Continuous availability

This chapter discusses the topics related to continuous availability, including:

- ▶ 4.1, “Firmware-assisted dump” on page 102
- ▶ 4.2, “User keys enhancements” on page 110
- ▶ 4.3, “Cluster Data Aggregation Tool” on page 111
- ▶ 4.4, “Cluster Aware AIX” on page 117
- ▶ 4.5, “SCTP component trace and RTEC adoption” on page 137
- ▶ 4.6, “Cluster aware perfstat library interfaces” on page 139

## 4.1 Firmware-assisted dump

This section discusses the differences in firmware-assisted dump in AIX v7.1.

### 4.1.1 Default installation configuration

The introduction of the POWER6 processor based systems allowed system dumps to be firmware assisted. When performing a firmware-assisted dump, system memory is frozen and the partition rebooted which allows a new instance of the operating system to complete the dump.

Firmware-assisted dump is now the default dump type in AIX V7.1, when the hardware platform supports firmware-assisted dump.

The traditional dump remains the default dump type for AIX V6.1, even when the hardware platform supports firmware-assisted dump.

Firmware-assisted dump offers improved reliability over the traditional dump type, by rebooting the partition and using a new kernel to dump data from the previous kernel crash.

Firmware-assisted dump requires:

- ▶ POWER6 processor based or later hard platform
- ▶ The LPAR must have a minimum of 1.5GB memory
- ▶ The dump logical volume must be in the root volume group
- ▶ Paging space cannot be defined as the dump logical volume

In the unlikely event that a firmware-assisted system may encounter a problem with execution, the firmware-assisted dump will be substituted by a traditional dump for this instance.

Example 4-1 shows the `sysdumpdev -l` command output from an AIX V6.1 LPAR. The system dump type has not been modified from the default installation setting. The field type of dump displays `traditional`. This shows that the partition default dump type is traditional and not firmware-assisted dump.

*Example 4-1 The `sysdumpdev -l` output in AIX V6.1*

---

```
# oslevel -s
6100-00-03-0808
# sysdumpdev -l
primary          /dev/lg_dumplv
secondary        /dev/sysdumpnu11
```

```

copy directory      /var/adm/ras
forced copy flag    TRUE
always allow dump   FALSE
dump compression    ON
type of dump        traditional
#

```

---

Example 4-2 shows the **sysdumpdev -l** command output from an AIX V7.1 LPAR. The system dump type has not been modified from the default installation setting. The field `type of dump` displays `fw-assisted`. This shows that the AIX V7.1 partition default dump type is firmware assisted and not traditional.

*Example 4-2 The `sysdumpdev -l` output in AIX V7.1*

```

# oslevel -s
7100-00-00-0000
# sysdumpdev -l
primary          /dev/lg_dumplv
secondary        /dev/sysdumpnull
copy directory   /var/adm/ras
forced copy flag TRUE
always allow dump FALSE
dump compression ON
type of dump     fw-assisted
full memory dump disallow
#

```

---

## 4.1.2 Full memory dump options

When firmware-assisted dump is enabled, the **sysdumpdev -l** command will display the full memory dump option. The full memory dump option can be set with the **sysdumpdev -f** command. The full memory dump option will only be displayed when the dump type is firmware-assisted dump.

The full memory dump option specifies the mode in which the firmware assisted dump will operate. The administrator can configure firmware-assisted dump to allow, disallow or require the dump of the full system memory.

Table 4-1 on page 104 lists the full memory dump options available with the **sysdumpdev -f** command:

Table 4-1 Full memory dump options available with the sysdumpdev -f command

Option	Description
disallow	Selective memory dump only. A full memory system dump is not allowed. This is the default
allow   allow_full	The full memory system dump mode is allowed but is performed only when the operating system cannot properly handle the dump request.
require   require_full	The full memory system dump mode is allowed and is always performed.

In Example 4-3 the full memory dump option is changed from disallow to require by using the **sysdumpdev -f** command. When modifying the full memory dump option from disallow to require, the next firmware-assisted dump will always perform a full system memory dump.

Example 4-3 Setting the full memory dump option with the sysdumpdev -f command

---

```
# sysdumpdev -l
primary          /dev/lg_dumplv
secondary       /dev/sysdumpnull
copy directory  /var/adm/ras
forced copy flag TRUE
always allow dump FALSE
dump compression ON
type of dump    fw-assisted
full memory dump disallow
# sysdumpdev -f require
# sysdumpdev -l
primary          /dev/lg_dumplv
secondary       /dev/sysdumpnull
copy directory  /var/adm/ras
forced copy flag TRUE
always allow dump FALSE
dump compression ON
type of dump    fw-assisted
full memory dump require
#
```

---

### 4.1.3 Changing the dump type on AIX V7.1

The firmware-assisted dump may be changed to traditional dump by using the `sysdumpdev -t` command. Using the traditional dump functionality will not allow the full memory dump options in Table 4-1 on page 104 to be executed, as these options are only available with firmware-assisted dump.

Changing from firmware-assisted to traditional dump will take effect immediately and does not require a reboot of the partition. Example 4-4 shows the `sysdumpdev -t` command being used to change the dump type from firmware-assisted to the traditional dump.

*Example 4-4 Changing to the traditional dump on AIX V7.1*

---

```
# sysdumpdev -l
primary          /dev/lg_dumplv
secondary       /dev/sysdumpnull
copy directory  /var/adm/ras
forced copy flag TRUE
always allow dump FALSE
dump compression ON
type of dump    fw-assisted
full memory dump require
# sysdumpdev -t traditional
# sysdumpdev -l
primary          /dev/lg_dumplv
secondary       /dev/sysdumpnull
copy directory  /var/adm/ras
forced copy flag TRUE
always allow dump FALSE
dump compression ON
type of dump    traditional
```

---

**Note:** When reverting to traditional dump, the full memory dump options are no longer available as these are options only available with firmware-assisted dump.

A partition configured to use the traditional dump may have the dump type changed to firmware-assisted. If the partition had previously been configured to use firmware-assisted dump, any full memory dump options will be preserved and defined when firmware-assisted dump is reinstated.

Changing from traditional to firmware-assisted dump requires a reboot of the partition for the dump changes to take effect.

**Note:** Firmware-assisted dump may be configured on POWER5™ based or earlier based hardware, but all system dumps will operate as traditional dump. POWER6 is the minimum hardware platform required to support firmware-assisted dump.

Example 4-5 shows the **sysdumpdev -t** command being used to reinstate firmware-assisted dump on a server configured to use the traditional dump.

*Example 4-5 Reinstating firmware-assisted dump with the sysdumpdev -t command*

---

```
# sysdumpdev -l
primary          /dev/lg_dumplv
secondary        /dev/sysdumpnull
copy directory   /var/adm/ras
forced copy flag TRUE
always allow dump FALSE
dump compression ON
type of dump     traditional
# sysdumpdev -t fw-assisted
Attention: the firmware-assisted system dump will be configured at the
next reboot.
# sysdumpdev -l
primary          /dev/lg_dumplv
secondary        /dev/sysdumpnull
copy directory   /var/adm/ras
forced copy flag TRUE
always allow dump FALSE
dump compression ON
type of dump     traditional
```

---

In Example 4-5 the message Attention: the firmware-assisted system dump will be configured at the next reboot is displayed once the **sysdumpdev -t fw-assisted** command has completed.

When a partition configured for firmware-assisted dump is booted, a portion of memory known as the *scratch area* is allocated to be used by the firmware-assisted dump functionality. For this reason, a partition configured to use the traditional system dump requires a reboot to allocate the *scratch area* memory that is required for a firmware-assisted dump to be initiated.

If the partition is not rebooted, firmware-assisted dump will not be activated until such a time as the partition reboot is completed.



**Note:** When the administrator attempts to switch from a traditional to firmware-assisted system dump, system memory is checked against the firmware-assisted system dump memory requirements. If these memory requirements are not met, then the **sysdumpdev -t** command output reports the required minimum system memory to allow for firmware-assisted dump to be configured.

Example 4-6 shows the partition reboot to allow for memory allocation and activation of firmware-assisted dump. Though firmware-assisted dump has been enabled, the **sysdumpdev -l** command displays the dump type as traditional because the partition has not yet been rebooted after the change to firmware-assisted dump.

*Example 4-6 Partition reboot to activate firmware-assisted dump*

---

```
# sysdumpdev -l
primary          /dev/lg_dumplv
secondary       /dev/sysdumpnull
copy directory  /var/adm/ras
forced copy flag TRUE
always allow dump FALSE
dump compression ON
type of dump     traditional
# shutdown -Fr

SHUTDOWN PROGRAM
...
...
Stopping The LWI Nonstop Profile...
Waiting for The LWI Nonstop Profile to exit...
Stopped The LWI Nonstop Profile.
0513-044 The sshd Subsystem was requested to stop.

Wait for 'Rebooting...' before stopping.
Error reporting has stopped.
Advanced Accounting has stopped...
Process accounting has stopped.
```

---

Example 4-7 on page 108 shows the partition after the reboot. The type of dump is displayed by using the **sysdumpdev -l** command, showing that the dump type is now set to fw-assisted.

As this is the same partition that we previously modified the full memory dump option to require, then changed the type of dump to traditional, the full memory dump option is reinstated once the dump type is reverted to firmware-assisted.

*Example 4-7 The sysdumpdev -l command after partition reboot*

---

```
# uptime
 06:15PM up 1 min, 1 user, load average: 1.12, 0.33, 0.12
# sysdumpdev -l
primary          /dev/lg_dumplv
secondary        /dev/sysdumpnull
copy directory   /var/adm/ras
forced copy flag TRUE
always allow dump FALSE
dump compression ON
type of dump     fw-assisted
full memory dump require
#
```

---

#### 4.1.4 Firmware-assisted dump on POWER5 and earlier hardware

The minimum supported hardware platform for firmware-assisted dump is the POWER6 processor based system.

In Example 4-8 we see a typical message output when attempting to enable firmware assisted dump on a pre POWER6 processor based system. In this example the AIX V7.1 is operating on a POWER5 model p550 system.

*Example 4-8 Attempting to enable firmware-assisted dump on a POWER5*

---

```
# oslevel -s
7100-00-00-0000
# uname -M
IBM,9113-550
# lsattr -El proc0
frequency 1654344000 Processor Speed False
smt_enabled true Processor SMT enabled False
smt_threads 2 Processor SMT threads False
state enable Processor state False
type PowerPC_POWER5 Processor type False
# sysdumpdev -l
primary          /dev/hd6
secondary        /dev/sysdumpnull
copy directory   /var/adm/ras
forced copy flag TRUE
```

```

always allow dump    FALSE
dump compression    ON
type of dump        traditional
# sysdumpdev -t fw-assisted
Cannot set the dump force_system_dump attribute.
    An attempt was made to set an attribute to an unsupported
value.
Firmware-assisted system dump is not supported on this platform.
# sysdumpdev -l
primary              /dev/hd6
secondary           /dev/sysdumpnull
copy directory      /var/adm/ras
forced copy flag    TRUE
always allow dump   FALSE
dump compression   ON
type of dump        traditional
#

```

---

In Example 4-8 on page 108, even though AIX V7.1 supports firmware-assisted dump as the default dump type, the POWER5 hardware platform does not support firmware-assisted dump, so the dump type at AIX V7.1 installation was set to traditional.

When the dump type was changed to firmware-assisted using the **sysdumpdev -t** command, the message Firmware-assisted system dump is not supported on this platform was displayed and the dump type remains set to traditional.

#### 4.1.5 Firmware-assisted dump support for non boot iSCSI device

The release of AIX Version 6.1 with the 6100-01 Technology Level introduced support for an iSCSI device to be configured as a dump device for firmware-assisted system dump.

The **sysdumpdev** command could be used to configure an iSCSI logical volume as a dump device. In AIX V6.1, it was mandatory that this dump device be located on an iSCSI boot device.

With the release of AIX V7.1, firmware-assisted dump also supports dump devices located on arbitrary non-boot iSCSI disks. This allows diskless servers to dump to remote iSCSI disks using firmware-assisted dump. The iSCSI disks must be members of the root volume group.

## 4.2 User keys enhancements

AIX 7.1 allows for configuring the number of user storage keys. It also allows a mode where all hardware keys are dedicated to user keys. This helps in developing large applications to use more user keys for application specific needs.

**Note:** By dedicating all of the hardware keys to user keys, kernel storage keys will get disabled. However we do *not* recommend this, because the kernel storage keys will not be able to help debugging the kernel memory problems any more if they are disabled.

Table 4-2 lists the maximum number of supported hardware keys on different hardware platforms.

Table 4-2 Number of storage keys supported

Power hardware platform	Maximized supported hardware keys on AIX
P5++	4
P6	8
P6+	15
P7	31

The **skctl** command is used to configure storage keys. Example 4-9 shows the usage of this command. It also shows how to view the existing settings and how to modify them.

The **smitty skctl** fastpath can also be used to configure storage keys. So one can use either **skctl** command or the **smitty skctl** interface for configuration.

Example 4-9 Configuring storage keys

---

```
# skctl -?
skctl: Not a recognized flag: ?
skctl: usage error
Usage: skctl [-D]
       skctl [-u <nukeys>/off] [-k on/off/default]
       skctl [-v [now|default|boot]
```

where:

```
no. of hardware keys) -u <nukeys> # number of user keys (2 - max.
                        -u off    # disable user keys
```

```

-k on/off      # enable/disable kernel keys
-k default    # set default kernel key state
-D            # use defaults
-v now        # view current settings
-v default    # view defaults
-v boot       # view settings for next boot

# skctl -v default
Default values for Storage Key attributes:

Max. number of hardware keys      = 31
Number of hardware keys enabled   = 31
Number of user keys                = 7
Kernel keys state                  = enabled

# skctl -v now
Storage Key attributes for current boot session:

Max. number of hardware keys      = 31
Number of hardware keys enabled   = 31
Number of user keys                = 12
Kernel keys state                  = enabled

# skctl -u 15
# skctl -v boot
Storage Key attributes for next boot session:

Max. number of hardware keys      = default
Number of hardware keys enabled   = default
Number of user keys                = 15
Kernel keys state                  = default

```

---

## 4.3 Cluster Data Aggregation Tool

First Failure Data Capture (FFDC) is a technique which ensures that when a fault is detected in a system (through error checkers or other types of detection methods), the root cause of the fault will be captured without the need to recreate the problem or run any sort of extended tracing or diagnostics program. Further information on FFDC can be found in *IBM AIX Continuous Availability Features*, REDP-4367.

FFDC has been enhanced to provide capabilities for quick analysis and root cause identification for problems that arise in workloads that span multiple systems. FFDC data will be collected on each of the configured nodes by the Cluster Data Aggregation Tool.

The Cluster Data Aggregation Tool environment is made of a central node and remote nodes. The central node is where the Cluster Data Aggregation Tool is installed on and executed from. It hosts the data collection repository, which is a new file system which contains collection of data from multiple remote nodes; The remote nodes are where FFDC data are collected, which are AIX LPARs (AIX 6.1 TL3), VIOS (2.1.1.0 based on AIX 6.1 TL3), or HMC (V7 R 3.4.2). The central node must be able to connect as an administrator user on the remote nodes. There is no need to install the Cluster Data Aggregation Tool on these remote nodes. For making a secure connection, SSH package should be installed on these nodes.

The Cluster Data Aggregation Tool is known by the **cdat** command. It is divided into several subcommands. The subcommands are **init**, **show**, **check**, **delete**, **discover-nodes**, **list-nodes**, **access**, **collect**, **list-types**, and **archive**. Only the **init** subcommand needs to be executed by the privileged user (root). The **init** subcommand creates the data infrastructure and defines the user used to run all other subcommands. It initializes the Cluster Data Aggregation repository.

**Note:** To prevent concurrent accesses to the Cluster Data Aggregation Tool configuration files, running multiple instances of the **cdat** command is forbidden and the repository is protected by a lock file.

The **smitty cdat** fastpath can also be used to configure Cluster Data Aggregation Tool. So one can use either **cdat** command or the **smitty cdat** interface for configuration.

Example 4-10 shows usage of **cdat** command in configuring Cluster Data Aggregation Tool.

*Example 4-10 Configuring Cluster Data Aggregation Tool*

---

```
# cdat -?
0965-030: Unknown sub-command: '-?'.
```

Usage: cdat sub-command [options]

Available sub-commands:

init	Initialize the repository
show	Display the content of the repository
check	Check consistency of the repository
delete	Remove collects from the repository
discover-nodes	Find LPARs or WPARs from a list of HMCs or LPARs
list-nodes	Display the list of configured nodes
access	Manage remote nodes authentication
collect	Collect data from remote nodes

list-types	Display the list of supported collect types
archive	Create a compressed archive of collects

```
# cdat init
Checking user cdat...Creating missing user.
Changing password for "cdat"
cdat's New password:
Enter the new password again:
Checking for SSH...found
Checking for SSH keys...generated
Checking directory /cdat...created
Checking XML file...created
Done.

# cdat show
Repository: /cdat
Local user: cdat

# cdat check
Repository is valid.

# cdat discover-nodes -?
Unknown option: ?
Usage: cdat discover-nodes -h
       cdat discover-nodes [-a|-w] [-f File] -n Type:[User@]Node ...

# cdat discover-nodes -n HMC:hscroot@192.168.100.111
Discovering nodes managed by hscroot@192.168.100.111...
The authenticity of host '192.168.100.111 (192.168.100.111)' can't be
established.
RSA key fingerprint is ee:5e:55:37:df:31:b6:78:1f:01:6d:f5:d1:67:d6:4f.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.100.111' (RSA) to the list of known
hosts.
Password:
Done.

# cat /cdat/nodes.txt
HMC:192.168.100.111
# LPARs of managed system 750_1-8233-E8B-061AA6P
LPAR:750_1_LP01
LPAR:750_1_LP02
LPAR:750_1_LP03
LPAR:750_1_LP04
VIOS:750_1_VIO_1
```

```
# Could not retrieve LPARs of managed system 750_2-8233-E8B-061AB2P
# HSCLO237 This operation is not allowed when the managed system is in
the No Connection state. After you have established a connection from
the HMC to the managed system and have entered a valid HMC access
password, try the operation again.
```

```
# cdat list-nodes
HMC 192.168.100.111
LPAR 750_1_LP01
LPAR 750_1_LP02
LPAR 750_1_LP03
LPAR 750_1_LP04
VIOS 750_1_VIO_1
```

```
# cdat list-types
List of available collect types:
```

```
perfpmr (/usr/lib/cdat/types/perfpmr):
  Retrieves the result of the perfpmr command from nodes of type
  LPAR.
```

```
psrasgrab (/usr/lib/cdat/types/psrasgrab):
  Harvests logs from a Centralized RAS Repository.
```

```
psrasinit (/usr/lib/cdat/types/psrasinit):
  Configures Centralized RAS pureScale clients.
```

```
psrasremove (/usr/lib/cdat/types/psrasremove):
  Unconfigures Centralized RAS pureScale clients.
```

```
snap (/usr/lib/cdat/types/snap):
  Gathers system configuration information from nodes of type LPAR or
  VIOS.
```

```
trace (/usr/lib/cdat/types/trace):
  Records selected system events from nodes of type LPAR or VIOS.
```

```
# cdat access -?
Unknown option: ?
Usage: cdat access -h
      cdat access [-dF] [-u User] -n Type:[User@]Node ...
      cdat access [-dF] [-u User] -f File ...
```

```
# cdat access -n LPAR:root@192.168.101.13 -n LPAR:root@192.168.101.11
The collect user will be created with the same password on all nodes.
```



```
Please enter a password for the collect user:
Re-enter the collect user password:
Initializing access to 'root' on host '192.168.101.13'...
Trying 'ssh'...found
The authenticity of host '192.168.101.13 (192.168.101.13)' can't be
established.
RSA key fingerprint is de:7d:f9:ec:8f:ee:e6:1e:8c:aa:18:b3:54:a9:d4:e0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.101.13' (RSA) to the list of known
hosts.
root@192.168.101.13's password:
Initializing access to 'root' on host '192.168.101.11'...
Trying 'ssh'...found
The authenticity of host '192.168.101.11 (192.168.101.11)' can't be
established.
RSA key fingerprint is 28:98:b8:d5:97:ec:86:84:d5:9e:06:ac:3b:b4:c6:5c.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.101.11' (RSA) to the list of known
hosts.
root@192.168.101.11's password:
Done.
```

```
# cdat collect -t trace -n LPAR:root@192.168.101.13 -n
LPAR:root@192.168.101.11
Is the collect for IBM support? (y/n) [y]: y
Please enter a PMR number: 12345,678,123
See file /cdat/00000003/logs.txt for detailed status.
Starting collect type "trace"
Collect type "trace" done, see results in "/cdat/00000003/trace/".
```

```
=====
Status report:
```

```
=====
```

```
192.168.101.11: SUCCEEDED
192.168.101.13: SUCCEEDED
```

```
# find /cdat/00000003/trace/
/cdat/00000003/trace/
/cdat/00000003/trace/192.168.101.11
/cdat/00000003/trace/192.168.101.11/logs.txt
/cdat/00000003/trace/192.168.101.11/trcfile
/cdat/00000003/trace/192.168.101.11/trcfmt
/cdat/00000003/trace/192.168.101.13
/cdat/00000003/trace/192.168.101.13/logs.txt
/cdat/00000003/trace/192.168.101.13/trcfile
/cdat/00000003/trace/192.168.101.13/trcfmt
```

```
# cdat show -v
Repository: /cdat
Local user: cdat

1: 2010-08-31T12:39:29

    PMR: 12345,123,123
    Location: /cdat/00000001/

2: 2010-08-31T12:40:24

    PMR: 12345,123,123
    Location: /cdat/00000002/

3: 2010-08-31T12:58:31

    PMR: 12345,678,123
    Location: /cdat/00000003/

192.168.101.11:
    type      : LPAR
    user      : root
    machine id : 00F61AA64C00
    lpar id   : 2
    timezone  : EDT

192.168.101.13:
    type      : LPAR
    user      : root
    machine id : 00F61AA64C00
    lpar id   : 4
    timezone  : EDT

# cdat archive -p 12345,678,123 -f archive
Compressed archive successfully created at archive.tar.Z.
```

---

It is possible to schedule periodic data collections using the **crontab** command. For instance, to run the snap collect type every day at midnight:

```
# crontab -e cdat

0 0 * * * /usr/bin/cdat collect -q -t snap -f /cdat/nodes.txt
```

With this configuration, `cdat` will create a new directory under `/cdat` (and a new collect id) every day at midnight that will contain the snap data for each node present in `/cdat/nodes.txt`.

Scheduled collects can also be managed transparently using the `smitty cdat_schedule` fastpath.

## 4.4 Cluster Aware AIX

The Cluster Aware AIX (CAA) services help in creating and managing a cluster of AIX nodes to build a highly available and an ideal architectural solution for a data center. IBM cluster products such as Reliable Scalable Cluster Technology (RSCT) and PowerHA use these services. CAA services can assist in the management and monitoring of an arbitrary set of nodes or in running a third-party cluster software.

The rest of the chapter discusses more details about each of these services together with examples using commands to configure and manage the cluster.

CAA services are basically a set of commands and services which the cluster software can exploit to provide high availability and disaster recovery support to external applications. The CAA services are broadly classified into the following:

Clusterwide event management	The AIX Event Infrastructure (5.12, “AIX Event Infrastructure extension and RAS” on page 189) allows event propagation across the cluster so that applications can monitor events from any node in the cluster.
Clusterwide storage naming service	When a cluster is defined or modified, the AIX interfaces automatically create a consistent shared device view across the cluster. A global device name, such as, <code>cdisk1</code> , would refer to the same physical disk from any node in the cluster.
Clusterwide command distribution	The <code>c1cmd</code> command provides a facility to distribute a command to a set of nodes that are members of a cluster. For example, the command <code>c1cmd date</code> returns the output of the <code>date</code> command from each of the nodes in the cluster.

Clusterwide communication	Communication between nodes within the cluster is achieved using multicasting over the IP based network and also using storage interface communication through Fibre Channel and SAS adapters. A new socket family (AF_CLUSTER) has been provided for reliable, in-order communication between nodes. When all network interfaces are lost, applications using these interfaces can still run.
---------------------------	--

The nodes which are part of the cluster should have common storage devices, either through the Storage Attached Network (SAN) or through the Serial-Attached SCSI (SAS) subsystems.

#### 4.4.1 Cluster configuration

This section describes the commands used to create and manage clusters. A sample cluster is created to explain the usage of these commands. Table 4-3 lists these commands together with their brief description.

Table 4-3 Cluster commands

Command	Description
<b>mkcluster</b>	Used to create a cluster.
<b>chcluster</b>	Used to change cluster configuration.
<b>rmcluster</b>	Used to remove cluster configuration.
<b>lscluster</b>	Used to list cluster configuration information.
<b>c1cmd</b>	Used to distribute a command to a set of nodes that are members of a cluster.

Below is a sample of creating a cluster on one of the nodes, nodeA. Before creating the cluster the `lscluster` command is used to make sure that no cluster already exists. The list of physical disks is displayed using the `lspv` command to help determine which disks to choose. Note down the names of the disks that will be used for the shared cluster disks, `hdisk4`, `hdisk5`, `hdisk6` and `hdisk7`. Example 4-11 shows the output of the commands used to determine the information needed before creating the cluster.

*Example 4-11 Before creating cluster*

---

```
# hostname
```

```

nodeA
# lscluster -m
Cluster services are not active.
# lspv
hdisk0          00cad74fd6d58ac1      rootvg          active
hdisk1          00cad74fa9d3b7e1      None
hdisk2          00cad74fa9d3b8de      None
hdisk3          00cad74f3964114a      None
hdisk4          00cad74f3963c575      None
hdisk5          00cad74f3963c671      None
hdisk6          00cad74f3963c6fa      None
hdisk7          00cad74f3963c775      None
hdisk8          00cad74f3963c7f7      None
hdisk9          00cad74f3963c873      None
hdisk10         00cad74f3963ca13      None
hdisk11         00cad74f3963caa9      None
hdisk12         00cad74f3963cb29      None
hdisk13         00cad74f3963cba4      None

```

---

The **mkcluster** command is used to create the cluster. Example 4-12 shows the use of **mkcluster** command.

The **-r** option is used to specify the repository disk used for storing cluster configuration information.

The **-d** option is used to specify cluster disks, each of which will be renamed to a new name beginning with **cldisk\***. Each of these cluster disks can be referenced by the new name from any of the nodes in the cluster. These new disk names refer to the same physical disk.

The **-s** option is used to specify the multicast address that is used for communication between the nodes in the cluster.

The **-m** option is used to specify the nodes which will be part of the cluster. Nodes are identified by the fully qualified hostnames as defined in DNS or with the local **/etc/hosts** file configuration.

The **lscluster** command is used to verify the creation of cluster. The **lspv** command shows the new names of the cluster disks.

*Example 4-12 Creating the cluster*

---

```

# mkcluster -r hdisk3 -d hdisk4,hdisk5,hdisk6,hdisk7 -s 227.1.1.211 -m
nodeA,nodeB,nodeC
Preserving 23812 bytes of symbol table [/usr/lib/drivers/ahafs.ext]
Preserving 19979 bytes of symbol table [/usr/lib/drivers/dpcomdd]
mkcluster: Cluster shared disks are automatically renamed to names such as
          cldisk1, [cldisk2, ...] on all cluster nodes. However, this cannot

```

take place while a disk is busy or on a node which is down or not reachable. If any disks cannot be renamed now, they will be renamed later by the clconfd daemon, when the node is available and the disks are not busy.

```
# lscluster -m
Calling node query for all nodes
Node query number of nodes examined: 3

Node name: nodeC
Cluster shorthand id for node: 1
uuid for node: 40752a9c-b687-11df-94d4-4eb040029002
State of node: UP
Smoothed rtt to node: 7
Mean Deviation in network rtt to node: 3
Number of zones this node is a member in: 0
Number of clusters node is a member in: 1
CLUSTER NAME      TYPE  SHID  UUID
SIRCOL_nodeA local      89320f66-ba9c-11df-8d0c-001125bfc896

Number of points_of_contact for node: 1
Point-of-contact interface & contact state
en0 UP

-----

Node name: nodeB
Cluster shorthand id for node: 2
uuid for node: 4001694a-b687-11df-80ec-000255d3926b
State of node: UP
Smoothed rtt to node: 7
Mean Deviation in network rtt to node: 3
Number of zones this node is a member in: 0
Number of clusters node is a member in: 1
CLUSTER NAME      TYPE  SHID  UUID
SIRCOL_nodeA local      89320f66-ba9c-11df-8d0c-001125bfc896

Number of points_of_contact for node: 1
Point-of-contact interface & contact state
en0 UP

-----

Node name: nodeA
Cluster shorthand id for node: 3
uuid for node: 21f1756c-b687-11df-80c9-001125bfc896
State of node: UP  NODE_LOCAL
Smoothed rtt to node: 0
Mean Deviation in network rtt to node: 0
```

```

Number of zones this node is a member in: 0
Number of clusters node is a member in: 1
CLUSTER NAME      TYPE  SHID  UUID
SIRCOL_nodeA local      89320f66-ba9c-11df-8d0c-001125bfc896

```

```

Number of points_of_contact for node: 0
Point-of-contact interface & contact state
n/a

```

```

# lspv
hdisk0          00cad74fd6d58ac1          rootvg          active
hdisk1          00cad74fa9d3b7e1          None
hdisk2          00cad74fa9d3b8de          None
caa_private0    00cad74f3964114a          caavg_private  active
cldisk4        00cad74f3963c575          None
cldisk3        00cad74f3963c671          None
cldisk2        00cad74f3963c6fa          None
cldisk1        00cad74f3963c775          None
hdisk8          00cad74f3963c7f7          None
hdisk9          00cad74f3963c873          None
hdisk10         00cad74f3963ca13          None
hdisk11         00cad74f3963caa9          None
hdisk12         00cad74f3963cb29          None
hdisk13         00cad74f3963cba4          None

```

**Note:** The `-n` option of `mkcluster` command can be used to specify an explicit name for the cluster. For a detailed explanation of these options, refer to the manpages.

As soon as the cluster is created, other active nodes of the cluster will configure and join into the cluster. The `lsccluster` command is executed from one of the other nodes in the cluster to verify cluster configuration. Example 4-13 shows the output from the `lsccluster` command from the node nodeB. Observe the State of node field in the `lsccluster` command. It gives you the latest status of the node as seen from the node where the `lsccluster` command is executed. A value of `NODE_LOCAL` indicates that this node is the local node where the `lsccluster` command is executed.

*Example 4-13 Verifying the cluster from another node*

```

# hostname
nodeB
# lsccluster -m
Calling node query for all nodes
Node query number of nodes examined: 3

```

```

Node name: nodeC

```

Cluster shorthand id for node: 1  
 uuid for node: 40752a9c-b687-11df-94d4-4eb040029002  
 State of node: UP  
 Smoothed rtt to node: 7  
 Mean Deviation in network rtt to node: 3  
 Number of zones this node is a member in: 0  
 Number of clusters node is a member in: 1  
 CLUSTER NAME      TYPE SHID    UUID  
 SIRCOL\_nodeA local            89320f66-ba9c-11df-8d0c-001125bfc896

Number of points\_of\_contact for node: 1  
 Point-of-contact interface & contact state  
 en0 UP

-----

Node name: nodeB  
 Cluster shorthand id for node: 2  
 uuid for node: 4001694a-b687-11df-80ec-000255d3926b  
**State of node: UP NODE\_LOCAL**  
 Smoothed rtt to node: 0  
 Mean Deviation in network rtt to node: 0  
 Number of zones this node is a member in: 0  
 Number of clusters node is a member in: 1  
 CLUSTER NAME      TYPE SHID    UUID  
 SIRCOL\_nodeA local            89320f66-ba9c-11df-8d0c-001125bfc896

Number of points\_of\_contact for node: 0  
 Point-of-contact interface & contact state  
 n/a

-----

Node name: nodeA  
 Cluster shorthand id for node: 3  
 uuid for node: 21f1756c-b687-11df-80c9-001125bfc896  
 State of node: UP  
 Smoothed rtt to node: 7  
 Mean Deviation in network rtt to node: 3  
 Number of zones this node is a member in: 0  
 Number of clusters node is a member in: 1  
 CLUSTER NAME      TYPE SHID    UUID  
 SIRCOL\_nodeA local            89320f66-ba9c-11df-8d0c-001125bfc896

Number of points\_of\_contact for node: 1  
 Point-of-contact interface & contact state



---

 en0 UP
 

---

Example 4-14 shows the output from the `lscluster -c` command to display basic cluster configuration information. The cluster name is `SIRCOL_nodeA`. An explicit cluster name can also be specified using the `-n` option to the `mkcluster` command. A unique Cluster `uuid` is generated for the cluster. Each of the nodes are assigned a unique Cluster `id`.

*Example 4-14 Displaying basic cluster configuration*

---

```
# lscluster -c
Cluster query for cluster SIRCOL_nodeA returns:
Cluster uuid: 89320f66-ba9c-11df-8d0c-001125bfc896
Number of nodes in cluster = 3
    Cluster id for node nodeC is 1
    Primary IP address for node nodeC is 9.126.85.51
    Cluster id for node nodeB is 2
    Primary IP address for node nodeB is 9.126.85.14
    Cluster id for node nodeA is 3
    Primary IP address for node nodeA is 9.126.85.13
Number of disks in cluster = 4
    for disk cldisk4 UUID = 60050763-05ff-c02b-0000-000000001114
cluster_major = 0 cluster_minor = 4
    for disk cldisk3 UUID = 60050763-05ff-c02b-0000-000000001115
cluster_major = 0 cluster_minor = 3
    for disk cldisk2 UUID = 60050763-05ff-c02b-0000-000000001116
cluster_major = 0 cluster_minor = 2
    for disk cldisk1 UUID = 60050763-05ff-c02b-0000-000000001117
cluster_major = 0 cluster_minor = 1
Multicast address for cluster is 227.1.1.211
```

---

Example 4-15 shows the output from the `lscluster -d` command displaying cluster storage interfaces. Observe the `state` field for each of the disks which gives the latest state of the corresponding disk. The `type` field is used to represent whether it is a cluster disk or a repository disk.

*Example 4-15 Displaying cluster storage interfaces*

---

```
# lscluster -d
Storage Interface Query

Cluster Name: SIRCOL_nodeA
Cluster uuid: 89320f66-ba9c-11df-8d0c-001125bfc896
Number of nodes reporting = 3
Number of nodes expected = 3
```

```

Node nodeA
Node uuid = 21f1756c-b687-11df-80c9-001125bfc896
Number of disk discovered = 5
    cldisk4
        state : UP
        uDid : 200B75CWLN1111407210790003IBMfcp
        uUid : 60050763-05ff-c02b-0000-000000001114
        type : CLUSDISK
    cldisk3
        state : UP
        uDid : 200B75CWLN1111507210790003IBMfcp
uUid : 60050763-05ff-c02b-0000-000000001115
        type : CLUSDISK
    cldisk2
        state : UP
        uDid : 200B75CWLN1111607210790003IBMfcp
        uUid : 60050763-05ff-c02b-0000-000000001116
        type : CLUSDISK
    cldisk1
        state : UP
        uDid : 200B75CWLN1111707210790003IBMfcp
        uUid : 60050763-05ff-c02b-0000-000000001117
        type : CLUSDISK
caa_private0
    state : UP
    uDid :
    uUid : 60050763-05ff-c02b-0000-000000001113
    type : REPDISK

Node
Node uuid = 00000000-0000-0000-0000-000000000000
Number of disk discovered = 0
Node
Node uuid = 00000000-0000-0000-0000-000000000000
Number of disk discovered = 0

```

---

Example 4-16 shows the output from the `lscluster -s` command displaying cluster network statistics on the local node. The command gives statistical information regarding the type and amount of packets received or sent to other nodes within the cluster.

*Example 4-16 Displaying cluster network statistics*

---

```

# lscluster -s
Cluster Statistics:

```

## Cluster Network Statistics:

pkts seen:71843	pkts passed:39429
IP pkts:33775	UDP pkts:32414
gossip pkts sent:16558	gossip pkts rcv:24296
cluster address pkts:0	CP pkts:32414
bad transmits:0	bad posts:0
short pkts:0	multicast pkts:32414
cluster wide errors:0	bad pkts:0
dup pkts:1	pkt fragments:0
fragments queued:0	fragments freed:0
requests dropped:0	pkts routed:0
pkts pulled:0	no memory:0
rxmit requests rcv:7	requests found:4
requests missed:0	ooo pkts:0
requests reset sent:0	reset rcv:0
requests lnk reset send :0	reset lnk rcv:0
rxmit requests sent:3	
alive pkts sent:3	alive pkts rcv:0
ahafs pkts sent:4	ahafs pkts rcv:1
nodedown pkts sent:8	nodedown pkts rcv:3
socket pkts sent:294	socket pkts rcv:75
cwide pkts sent:33	cwide pkts rcv:45
socket pkts no space:0	pkts rcv notforhere:1918
stale pkts rcv:0	other cluster pkts:0
storage pkts sent:1	storage pkts rcv:1
out-of-range pkts rcv:0	

---

Example 4-17 shows the output from the `lscluster -i` command listing cluster configuration interfaces on the local node. The `Interface state` gives the latest state of the corresponding interfaces of each of the nodes.

*Example 4-17 Displaying cluster configuration interfaces*

---

```
# lscluster -i
Network/Storage Interface Query

Cluster Name:  SIRCOL_nodeA
Cluster uuid:  89320f66-ba9c-11df-8d0c-001125bfc896
Number of nodes reporting = 3
Number of nodes expected = 3
Node nodeA
Node uuid = 21f1756c-b687-11df-80c9-001125bfc896
Number of interfaces discovered = 2
    Interface number 1 en0
        ifnet type = 6 ndd type = 7
```

```

Mac address length = 6
Mac address = 0.11.25.bf.c8.96
Smoothed rrt across interface = 7
Mean Deviation in network rrt across interface = 3
Probe interval for interface = 100 ms
ifnet flags for interface = 0x5e080863
ndd flags for interface = 0x63081b
Interface state UP
Number of regular addresses configured on interface = 1
IPV4 ADDRESS: 9.126.85.13 broadcast 9.126.85.255 netmask
255.255.255.0
Number of cluster multicast addresses configured on interface =
1
IPV4 MULTICAST ADDRESS: 227.1.1.211 broadcast 0.0.0.0 netmask
0.0.0.0
Interface number 2 dpcom
ifnet type = 0 ndd type = 305
Mac address length = 0
Mac address = 0.0.0.0.0.0
Smoothed rrt across interface = 750
Mean Deviation in network rrt across interface = 1500
Probe interval for interface = 22500 ms
ifnet flags for interface = 0x0
ndd flags for interface = 0x9
Interface state UP RESTRICTED AIX_CONTROLLED
Node nodeC
Node uuid = 40752a9c-b687-11df-94d4-4eb040029002
Number of interfaces discovered = 2
Interface number 1 en0
ifnet type = 6 ndd type = 7
Mac address length = 6
Mac address = 4e.b0.40.2.90.2
Smoothed rrt across interface = 8
Mean Deviation in network rrt across interface = 3
Probe interval for interface = 110 ms
ifnet flags for interface = 0x1e080863
ndd flags for interface = 0x21081b
Interface state UP
Number of regular addresses configured on interface = 1
IPV4 ADDRESS: 9.126.85.51 broadcast 9.126.85.255 netmask
255.255.255.0
Number of cluster multicast addresses configured on interface =
1
IPV4 MULTICAST ADDRESS: 227.1.1.211 broadcast 0.0.0.0 netmask
0.0.0.0
Interface number 2 dpcom
ifnet type = 0 ndd type = 305
Mac address length = 0
Mac address = 0.0.0.0.0.0

```

```

        Smoothed rrt across interface = 750
Mean Deviation in network rrt across interface = 1500
        Probe interval for interface = 22500 ms
        ifnet flags for interface = 0x0
        ndd flags for interface = 0x9
        Interface state UP RESTRICTED AIX_CONTROLLED
Node nodeB
Node uuid = 4001694a-b687-11df-80ec-000255d3926b
Number of interfaces discovered = 2
    Interface number 1 en0
        ifnet type = 6 ndd type = 7
        Mac address length = 6
        Mac address = 0.2.55.d3.92.6b
        Smoothed rrt across interface = 7
        Mean Deviation in network rrt across interface = 3
        Probe interval for interface = 100 ms
        ifnet flags for interface = 0x5e080863
        ndd flags for interface = 0x63081b
        Interface state UP
        Number of regular addresses configured on interface = 1
        IPV4 ADDRESS: 9.126.85.14 broadcast 9.126.85.255 netmask
255.255.255.0
        Number of cluster multicast addresses configured on interface =
1
        IPV4 MULTICAST ADDRESS: 227.1.1.211 broadcast 0.0.0.0 netmask
0.0.0.0
    Interface number 2 dpcom
        ifnet type = 0 ndd type = 305
        Mac address length = 0
        Mac address = 0.0.0.0.0.0
        Smoothed rrt across interface = 750
        Mean Deviation in network rrt across interface = 1500
        Probe interval for interface = 22500 ms
        ifnet flags for interface = 0x0
        ndd flags for interface = 0x9
        Interface state UP RESTRICTED AIX_CONTROLLED

```

---

Cluster configuration can be modified using the **chcluster** command. Example 4-18 shows the use of the **chcluster** command. Here, the node nodeC is removed from the cluster. The **lcluster** command is used to verify the removal of nodeC from the cluster.

*Example 4-18 Deletion of node from cluster*

---

```

# chcluster -n SIRCOL_nodeA -m -nodeC
# lcluster -m
Calling node query for all nodes
Node query number of nodes examined: 2

```

```

Node name: nodeB
Cluster shorthand id for node: 2
uuid for node: 4001694a-b687-11df-80ec-000255d3926b
State of node: UP
Smoothed rtt to node: 7
Mean Deviation in network rtt to node: 3
Number of zones this node is a member in: 0
Number of clusters node is a member in: 1
CLUSTER NAME      TYPE SHID  UUID
SIRCOL_nodeA local      c5ea0c7a-bab9-11df-a75b-001125bfc896

Number of points_of_contact for node: 1
Point-of-contact interface & contact state
en0 UP

```

---

```

Node name: nodeA
Cluster shorthand id for node: 3
uuid for node: 21f1756c-b687-11df-80c9-001125bfc896
State of node: UP  NODE_LOCAL
Smoothed rtt to node: 0
Mean Deviation in network rtt to node: 0
Number of zones this node is a member in: 0
Number of clusters node is a member in: 1
CLUSTER NAME      TYPE SHID  UUID
SIRCOL_nodeA local      c5ea0c7a-bab9-11df-a75b-001125bfc896

Number of points_of_contact for node: 0
Point-of-contact interface & contact state
n/a

```

---

Similarly, Example 4-19 shows removal of cluster disk cldisk3 from the cluster.

*Example 4-19 Deletion of cluster disk from cluster*

---

```

# lspv |grep cldisk3
cldisk3          00cad74f3963c6fa          None
# chcluster -n SIRCOL_nodeA -d -cldisk3
chcluster: Removed cluster shared disks are automatically renamed to names such
as hdisk10, [hdisk11, ...] on all cluster nodes. However, this cannot
take place while a disk is busy or on a node which is down or not
reachable. If any disks cannot be renamed now, you must manually
rename them by removing them from the ODM database and then running
the cfgmgr command to recreate them with default names. For example:

```

```

        rmdev -l cldisk1 -d
        rmdev -l cldisk2 -d
        cfgmgr
# lspv |grep cldisk3
# lspv |grep cldisk*
cdisk1      00cad74f3963c575      None
cdisk4      00cad74f3963c671      None
cdisk2      00cad74f3963c775      None

```

---

Example 4-20 is another example showing addition of a new disk hdisk9 as a cluster disk. Notice that hdisk9 is renamed to cldisk5 after executing the **chcluster** command.

*Example 4-20 Addition of a disk to the cluster*

---

```

# chcluster -n SIRCOL_nodeA -d +hdisk9
chcluster: Cluster shared disks are automatically renamed to names such as
          cldisk1, [cdisk2, ...] on all cluster nodes. However, this cannot
          take place while a disk is busy or on a node which is down or not
          reachable. If any disks cannot be renamed now, they will be renamed
          later by the clconfd daemon, when the node is available and the disks
          are not busy.
# lspv |grep cldisk*
cdisk1      00cad74f3963c575      None
cdisk4      00cad74f3963c671      None
cdisk2      00cad74f3963c775      None
cdisk5      00cad74f3963c873      None

```

---

Example 4-21 shows use of the **rmcluster** command to remove cluster configuration. Note the output from the **lscluster** and **lspv** commands after the removal of the cluster.

*Example 4-21 Removal of cluster*

---

```

# rmcluster -n SIRCOL_nodeA
rmcluster: Removed cluster shared disks are automatically renamed to names such
          as hdisk10, [hdisk11, ...] on all cluster nodes. However, this cannot
          take place while a disk is busy or on a node which is down or not
          reachable. If any disks cannot be renamed now, you must manually
          rename them by removing them from the ODM database and then running
          the cfgmgr command to recreate them with default names. For example:
          rmdev -l cldisk1 -d
          rmdev -l cldisk2 -d
          cfgmgr
# lscluster -m
Cluster services are not active.
# lspv |grep cldisk*

```

---

The **c1cmd** command is used to distribute commands to one or more nodes which are part of the cluster. In Example 4-22, **c1cmd** command executes the **date** command on each of the nodes in the cluster and returns with their outputs.

*Example 4-22 Usage of c1cmd command*

---

```
# c1cmd -n SIRC0L_nodeA date
-----
NODE nodeA
-----
Wed Sep  8 02:13:58 PAKDT 2010
-----
NODE nodeB
-----
Wed Sep  8 02:14:00 PAKDT 2010
-----
NODE nodeC
-----
Wed Sep  8 02:13:58 PAKDT 2010
```

---

## 4.4.2 Cluster system architecture flow

When a cluster is created, various subsystems will get configured. The following list describes the process of the clustering subsystem:

- ▶ The cluster is created using the **mkcluster** command.
- ▶ The cluster configuration is written to the raw section of one of the shared disks designated as the cluster repository disk.
- ▶ Primary and secondary database nodes are selected from the list of candidate nodes in the **mkcluster** command. For the primary or secondary database failure, an alternate node is started to perform the role of a new primary or new secondary database node.
- ▶ Special volume groups and logical volumes are created on the cluster repository disk.
- ▶ Cluster file systems are created on the special volume group.
- ▶ The cluster repository database is created on both primary and secondary nodes.
- ▶ The cluster repository database is started.
- ▶ Cluster services are made available to other functions in the operating system, such as Reliable Scalable Cluster Technology (RSCT) and PowerHA SystemMirror.



- ▶ Storage framework register lists are created on the cluster repository disk.
- ▶ A global device namespace is created and interaction with LVM starts for handling associated volume group events.
- ▶ A clusterwide multicast address is established.
- ▶ The node discovers all of the available communication interfaces.
- ▶ The cluster interface monitoring starts.
- ▶ The cluster interacts with AIX Event Infrastructure for clusterwide event distribution.
- ▶ The cluster exports cluster messaging and cluster socket services to other functions in the operating system, such as Reliable Scalable Cluster Technology (RSCT) and PowerHA SystemMirror.

### 4.4.3 Cluster event management

The AIX event infrastructure is used for event management on AIX. For a detailed description, refer to 5.12, “AIX Event Infrastructure extension and RAS” on page 189. Table 4-4 lists the cluster specific events.

Table 4-4 Cluster events

Cluster events	Description
nodeList	Monitors changes in cluster membership
clDiskList	Monitors changes in cluster disk membership
nodeContact	Monitors the last contact status of the node in a cluster
nodeState	Monitors the state of the node in the cluster
nodeAddress	Alias is added or removed from a network interface
networkAdapterState	Monitors the network interface of a node in the cluster
clDiskState	Monitors clustered disks
repDiskState	Monitors the repository disk
diskState	Monitors the local disk changes
vgState	Verifies the status of the volume group on a disk

These events are propagated to all nodes in the cluster so that event monitoring applications will be notified as and when event happens on any node in the cluster.

## 4.4.4 Cluster socket programming

Cluster communications can operate over the traditional networking interfaces (IP-based) or using the storage interfaces (Fibre Channel or SAS).

When cluster communications is configured over both transports the redundancy and high availability of the underlying cluster node software and hardware configuration can be maximized by using all the paths for communications. In case of network interface failures, you can use the storage framework (Fibre Channel or SAS) to maintain communication between the cluster nodes. Cluster communications is achieved by exploiting the multicast capabilities of the networking and storage subsystems.

Example 4-23 provides a sample cluster family socket server and client program that is used to communicate between two nodes in the cluster.

The server will define port 29 to be used for communications.

Node A is identified as node 3 (the shorthand ID for node from the `lscluster -m` output).

Node B is identified as node 2 (the shorthand ID for node from the `lscluster -m` output).

### *Example 4-23 Cluster messaging example*

---

```
# hostname
nodeA
# ./server 29
```

```
Server Waiting for client on port 29
From cluster node: 2
Message: this is test message
```

```
# hostname
nodeB
# ./client 3 29 "this is test message"
```

```
->cat server.c
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <stdio.h>
#include <unistd.h>
#include <errno.h>
```

```

#include <string.h>
#include <stdlib.h>
#include <sys/cluster.h>
#include <cluster/cluster_var.h>

int
main(int argc, char *argv[])
{
    int          sock;
    unsigned long int addr_len, bytes_read;
    char         recv_data[1024];
    struct sockaddr_clust server_addr, client_addr;
    int          port;

    if (argc != 2) {
        fprintf(stdout, "Usage: ./server <port num>\n");
        exit(1);
    }
    if ((sock = socket(AF_CLUST, SOCK_DGRAM, 0)) == -1) {
        perror("Socket");
        exit(1);
    }
    port = atoi(argv[1]);
    bzero((char *) &server_addr, sizeof(server_addr));
    server_addr.sclust_family = AF_CLUST;
    server_addr.sclust_port = port;
    server_addr.sclust_cluster_id = WWID_LOCAL_CLUSTER;
    server_addr.sclust_addr = get_clusterid();
    if (bind(sock, (struct sockaddr *) & server_addr, sizeof(struct
sockaddr_clust)) == -1) {
        perror("Bind");
        exit(1);
    }
    addr_len = sizeof(struct sockaddr_clust);
    fprintf(stdout, "\nServer Waiting for client on port %d",
port);
    fflush(stdout);
    while (1) {
        bytes_read = recvfrom(sock, recv_data, 1024, 0, (struct
sockaddr *) & client_addr, &addr_len);
        recv_data[bytes_read] = '\0';
        fprintf(stdout, "\nFrom cluster node: %d",
client_addr.sclust_addr);
        fprintf(stdout, "\nMessage: %s\n", recv_data);
    }
}

```

```

        return 0;
    }

->cat client.c
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <netdb.h>
#include <stdio.h>
#include <unistd.h>
#include <errno.h>
#include <string.h>
#include <stdlib.h>
#include <sys/cluster.h>
#include <cluster/cluster_var.h>

#define MAX_MSG 100
int
main(int argc, char *argv[])
{
    int          sock, rc, i;
    struct sockaddr_clust sclust;
    struct hostent *host;
    char         send_data[1024];

    if (argc <= 3) {
        fprintf(stdout, "Usage: ./client <cluster ID of server>
<port> < MSG >");
        exit(1);
    }

    if ((sock = socket(AF_CLUSTER, SOCK_DGRAM, 0)) == -1) {
        perror("socket");
        exit(1);
    }

    bzero((char *) &sclust.sclust_len, sizeof(struct
sockaddr_clust));
    sclust.sclust_addr = atoi(argv[1]);
    sclust.sclust_len = sizeof(struct sockaddr_clust);
    sclust.sclust_family = AF_CLUSTER;
    sclust.sclust_cluster_id = WWID_LOCAL_CLUSTER;
    sclust.sclust_port = atoi(argv[2]);

```

```

rc = bind(sock, (struct sockaddr *) & sclust, sizeof(sclust));
if (rc < 0) {
    printf("%s: cannot bind port\n", argv[0]);
    exit(1);
}
/* send data */
for (i = 3; i < argc; i++) {
    rc = sendto(sock, argv[i], strlen(argv[i]) + 1, 0,
(struct sockaddr *) & sclust, sizeof(sclust));
    if (rc < 0) {
        printf("%s: cannot send data %d \n", argv[0], i
- 1);
        close(sock);
        exit(1);
    }
}
return 1;
}

```

---

#### 4.4.5 Cluster storage communication configuration

In order to be able to communicate using storage communication interfaces for high availability and redundancy of communication paths between nodes in the cluster, the storage adapters need to be configured.

The following information only applies to Fibre Channel adapters. No setup is necessary for SAS adapters. The following Fibre Channel adapters are supported:

- ▶ 4 GB Single-Port Fibre Channel PCI-X 2.0 DDR Adapter (FC 1905; CCIN 1910)
- ▶ 4 GB Single-Port Fibre Channel PCI-X 2.0 DDR Adapter (FC 5758; CCIN 280D)
- ▶ 4 GB Single-Port Fibre Channel PCI-X Adapter (FC 5773; CCIN 5773)
- ▶ 4 GB Dual-Port Fibre Channel PCI-X Adapter (FC 5774; CCIN 5774)
- ▶ 4 Gb Dual-Port Fibre Channel PCI-X 2.0 DDR Adapter (FC 1910; CCIN 1910)
- ▶ 4 Gb Dual-Port Fibre Channel PCI-X 2.0 DDR Adapter (FC 5759; CCIN 5759)
- ▶ 8 Gb PCI Express Dual Port Fibre Channel Adapter (FC 5735; CCIN 577D)
- ▶ 8 Gb PCI Express Dual Port Fibre Channel Adapter 1Xe Blade (FC 2B3A; CCIN 2607)

- ▶ 3 Gb Dual-Port SAS Adapter PCI-X DDR External (FC 5900 and 5912; CCIN 572A)

**Note:** For the most current list of supported Fibre Channel adapters, contact your IBM representative.

To configure the Fibre Channel adapters that will be used for cluster storage communications, complete the following steps:

**Note:** In the following steps the X in fcsX represents the number of your Fibre Channel adapters, for example, fcs1, fcs2, or fcs3.

1. Run the following command:

```
rmdev -R1 fcsX
```

**Note:** If you booted from the Fibre Channel adapter, you do not need to complete this step.

2. Run the following command:

```
chdev -l fcsX -a tme=yes
```

**Note:** If you booted from the Fibre Channel adapter, add the -P flag.

3. Run the following command:

```
chdev -l fscsiX -a dyntrk=yes -a fc_err_recov=fast_fail
```

4. Run the **cfgmgr** command.

**Note:** If you booted from the Fibre Channel adapter and used the -P flag, you must reboot.

5. Verify the configuration changes by running the following command:

```
lsdev -C | grep sfwcom
```

After you create the cluster, you can list the cluster interfaces and view the storage interfaces by running the following command:

```
lsccluster -i
```

*Example 4-24 Cluster storage communication configuration*


---

```

# rmdev -Rl fcs0
fcnet0 Defined
hdisk1 Defined
hdisk2 Defined
hdisk3 Defined
hdisk4 Defined
hdisk5 Defined
hdisk6 Defined
hdisk7 Defined
hdisk8 Defined
hdisk9 Defined
hdisk10 Defined
sfwcomm0 Defined
fscsi0 Defined
fcs0 Defined
# chdev -l fcs0 -a tme=yes
fcs0 changed
# chdev -l fscsi0 -a dyntrk=yes -a fc_err_recov=fast_fail
fscsi0 changed
# cfgmgr >cfg.out 2>&1
# lsdev -C | grep sfwcom
sfwcomm0   Defined   00-00-02-FF Fiber Channel Storage Framework Comm
sfwcomm1   Available 00-01-02-FF Fiber Channel Storage Framework Comm

```

---

**Note:** We recommend configuring cluster storage interfaces. The above set of commands used to configure the storage interfaces should be executed on all the nodes which are part of the cluster. The cluster should be created after configuring the interfaces on all the nodes.

## 4.5 SCTP component trace and RTEC adoption

The AIX enterprise Reliability Availability Serviceability (eRAS) infrastructure defines a component definition framework. This framework supports three distinct domains:

- ▶ Runtime Error Checking (RTEC)
- ▶ Component Trace (CT)
- ▶ Component Dump (CD)

The Stream Control Transmission Protocol (SCTP) implementation in AIX V7.1 and AIX V6.1 TL 6100-06 significantly enhances the adoption of the RAS

component framework for the RTEC and CT domains. To that extent the following two new trace hooks are defined:

- ▶ Event ID 6590 (0x659) with event label SCTP
- ▶ Event ID 65a0 (0x65a) with event label SCTP\_ERR

The previously existing base component *sctp* of the CT and RTEC component tree is complemented by an additional sub-component *sctp\_err*.

The integration into the component trace framework enables both the memory trace mode (private memory trace) and the user trace mode (system trace) for the base component and its new sub-component.

The CT SCTP component hierarchy of a given AIX configuration and the current settings for the memory trace mode and the user trace mode can be listed by the **ctctrl** command. The **ctctrl** command also allows you to modify the component trace related configuration parameters. The following **ctctrl** command output shows the default component trace configuration for the SCTP component just after the SCTP kernel extension has been loaded through the **sctpctrl load** command. As you can see the memory trace is set to normal (level=3) and the system trace level to detailed (level=7) for the sctp component and for the sctp.sctp\_err sub-component the memory trace level is set to minimal (level=1) and the system trace level to detailed (level=7):

```
75011p01:/> ctctrl -c sctp -q -r
```

Component name	Have alias	Mem Trc /level	Sys Trc /level	Buffer size /Allocated
sctp	NO	ON/3	ON/7	40960/YES
.sctp_err	NO	ON/1	ON/7	10240/YES

The RTEC SCTP component hierarchy of a given AIX configuration and the current settings for error checking level, disposition for low-severity errors, and disposition for medium-severity errors can be listed by the **errctrl** command. The **errctrl** command also allows you to modify the runtime error checking related configuration parameters. The following **errctrl** command output shows that the default error checking level for all SCTP components is normal (level=3), and that low-severity errors (LowSevDis=64), and medium-severity errors (MedSevDisp=64) are logged (collect service data and continue):

```
75011p01:/> errctrl -c sctp -q -r
```

Component name	Have alias	ErrChk /level	LowSev Disp	MedSev Disp
sctp	NO	ON/3	64	64



.sctp_err		NO		ON/3		64		64
-----------	--	----	--	------	--	----	--	----

The AIX SCTP implementation is intentionally not integrated with the AIX enterprise RAS Component Dump domain. A component dump temporarily suspends execution and the Stream Control Transmission Protocol may react negatively by false time-outs and failovers being perceived by peer nodes. However, a functionality similar to the component dump is delivered through the **dump** parameter of the **sctpctrl** command. This command has also been enhanced in AIX V7.1 and AIX V6.1 TL 6100-06 to provide an improved formatting of the command output.

## 4.6 Cluster aware perfstat library interfaces

IBM PowerHA is a high availability solution for AIX that provides automated failure detection, diagnosis, application recovery, and node reintegration.

It consists of two components:

- ▶ **High availability:** The process of ensuring an application is available for use through the use of duplicated and/or shared resources.
- ▶ **Cluster multi-processing:** Multiple applications running on the same nodes with shared or concurrent access to the data.

This High Availability solution demands two very important needs from the performance monitoring perspective

1. Ability to collect and analyze the performance data of the entire cluster at the aggregate level (from any node in the cluster).
2. Ability to collect and analyze the performance data of an individual node in the cluster (from any node in the cluster).

The **perfstat** application programming interface (API) is a collection of C programming language subroutines that execute in user space and uses the perfstat kernel extension to extract various AIX performance metrics.

Beginning with AIX v7.1 and AIX 6.1 TL06, the existing **perfstat** library is enhanced to support performance data collection and analysis for a single node or multiple nodes in a cluster. The enhanced perfstat library provides APIs to obtain performance metrics related to CPU, memory, I/O and others to provide performance statistics about a node within a cluster.

The perfstat library is also updated with a new interface called **perfstat\_cluster\_total** (similar to **perfstat\_partition\_total** interface) that provides cluster level aggregate data.

A separate interface called `perfstat_node_list` is also added to retrieve the list of nodes available in the cluster.

A new set of APIs (NODE interfaces) are available which return usage metrics related to a set of components or individual components specific to a remote node in a cluster.

**Note:** The `perfstat_config` (`PERFSTAT_ENABLE` | `PERFSTAT_CLUSTER_STATS`, `NULL`) must be used to enable the remote node statistics collection (available only in a cluster environment).

Once node related performance data is collected, `perfstat_config`(`PERFSTAT_DISABLE` | `PERFSTAT_CLUSTER_STATS`, `NULL`) must be used to disable collection of node/cluster statistics.

Here are the list of node interfaces that are added:

`perfstat_<subsystem>_node` Subroutines

### Purpose

Retrieve remote node's performance statistics of subsystem type. The list of subroutines are as follows:

- ▶ `perfstat_cpu_total_node`
- ▶ `perfstat_disk_node`
- ▶ `perfstat_disk_total_node`
- ▶ `perfstat_diskadapter_node`
- ▶ `perfstat_diskpath_node`
- ▶ `perfstat_logicalvolume_node`
- ▶ `perfstat_memory_page_node`
- ▶ `perfstat_memory_total_node`
- ▶ `perfstat_netbuffer_node`
- ▶ `perfstat_netinterface_node`
- ▶ `perfstat_netinterface_total_node`
- ▶ `perfstat_pagingpace_node`
- ▶ `perfstat_partition_total_node`
- ▶ `perfstat_protocol_node`
- ▶ `perfstat_tape_node`
- ▶ `perfstat_tape_total_node`
- ▶ `perfstat_volumegroup_node`

### Library

Perfstat Library (`libperfstat.a`)

## Syntax

```
#include <libperfstat.h>
```

```
int perfstat_cpu_node ( name, userbuff, sizeof_userbuff, desired_number  
)
```

```
perfstat_id_node_t *name;  
perfstat_cpu_t *userbuff;  
int sizeof_userbuff;  
int desired_number;
```

```
int perfstat_cpu_total_node ( name, userbuff, sizeof_userbuff,  
desired_number )
```

```
perfstat_id_node_t *name;  
perfstat_cpu_total_t *userbuff;  
int sizeof_userbuff;  
int desired_number;
```

```
int perfstat_disk_node ( name, userbuff, sizeof_userbuff,  
desired_number )
```

```
perfstat_id_node_t *name;  
perfstat_disk_t *userbuff;  
int sizeof_userbuff;  
int desired_number;
```

```
int perfstat_disk_total_node ( name, userbuff, sizeof_userbuff,  
desired_number )
```

```
perfstat_id_node_t *name;  
perfstat_disk_total_t *userbuff;  
int sizeof_userbuff;  
int desired_number;
```

```
int perfstat_diskadapter_node ( name, userbuff, sizeof_userbuff,  
desired_number )
```

```
perfstat_id_node_t *name;  
perfstat_diskadapter_t *userbuff;  
int sizeof_userbuff;  
int desired_number;
```

```
int perfstat_diskpath_node ( name, userbuff, sizeof_userbuff,  
desired_number )
```

```
perfstat_id_node_t *name;  
perfstat_diskpath_t *userbuff;  
int sizeof_userbuff;
```

```
int desired_number;

int perfstat_logicalvolume_node ( name, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_logicalvolume_t *userbuff;
int sizeof_userbuff;
int desired_number;

int perfstat_memory_page_node ( name, psize, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_psize_t *psize;
perfstat_memory_page_t *userbuff;
int sizeof_userbuff;
int desired_number;

int perfstat_memory_total_node ( name, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_memory_total_t *userbuff;
int sizeof_userbuff;
int desired_number;

int perfstat_netbuffer_node ( name, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_netbuffer_t *userbuff;
int sizeof_userbuff;
int desired_number;

int perfstat_netinterface_node ( name, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_netinterface_t *userbuff;
int sizeof_userbuff;
int desired_number;

int perfstat_netinterface_total_node ( name, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_netinterface_total_t *userbuff;
int sizeof_userbuff;
int desired_number;
```

```
int perfstat_pagingspace_node ( name, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_pagingspace_t *userbuff;
int sizeof_userbuff;
int desired_number;

int perfstat_partition_total_node ( name, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_partition_total_t *userbuff;
int sizeof_userbuff;
int desired_number;

int perfstat_protocol_node ( name, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_protocol_t *userbuff;
int sizeof_userbuff;
int desired_number;

int perfstat_tape_node ( name, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_tape_t *userbuff;
int sizeof_userbuff;
int desired_number;

int perfstat_tape_total_node ( name, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_tape_total_t *userbuff;
int sizeof_userbuff;
int desired_number;

int perfstat_volume_group_node ( name, userbuff, sizeof_userbuff,
desired_number )
perfstat_id_node_t *name;
perfstat_volume_group_t *userbuff;
int sizeof_userbuff;
int desired_number;
```

## Description

These subroutines return remote node's performance statistics in their corresponding perfstat\_<subsystem>\_t structure.

To get statistics from any particular node in a cluster, the Node ID or the Node name must be specified in the name parameter. The userbuff parameter must be allocated and the desired\_number parameter must be set.

**Note:** The remote node should belong to one of the clusters in which the current node (the perfstat API call is run) is participating.

Refer to the AIX version 7.1 documentation for additional details.

<http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.doc/doc/base/technicalreferences.htm>



# System management

In this chapter, the following system management enhancements are discussed:

- ▶ 5.1, “CPU interrupt disablement” on page 146
- ▶ 5.2, “Distributed System Management” on page 147
- ▶ 5.3, “AIX System Configuration Structure Expansion” on page 165
- ▶ 5.4, “AIX Runtime Expert” on page 166
- ▶ 5.5, “Removal of CSM” on page 176
- ▶ 5.6, “Removal of IBM Text-to-Speech” on page 179
- ▶ 5.7, “AIX device renaming” on page 180
- ▶ 5.8, “1024 Hardware thread enablement” on page 181
- ▶ 5.9, “Kernel Memory Pinning” on page 185
- ▶ 5.10, “ksh93 enhancements” on page 188
- ▶ 5.11, “DWARF” on page 188
- ▶ 5.12, “AIX Event Infrastructure extension and RAS” on page 189

## 5.1 CPU interrupt disablement

AIX 6.1 TL6 and 7.1 provides facility to quiesce external I/O interrupts on a given set of logical processors. It helps reduce interrupt jitter that affects application performance.

When co-scheduling Parallel Operation Environment (POE) jobs or even in non-POE commercial environment, administrators can control the process scheduling and interrupt handling across all the processors. It is desirable to quiesce interrupts on the SMT threads which are running POE jobs to avoid interrupting the jobs. By doing so, the user applications can run on a given set of processors without being affected by any external interrupts.

The CPU interrupt disablement function can be configured using the kernel service, system call or user command as listed below:

- ▶ Kernel service: `k_cpuextintr_ctl()`
- ▶ System call: `cpuextintr_ctl()`
- ▶ Command line: `cpuextintr_ctl`

This functionality is supported on Power5/Power6/Power7 and any future System P hardware. It is supported both dedicated or shared processor logical partitions.

Example 5-1 shows the output of `cpuextintr_ctl` command used to disable external interrupts on the CPU 1 on a system which has 2 CPUs.

**Note:** The changes are reflected dynamically without requiring a reboot of the system. Also, the changes are *not* persistent across reboots of the system.

### *Example 5-1 Disabling interrupts*

---

```
# bindprocessor -q
The available processors are: 0 1

# cpuextintr_ctl -Q
The CPUs that have external interrupt enabled:

    0    1

The CPUs that have external interrupt disabled:

# cpuextintr_ctl -C 1 -i disable

# cpuextintr_ctl -Q
```



The CPUs that have external interrupt enabled:

0

The CPUs that have external interrupt disabled:

1

---

**Note:**

- ▶ When the request for external interrupt is *disable*, only external interrupt priority more favored than INTCLASS0 may be delivered to the controlled CPU, which includes Environmental and POver Warning (EPOW) interrupt and IPI (MPC) interrupt.
- ▶ Even though the external interrupt has been disabled using these interfaces, CPU can still be interrupted by IPI/MPC or EPOW interrupt or any priority registered at INTMAX.
- ▶ CPU interrupt disablement works with CPU DR add/removal (DLPAR operation). Once a CPU DR added to the partition, the external interrupt will be enabled by default.
- ▶ CPU interrupt disablement works with CPU Intelligent folding.
- ▶ It guarantees that at least one of the CPU on the system will have external interrupt enabled.

## 5.2 Distributed System Management

Starting with AIX 6.1 TL3 a new package is shipped with the base media called Distributed System Management (DSM). In AIX 7.1 this new DSM package replaces the Cluster Systems Management package (CSM). The CSM package is no longer available on AIX 7.1. Commands such as dcp and dsh are not available on AIX 7.1 without installing the DSM package. This package is not installed by default but is on the base installation media. The DSM package is in the filesets, dsm.core and dsm.dsh.

Selecting the dsm package from the install media will install the following:

Table 5-1 DSM components

dsm.core	Distributed Systems Management Core
dsm.dsh	Distributed Systems Management Dsh

The new DSM programs found in the fileset dsm.core are:

<b>dpasswd</b>	creates an encrypted password file for an access point
<b>dkeyexch</b>	exchanges default ssh keys with an access point
<b>dgetmacs</b>	collects MAC address information from a machine
<b>dconsole</b>	opens a remote console to a machine

## 5.2.1 dpasswd command

The **dpasswd** command is used to create the DSM password file. The password file contains a user ID and associated encrypted password. The command will generate an AES key and write it to the file `/etc/ibm/sysmgt/dsm/config/.key`, if this file does not already exist. The default key size will be 128 bits. The command can generate a 256-bit key if the unrestricted Java™ security files have been installed. For more information on these policy files, refer to the Java Security Guide, which ships with the Java Runtime package.

The key will be used to encrypt the password before writing it to the file. It will also be used by the other DSM programs to decrypt the password. If the key file is removed, it will be re-created with a new key the next time the command is run.

**Note:** If the `.key` file is removed, password files created with that key can not be decrypted. If the `.key` file is removed, the existing password files must be recreated with the **dpasswd** command

If the password file name is given with no path information the password file will be written to the `/etc/ibm/sysmgt/dsm/config` directory.

Run the **dpasswd -h** command to view the command syntax

Example 5-2 shows the use of the **dpasswd** command to create the password file.

*Example 5-2 Creating a password file*

---

```
# dpasswd -f my_password_file -U userID
Password file is /etc/ibm/sysmgt/dsm/config/my_password_file
Password:
Re-enter password:
Password file created.
#
```

---

## 5.2.2 dkeyexch command

The **dkeyexch** command is used to exchange ssh keys between the NIM master and a client access point. The command will require the encrypted password file created by the **dpasswd** command. The information in the password file is used to exchange ssh keys with the access points specified in the command.

This command will exchange the default ssh RSA and DSA keys located in the user's \$HOME/.ssh directory as generated by the ssh-keygen command. It will exchange keys stored in user named files.

**Note:** openssl (openss.base) and openssh (openssh.base) must be installed

The command can also be used to remove keys from an access point.

**Note:** BladeCenter® currently limits the number of installed keys to 12. When adding keys to a BladeCenter, the command will verify there are keyslots available for the new keys. If only one slot is available, only the DSA key will be exchanged.

Run the **dkeyexch -h** command to see the command syntax.

Example 5-3 shows a key exchange between the NIM master and an HMC. The password file must exist and contain a valid user ID and encrypted password for this HMC. Following the key exchange, an ssh session can be established with no password prompt.

*Example 5-3 Key exchange between NIM and an HMC*

---

```
# dkeyexch -f /etc/ibm/sysmgt/dsm/config/hmc_password_file -I hmc -H
hmc01.clusters.com
# ssh hscroot@hmc01.clusters.com
Last login: Tue Dec 23 11:57:55 2008 from nim_master.clusters.com
hscroot@hmc01:~>
```

---

## 5.2.3 dgetmacs command

The **dgetmacs** command is used to query a client node for its network adapter information. This information is gathered even if the node has no operating system on it or is powered off. This command requires AIX 7.1 SP 1

**Note:** When the `open_firmware` mode is used (either when specified on the command line or if the `dsh` and `arp` modes failed), the command will cause the client node to be rebooted into a special state so that the adapter information can be obtained. This only applies to client nodes managed by a HMC or an IVM. Ensure the client node is not in use before running this command.

Run the `dgetmacs -h` command to view the command syntax

Example 5-4 shows an example that uses the `dsh` method.

*Example 5-4 Using the dsh method*

---

```
# dgetmacs -m dsh -n canif3_obj -C NIM
Using an adapter type of "ent".
Attempting to use dsh method to collect MAC addresses.
#
Node::adapter_type::interface_name::MAC_address::location::media_speed::adapter_
duplex::UNUSED::install_gateway::ping_status::machine_type::netaddr::subnet_mask
canif3_obj::ent_v::en0::001A644486E1:::1000::full:::172.16.143.250:::secondar
y::172.16.128.91:::255.255.240.0
canif3_obj::ent_v::en1::1E9E18F60404:::172.16.143.250:::secondary:::
```

---

Additional examples can be found in the tech note document located at [/opt/ibm/sysmgmt/dsm/doc/dsm\\_tech\\_note.pdf](/opt/ibm/sysmgmt/dsm/doc/dsm_tech_note.pdf)

## 5.2.4 dconsole command

The `dconsole` command is used to open a remote console to a client node. The command operates in both the `DEFAULT` and `NIM` contexts. The command supports read-only consoles and console logging.

The command is supported by a daemon program that is launched when the `dconsole` command is invoked for the first time. This console daemon will remain running as long as there are consoles open. When the last console is closed, the console daemon will terminate. By default, the daemon listens on TCP port number 9085, which has been reserved from IANA for this purpose. The port number may be changed by overriding the `dconsole_Port_Number` entry in the DSM properties file.

Run the `dconsole -h` command to view the syntax

### **dconsole display modes**

The command operates in one of two display modes, “default” and “text”.

In the default display mode, the command uses an xterm window to display the console. In this mode, consoles to multiple client nodes can be opened from a single command. A separate window will be opened for each node. The default display mode requires that the DISPLAY environment variable be set before the dconsole command is invoked. The variable must be set to the address of an X-Windows server where the console will be displayed. By default, the console window is launched using the fixed font.

The remote console session is closed by closing the xterm window. Issuing Ctrl-x within the console window will also close the console session.

The text display mode is invoked by adding the -t flag to the command line. In this mode, no XWindows server is required. The console will be opened in the current session. The text mode console session is closed by issuing Ctrl-X.

DSM offers the ability to log remote console sessions on client nodes. By default logging is disabled. It may be enabled on a console-by-console basis by issuing the dconsole command with the -l (lower-case L) flag. It may also be enabled globally by overriding the n entry in the DSM properties file (setting the value to “Yes” enables global console logging). When logging is enabled, any data that is visible on the console will also be written to a log file. The console must be open for logging to take place.

**Note:** Changing the global setting has no impact on console sessions that are already open when the setting was changed. Any open consoles must be closed and re-opened for the updated setting to take effect.

By default, console log files are written to the /var/ibm/sysmgmt/dsm/log/console directory. Both the log directory and console log subdirectory may be changed by overriding the dconsole\_Log\_File\_Subdirectory entry in the DSM properties file.

By default these files will rotate. The maximum file size is about 256 kilobytes, and up to four files will be kept for each console log. The number of rotations may be changed by overriding the Log\_File\_Rotation entry in the DSM properties file. Setting the value to zero disables log rotation and will allow the logs to grow in size up to the available file system space.

Example 5-5 shows the **dconsole** command starting in text mode with logging enabled.

*Example 5-5 Starting dconsole in text mode with logging*

---

```
# dconsole -n 9.47.93.94 -t -l
Starting console daemon
[read-write session]
```

Open in progress

Open Completed.

AIX Version 6  
Copyright IBM Corporation, 1982, 2009.  
Console login:

---

For Example 5-5 on page 151 an entry was made in the node info file to define the target system and access point information. The node info file is found in /etc/ibm/sysmgt/dsm directory.

Example 5-6 shows the format of the fnode info file used in Example 5-5 on page 151.

*Example 5-6 Contents of the node info file*

---

```
# cat /etc/ibm/sysmgt/dsm/nodeinfo
9.47.93.94|hmc|9.47.91.240|TargetHWTypeModel=9117-570:TargetHWSerialNum
=1038FEA:TargetLPARID=11|/etc/ibm/sysmgt/dsm/config/hsc_password
#
```

---

Additional options and usages of the the console command along with information on using DSM and NIM to install new clients can be found in the DSM tech note. This tech note document is located at /opt/ibm/sysmgt/dsm/doc/dsm\_tech\_note.pdf

## 5.2.5 dcp command

The **dcp** command works the same as it did in AIX 6.1 The command will copy files to or from multiple nodes. The node list is not the same as the DSM node info file.

Example 5-7 shows the use of **dcp** command to copy the testdata.log file to a new file on the nodes listed in the node list file.

*Example 5-7 Example use of the dcp command*

---

```
# dcp /tmp/testdata.log /tmp/testdata_copy4.log
```

---

For Example 5-7 the location of the node list was specified in an environment variable as shown in Example 5-8 on page 153.

*Example 5-8 Checking dsh environment variables*

---

```
# env | grep -i dsh
DSH_REMOTE_CMD=/usr/bin/ssh
DSH_NODE_LIST=/etc/ibm/sysmgmt/dsm/nodelist
DSH_NODE_RSH=/usr/bin/ssh
#
```

---

The nodelist of the **dcp** command was a simple list of target ip addresses as seen in Example 5-9.

*Example 5-9 Sample node list*

---

```
# cat /etc/ibm/sysmgmt/dsm/nodelist
9.47.93.94
9.47.93.60

#
```

---

## 5.2.6 dsh command

The **dsh** command works the same as it did in AIX 6.1 The command will run commands concurrently on multiple nodes. The node list is not the same as the DSM node info file.

Example 5-10 shows the use of **dsh** command to run the **date** command on the nodes listed in the node list file.

*Example 5-10 Example using the dsh command*

---

```
# dsh -a date
e19-93-60.ent.beaverton.ibm.com: Tue Sep 14 16:07:51 PDT 2010
e19-93-94.ent.beaverton.ibm.com: Tue Sep 14 16:08:02 PDT 2010
```

---

For Example 5-10 the location of the node list was specified in an environment variable as shown in Example 5-11

*Example 5-11 Setting up the environment variables*

---

```
# env | grep -i dsh
DSH_REMOTE_CMD=/usr/bin/ssh
DSH_NODE_LIST=/etc/ibm/sysmgmt/dsm/nodelist
DSH_NODE_RSH=/usr/bin/ssh
```

---

#

The node list for the **dsh** command was a simple list of target ip addresses as seen in Example 5-12.

*Example 5-12 Sample node list*

---

```
# cat /etc/ibm/sysmgmt/dsm/nodelist
9.47.93.94
9.47.93.60
```

#

---

## 5.2.7 Using DSM and NIM

The AIX Network Installation Manager (NIM) has been enhanced to work with the Distributed System Management (DSM) commands. This integration enables the automatic installation of new AIX systems that are either currently powered on or off.

The example that follows will demonstrate this functionality. We will follow a sequence of steps to use NIM to install the AIX operating system onto a new NIM client LPAR, using DSM. We will be installing AIX onto an HMC controlled LPAR.

The steps are as follows:

1. Collect information for console access points, such as the IP address or hostname of the HMC, and the HMC administrator user ID and password.
2. Collect information relating to the new NIM client LPAR, such as the hostname, IP address, hardware type-model, serial number of the system, and LPAR id.
3. Run the **dpasswd** command to generate the password file for the HMC access point. Run the **dkeyexch** command to exchange the NIM master SSH key with the HMC.
4. Define a new NIM HMC and management object for the HMC and the CEC. Specifying the password file that was created in the previous step.
5. Obtain the MAC address for the network adapter of the new LPAR using the **dgetmacs** command.
6. Define a new NIM machine object for the new NIM client LPAR.
7. Perform a NIM **bos\_inst** operation on the NIM client to install the AIX operating system.



8. From the NIM master, open a console window, with the **dconsole** command and monitor the NIM installation.
9. The final step is to verify that AIX has installed successfully.

In this scenario, the HMC IP address is 10.52.52.98 and its hostname is hmc5. The system type, model and serial number information is collected from the HMC, as shown in Example 5-13

*Example 5-13 Collecting the system type, model and serial number from HMC*

---

```
hscroot@hmc5:~> lssyscfg -r sys -F name,type_model,serial_num
750_2-8233-E8B-061AB2P,8233-E8B,061AB2P
```

---

The LPAR id is also collected from the HMC as shown in Example 5-14.

*Example 5-14 Collecting the LPAR id information from the HMC*

---

```
hscroot@hmc5:~> lssyscfg -r lpar -m 750_2-8233-E8B-061AB2P -F
name,lp_ar_id
750_2_LP04,5
750_2_LP03,4
750_2_LP02,3
750_2_LP01,2
750_2_VIO_1,1
orion,6
```

---

The HMC admin user ID is hscroot and the password is abc123. The **dpasswd** command is run to store the user password. The NIM master SSH key is generated and exchanged with the HMC with the **dkeyexch** command. We confirmed we could ssh to the HMC without being prompted for a password, as shown in Example 5-15.

*Example 5-15 Configuring ssh access to the HMC from the NIM master*

---

```
# dpasswd -f my_password_file -U hscroot
# dkeyexch -f /etc/ibm/sysmgt/dsm/config/my_password_file -I hmc -H 10.52.52.98
# ssh hscroot@hmc5
Last login: Fri Sep 10 09:46:03 2010 from 10.52.52.101
hscroot@hmc5:~>
```

---

The new NIM client LPAR IP address is 10.52.52.200 and the hostname is orion. The LPAR id is 6. This information and the hardware type-model and serial number of the target Power system were recorded in the `/etc/ibm/sysmgt/dsm/nodeinfo` file, as shown in Example 5-16 on page 156.

*Example 5-16 Entry in the nodeinfo file for the new host, Power System and HMC*


---

```
# cat /etc/ibm/sysmgmt/dsm/nodeinfo
75021p01|hmc|10.52.52.98|TargetHWTypeModel=8233-E8B:TargetHWSerialNum=061AB2P:TargetLPARID=2|/etc/ibm
/sysmgmt/dsm/config/my_password_file
75021p02|hmc|10.52.52.98|TargetHWTypeModel=8233-E8B:TargetHWSerialNum=061AB2P:TargetLPARID=3|/etc/ibm
/sysmgmt/dsm/config/my_password_file
75021p03|hmc|10.52.52.98|TargetHWTypeModel=8233-E8B:TargetHWSerialNum=061AB2P:TargetLPARID=4|/etc/ibm
/sysmgmt/dsm/config/my_password_file
75021p04|hmc|10.52.52.98|TargetHWTypeModel=8233-E8B:TargetHWSerialNum=061AB2P:TargetLPARID=5|/etc/ibm
/sysmgmt/dsm/config/my_password_file
orion|hmc|10.52.52.98|TargetHWTypeModel=8233-E8B:TargetHWSerialNum=061AB2P:TargetLPARID=6|/etc/ibm/sy
smgmt/dsm/config/my_password_file
```

---

We defined a new NIM HMC and management object for the HMC and the CEC, as shown in Example 5-17.

*Example 5-17 Defining the HMC and CEC NIM objects*


---

```
# nim -o define -t hmc -a if1="find_net hmc5 0" -a
passwd_file="/etc/ibm/sysmgmt/dsm/config/my_password_file" hmc5

# lsnim -Fl hmc5
hmc5:
  id          = 1284061389
  class       = management
  type        = hmc
  if1         = net_10_52_52 hmc5 0
  Cstate      = ready for a NIM operation
  prev_state  =
  Mstate      = currently running
  manages     = cec0
  passwd_file = /etc/ibm/sysmgmt/dsm/config/my_password_file

# nim -o define -t cec -a hw_type=8233 -a hw_model=E8B -a hw_serial=061AB2P -a
mgmt_source=hmc5 cec0

# lsnim -Fl cec0
cec0:
  id          = 1284061538
  class       = management
  type        = cec
  Cstate      = ready for a NIM operation
  prev_state  =
  manages     = 75021p02
  manages     = orion
  hmc         = hmc5
  serial      = 8233-E8B*061AB2P
```

---

We obtained the MAC address for the virtual network adapter in the new LPAR. The **dgetmacs** command is used to obtain this information. This command will

power on the LPAR in *Open Firmware* mode to query the network adapter MAC address information. The LPAR in this example was in a *Not Activated* state prior to running the **dgetmacs** command.

**Note:** If the MAC address of the network adapter is unknown, you can define the client with a MAC address of 0 and use the **dgetmacs** command to retrieve it. Once the MAC address is identified, the NIM standalone object if1 attribute can be changed with the **nim -o change** command.

This MAC address is required for the bos\_inst NIM operation for clients that can not be reached.

If the LPAR is in a *Running* state the LPAR will be powered down and restarted in *Open Firmware* mode. Once the MAC address has been acquired, the LPAR will be powered down again.

---

*Example 5-18 Obtaining the MAC address for the LPARs virtual network adapter*

---

```
# dgetmacs -n orion
Using an adapter type of "ent".
Could not dsh to node orion.
Attempting to use openfirmware method to collect MAC addresses.
Acquiring adapter information from Open Firmware for node orion.

#
Node::adapter_type::interface_name::MAC_address::location::media_speed:
:adapter_duplex::UNUSED::install_gateway::ping_status::machine_type::ne
taddr::subnet_mask

orion::ent_v::::6E8DD877B814::U8233.E8B.061AB2P-V6-C20-T1::auto::auto::
::::n/a::secondary::::
```

---

We defined a new NIM machine object for the new LPAR, as shown in Example 5-19.

---

*Example 5-19 Defining new NIM machine object with HMC, LPAR and CEC options*

---

```
# nim -o define -t standalone -a if1="net_10_52_52 orion 6E8DD877B814" -a
net_settings1="auto auto" -a mgmt_profile1="hmc5 6 cec0" orion
# lsrim -Fl orion
orion:
  id           = 1284075145
  class        = machines
  type         = standalone
  connect      = nimsh
  platform     = chrp
```

```

netboot_kernel = 64
if1             = net_10_52_52 orion 6E8DD877B814
net_settings1  = auto auto
cable_type1    = N/A
mgmt_profile1  = hmc5 6 cec0
Cstate         = ready for a NIM operation
prev_state     = not running
Mstate        = currently running
cpuid          = 00F61AB24C00
Cstate_result  = success
default_profile =
type=hmc,ip=10.52.52.98,passwd_file=/etc/ibm/sysmgt/dsm/config/my_password_file
:type=lpar,identity=6:type=cec,serial=8233-E8B*061AB2P:

```

---

The LPAR was in a Not Activated state. We enabled the NIM client for BOS installation as shown in Example 5-20. This initiated a network boot of the LPAR.

*Example 5-20 Displaying LPAR state and enabling NIM bos\_inst on the NIM client*

---

```

# ssh hscroot@hmc5
Last login: Fri Sep 10 15:57:24 2010 from 10.52.52.101
hscroot@hmc5:~> vtmenu

-----
Partitions On Managed System: 750_2-8233-E8B-061AB2P
OS/400 Partitions not listed
-----

1) 750_2_LP01 Running
2) 750_2_LP02 Running
3) 750_2_LP03 Running
4) 750_2_LP04 Running
5) 750_2_VIO_1 Running
6) orion Not Activated

Enter Number of Running Partition (q to quit): q
hscroot@hmc5:~> exit
exit
Connection to hmc5 closed.
#
# nim -o bos_inst -a bosinst_data=noprompt_bosinst -a source=rte -a
installp_flags=agX -a accept_licenses=yes -a spot=spotaix7100 -a
lpp_source=aix7100 orion
dnetboot Status: Invoking /opt/ibm/sysmgt/dsm/dsmbin/lpar_netboot orion
dnetboot Status: Was successful network booting node orion.
#

```

---

We opened a console window (in read-only mode with session logging enabled) using the **dconsole** command to monitor the NIM installation as shown in Example 5-21. Only partial output is shown as the actual log is extremely verbose.

*Example 5-21 Monitoring the NIM installation with the dconsole command*

---

```
# dconsole -n orion -t -l -r
Starting console daemon
[read only session, user input discarded]

Open in progress

Open Completed.
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM

          1 = SMS Menu                5 = Default Boot List
          8 = Open Firmware Prompt    6 = Stored Boot List

Memory      Keyboard      Network      SCSI      Speaker
.....
10.52.52.200: 24 bytes from 10.52.52.101: icmp_seq=9 ttl=? time=11 ms

10.52.52.200: 24 bytes from 10.52.52.101: icmp_seq=10 ttl=? time=11 ms

PING SUCCESS.
ok
0 > 0 to my-self ok
0 > boot
/vdevice/1-lan@30000014:speed=auto,duplex=auto,bootp,10.52.52.101,,10.52.52.200
,10.52.52.101

.....

TFTP BOOT -----
Server IP.....10.52.52.101
Client IP.....10.52.52.200
Gateway IP.....10.52.52.101
Subnet Mask.....255.255.254.0
( 1 ) Filename...../tftpboot/orion
TFTP Retries.....5
Block Size.....512
PACKET COUNT = 12900

.....
```

## Installing Base Operating System

Please wait...

Approximate % tasks complete	Elapsed time (in minutes)
---------------------------------	------------------------------

---

On the NIM master, the NIM client status during the installation was monitored, as shown in Example 5-22.

*Example 5-22 Monitoring the NIM client installation status from the NIM master*

---

```
# lsnim -Fl orion
orion:
  id           = 1284075145
  class        = machines
  type         = standalone
  connect      = nimsh
  platform     = chrp
  netboot_kernel = 64
  if1          = net_10_52_52 orion 6E8DD877B814
  net_settings1 = auto auto
  cable_type1  = N/A
  mgmt_profile1 = hmc5 6 cec0
  Cstate       = Base Operating System installation is being performed
  prev_state   = BOS installation has been enabled
  Mstate       = in the process of booting
  info         = BOS install 21% complete : Installing additional software.
  boot         = boot
  bosinst_data = noprompt_bosinst
  lpp_source   = aix7100
  nim_script   = nim_script
  spot         = spotaix7100
  exported     = /export/lppsrc/aix7100
  exported     = /export/nim/scripts/orion.script
  exported     = /export/spot/spotaix7100/usr
  exported     = /tmp/cg/bosinst.data
  cpuid        = 00F61AB24C00
  control      = master
  Cstate_result = success
  boot_info    = -aip=10.52.52.200 -aha=6E8DD877B814 -agw=10.52.52.101
               -asm=255.255.254.0 -asa=10.52.52.101
```

```

trans1      = 86 1 6 master /usr/sbin/nim -o deallocate -F
-asubclass=all -aasync=yes orion
trans2      = 86 14 1 master
/usr/lpp/bos.sysmgt/nim/methods/m_destroy_res -aforce=yes -aignore_state=yes -a
ignore_lock=yes orion
default_profile =
type=hmc,ip=10.52.52.98,passwd_file=/etc/ibm/sysmgt/dsm/config/my_password_file
:type=lpar,identity=6:type=cec,serial=8233-E8B*061AB2P:

```

---

On the NIM master, the DSM network boot output is logged to the `/var/ibm/sysmgt/dsm/log/dnetboot.name.log.XXX`, where `name` is the node name and `XXX` is the log sequence number.

*Example 5-23 DSM network boot log file output*

---

```

# cd /var/ibm/sysmgt/dsm/log/
# cat dnetboot.orion.log.253
Output log for dnetboot is being written to
/var/ibm/sysmgt/dsm/log//dnetboot.orion.log.253.
-----
dnetboot: Logging started Fri Sep 10 16:03:21 EDT 2010.
-----

dnetboot Status: Invoking /opt/ibm/sysmgt/dsm/dsmbin/lpar_netboot orion
16:3:21 dnetboot Status: Invoking /opt/ibm/sysmgt/dsm/dsmbin/lpar_netboot
orion
-----
dnetboot: Logging stopped Fri Sep 10 16:03:21 EDT 2010.
-----

dnetboot Status: Invoking /opt/ibm/sysmgt/dsm/dsmbin/lpar_netboot -i -t ent -D
-S 10.52.52.101 -G 10.52.52.101 -C 10.52.52.200 -m 6E8DD877B814 -s auto -d auto
-F /etc/ibm/sysmgt/dsm/config/my_password_file -j hmc -J 10.52.52.98 6 061AB2P
8233-E8B
# Connected
# Checking for OF prompt.
# Timeout waiting for OF prompt; rebooting.
# Checking for power off.
# Client IP address is 10.52.52.200.
# Server IP address is 10.52.52.101.
# Gateway IP address is 10.52.52.101.
# Getting adapter location codes.
# /vdevice/l-lan@30000014 ping successful.
# Network booting install adapter.
# bootp sent over network.
# Network boot proceeding, lpar_netboot is exiting.
# Finished.
16:4:41 dnetboot Status: Was successful network booting node orion.

```

---

The **dconsole** command can log session output if called with the **-l** flag. The log file is located on the NIM master, in the `/var/ibm/sysmgmt/dsm/log/console/name.X` file, where `name` is the node name and `X` is the log sequence number. This file can be monitored using the **tail** command, as shown in Example 5-24.

*Example 5-24 DSM dconsole log file*

---

```
# cd /var/ibm/sysmgmt/dsm/log/console/
# ls -ltr
total 1664
-rw-r--r--  1 root    system      1464 Sep 09 15:39 75021p01.0
-rw-r--r--  1 root    system     34118 Sep 09 19:27 75021p02.0
-rw-r--r--  1 root    system    262553 Sep 10 12:12 orion.3
-rw-r--r--  1 root    system    262202 Sep 10 12:46 orion.2
-rw-r--r--  1 root    system         0 Sep 10 16:01 orion.0.lck
-rw-r--r--  1 root    system    262282 Sep 10 16:09 orion.1
-rw-r--r--  1 root    system    11708 Sep 10 16:09 orion.0
# tail -f orion.0

5724X1301
  Copyright IBM Corp. 1991, 2010.
  Copyright AT&T 1984, 1985, 1986, 1987, 1988, 1989.
  Copyright Unix System Labs, Inc., a subsidiary of Novell, Inc. 1993.
  All Rights Reserved.
  US Government Users Restricted Rights - Use, duplication or disclosure
  restricted by GSA ADP Schedule Contract with IBM Corp.
  . . . . . << End of copyright notice for x1C.rte >>. . . . .

Filesets processed:  344 of 591
System Installation Time: 5 minutes          Tasks Complete: 61%

installp: APPLYING software for:
          x1C.msg.en_US.rte 11.1.0.1

. . . . . << Copyright notice for x1C.msg.en_US >> . . . . .
  Licensed Materials - Property of IBM

5724X1301
  Copyright IBM Corp. 1991, 2010.
  Copyright AT&T 1984, 1985, 1986, 1987, 1988, 1989.
  Copyright Unix System Labs, Inc., a subsidiary of Novell, Inc. 1993.
  All Rights Reserved.
  US Government Users Restricted Rights - Use, duplication or disclosure
  restricted by GSA ADP Schedule Contract with IBM Corp.
  . . . . . << End of copyright notice for x1C.msg.en_US >>. . . . .
```

---



Another log file, related to network boot is also available on the NIM master. It contains extended network boot information and is located in `/tmp/lpar_netboot.PID.exec.log`, where PID is the process ID of the `lpar_netboot` process, as shown in Example 5-25. Only partial output is shown as the actual log file is extremely verbose.

*Example 5-25 lpar\_netboot log file*

---

```
# cd /tmp
# cat lpar_netboot.16056500.exec.log
lpar_netboot Status: node = 6, profile = 061AB2P, manage = 8233-E8B
lpar_netboot Status: process id is 16056500
lpar_netboot Status: -t List only ent adapters
lpar_netboot Status: -D (discovery) flag detected
lpar_netboot Status: -i (force immediate shutdown) flag detected
lpar_netboot Status: using adapter speed of auto
lpar_netboot Status: using adapter duplex of auto
lpar_netboot Status: using server IP address of 10.52.52.101
lpar_netboot Status: using client IP address of 10.52.52.200
lpar_netboot Status: using gateway IP address of 10.52.52.101
lpar_netboot Status: using macaddress of 6E8DD877B814
lpar_netboot Status: ck_args start
lpar_netboot Status: node 6
lpar_netboot Status: managed system 8233-E8B
lpar_netboot Status: username
lpar_netboot Status: password_file /etc/ibm/sysmgt/dsm/config/my_password_file
lpar_netboot Status: password
lpar_netboot Status: hmc-controlled node detected
lpar_netboot Status: node type is hmc
lpar_netboot Status: open port
lpar_netboot Status: open S1 port
lpar_netboot Status: console command is /opt/ibm/sysmgt/dsm/bin//dconsole -c -f -t -n
....
lpar_netboot Status: power reported as off, checking power state
lpar_netboot Status: power state is 6 Not Activated
lpar_netboot Status: power off complete
lpar_netboot Status: power on the node to Open Firmware
lpar_netboot Status: wait for power on
lpar_netboot Status: power on complete
lpar_netboot Status: waiting for RS/6000 logo
lpar_netboot Status: at RS/6000 logo
lpar_netboot Status: Check for active console.
.....
lpar_netboot Status: ping_server start
lpar_netboot Status: full_path_name : /vdevice/1-lan@30000014
lpar_netboot Status: phandle : 0000021cf420
lpar_netboot Status: get_adap_prop start
lpar_netboot Status: get_adap_prop start
lpar_netboot Status: get_adap_prop command is " supported-network-types" 0000021cf420
....
lpar_netboot Status: ping_server command is ping
/vdevice/1-lan@30000014:10.52.52.101,10.52.52.200,10.52.52.101
send_command start:ping /vdevice/1-lan@30000014:10.52.52.101,10.52.52.200,10.52.52.101
ping /vdevice/1-lan@30000014:10.52.52.101,10.52.52.200,10.52.52.101
ping /vdevice/1-lan@30000014:10.52.52.101,10.52.52.200,10.52.52.101
10.52.52.200: 24 bytes from 10.52.52.101: icmp_seq=1 ttl=? time=10 ms

10.52.52.200: 24 bytes from 10.52.52.101: icmp_seq=2 ttl=? time=10 ms

10.52.52.200: 24 bytes from 10.52.52.101: icmp_seq=3 ttl=? time=10 ms

10.52.52.200: 24 bytes from 10.52.52.101: icmp_seq=4 ttl=? time=11 ms
....
PING SUCCESS.
```

```

ok
....

TFTP1par_netboot Status: network boot initiated
/usr/bin/dspmsg -s 1 /usr/lib/nls/msg/en_US/IBMhsc.netboot.cat 55 '# bootp sent over network.
.....
FINAL PACKET COUNT = 34702 1UNT = 17700
FINAL FILE SIZE = 17766912 BYTES

Elapsed time since release of system processors: 15840 mins 39 secs

-----
                Welcome to AIX.
                boot image timestamp: 15:00 09/09
                The current time and date: 20:04:40 09/10/2010
                processor count: 2; memory size: 2048MB; kernel size: 35060743
boot device:
/vdevice/1-1an@30000014:speed=auto,duplex=auto,bootp,10.52.52.101,,10.52.52.200,10.52.52.101
/usr/bin/dspmsg -s 1 /usr/lib/nls/msg/en_US/IBMhsc.netboot.cat 56 '# Finished.

```

Once the AIX installation is complete a login prompt is displayed in the console window. We then logged into the LPAR and confirmed AIX was installed as expected. We started a read-write console session with the **dconsole** command, as shown in Example 5-26.

*Example 5-26 Verifying AIX installed successfully from a dconsole session*

```

# dconsole -n orion -t -l
Starting console daemon
[read-write session]

Open in progress

Open Completed.

AIX Version 7
Copyright IBM Corporation, 1982, 2010.
Console login: root
*****
*                                                                 *
*                                                                 *
*  Welcome to AIX Version 7.1!                                   *
*                                                                 *
*                                                                 *
*  Please see the README file in /usr/lpp/bos for information pertinent to *
*  this release of the AIX Operating System.                     *
*                                                                 *
*                                                                 *
*****

# oslevel -s
7100-00-00-0000

```

## 5.3 AIX System Configuration Structure Expansion

New hardware and operating system capabilities required enhancements of the system configuration structure defined on AIX in `/usr/include/sys/systemcfg.h`.

As result a new kernel service called `kgetsystemcfg()` and a new library function called `getsystemcfg()` have been implemented.

This new facility should be used in place of the existing `__system_configuration` structure that is accessible via memory as this new facility will be used for new configuration information in the future that will not be accessible via the `__system_configuration` structure.

The new facility however gives access to all the data in `__system_configuration` plus new (future) configuration data.

### 5.3.1 `kgetsystemcfg` kernel service

This kernel service man page provides the following information

*Example 5-27 `kgetsystemcfg` man page header*

---

**Purpose**

Displays the system configuration information.

**Syntax**

```
#include <systemcfg.h>
uint64_t kgetsystemcfg ( int name)
```

**Description**

Displays the system configuration information.

**Parameters**

**name**

Specifies the system variable setting to be returned. Valid values for the name parameter are defined in the `systemcfg.h` file.

**Return value**

**EINVAL**

The value of the name parameter is invalid.

---

### 5.3.2 `getsystemcfg` Subroutine

This libc subroutine man page provides the following information

*Example 5-28 getsystemcfg libc sub-routine man page header.*

---

**Purpose**

Displays the system configuration information.

**Syntax**

```
#include <systemcfg.h>
uint64_t getsystemcfg ( int name)
```

**Parameters**

**name**

Specifies the system variable setting to be returned. Valid values for the name parameter are defined in the systemcfg.h file.

**Return value**

**EINVAL**

The value of the name parameter is invalid.

---

## 5.4 AIX Runtime Expert

AIX 6.1 TL4 included a tool called AIX Runtime Expert. It provides the ability to collect, apply and verify the runtime environment for one or more AIX instances. This can be a valuable tool to use if a system needs to be cloned or if a comparison is needed of the tunables between different AIX instances. This tool allows for the creation of a configuration profile (in XML format) capturing several settings and customizations done to an AIX instance.

With this AIX configuration profile, the system administrator will be able to apply it to new AIX servers or compare it to other configuration servers in order to track any change. From deploying a medium to large server infrastructure or to maintain server farms in a timely fashion, AIX Runtime Expert is the preferred tool for an efficient system administration with its “one-button” approach in managing and configuring numerous AIX instances.

AIX 6.1 TL6 and AIX 7.1 extends the tool with two new capabilities:

- ▶ Consolidate the management of AIX configuration profiles into a single control template.
- ▶ Ease the creation of a configuration template that can be deployed across a network of AIX O/S instances in a scale-out configuration.

Example 5-29 lists the AIX Runtime Expert filesets for AIX 7.1

*Example 5-29 AIX 7.1 AIX Runtime Expert filesets*

---

```
# ls|pp -l | grep -i artex
```

artex.base.agent	7.1.0.0	COMMITTED	AIX Runtime Expert CAS agent
artex.base.rte	7.1.0.0	COMMITTED	AIX Runtime Expert
artex.base.samples	7.1.0.0	COMMITTED	AIX Runtime Expert sample

---

## 5.4.1 AIX Runtime Expert overview

AIX components and sub-systems provide a diversity of control points to manage runtime behavior. These control points can be configuration files, command line and environment variables. These control points are independent of each other and are managed separately. AIX Runtime Expert is a tool to help manage these control points.

AIX Runtime Expert uses an XML file called a profile to manage these control points. The user can create one or multiple profile files depending on the desired results. The user can create a unique profile to suit their needs. These profiles can be created, edited and used to tune a second AIX instance to match an existing AIX instance. The AIX Runtime Expert can also compare two profiles or compare a profile to a running system to see the differences.

The user creates these profiles using the AIX Runtime Expert tool along with two types of read only files that are used to build the profiles. These two types of files are called profile templates and catalogs.

### AIX Runtime Expert profile templates

AIX Runtime Expert profile templates are XML files that include a list of tunable parameters. Each XML profile template is used to control any changeable tunable of a system. For example the vmoProfile.xml file is used for the vmo system tuning. The iooProfile.xml file is used for the I/O system tuning.

There are many profile templates. They can be found in /etc/security/artex/samples directory. They are read only files. The templates are not meant to be edited. It is also possible to see a list of all available profile templates using the **artexlist** command as shown in Example 5-30.

*Example 5-30 AIX Runtime Expert profile template listing*

---

```
# artexlist
/etc/security/artex/samples/acctctlProfile.xml
/etc/security/artex/samples/aixpertProfile.xml
/etc/security/artex/samples/all.xml
/etc/security/artex/samples/alogProfile.xml
/etc/security/artex/samples/authProfile.xml
...
/etc/security/artex/samples/sysdumpdevProfile.xml
```

```

/etc/security/artex/samples/trcctlProfile.xml
/etc/security/artex/samples/trustchkProfile.xml
/etc/security/artex/samples/tsdProfile.xml
/etc/security/artex/samples/viosdevattrProfile.xml
/etc/security/artex/samples/vmoProfile.xml

```

---

These profile templates do not have any parameter values. They are used as template to extract the current systems values and create a new profile the the user may edit.

As new configuration options become available, new templates can be added to expand the value of the AIX Runtime Expert capabilities.

### AIX Runtime Expert catalog

The AIX Runtime Expert catalogs are read-only files located in `/etc/security/artex/catalogs` directory. They define how to map configuration profile values to parameters that run commands and configuration actions. They also identify values that can be modified.

Each catalog contains parameters for one component. However, some catalogs can contain parameters for multiple closely related components. To list all the catalogs use the `artexlist -c` command as shown in Example 5-31

#### *Example 5-31 AIX Runtime Expert catalog listing*

---

```

# artexlist -c
/etc/security/artex/catalogs/acctctlParam.xml
/etc/security/artex/catalogs/aixpertParam.xml
/etc/security/artex/catalogs/alogParam.xml
/etc/security/artex/catalogs/authParam.xml
...
/etc/security/artex/catalogs/trcctlParam.xml
/etc/security/artex/catalogs/trustchkParam.xml
/etc/security/artex/catalogs/tsdParam.xml
/etc/security/artex/catalogs/viosdevattrParam.xml
/etc/security/artex/catalogs/vmoParam.xml
#

```

---

The names of the catalogs describe the components that are contained in the catalog. The example of catalog named `schedoParam.xml` of Example 5-32 on page 169 gives the command name `schedo` and the short description `schedo` parameters. It will allow `schedo` command sub-parameters configuration.

In each file the <description>.xml element provides a description of the catalog

*Example 5-32 Catalog file schedoParam.xml*

---

```
# head /etc/security/artex/catalogs/schedoParam.xml
<?xml version="1.0" encoding="UTF-8"?>
<Catalog id="schedoParam" version="2.0">
<ShortDescription><NLSCatalog catalog="artexcat.cat" setNum="41"
msgNum="1">schedo parameters</NLSCatalog></ShortDescription>
    <Description><NLSCatalog catalog="artexcat.cat" setNum="41"
msgNum="2">Parameter definition for the schedo
command</NLSCatalog></Description>
<CfgMethod id="schedo">
    <Get type="current">
        <Command>/usr/sbin/schedo -a</Command>
        <Filter>/usr/bin/grep -v '= n/a$'</Filter>
    ...
```

---

Profiles file may reference one or multiple catalogs. For example schedoProfile.xml profile will only reference the schedoParam catalog. The all.xml profile file will reference all catalogs since it want's to contains all the system tunables. Beginning of these two files are listed in Example 5-33.

*Example 5-33 Profiles file referenceing catalogs*

---

```
# head /etc/security/artex/samples/schedoProfile.xml
<?xml version="1.0" encoding="UTF-8"?>
<Profile origin="reference" readOnly="true" version="2.0.0">
    <Catalog id="schedoParam" version="2.0">
        <Parameter name="affinity_lim"/>
        <Parameter name="big_tick_size"/>
        <Parameter name="ded_cpu_donate_thresh"/>
        <Parameter name="fixed_pri_global"/>
    ...

# head /etc/security/artex/samples/all.xml
<?xml version="1.0" encoding="UTF-8"?>
<Profile origin="merge: acctctlProfile.xml, aixpertProfile.xml,
alogProfile.xml, authProfile.xml, authentProfile.xml,
chconsProfile.xml, chdevProfile.xml, chlicenseProfile.xml,
chservicesProfile.xml, chssysProfile.xml, chsubserverProfile.xml,
chuserProfile.xml, classProfile.xml, coreProfile.xml,
dumpctrlProfile.xml, envProfile.xml, errdemonProfile.xml,
ewlmProfile.xml, ffdcProfile.xml, filterProfile.xml,
gencopyProfile.xml, iooProfile.xml, krecoveryProfile.xml,
login.cfgProfile.xml, lvmoProfile.xml, mktcpipProfile.xml,
mkuser.defaultProfile.xml, namerslvProfile.xml, nfsProfile.xml,
```

```

nfsoProfile.xml, nisProfile.xml, noProfile.xml, probevueProfile.xml,
rasoProfile.xml, roleProfile.xml, ruserProfile.xml, schedoProfile.xml,
secattrProfile.xml, shconfProfile.xml, smtctlProfile.xml,
syscorepathProfile.xml, sysdumpdevProfile.xml, trcctlProfile.xml,
trustchkProfile.xml, tsdProfile.xml, vmoProfile.xml" version="2.0.0"
date="2010-08-20T01:11:26Z" readOnly="true">
<Catalog id="acctlParam" version="2.0">
  <Parameter name="turacct"/>
  <Parameter name="agarm"/>
  <Parameter name="agke"/>
  <Parameter name="agproc"/>
  <Parameter name="isystem"/>
  <Parameter name="iprocess"/>
  <Parameter name="email_addr"/>
....

```

---

As new tunable parameters become available new catalogs can be created to expand the value of the AIX Runtime Expert capabilities.

## AIX Runtime Expert commands

The current commands available in AIX Runtime Expert to manipulate profiles and use catalogs are:

<b>artexget</b>	Extract rconfiguration and tuning parameter information from a running system or from a specified configuration profile
<b>artexset</b>	Set values on a system from a profile to take effect immediately or after system restart
<b>artexdiff</b>	Compare values between a running system and a profile, or compare between two profiles
<b>artexmerge</b>	Combine the contents of two or more profiles into a single profile
<b>artexlist</b>	List configuration profiles or catalog that exist on local system or on the LDAP server.

The **artexget** command output can be in the following formats:

- ▶ The txt variable specifies plain text format.
- ▶ The csv variable specifies comma separated values format.
- ▶ The xml format specifies xml format. This is the default format.

The **artexset** command will dynamically set the specified tunables if none of them are restricted. It can also specify it must be applied at each boot of the



system. By default, this command also creates a rollback profile which allows you to undo a profile change if needed.

For detailed parameters see the man pages or info center:

[http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.baseadm/doc/baseadmndita/aix\\_ev.htm](http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.baseadm/doc/baseadmndita/aix_ev.htm)

## Building a AIX Runtime Expert profile

The following steps are will create a profile on a system

1. Create a profile from the running system based on default profile and catalog using the **artexget** command. The result of that command is a XML file that can be modified with any XML editor or any text editor.
2. User created profiles can be customized by changing the values of the parameters or by removing some of the parameters that are not required.
3. Verify that the profile changes have been saved correctly by comparing them against the current system settings using the the **artexdiff** command. It will displays the parameters that were modified. The `<FirstValue>` displays the value of the profile, and the `<SecondValue>` displays the value of the current system.
4. Use the **artexset** command to set a system with the parameters from the new profile. The **artexset** command allows for specifying when the new parameters are to take effect, immediately, at next boot or at each system restart.

**Note:** When the `-t` option is specified, the command **artexset** tests the correctness of the profile. The command checks whether the profile has the correct XML format. Also, it checks whether the parameters defined in the profile are valid and supported by AIX Runtime Expert.

The following sections cover two example of the use of the AIX Runtime Expert commands.

### 5.4.2 Changing mkuser defaults example

In this example the desire is to change the following default parameters when creating users:

- ▶ User home directory to be located in `/userhome` directory
- ▶ Set the shell to `/usr/bin/ksh93`

Using AIX Runtime Expert a new profile can be created with the desired changes. It is also possible to return to the default system (rollback) without knowing which system config file needs to be modified.

### Listing of current environment setting

To get the default environment setting for mkuser setting, the **artexget** command is used with the profile called `mkuser.defaultProfile.xml` as shown in Example 5-34.

#### *Example 5-34 Default mkuser profile*

---

```
# cd /etc/security/artex/samples
# artexget -r mkuser.defaultProfile.xml
<?xml version="1.0" encoding="UTF-8"?>
<Profile origin="get" version="2.0.1" date="2010-09-07T20:43:32Z">
  <Catalog id="mkuser.default.adminParam" version="2.0">
    <Parameter name="account_locked" value=""/>
    ...
    <Parameter name="home" value="/home/$USER"/>
    ...
    <Parameter name="shell" value="/usr/bin/ksh"/>
    ...
  </Catalog>
</Profile>
```

---

Note that the default home is `/home/$USER` and the default shell is `/usr/bin/ksh`. Creating the user `user1` with that default profile would result in an entry in `/etc/passwd`:

#### *Example 5-35 Default user creation*

---

```
# grep user1 /etc/passwd
user1:*:204:1::/home/user1:/usr/bin/ksh
```

---

### Modify current setting

In order to create a new profile the **artexget** command is used to create a new profile based on the system defaults and then edit the new profile is edited with the desired changes. Example 5-36 shows these steps.

#### *Example 5-36 Building a new profile based on the*

---

```
# cd /etc/security/artex/samples
# artexget -r mkuser.defaultProfile.xml > /tmp/mkuser1.xml
vi /tmp/mkuser1.xml
```

---

**Note:** For this particular example the `mkuser.defaultProfile.xml` file has two sets of parameters. One for the admin user and the other for an ordinary user. The home directory and shell changes were only made to the parameters for the ordinary user.

After updating the new profile with new values for the home directory and shell the `artexdiff -c -r` command is used to check the changes. Example 5-34 on page 172 shows the results of this command.

*Example 5-37 XLM output of new profile and running system differences*

---

```
# artexdiff -c -r /tmp/mkuser1.xml
<?xml version="1.0" encoding="UTF-8"?>
<DifferenceData>
  <Parameter name="shell" catalogName="mkuser.default.userParam"
result="value">
  <FirstValue>/usr/bin/ksh93</FirstValue>
  <SecondValue>/usr/bin/ksh</SecondValue>
</Parameter>
  <Parameter name="home" catalogName="mkuser.default.userParam"
result="value">
  <FirstValue>/userhome/$USER</FirstValue>
  <SecondValue>/home/$USER</SecondValue>
</Parameter>
</DifferenceData>
```

---

A summary listing is available with `artexdiff -c -r -f txt` command as shown in Example 5-38.

*Example 5-38 Text output of the new profile and the running system differences*

---

```
# artexdiff -c -r -f txt /tmp/mkuser1.xml
/tmp/mkuser1.xml | System Values
mkuser.default.userParam:shell /usr/bin/ksh93 | /usr/bin/ksh
mkuser.default.userParam:home /userhome/$USER | /home/$USER
```

---

## Apply the new profile and checking the result

Use the `artexset` command with the new profile to change the system defaults as shown in Example 5-39.

*Example 5-39 Applying the new profile*

---

```
# artexset /tmp/mkuser1.xml
```

---

Now any user created will use the new defaults as shown in Example 5-40

*Example 5-40 Creating a new user with the new defaults*

---

```
# mkuser user3
# grep user3 /etc/passwd
user3:*:206:1::/userhome/user3:/usr/bin/ksh93
```

---

Note the new user is now using the /userhome directory instead of the /home directory and the new user is also using the ksh93 shell.

### Profile rollback

In case there is a need to remove the new configuration from the system, the **artexset -u** command will restore parameter values to the value of the last applied profile. The **artexdiff** command can be used to verify the result.

## 5.4.3 Schedo and ioo profile merging example

In this example it is desired to configure the two tunables that are in different profiles. First is the affinity\_lim tunable and the second is posix\_aio\_maxservers. These values are described in the /etc/security/artex/samples default profile directory in multiple profile files:

- ▶ all.xml
- ▶ default.xml
- ▶ iooProfile.xml for posix\_aio\_maxservers
- ▶ schedoProfile.xml for affinity\_lim

It is possible to get the current values for all.xml or default.xml and remove all non needed entries, but it is easier to create a new profile file using the profile templates iioProfile.xml and schedoProfile.xml and then merging them. The following steps are:

- ▶ Get the runtime values for ioo command
- ▶ Get the runtime values for schedo command
- ▶ Create a merge profile
- ▶ Edit profile to remove all <Parameter name= > entries not needed. But don't remove the catalog entries.
- ▶ Check the profile for correctness using the **artexset -t** command
- ▶ Check the current system values with the **artexget -r -f txt** command
- ▶ Check to see actions would be required, like a system restart, when these parameters are changed with the **artexset -p** command
- ▶ Check the running system values with the new profile using the **artexdiff -r -c -f txt** command.

Example 5-41 shows the execution of these steps. In this example the `affinity_lim` is changed from 7 to 6 and the `posix_aio_maxservers` is changed from 30 to 60 using the vi editor.

*Example 5-41 Creating a new merged profile*

---

```
# cd /etc/security/artex/samples
# artexget -r iooProfile.xml > /tmp/1.xml
# artexget -r schedoProfile.xml > /tmp/2.xml
# artexmerge /tmp/1.xml /tmp/2.xml > /tmp/3.x>
# vi /tmp/3.xml

# cat /tmp/3.xml
<?xml version="1.0" encoding="UTF-8"?>
<Profile origin="merge: /tmp/1.xml, /tmp/2.xml" version="2.0.0"
date="2010-09-09T04:45:19Z">
  <Catalog id="iooParam" version="2.0">
    <Parameter name="posix_aio_maxservers" value="60"/>
  </Catalog>
  <Catalog id="schedoParam" version="2.0">
    <Parameter name="affinity_lim" value="6"/>
  </Catalog>
</Profile>

# artexset -t /tmp/3.xml
Profile correctness check successful.

# artexget -r -f txt /tmp/3.xml
Parameter name      Parameter value
-----
##Begin: schedoParam
affinity_lim        7
posix_aio_maxservers 30
##End: iooParam

# artexset -p /tmp/3.xml
#Parameter name:Parameter value:Profile apply type:Catalog apply
type:Additional Action
affinity_lim:6:now_perm:now_perm:
posix_aio_maxservers:60:now_perm:now_perm:

# artexdiff -r -c -f txt /tmp/3.xml
/tmp/3.xml | System Values

schedoParam:affinity_lim 6 | 7
iooParam:posix_aio_maxservers 60 | 30
```

---

## 5.4.4 Latest enhancement

With the AIX 6.1 TL 6, new enhancements to AIX Runtime Expert includes

- ▶ LDAP support to distribute files accross the network
- ▶ NIM server remote setting
- ▶ Capability to do profile versioning meaning that output profiles can have customized version numbers (**artexget -V** option)
- ▶ Custom profile description can be added to the profile output by using the **artexget -m** command option
- ▶ Prioritization of parameters and catalogs for set operation
- ▶ Snap command updates
- ▶ Director plug-in enablement (see fileset artex.base.agent)

The Director plug-in is also known as AIX Profile Manager (APM) which allows views and runtime configuration profile management over groups of systems across the data center.

It uses LDAP for distributing files across the network. See **mksecldap**, **secldapcintd** and **ldapadd** commands. The configuration ldap file is found as `/etc/security/ldap/ldap.cfg`

Use of APM allows retrieval, copy, modification and delete of profile in an easy GUI way like using check box style over AIX Runtime Expert templates.

See Director plug-in documentation for more information in the System Director Information Center.

On a NIM server the **artexremset** provides the ability to execute **artexset** commands on each client with a designated profile provided by the server or a profile stored on an LDAP server. The command syntax would be similar to

```
artexremset -L ldap://profile1.xml client1 client2
```

To retrieve a profile on LDAP server you can use the command:

```
artexget ldap://profile1.xml
```

## 5.5 Removal of CSM

Starting with AIX V7.1, the Cluster Systems Management (CSM) software will no longer ship with AIX media.CSM will not be supported with AIX V7.1. Table 5-2 on page 177 lists the filesets that have been removed:

Table 5-2 Removed CSM fileset packages

Fileset	Description
csm.bluegene	CSM support on Blue Gene®
csm.client	Cluster Systems Management Client
csm.core	Cluster Systems Management Core
csm.deploy	Cluster Systems Management Deployment Component
csm.diagnostics	Cluster Systems Management Probe Manager / Diagnostics
csm.dsh	Cluster Systems Management Dsh
csm.essl	Cluster Systems Management ESSL Solution Pack
csm.gpfs	Cluster Systems Management GPFS™ Solution Pack
csm.gui.dcem	Distributed Command Execution Manager Runtime Environment
csm.gui.websm	CSM Graphical User Interface.
csm.hams	Cluster Systems Management HA
csm.hc_utils	Cluster Systems Management Hardware Control Utils
csm.hpsnm	IBM Switch Network Manager
csm.ll	Cluster Systems Management LoadLeveler® Solution Pack
csm.msg.*	CSM Core Function Messages
csm.pe	Cluster Systems Management PE Solution Pack
csm.pessl	CSM Parallel Engineering Scientific Subroutines Library
csm.server	Cluster Systems Management Server

IBM is shifting to a dual-prong strategy for the system management of IBM server clusters. The strategy and plans have diverged to meet the unique requirements of High Performance Computing (HPC) customers as compared to that of general computing customers.

### High Performance Computing

For HPC customers, the Extreme Cloud Administration Toolkit (xCAT), an open source tool originally developed for IBM system x clusters, has been enhanced to

support all of the HPC capabilities of CSM on all of the platforms that CSM currently supports. Customers can begin planning to transition to this strategic cluster system management tool for HPC. IBM will continue to enhance xCAT to meet the needs of the HPC customer set.

xCAT provides some improvements over CSM. These include:

- ▶ Better scalability, including hierarchical management.
- ▶ Support for a broader range of hardware and operating systems.
- ▶ iSCSI support.
- ▶ Automatic setup of additional services: DNS, syslog, NTP, LDAP.
- ▶ Automatic node definition through discovery process.

Refer to the following publications for detailed information relating to xCAT:

- ▶ *xCAT 2 Guide for the CSM System Administrator*  
<http://www.redbooks.ibm.com/redpapers/pdfs/redp4437.pdf>

## General Computing

For general computing customers who operate non-HPC clustering infrastructure, IBM Systems Director and its family of products are IBM's strategic cross-platform system management solution.

IBM Systems Director helps customers achieve the full benefits of virtualization within their data center by reducing the complexity of systems management. IBM Systems Director VMControl™ Image Manager V2.2, a plug-in to IBM Systems Director, provides support to manage and automate the deployment of virtual appliances from a centralized location.

Together, IBM Systems Director and VMControl provide many cluster management capabilities found in CSM, such as systems discovery, node inventory, node groups, event monitoring, firmware flashing, and automated responses. They also provide many cluster management capabilities like CSM's distributed command execution and remote console, NIM-based AIX mksysb installation for HMC and IVM managed LPARs and the deployment of one or many AIX and/or Linux® virtual server images. IBM Systems Director includes a command-line interface (CLI) for scripting most cluster management functions.

For more information relating to IBM Systems Director, please refer to the following websites:

<http://www.ibm.com/systems/management/director/>

<http://www.ibm.com/power/software/management/>



Other functions of CSM have been ported to the Distributed Systems Management (DSM) package. For example, commands such as dsh and dcp are located in this package. This component is required in an IBM Systems Director environment. The dsm.core package was first shipped with AIX V6.1 with the 6100-03 Technology Level. Documentation relating to configuration and usage is located in the /opt/ibm/sysmgt/dsm/doc/dsm\_tech\_note.pdf file from the dsm.core fileset. Please refer to the following websites for install and usage information relating to this fileset:

[http://publib.boulder.ibm.com/infocenter/director/v6r2x/index.jsp?topic=/com.ibm.director.install.helps.doc/fqm0\\_t\\_preparing\\_to\\_install\\_ibm\\_director\\_on\\_aix.html](http://publib.boulder.ibm.com/infocenter/director/v6r2x/index.jsp?topic=/com.ibm.director.install.helps.doc/fqm0_t_preparing_to_install_ibm_director_on_aix.html)

[http://publib.boulder.ibm.com/infocenter/director/v6r2x/index.jsp?topic=/com.ibm.director.cli.helps.doc/fqm0\\_r\\_cli\\_remote\\_access\\_cmds.html](http://publib.boulder.ibm.com/infocenter/director/v6r2x/index.jsp?topic=/com.ibm.director.cli.helps.doc/fqm0_r_cli_remote_access_cmds.html)

Functionality relating to Dynamic Logical Partitioning (DLPAR), previously provided by CSM, has been ported to Reliable Scalable Cluster Technology (RSCT). Previous releases of AIX required the csm.core fileset be installed in order to support DLPAR functions. This functionality is now provided by the rsct.core.rmc fileset. This fileset is automatically installed by default.

## 5.6 Removal of IBM Text-to-Speech

The IBM Text-to-Speech (TTS) package is a speech engine that allows applications to produce speech. Starting with AIX V7.1, the IBM TTS will no longer ship with the AIX Expansion Pack. The contents of the Expansion Pack vary over time. New software products can be added, changed, or removed. Changes to the content of the AIX Version 7.1 Expansion Pack are announced either as part of an AIX announcement or independently of the release announcement.

TTS is installed in the /usr/opt/ibmtts directory. The following filesets will no longer be included with this media:

### **tts\_access.base**

IBM TTS runtime base

### **tts\_access.base.en\_US**

IBM TTS runtime (U.S. English)

Refer to the following website for latest information relating to the contents of the AIX Expansion Pack:

<http://www.ibm.com/systems/power/software/aix/expansionpack/>

## 5.7 AIX device renaming

Devices can be renamed in AIX 6.1 TL6 and 7.1. The **rendev** command is used for renaming the devices. One of the use cases would be to rename a group of disks on which application data may reside, to be able to distinguish them from other disks on the system.

Once the device is renamed using **rendev** command, the device entry under `/dev/` corresponding to the old name will go away. A new entry under `/dev/` will be seen corresponding to the new name. Applications should refer to the device using the new name.

**Note:** Certain devices such as `/dev/console`, `/dev/mem`, `/dev/null`, and others that are identified only with `/dev` special files cannot be renamed. These devices typically do not have any entry in ODM configuration database.

Some devices may have special requirements on their names in order for other devices or applications to use them. Using the **rendev** command to rename such a device may result in the device being unusable.

The devices cannot be renamed if they are in use.

Example 5-42 shows how the disk `hdisk11` is renamed to `testdisk1`.

*Example 5-42 Renaming device*

---

```
# lspv
hdisk0          00cad74f7904d234          rootvg          active
hdisk1          00cad74fa9d4a6c2          None
hdisk2          00cad74fa9d3b8de          None
hdisk3          00cad74f3964114a          None
hdisk4          00cad74f3963c575          None
hdisk5          00cad74f3963c671          None
hdisk6          00cad74f3963c6fa          None
hdisk7          00cad74f3963c775          None
hdisk8          00cad74f3963c7f7          None
hdisk9          00cad74f3963c873          None
hdisk10         00cad74f3963ca13          None
hdisk11         00cad74f3963caa9          None
hdisk12         00cad74f3963cb29          None
```

```

hdisk13      00cad74f3963cba4      None
# rendez -l hdisk11 -n testdisk1
# lspv
hdisk0      00cad74f7904d234      rootvg      active
hdisk1      00cad74fa9d4a6c2      None
hdisk2      00cad74fa9d3b8de      None
hdisk3      00cad74f3964114a      None
hdisk4      00cad74f3963c575      None
hdisk5      00cad74f3963c671      None
hdisk6      00cad74f3963c6fa      None
hdisk7      00cad74f3963c775      None
hdisk8      00cad74f3963c7f7      None
hdisk9      00cad74f3963c873      None
hdisk10     00cad74f3963ca13      None
testdisk1  00cad74f3963caa9      None
hdisk12     00cad74f3963cb29      None
hdisk13     00cad74f3963cba4      None

```

---

## 5.8 1024 Hardware thread enablement

AIX 7.1 provides support to run the partition with upto 1024 logical CPUs, both in dedicated and shared processor modes. This has been tested on the IBM,9119-FHB system. The earlier limit on the number of supported CPUs was 256 on AIX 6.1 TL4 on Power 7 systems.

Example 5-43 shows sample output from few commands executed on Power 795 system giving details about the system configuration. The **lsattr** command gives information such as modelname. Processor and memory information is seen under the **lparstat** command output. Scheduler Resource Allocation Domains (SRAD) information is seen under the **lssrad** command output.

### *Example 5-43 Power 795 system configuration*

---

```

# lsattr -El sys0
SW_dist_intr  false          Enable SW distribution of interrupts      True
autorestart  true           Automatically REBOOT OS after a crash     True
boottype     disk          N/A                                       False
capacity_inc  1.00         Processor capacity increment             False
capped       true          Partition is capped                       False
conslogin    enable        System Console Login                     False
cpuguard     enable        CPU Guard                                 True

```

dedicated	true	Partition is dedicated	False
enhanced_RBAC	true	Enhanced RBAC Mode	True
ent_capacity	256.00	Entitled processor capacity	False
frequency	6400000000	System Bus Frequency	False
fullcore	true	Enable full CORE dump	True
fwversion	IBM,ZH720_054	Firmware version and revision levels	False
ghostdev	0	Recreate devices in ODM on system change	True
id_to_partition	0X80000D2F7C100002	Partition ID	False
id_to_system	0X80000D2F7C100000	System ID	False
iostat	false	Continuously maintain DISK I/O history	True
keylock	normal	State of system keylock at boot time	False
log_pg_dealloc	true	Log predictive memory page deallocation events	True
max_capacity	256.00	Maximum potential processor capacity	False
max_logname	9	Maximum login name length at boot time	True
maxbuf	20	Maximum number of pages in block I/O BUFFER CACHE	True
maxmbuf	0	Maximum Kbytes of real memory allowed for MBUFS	True
maxpout	8193	HIGH water mark for pending write I/Os per file	True
maxuproc	64000	Maximum number of PROCESSES allowed per user	True
min_capacity	1.00	Minimum potential processor capacity	False
minpout	4096	LOW water mark for pending write I/Os per file	True
modelname	IBM,9119-FHB	Machine name	False
ncargs	256	ARG/ENV list size in 4K byte blocks	True
nfs4_acl_compat	secure	NFS4 ACL Compatibility Mode	True
ngroups_allowed	128	Number of Groups Allowed	True
pre430core	false	Use pre-430 style CORE dump	True
pre520tune	disable	Pre-520 tuning compatibility mode	True
realmem	4219994112	Amount of usable physical memory in Kbytes	False
rtasversion	1	Open Firmware RTAS version	False
sed_config	select	Stack Execution Disable (SED) Mode	True
systemid	IBM,020288C75	Hardware system identifier	False
variable_weight	0	Variable processor capacity weight	False
# lparstat -i			
Node Name		: test1	

Partition Name	: test1new
Partition Number	: 2
Type	: Dedicated
Mode	: Capped
Entitled Capacity	: 256.00
Partition Group-ID	: 32770
Shared Pool ID	: -
Online Virtual CPUs	: 256
Maximum Virtual CPUs	: 256
Minimum Virtual CPUs	: 1
Online Memory	: 4121088 MB
Maximum Memory	: 4194304 MB
Minimum Memory	: 256 MB
Variable Capacity Weight	: -
Minimum Capacity	: 1.00
Maximum Capacity	: 256.00
Capacity Increment	: 1.00
Maximum Physical CPUs in system	: 256
Active Physical CPUs in system	: 256
Active CPUs in Pool	: -
Shared Physical CPUs in system	: 0
Maximum Capacity of Pool	: 0
Entitled Capacity of Pool	: 0
Unallocated Capacity	: -
Physical CPU Percentage	: 100.00%
Unallocated Weight	: -
Memory Mode	: Dedicated
Total I/O Memory Entitlement	: -
Variable Memory Capacity Weight	: -
Memory Pool ID	: -
Physical Memory in the Pool	: -
Hypervisor Page Size	: -
Unallocated Variable Memory Capacity Weight:	-

```

Unallocated I/O Memory entitlement      : -
Memory Group ID of LPAR                 : -
Desired Virtual CPUs                     : 256
Desired Memory                           : 4121088 MB
Desired Variable Capacity Weight         : -
Desired Capacity                         : 256.00
Target Memory Expansion Factor           : -
Target Memory Expansion Size              : -
Power Saving Mode                         : Disabled

```

```
# lssrad -av
```

REF1	SRAD	MEM	CPU
0			
	0	94341.00	0 4 8 12 16 20 24 28
	1	94711.00	32 36 40 44 48 52 56 60
	2	94711.00	64 68 72 76 80 84 88 92
	3	94711.00	96 100 104 108 112 116 120 124
1			
	4	94711.00	128 132 136 140 144 148 152 156
	5	94695.00	160 164 168 172 176 180 184 188
	6	94695.00	192 196 200 204 208 212 216 220
	7	94695.00	224 228 232 236 240 244 248 252
2			
	8	94695.00	256 260 264 268 272 276 280 284
	9	94695.00	288 292 296 300 304 308 312 316
	10	94695.00	320 324 328 332 336 340 344 348
	11	94695.00	352 356 360 364 368 372 376 380
3			
	12	94695.00	384 388 392 396 400 404 408 412
	13	94695.00	416 420 424 428 432 436 440 444
	14	94695.00	448 452 456 460 464 468 472 476
	15	94695.00	480 484 488 492 496 500 504 508
4			
	16	93970.94	512 516 520 524 528 532 536 540

17	45421.00	544 548 552 556 560 564 568 572
18	94695.00	576 580 584 588 592 596 600 604
19	94695.00	608 612 616 620 624 628 632 636
5		
20	94695.00	640 644 648 652 656 660 664 668
21	94695.00	672 676 680 684 688 692 696 700
22	94695.00	704 708 712 716 720 724 728 732
23	94695.00	736 740 744 748 752 756 760 764
6		
24	94695.00	768 772 776 780 784 788 792 796
25	94695.00	800 804 808 812 816 820 824 828
26	94695.00	832 836 840 844 848 852 856 860
27	94864.00	864 868 872 876 880 884 888 892
7		
28	94896.00	896 900 904 908 912 916 920 924
29	94880.00	928 932 936 940 944 948 952 956
30	94896.00	960 964 968 972 976 980 984 988
31	94309.00	992 996 1000 1004 1008 1012 1016 1020

---

## 5.9 Kernel Memory Pinning

AIX 6.1 TL6 and 7.1 provides facility to keep AIX kernel and kernel extension data in physical memory for as long as possible. This feature is referred to as Kernel Memory Pinning or Locking. On systems running with sufficiently large amount of memory, locking avoids unnecessary kernel page faults thereby providing improved performance.

Kernel memory locking differs from traditional pinning of memory in the following ways:

- ▶ Pinning is an explicit operation performed using the kernel services such as `pin()`, `ltpin()`, `xlate_pin()`, and others. A pinned page is never unpinned until it is explicitly unpinned using the kernel services. Kernel locking is an implicit operation. There are no kernel services to lock and unlock a page.
- ▶ Pinned memory is never eligible for stealing by the Least Recently Used (LRU) page replacement demon. Locked memory on the other hand is eligible

for stealing when no other pages are available for stealing. The real advantage of locked memory is that it is not stolen until no other option is left. Because of this, there are more chances of retaining kernel data in memory for a longer period.

- ▶ Pinned memory has a hard limit. Once the limit is reached, the pin service can fail with ENOMEM. Locking enforces a soft limit in the sense that if a page frame can be allocated for the kernel data, it is automatically locked. Never is the case that a page frame is not locked due to some locking limit, because there is no such limit.
- ▶ User memory can be pinned using the `mlock()` system call. User memory on the other hand are not locked.

The following are considered as kernel memory which are eligible for locking:

- ▶ Kernel segment where the kernel itself resides
- ▶ All global kernel space such as kernel heaps, message buffer (mbuf) heaps, Ldata heap, mtrace buffers, scb pool and others.
- ▶ All kernel space private to a process such as Process private segments for 64-bit processes, kernel thread segments, loader overflow segment and others.
- ▶ And few others

The following are *not* considered as kernel memory and are *not* locked:

- ▶ Process text and data (heaps and user-space stacks)
- ▶ Shared library text and data
- ▶ Shared memory segments, mmaped segments
- ▶ File cache segments
- ▶ And few others

The following Virtual Memory Management (VMM) VMM tunables were added or modified to support kernel memory locking.

<code>vmm_klock_mode</code>	New tunable to enable and disable kernel memory locking.
<code>maxpin</code>	Kernel's locked memory is treated like pinned memory. Therefore, the default <code>maxpin%</code> is raised from 80% to 90% if kernel locking is enabled.

Example 5-44 on page 187 shows how to configure kernel memory locking using the `vmo` tunable.



*Example 5-44 Configuring kernel memory locking*

```
# vmo -h vmm_klock_mode
Help for tunable vmm_klock_mode:
Purpose:
Select the kernel memory locking mode.
Values:
    Default: 2
    Range: 0 - 3
    Type: Bosboot
    Unit: numeric

Tuning:
Kernel locking prevents paging out kernel data. This improves system performance in many
cases. If set to 0, kernel locking is disabled. If set to 1, kernel locking is enabled
automatically if Active Memory Expansion (AME) feature is also enabled. In this mode,
only a subset of kernel memory is locked. If set to 2, kernel locking is enabled
regardless of AME and all of kernel data is eligible for locking. If set to 3, only the
kernel stacks of processes are locked in memory. Enabling kernel locking has the most
positive impact on performance of systems that do paging but not enough to page out
kernel data or on systems that do not do paging activity at all. Note that 1, 2, and 3
are only advisory. If a system runs low on free memory and performs extensive paging
activity, kernel locking is rendered ineffective by paging out kernel data. Kernel
locking only impacts pageable page-sizes in the system.
```

```
# vmo -L vmm_klock_mode
NAME                CUR  DEF  BOOT  MIN  MAX  UNIT      TYPE
DEPENDENCIES
-----
vmm_klock_mode      2    2    2     0    3   numeric    B
-----
```

```
# vmo -o vmm_klock_mode
vmm_klock_mode = 2

# vmo -r -o vmm_klock_mode=1
Modification to restricted tunable vmm_klock_mode, confirmation required yes/no yes
Setting vmm_klock_mode to 1 in nextboot file
Warning: some changes will take effect only after a bosboot and a reboot
Run bosboot now? yes/no yes
```

```
bosboot: Boot image is 45651 512 byte blocks.
Warning: changes will take effect only at next reboot
```

```
# vmo -L vmm_klock_mode
NAME                CUR  DEF  BOOT  MIN  MAX  UNIT      TYPE
DEPENDENCIES
-----
vmm_klock_mode      2    2    1     0    3   numeric    B
-----
```

The following are few guidelines for setting `vmm_klock_mode` tunable:

- ▶ Setting `vmm_klock_mode` to value 2 or 3 is recommended for those systems where applications are sensitive to page-faults inside the kernel.
- ▶ Value 2 is recommended for systems where no page-faults of any kind are expected, because kernel is already locked in memory. However, by setting value 2 the system is better prepared for future optimization in the kernel that require a fully-pinned kernel.
- ▶ For systems where value 2 results in excessive paging of user-space data, value 3 is recommended.
- ▶ Systems that see their paging spaces getting filled up such that the overall usage does not decrease much even when no applications are running may benefit from using value 3. This is because a nearly full paging space whose usage does not seem to track the usage by applications, is most likely experiencing heavy paging of kernel data. For such systems, value 2 is also worth an experiment; however, the risk of excessive paging of user-space data may be greatly increased.

## 5.10 ksh93 enhancements

In addition to the default system Korn shell (`/usr/bin/ksh`), AIX provides an enhanced version available as Korn Shell (`/usr/bin/ksh93`) shipped as a 32-bit binary. This enhanced version is mostly upward compatible with current default version, and includes additional features that are not available in `/usr/bin/ksh`.

Starting AIX 7.1, *ksh93* is shipped as a 64-bit binary (Version M 93t+ 2009-05-05). This 64-bit binary is built from a more recent code base to include additional features.

For complete list of information on `ksh93`, refer to `/usr/bin/ksh93` man pages.

## 5.11 DWARF

AIX V7.1 adds support for the standard DWARF debugging format, which is a modern standard for specifying the format of debugging information in executables. It is used by a wide variety of operating systems and provides greater extensibility and compactness. The widespread use of DWARF also increases the portability of software for developers of compilers and other debugging tools between AIX and other operating systems.

**Author Comment:** The IBM XL C/C++ version 11 compiler generates DWARF debug format when compiled with `-qdbgfmt=dwarf` option. **(PTF available Sept.24, 2010)**

Detailed DWARF debugging information format can be found at the following:

<http://www.dwarfstd.org>

## 5.12 AIX Event Infrastructure extension and RAS

This feature first became available in AIX 6.1 TL 04 and has been enhanced in AIX 6.1 TL 06.

AIX Event Infrastructure is an event monitoring framework for monitoring pre-defined and user-defined events.

Within the context of the AIX Event Infrastructure, an event is defined as:

- ▶ any change of state which can be detected by the kernel or a kernel extension at the exact moment (or an approximation) the change occurs.
- ▶ any change of value which can be detected by the kernel or a kernel extension at the exact moment (or an approximation) the change occurs.

In both the change of state and change of value, the events which may be monitored are represented as a pseudo file system.

### 5.12.1 Some advantages of AIX Event Infrastructure

Advantages of the AIX Event infrastructure include:

- ▶ No need for constant polling. Users monitoring the events are notified when those events occur
- ▶ Detailed information about an event (such as stack trace and user and process information) is provided to the user monitoring the event
- ▶ Existing file system interfaces are used so that there is no need for a new API
- ▶ Control is handed to the AIX Event Infrastructure at the exact time the event occurs

For further information on the AIX Event Infrastructure, visit:  
[http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.baseadm/doc/baseadmndita/aix\\_ev.htm](http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.baseadm/doc/baseadmndita/aix_ev.htm)

## 5.12.2 Configuring the AIX Event Infrastructure

The following procedure outlines the activities required to configure the AIX Event Infrastructure:

1. Install the `bos.ahafs` file set (available in AIX 6.1 TL 6 and later).

The AIX V7.1 `bos.ahafs` package content is listed with the `ls1pp -f` command in Example 5-45

*Example 5-45 The `ls1pp -f bos.ahafs` package listing*

---

```
# ls1pp -f bos.ahafs
  Fileset                File
-----
Path: /usr/lib/objrepos
  bos.ahafs 7.1.0.0      /usr/samples/ahafs/samplePrograms/kextEvProd
                        /usr/samples/ahafs/bin/aha.inp
  /usr/samples/ahafs/samplePrograms/evMon/mon_1event
                        /usr/samples/ahafs/samplePrograms/evMon
  /usr/samples/ahafs/samplePrograms/kextEvProd/Makefile
  /usr/samples/ahafs/samplePrograms/evMon/mon_1event.c
                        /usr/include/sys/ahafs_evProds.h
                        /usr/lib/drivers/ahafs.ext
                        /usr/samples/ahafs/README
  /usr/samples/ahafs/samplePrograms/kextEvProd/triggerEvent
                        /usr/lib/ras/autoload/ahafs64.kdb
                        /usr/samples/ahafs/bin/aha
                        /usr/samples/ahafs/samplePrograms
                        /usr/samples/ahafs
  /usr/samples/ahafs/samplePrograms/evMon/Makefile
  /usr/samples/ahafs/samplePrograms/kextEvProd/testKext.c
                        /usr/samples/ahafs/samplePrograms/evMon/aha.pl
                        /usr/samples/ahafs/bin
  /usr/samples/ahafs/samplePrograms/kextEvProd/triggerEvent.c
  /usr/samples/ahafs/samplePrograms/kextEvProd/README.howToTest
  /usr/samples/ahafs/samplePrograms/kextEvProd/testKext
                        /usr/samples/ahafs/bin/aha.c
  /usr/samples/ahafs/samplePrograms/kextEvProd/testKext_syscalls.exp
  /usr/samples/ahafs/samplePrograms/kextEvProd/kexload
  /usr/samples/ahafs/samplePrograms/evMon/mon_1event.java
```

```

/usr/samples/ahafs/bin/Makefile
/usr/samples/ahafs/samplePrograms/kextEvProd/kexload.c
/usr/samples/ahafs/samplePrograms/evMon/mon_1event.pl

```

```

Path: /etc/objrepos
bos.ahafs 7.1.0.0 NONE

```

---

2. Create the directory for the desired mount point using the **mkdir** command:

```
mkdir /aha
```

3. Run the **mount** command for the file system of type ahafs on the desired mount point in order to load the AHAFS kernel extension and create the file system structure needed by the AIX Event Infrastructure environment as shown in Example 5-46

*Example 5-46 Mounting the file system*

---

```

# mount -v ahafs /aha /aha
# df | grep aha
/aha          -          -          -          15      1% /aha
# genkex | grep aha
f1000000c033c000 19000 /usr/lib/drivers/ahafs.ext

```

---

**Note:** Only one instance of an AHAFS file system may be mounted at a time.

An AHAFS file system may be mounted on any regular directory, but it is suggested that users use /aha mount point.

**Note:** Currently, all directories in AHAFS have a mode of 01777 and all files have a mode of 0666. These modes cannot be changed, but ownership of files and directories may be changed.

Access control for monitoring events is done at the event producer level.

Creation and modification times are not maintained in AHAFS and are always returned as the current time when issuing `stat()` on a file. Any attempt to modify these times will return an error.

### 5.12.3 Use of monitoring sample

For our purpose will use an event monitoring called `evMon` with a C program called `mon_1event` (see package content in Example 5-45 on page 190).

The `mon_1event` monitor is used to monitor a single event occurrence only.

Once the monitor is triggered and the event is reported, the `mon_1event` monitor exits. Any new monitor will need to be re-initiated.

*Example 5-47 The syntax output from the `mon_1event` C program*

---

# `./mon_1event`

SYNTAX: `./mon_1event <aha-monitor-file>`  
`[<key1>=<value1>[;<key2>=<value2>;...]]`

where:

`<aha-monitor-file>` : Pathname of an AHA file with suffix ".mon".  
 The possible keys and their values are:

Keys	values	comments
WAIT_TYPE	WAIT_IN_SELECT (default) WAIT_IN_READ	uses select() to wait. uses read() to wait.
CHANGED	YES (default) or not-YES	monitors state-change. It cannot be used with THRESH_HI.
THRESH_HI	positive integer	monitors high threshold.

Examples:

```
1: ./mon_1event /aha/fs/utilFs.monFactory/var.mon "THRESH_HI=95"
2: ./mon_1event /aha/fs/modFile.monFactory/etc/passwd.mon
"CHANGED=YES"
3: ./mon_1event /aha/mem/vmo.monFactory/npskill.mon
4: ./mon_1event /aha/cpu/waitTmCPU.monFactory/waitTmCPU.mon
"WAIT_TYPE=WAIT_IN_READ;THRESH_HI=50"
```

---

## Creating the monitor file:

Before monitoring an event, the monitor file corresponding to the event must be created. AHAFS does support `open()` with the `O_CREAT` flag.

Example 5-48 on page 193 shows the steps required to monitor the `/tmp` file system for a threshold utilization of 45 %.

In the Example 5-48 on page 193, the following definitions are used:

- ▶ the `mon_1event` C program has been used to open the monitor file

- ▶ the monitor file is the `/aha/fs/utilFs.monFactory/tmp.mo` file
- ▶ the monitor event is the value `THRESH_HI=45`

Generally, the necessary subdirectories may need to be created when the mount point is not the `/` file system. In this example, `/tmp` has not direct subdirectories from the `/`, so there is no need to create and subdirectories.

Next, create the monitoring file called `tmp.mon` for the `/tmp` file system.

**Note:** Monitoring the root file system would require the creation of a monitor called `.mon` in `/aha/fs/utilFs.monFactory`.

*Example 5-48 Creating a monitoring the event*

---

```
# df /tmp
Filesystem      512-blocks      Free %Used    Iused %Iused Mounted on
/dev/hd3         262144    255648    3%      42     1% /tmp
# ls /aha/fs/utilFs.monFactory/tmp.mon
/aha/fs/utilFs.monFactory/tmp.mon
# cat /aha/fs/utilFs.monFactory/tmp.mon
# ./mon_1event /aha/fs/utilFs.monFactory/tmp.mon "THRESH_HI=45"
Monitor file name: /aha/fs/utilFs.monFactory/tmp.mon
Write String   : THRESH_HI=45
Entering select() to wait till the event corresponding to the AHA node
/aha/fs/utilFs.monFactory/tmp.mon occurs.
Please issue a command from another window to trigger this event.
```

---

At this stage, the console in Example 5-48 is paused awaiting the event to trigger.

On another window we issue the `dd` command to create the `/tmp/TEST` file. By using the `dd` command to create the `/tmp/TEST` file, the `/tmp` file system utilization increases to 29 %.

Example 5-49 shows the `dd` command being used to create the `/tmp/TEST` file:

*Example 5-49 Using dd command to increase /tmp filesystem utilization*

---

```
# dd if=unix of=/tmp/TEST
68478+1 records in.
68478+1 records out.
# df /tmp
Filesystem      512-blocks      Free %Used    Iused %Iused Mounted on
/dev/hd3         262144    187168    29%      43     1% /tmp
```

---

Because the /tmp file system did not reach the 45 % threshold limit defined by the THRESH\_HI value, no activity or response was seen the on initial window.

In Example 5-50, a second **dd** command is used to create the /tmp/TEST2 file:

*Example 5-50 Increase of /tmp filesystem utilization to 55%*

---

```
# df /tmp
Filesystem      512-blocks      Free %Used      Iused %Iused Mounted on
/dev/hd3         262144      187168   29%          43      1% /tmp
# dd if=unix of=/tmp/TEST2
68478+1 records in.
68478+1 records out.
# df /tmp
Filesystem      512-blocks      Free %Used      Iused %Iused Mounted on
/dev/hd3         262144      118688   55%          44      1% /tmp
#
```

---

In Example 5-50, the /tmp filesystem utilization has now reached 55 %, which is above the 45 % trigger defined in the value THRESH\_HI, in Example 5-48 on page 193.

The mon\_1event C program will now complete and the initial window will display the response seen in Example 5-51:

*Example 5-51 THRESH\_HI threshold is reached or exceeded*

---

```
The select() completed.
The event corresponding to the AHA node
/aha/fs/utilFs.monFactory/tmp.mon has occurred.
```

```
BEGIN_EVENT_INFO
```

---

To summarize, once a successful write has been performed to the monitor file /aha/fs/utilFs.monFactory/tmp.mon , the monitor will wait on the event in read() or in select().

The select() call will return indicating that the event has occurred. Monitors waiting in select() will need to perform a separate read() to obtain the event data.

Once the event occurs, it will no longer be monitored by the monitor process.

If another monitoring of the event is required, another monitor needs to be initiated to again specify how and when to notify of the alert process.



**Note:** Writing information to the monitor file only prepares AHAFS for a subsequent `select()` or `blocking read()`. Monitoring does not start until a `select()` or `blocking read()` is done.

To prevent multiple threads from overwriting each other's data, if a process already has a thread waiting in a `select()` or `read()` call, another thread's write to the file will return `EBUSY`.

### **Available pre-defined event producers**

A set of pre-defined event producer is available in the system. They are called `modFile`, `modDir`, `utilFs`, `waitTmCPU`, `waitersFreePg`, `waitTmPgInOut`, `vmo`, `schedo`, `pidProcessMon`, `processMon`.

When the system is part of an active cluster, more pre-defined event producers are available such as:

`nodeList`, `clDiskList`, `linkedCl`, `nodeContact`, `nodeState`, `nodeAddress`, `networkAdapterState`, `clDiskState`, `repDiskState`, `diskState`, `vgState`.



## 6



# Performance management

The performance of a computer system is evaluated based on clients expectations and the ability of the system to fulfill these expectations. The objective of performance management is to balance between appropriate expectations and optimizing the available system resources.

Many performance-related issues can be traced back to operations performed by a person with limited experience and knowledge who unintentionally restricts some vital logical or physical resource of the system. Most of these actions may at first be initiated to optimize the satisfaction level of some users, but in the end, they degrade the overall satisfaction of other users.

This chapter discusses the following performance management enhancements:

- ▶ 6.1, “Support for Active Memory Expansion” on page 198
- ▶ 6.2, “Hot Files Detection and filemon” on page 227
- ▶ 6.3, “Memory affinity API enhancements” on page 241
- ▶ 6.4, “iostat command enhancement” on page 244

## 6.1 Support for Active Memory Expansion

Active Memory™ Expansion (AME) is a technology available on IBM POWER7 based processor systems. It provides the capability for expanding a system's effective memory capacity. AME employs memory compression technology to transparently compress in-memory data, allowing more data to be placed into memory. This has the positive effect of expanding the memory capacity for a given system. Please refer to the following website for detailed information relating to AME:

[http://www.ibm.com/systems/power/hardware/whitepapers/am\\_exp.html](http://www.ibm.com/systems/power/hardware/whitepapers/am_exp.html)

With the introduction of AME a tool was required to monitor, report and plan for an AME environment. To assist in planning the deployment of a workload in an AME environment, a tool known as the Active Memory Expansion Planning and Advisory Tool (**amepat**) has been introduced. Several existing AIX performance tools have been modified to monitor AME statistics. This section will discuss the performance monitoring tools related to AME monitoring and reporting.

### 6.1.1 amepat command

To assist in planning the deployment of a workload in an AME environment, a tool known as the Active Memory Expansion Planning and Advisory Tool (**amepat**) has been introduced. This tool is available in AIX V7.1 and in AIX V6.1 with the 6100-04 Technology Level, Service Pack 2. The utility is able to monitor global memory usage for an individual LPAR. The **amepat** command serves two key functions:

- Workload Planning** The **amepat** command can be run to determine if a workload would benefit from AME, and also to provide a list of possible AME configurations for a particular workload.
- Monitoring** When AME is enabled, the **amepat** command can be used to monitor the workload and AME performance statistics.

The tool can be invoked in two different modes:

- Recording** In this mode **amepat** records system configurations and various performance statistics into a user specified recording file.
- Reporting** In this mode the **amepat** command analyzes the system configuration and performance statistics, collected in real time or from the user specified recording file, to generate workload utilization and planning reports.

When considering using AME for an existing workload, the **amepat** command can be used to provide guidance on possible AME configurations. You can run the **amepat** command on an existing system that is not currently using AME. The tool will monitor the memory usage, memory reference patterns, and data compressibility over a (user-configurable) period of time. A report is generated with a list of possible AME configurations for the given workload. Estimated CPU utilization impacts for the different AME configurations are also shown.

The tool can be run on all versions of IBM Power Systems supported by AIX V6.1 and AIX V7.1. This includes POWER4™, POWER5, POWER6 and POWER7 processors.

Two key considerations when running the **amepat** command, when planning for a given workload, are time and duration.

**Time** The time at which to run the tool. To get the best possible results from the tool, it must be run during a period of peak utilization on the system. This ensures that the tool captures peak utilization of memory for the specific workload.

**Duration** The duration to run the tool. A monitoring duration must be specified when starting the **amepat** command. For the best possible results from the tool, it must be run for the duration of peak utilization on the system.

The tool can also be used on AME enabled systems to provide a report of other possible AME configurations for a workload.

The **amepat** command requires privileged access to run in *Workload Planning* mode. If the tool is invoked without the necessary privilege then the planning capability is disabled (-N flag will be turned on implicitly), as shown in Example 6-1.

*Example 6-1 Running amepat without privileged access*

---

```
$ amepat
```

```
WARNING: Running in no modeling mode.
```

```
Command Invoked           : amepat
```

```
Date/Time of invocation   : Mon Aug 30 17:21:25 EDT 2010
```

```
Total Monitored time     : NA
```

```
Total Samples Collected  : NA
```

```
System Configuration:
```

```
-----
```

```

Partition Name           : 75021p01
Processor Implementation Mode : POWER7
Number Of Logical CPUs   : 16
Processor Entitled Capacity : 1.00
Processor Max. Capacity  : 4.00
True Memory              : 8.00 GB
SMT Threads              : 4
Shared Processor Mode    : Enabled-Uncapped
Active Memory Sharing    : Disabled
Active Memory Expansion  : Enabled
Target Expanded Memory Size : 16.00 GB
Target Memory Expansion factor : 2.00

```

System Resource Statistics:	Current
-----	-----
CPU Util (Phys. Processors)	0.10 [ 2%]
Virtual Memory Size (MB)	1697 [ 10%]
True Memory In-Use (MB)	1621 [ 20%]
Pinned Memory (MB)	1400 [ 17%]
File Cache Size (MB)	30 [ 0%]
Available Memory (MB)	14608 [ 89%]

AME Statistics:	Current
-----	-----
AME CPU Usage (Phy. Proc Units)	0.00 [ 0%]
Compressed Memory (MB)	203 [ 1%]
Compression Ratio	2.35
Deficit Memory Size (MB)	74 [ 0%]

This tool can also be used to monitor CPU and memory usage statistics only. In this mode, the **amepat** command will gather CPU and memory utilization statistics but will not provide any workload planning data or reports. If it is invoked without any duration or interval, the **amepat** command will provide a snapshot report of the LPARs memory and CPU utilization, as shown in Example 6-2.

*Example 6-2 CPU and memory utilization snapshot from amepat*

```
# amepat
```

```

Command Invoked           : amepat

Date/Time of invocation   : Mon Aug 30 17:37:58 EDT 2010
Total Monitored time      : NA
Total Samples Collected  : NA

```

## System Configuration:

```

-----
Partition Name           : 75021p01
Processor Implementation Mode : POWER7
Number Of Logical CPUs   : 16
Processor Entitled Capacity : 1.00
Processor Max. Capacity  : 4.00
True Memory              : 8.00 GB
SMT Threads              : 4
Shared Processor Mode    : Enabled-Uncapped
Active Memory Sharing    : Disabled
Active Memory Expansion  : Enabled
Target Expanded Memory Size : 16.00 GB
Target Memory Expansion factor : 2.00

```

System Resource Statistics:	Current
-----	-----
CPU Util (Phys. Processors)	0.45 [ 11%]
Virtual Memory Size (MB)	1706 [ 10%]
True Memory In-Use (MB)	1590 [ 19%]
Pinned Memory (MB)	1405 [ 17%]
File Cache Size (MB)	11 [ 0%]
Available Memory (MB)	13994 [ 85%]

AME Statistics:	Current
-----	-----
AME CPU Usage (Phy. Proc Units)	0.02 [ 1%]
Compressed Memory (MB)	237 [ 1%]
Compression Ratio	2.25
Deficit Memory Size (MB)	700 [ 4%]

---

Example 6-3 demonstrates how to generate a report with a list of possible AME configurations for a workload. The tool will include an estimate of the CPU utilization impacts for the different AME configurations.

*Example 6-3 List possible AME configurations for an LPAR with amepat*

---

```
# amepat 1
```

```

Command Invoked           : amepat 1

Date/Time of invocation   : Tue Aug 31 12:35:17 EDT 2010
Total Monitored time     : 1 mins 51 secs
Total Samples Collected  : 1

```

## System Configuration:

```

-----
Partition Name           : 75021p02
Processor Implementation Mode : POWER7
Number Of Logical CPUs   : 16
Processor Entitled Capacity : 1.00
Processor Max. Capacity  : 4.00
True Memory              : 8.00 GB
SMT Threads              : 4
Shared Processor Mode    : Enabled-Uncapped
Active Memory Sharing    : Disabled
Active Memory Expansion  : Disabled

```

## System Resource Statistics: Current

```

-----
CPU Util (Phys. Processors) 1.74 [ 44%]
Virtual Memory Size (MB)     5041 [ 62%]
True Memory In-Use (MB)     5237 [ 64%]
Pinned Memory (MB)          1448 [ 18%]
File Cache Size (MB)        180 [ 2%]
Available Memory (MB)       2939 [ 36%]

```

## Active Memory Expansion Modeled Statistics :

```

-----
Modeled Expanded Memory Size : 8.00 GB
Achievable Compression ratio :2.12

```

Expansion Factor	Modeled True Memory Size	Modeled Memory Gain	CPU Usage Estimate
1.00	8.00 GB	0.00 KB [ 0%]	0.00 [ 0%]
1.11	7.25 GB	768.00 MB [ 10%]	0.00 [ 0%]
1.19	6.75 GB	1.25 GB [ 19%]	0.00 [ 0%]
1.34	6.00 GB	2.00 GB [ 33%]	0.00 [ 0%]
1.40	5.75 GB	2.25 GB [ 39%]	0.00 [ 0%]
1.53	5.25 GB	2.75 GB [ 52%]	0.00 [ 0%]
1.60	5.00 GB	3.00 GB [ 60%]	0.00 [ 0%]

## Active Memory Expansion Recommendation:

```

-----
The recommended AME configuration for this workload is to configure the LPAR with a memory size of 5.00 GB and to configure a memory expansion factor of 1.60. This will result in a memory gain of 60%. With this configuration, the estimated CPU usage due to AME is approximately 0.00 physical processors, and the estimated overall peak CPU resource required for the LPAR is 1.74 physical processors.

```

NOTE: amepat's recommendations are based on the workload's utilization level



during the monitored period. If there is a change in the workload's utilization level or a change in workload itself, `amepat` should be run again.

The modeled Active Memory Expansion CPU usage reported by `amepat` is just an estimate. The actual CPU usage used for Active Memory Expansion may be lower or higher depending on the workload.

---

The `amepat` report consists of six different sections.

## Command Information Section

This section provides details about the arguments passed to the tool, such as time of invocation, total time the system was monitored and the number of samples collected.

## System Configuration Section

In this section, details relating to the systems configuration are shown. The details are listed in Table 6-1:

*Table 6-1 System Configuration details reported by `amepat`*

System Configuration Detail	Description
<b>Partition Name</b>	The node name from where the <code>amepat</code> command is invoked.
<b>Processor Implementation Mode</b>	The processor mode. The mode can be POWER4, POWER5, POWER6 and POWER7.
<b>Number of Logical CPUs</b>	The total number of logical CPUs configured and active in the partition.
<b>Processor Entitled Capacity</b>	Capacity Entitlement of the partition, represented in physical processor units.  <b>Note:</b> The physical processor units can be expressed in fractions of CPU, for example, 0.5 of a physical processor.
<b>Processor Max. Capacity</b>	Maximum Capacity this partition can have, represented in physical processors units.  <b>Note:</b> The physical processor unit can be expressed in fractions of CPU, for example, 0.5 physical processor.

<b>True Memory</b>	The true memory represents real physical or logical memory configured for this LPAR.
<b>SMT Threads</b>	Number of SMT threads configured in the partition. This can be 1, 2 or 4.
<b>Shared Processor Mode</b>	Indicates whether the Shared Processor Mode is configured for this partition. Possible values are:  <b>Disabled</b> - Shared Processor Mode is not configured.  <b>Enabled-Capped</b> - Shared Processor Mode is enabled and running in capped mode.  <b>Enabled-Uncapped</b> - Shared Processor Mode is enabled and running in uncapped mode.
<b>Active Memory Sharing</b>	Indicates whether Active Memory Sharing is <b>Enabled</b> or <b>Disabled</b> .
<b>Active Memory Expansion</b>	Indicates whether Active Memory Expansion is <b>Enabled</b> or <b>Disabled</b> .
<b>Target Expanded Memory Size</b>	Indicates the target expanded memory size in MegaBytes for the LPAR. The Target Expanded Memory Size is the True Memory Size multiplied by the Target Memory Expansion Factor.  <b>Note:</b> This is displayed only when AME is enabled.
<b>Target Memory Expansion Factor</b>	Indicates the target expansion factor configured for the LPAR.  <b>Note:</b> This is displayed only when AME is enabled.

## System Resource Statistics

In this section, details relating to the systems resource utilization, from a CPU and memory perspective, are displayed.

Table 6-2 System resource statistics reported by amepat

System Resource	Description
<b>CPU Util</b>	The Partition's CPU utilization in units of number of physical processors. The percentage of utilization against the Maximum Capacity is also reported.  <b>Note:</b> If AME is enabled, the CPU utilization due to memory compression/decompression is also included.
<b>Virtual Memory Size</b>	The Active Virtual Memory size in MegaBytes. The percentage against the True Memory size is also reported.
<b>True Memory In-Use</b>	This is the amount of the LPARs real physical (or logical) memory in MegaBytes. The percentage against the True Memory size is also reported.
<b>Pinned Memory</b>	This represents the pinned memory size in MegaBytes. The percentage against the True Memory size is also reported.
<b>File Cache Size</b>	This represents the non-computational file cache size in MegaBytes. The percentage against the True Memory size is also reported.
<b>Available Memory</b>	This represents the size of the memory available, in MegaBytes, for application usage. The percentage against the True Memory Size is also reported.

**Note:** If `amepat` is run with a duration and interval then Average, Minimum and Maximum utilization metrics are displayed.

### Active Memory Expansion Statistics

If AME is enabled, then AME usage statistics are displayed in this section. Table 6-3 describes the various statistics that are reported.

Table 6-3 AME statistics reported using amepat

Statistic	Description
-----------	-------------

<b>AME CPU Usage</b>	The CPU utilization for AME activity in units of physical processors. It indicates the amount of processing capacity used for memory compression activity. The percentage of utilization against the Maximum Capacity is also reported.
<b>Compressed Memory</b>	The total amount of virtual memory that is compressed. This is measured in MegaBytes. The percentage against the Target Expanded Memory Size is also reported.
<b>Compression Ratio</b>	This represents how well the data is compressed in memory. A higher compression ratio indicates that the data compresses to a smaller size. For example, if 4 Kilobytes of data can be compressed down to 1 Kilobyte, then the compression ration is 4.0.
<b>Deficit Memory Size</b>	The size of the expanded memory, in MegaBytes, deficit for the LPAR. This is only displayed if the LPAR has a memory deficit. The percentage against the Target Expanded Memory Size is also reported.

**Note:** The AME statistics section is only displayed when the tool is invoked on a AME enabled machine. It also displays the Average, Minimum and Maximum values when run with a duration and interval.

## Active Memory Expansion Modeled Statistics

This section provide details for the modeled statistics for AME. Table 6-4 describes the information relating to modeled statistics.

Table 6-4 AME modeled statistics

<b>Modeled Expanded Memory Size</b>	Represents the expanded memory size that is used to produce the modeled statistics.
<b>Average Compression Ratio</b>	Represents the average compression ratio of the in-memory data of the workload. This compression ratio is used to produce the modeled statistics.
<b>Modeled Expansion Factor</b>	Represents the modeled target memory expansion factor.

<b>Modeled True Memory Size</b>	Represents the modeled true memory size (real physical or logical memory).
<b>Modeled Memory Gain</b>	Represents the amount of memory the partition can gain by enabling AME for the reported modeled expansion factor.
<b>AME CPU Usage Estimate</b>	<p>Represents an estimate of the CPU that would be used for memory compression activity. The CPU usage is reported in units of physical processors. The percentage of utilization against the Maximum Capacity is also reported.</p> <p><b>Note:</b> This is an estimate and should only be used as a guide. The actual usage can be higher or lower depending on the workload.</p>

## Recommendation

This section provides information relating to optimal AME configurations and the benefits they may provide to the current running workload. These recommendations are based on the behavior of the system during the monitoring period. They can be used for guidance when choosing an optimal AME configuration for the system. Actual statistics can vary based on the real time behavior of the workload. AME statistics and recommendations are used for workload planning.

**Note:** Only one instance of the **amepat** command is allowed to run, in *Workload Planning* mode, at a time.

If you attempt to run two instances of the tool in this mode, the following message will be displayed:

```
amepat: Only one instance of amepat is allowed to run at a time.
```

The tool can also be invoked using the **smit** fast path, **smit amepat**.

The command is restricted in a WPAR environment. If you attempt to run the tool from a WPAR an error message is displayed, as shown in Example 6-4.

*Example 6-4 Running amepat within a WPAR*

---

```
# amepat
amepat: amepat cannot be run inside a WPAR
```

---

The optional **amepat** command line flags and their descriptions are listed in Table 6-5

Table 6-5 *Optional command line flags to amepat*

Flag	Description
<b>-a</b>	<p>Specifies to auto-tune the expanded memory size for AME Modeled Statistics. When this option is selected, the Modeled Expanded Memory Size is estimated based on the current memory usage of the workload (excludes the available memory size).</p> <p><b>Note:</b> -a -t are mutually exclusive.</p>
<b>-c</b> <i>max_ame_cpuusage%</i>	<p>Specifies the maximum AME CPU usage in terms of percentage to be used for producing the modeled statistics and recommendations.</p> <p><b>Note:</b> The default maximum used is 15%. The -C and -c option cannot be specified together. The -c and -e options are mutually exclusive.</p>
<b>-C</b> <i>max_ame_cpuusage%</i>	<p>Specifies the maximum AME CPU usage in terms of number of physical processors to be used for producing the modeled statistics and recommendations.</p> <p><b>Note:</b> The -C and -c option cannot be specified together. The -C and -e options are mutually exclusive.</p>

Flag	Description
<p><b>-e</b> <i>startexpfactor:stopexpfactor:incexpfactor</i></p>	<p>Specifies the range of expansion factors to be reported in the AME Modeled Statistics section.</p> <p><b>Startexpfactor</b> Starting expansion factor. This field is mandatory if -e is used.</p> <p><b>Stopexpfactor</b> Stop expansion factor. If not specified then the modeled statistics is generated for the start expansion factor alone.</p> <p><b>incexpfactor</b> Incremental expansion factor. Allowed range is 0.01-1.0. Default is 0.5. Stop expansion factor needs to be specified in order to specify the incremental expansion factor.</p> <p><b>Note:</b> The -e option cannot be combined with -C or -c options.</p>
<p><b>-m</b> <i>min_mem_gain</i></p>	<p>Specifies the Minimum Memory Gain. This value is specified in Megabytes. This value is used in determining the various possible expansion factors reported in the Modeled Statistics and also influences the produced recommendations.</p>
<p><b>-n</b> <i>num_entries</i></p>	<p>Specifies the number of entries that need to be displayed in the Modeled Statistics.</p> <p><b>Note:</b> When the -e option is used with <b>incexpfactor</b> then the -n value is ignored.</p>
<p><b>-N</b></p>	<p>Disable AME modeling (Workload Planning Capability).</p>
<p><b>-P</b> <i>recfile</i></p>	<p>Process the specified recording file and generate a report.</p>

Flag	Description
<b>-R</b> <i>recfile</i>	Record the active memory expansion data in the specified recording file. The recorded data can be post processed later using the -P option.  <b>Note:</b> Only the -N option can be combined with -R.
<b>-t</b> <i>tgt_expmem_size</i>	Specifies the Modeled Target Expanded Memory Size. This makes the tool to use the user specified size for modeling instead of the calculated one. Note: The -t and -a options are mutually exclusive.
<b>-u</b> <i>minuncompressdpoolsize</i>	Specifies the minimum uncompressed pool size in Megabytes. This value over-rides the tool calculated value for producing Modeled Statistics.  <b>Note:</b> This flag can be used only when AME is disabled.
<b>-v</b>	Enables Verbose Logging. When specified a verbose log file is generated, named as amepat_yyyymmddhmm.log, where yyyymmddhmm represents the time of invocation.  <b>Note:</b> The verbose log also contains detailed information on various samples collected and hence the file will be larger than the output generated by the tool.
<b>Duration</b>	Duration represents the amount of total time the tool required to monitor the system before generating any reports.  <b>Note:</b> When duration is specified, interval/samples cannot be specified. The interval and samples will be determined by the tool automatically. The actual monitoring time can be higher than the duration specified based on the memory usage and access patterns of the workload.



Flag	Description
<b>Interval &lt;Samples&gt;</b>	<p>Interval represents the amount of sampling time, Samples represents the number of samples need to be collected.</p> <p><b>Note:</b> When interval samples are specified, duration is calculated automatically as (interval x Samples). The actual monitoring time can be higher than the duration specified based on the memory usage and access patterns of the workload.</p>

To display the AME monitoring report, run the **amepat** command without any flags or options, as shown in Example 6-5.

*Example 6-5 Displaying the amepat monitoring report*

```
# amepat
```

```
Command Invoked           : amepat

Date/Time of invocation   : Mon Aug 30 17:22:00 EDT 2010
Total Monitored time     : NA
Total Samples Collected  : NA
```

System Configuration:

```
-----
Partition Name           : 75021p01
Processor Implementation Mode : POWER7
Number Of Logical CPUs   : 16
Processor Entitled Capacity : 1.00
Processor Max. Capacity  : 4.00
True Memory              : 8.00 GB
SMT Threads              : 4
Shared Processor Mode    : Enabled-Uncapped
Active Memory Sharing    : Disabled
Active Memory Expansion  : Enabled
Target Expanded Memory Size : 16.00 GB
Target Memory Expansion factor : 2.00
```

```
System Resource Statistics:           Current
-----
CPU Util (Phys. Processors)          0.10 [ 2%]
```

Virtual Memory Size (MB)	1697 [ 10%]
True Memory In-Use (MB)	1620 [ 20%]
Pinned Memory (MB)	1400 [ 17%]
File Cache Size (MB)	30 [ 0%]
Available Memory (MB)	14608 [ 89%]

AME Statistics:	Current
-----	-----
AME CPU Usage (Phy. Proc Units)	0.00 [ 0%]
Compressed Memory (MB)	203 [ 1%]
Compression Ratio	2.35
Deficit Memory Size (MB)	74 [ 0%]

In Example 6-6 the **amepat** command will monitor the workload on a system for a duration of 10 minutes with 5 minute sampling intervals and 2 samples:

*Example 6-6 Monitoring the workload on a system with amepat for 10 minutes*

```
# amepat 5 2
```

```
Command Invoked      : amepat 5 2
Date/Time of invocation : Mon Aug 30 17:26:20 EDT 2010
Total Monitored time  : 10 mins 48 secs
Total Samples Collected : 2
```

System Configuration:

```
-----
Partition Name       : 75021p01
Processor Implementation Mode : POWER7
Number Of Logical CPUs : 16
Processor Entitled Capacity : 1.00
Processor Max. Capacity : 4.00
True Memory          : 8.00 GB
SMT Threads          : 4
Shared Processor Mode : Enabled-Uncapped
Active Memory Sharing : Disabled
Active Memory Expansion : Enabled
Target Expanded Memory Size : 16.00 GB
Target Memory Expansion factor : 2.00
```

System Resource Statistics:	Average	Min	Max
-----	-----	-----	-----
CPU Util (Phys. Processors)	2.39 [ 60%]	1.94 [ 48%]	2.84 [ 71%]
Virtual Memory Size (MB)	1704 [ 10%]	1703 [ 10%]	1706 [ 10%]
True Memory In-Use (MB)	1589 [ 19%]	1589 [ 19%]	1590 [ 19%]
Pinned Memory (MB)	1411 [ 17%]	1405 [ 17%]	1418 [ 17%]
File Cache Size (MB)	10 [ 0%]	10 [ 0%]	11 [ 0%]
Available Memory (MB)	14057 [ 86%]	13994 [ 85%]	14121 [ 86%]

AME Statistics:	Average	Min	Max
-----	-----	-----	-----
AME CPU Usage (Phy. Proc Units)	0.11 [ 3%]	0.02 [ 1%]	0.21 [ 5%]
Compressed Memory (MB)	234 [ 1%]	230 [ 1%]	238 [ 1%]
Compression Ratio	2.25	2.25	2.26
Deficit Memory Size (MB)	701 [ 4%]	701 [ 4%]	702 [ 4%]

```
Active Memory Expansion Modeled Statistics :
```

```

-----
Modeled Expanded Memory Size : 16.00 GB
Achievable Compression ratio :2.25

```

Expansion Factor	Modeled True Memory Size	Modeled Memory Gain	CPU Usage Estimate
1.02	15.75 GB	256.00 MB [ 2%]	0.00 [ 0%]
1.17	13.75 GB	2.25 GB [ 16%]	0.00 [ 0%]
1.31	12.25 GB	3.75 GB [ 31%]	0.00 [ 0%]
1.46	11.00 GB	5.00 GB [ 45%]	0.75 [ 19%]
1.60	10.00 GB	6.00 GB [ 60%]	1.54 [ 39%]
1.73	9.25 GB	6.75 GB [ 73%]	2.14 [ 53%]
1.89	8.50 GB	7.50 GB [ 88%]	2.73 [ 68%]

Active Memory Expansion Recommendation:

WARNING: This LPAR currently has a memory deficit of 701 MB. A memory deficit is caused by a memory expansion factor that is too high for the current workload. It is recommended that you reconfigure the LPAR to eliminate this memory deficit. Reconfiguring the LPAR with one of the recommended configurations in the above table should eliminate this memory deficit.

The recommended AME configuration for this workload is to configure the LPAR with a memory size of 12.25 GB and to configure a memory expansion factor of 1.31. This will result in a memory gain of 31%. With this configuration, the estimated CPU usage due to AME is approximately 0.00 physical processors, and the estimated overall peak CPU resource required for the LPAR is 2.64 physical processors.

NOTE: amepat's recommendations are based on the workload's utilization level during the monitored period. If there is a change in the workload's utilization level or a change in workload itself, amepat should be run again.

The modeled Active Memory Expansion CPU usage reported by amepat is just an estimate. The actual CPU usage used for Active Memory Expansion may be lower or higher depending on the workload.

---

To cap AME CPU usage to 30%, when capturing Workload Planning data for 5 minutes, you would enter the command shown in Example 6-7.

*Example 6-7 Capping AME CPU usage to 30%*

```

# amepat -c 30 5

Command Invoked          : amepat -c 30 5
Date/Time of invocation  : Mon Aug 30 17:43:28 EDT 2010
Total Monitored time     : 6 mins 7 secs
Total Samples Collected  : 3

System Configuration:
-----
Partition Name           : 75021p01
Processor Implementation Mode : POWER7
Number Of Logical CPUs   : 16
Processor Entitled Capacity : 1.00
Processor Max. Capacity  : 4.00
True Memory              : 8.00 GB
SMT Threads              : 4
Shared Processor Mode    : Enabled-Uncapped
Active Memory Sharing    : Disabled
Active Memory Expansion  : Enabled
Target Expanded Memory Size : 16.00 GB

```

Target Memory Expansion factor : 2.00

System Resource Statistics:	Average	Min	Max
CPU Util (Phys. Processors)	0.02 [ 0%]	0.01 [ 0%]	0.02 [ 1%]
Virtual Memory Size (MB)	1780 [ 11%]	1780 [ 11%]	1781 [ 11%]
True Memory In-Use (MB)	1799 [ 22%]	1796 [ 22%]	1801 [ 22%]
Pinned Memory (MB)	1448 [ 18%]	1448 [ 18%]	1448 [ 18%]
File Cache Size (MB)	83 [ 1%]	83 [ 1%]	84 [ 1%]
Available Memory (MB)	14405 [ 88%]	14405 [ 88%]	14407 [ 88%]

AME Statistics:	Average	Min	Max
AME CPU Usage (Phy. Proc Units)	0.00 [ 0%]	0.00 [ 0%]	0.00 [ 0%]
Compressed Memory (MB)	198 [ 1%]	198 [ 1%]	199 [ 1%]
Compression Ratio	2.35	2.35	2.36
Deficit Memory Size (MB)	116 [ 1%]	116 [ 1%]	116 [ 1%]

Active Memory Expansion Modeled Statistics :

Modeled Expanded Memory Size : 16.00 GB  
Achievable Compression ratio :2.35

Expansion Factor	Modeled True Memory Size	Modeled Memory Gain	CPU Usage Estimate
1.02	15.75 GB	256.00 MB [ 2%]	0.00 [ 0%]
1.17	13.75 GB	2.25 GB [ 16%]	0.00 [ 0%]
1.34	12.00 GB	4.00 GB [ 33%]	0.00 [ 0%]
1.49	10.75 GB	5.25 GB [ 49%]	0.00 [ 0%]
1.65	9.75 GB	6.25 GB [ 64%]	0.00 [ 0%]
1.78	9.00 GB	7.00 GB [ 78%]	0.00 [ 0%]
1.94	8.25 GB	7.75 GB [ 94%]	0.00 [ 0%]

Active Memory Expansion Recommendation:

WARNING: This LPAR currently has a memory deficit of 116 MB. A memory deficit is caused by a memory expansion factor that is too high for the current workload. It is recommended that you reconfigure the LPAR to eliminate this memory deficit. Reconfiguring the LPAR with one of the recommended configurations in the above table should eliminate this memory deficit.

The recommended AME configuration for this workload is to configure the LPAR with a memory size of 8.25 GB and to configure a memory expansion factor of 1.94. This will result in a memory gain of 94%. With this configuration, the estimated CPU usage due to AME is approximately 0.00 physical processors, and the estimated overall peak CPU resource required for the LPAR is 0.02 physical processors.

NOTE: amepat's recommendations are based on the workload's utilization level during the monitored period. If there is a change in the workload's utilization level or a change in workload itself, amepat should be run again.

The modeled Active Memory Expansion CPU usage reported by amepat is just an estimate. The actual CPU usage used for Active Memory Expansion may be lower or higher depending on the workload.

To start modeling a memory gain of 1000MB for a duration of 5 minutes and generate a AME Workload Planning report, you would enter the command shown in Example 6-8 on page 215.

*Example 6-8 AME modeling memory gain of 100MB*

```
# amepat -m 1000 5
```

```
Command Invoked      : amepat -m 1000 5
Date/Time of invocation : Mon Aug 30 18:42:46 EDT 2010
Total Monitored time   : 6 mins 9 secs
Total Samples Collected : 3
```

## System Configuration:

```
-----
Partition Name       : 75021p01
Processor Implementation Mode : POWER7
Number Of Logical CPUs : 16
Processor Entitled Capacity : 1.00
Processor Max. Capacity : 4.00
True Memory          : 8.00 GB
SMT Threads          : 4
Shared Processor Mode : Enabled-Uncapped
Active Memory Sharing : Disabled
Active Memory Expansion : Enabled
Target Expanded Memory Size : 16.00 GB
Target Memory Expansion factor : 2.00
```

System Resource Statistics:	Average	Min	Max
-----	-----	-----	-----
CPU Util (Phys. Processors)	0.02 [ 0%]	0.01 [ 0%]	0.02 [ 1%]
Virtual Memory Size (MB)	1659 [ 10%]	1658 [ 10%]	1661 [ 10%]
True Memory In-Use (MB)	1862 [ 23%]	1861 [ 23%]	1864 [ 23%]
Pinned Memory (MB)	1362 [ 17%]	1362 [ 17%]	1363 [ 17%]
File Cache Size (MB)	163 [ 2%]	163 [ 2%]	163 [ 2%]
Available Memory (MB)	14538 [ 89%]	14536 [ 89%]	14539 [ 89%]

AME Statistics:	Average	Min	Max
-----	-----	-----	-----
AME CPU Usage (Phy. Proc Units)	0.00 [ 0%]	0.00 [ 0%]	0.00 [ 0%]
Compressed Memory (MB)	0 [ 0%]	0 [ 0%]	0 [ 0%]
Compression Ratio	N/A		

## Active Memory Expansion Modeled Statistics :

```
-----
Modeled Expanded Memory Size : 16.00 GB
Achievable Compression ratio :0.00
```

## Active Memory Expansion Recommendation:

```
-----
The amount of compressible memory for this workload is small. Only 1.81% of the current memory size is compressible. This tool analyzes compressible memory in order to make recommendations. Due to the small amount of compressible memory, this tool cannot make a recommendation for the current workload.
```

```
This small amount of compressible memory is likely due to the large amount of free memory. 38.63% of memory is free (unused). This may indicate the load was very light when this tool was run. If so, please increase the load and run this tool again.
```

To start modeling a minimum uncompressed pool size of 2000MB for a duration of 5 minutes and generate a AME Workload Planning report, you would enter the command shown in Example 6-9

**Note:** This command can only be run on a system with AME disabled. If you attempt to run it on an AME enabled system you will see the following message: amepat: -u option is not allowed when AME is ON.

#### Example 6-9 Modeling a minimum uncompressed pool size of 2000MB

```
# amepat -u 2000 5
```

Command Invoked : amepat -u 2000 5

Date/Time of invocation : Mon Aug 30 18:51:46 EDT 2010  
 Total Monitored time : 6 mins 8 secs  
 Total Samples Collected : 3

System Configuration:  
 -----  
 Partition Name : 75021p02  
 Processor Implementation Mode : POWER7  
 Number Of Logical CPUs : 16  
 Processor Entitled Capacity : 1.00  
 Processor Max. Capacity : 4.00  
 True Memory : 8.00 GB  
 SMT Threads : 4  
 Shared Processor Mode : Enabled-Uncapped  
 Active Memory Sharing : Disabled  
 Active Memory Expansion : Disabled

System Resource Statistics:

	Average	Min	Max
CPU Util (Phys. Processors)	0.01 [ 0%]	0.01 [ 0%]	0.02 [ 0%]
Virtual Memory Size (MB)	1756 [ 21%]	1756 [ 21%]	1756 [ 21%]
True Memory In-Use (MB)	1949 [ 24%]	1949 [ 24%]	1949 [ 24%]
Pinned Memory (MB)	1446 [ 18%]	1446 [ 18%]	1446 [ 18%]
File Cache Size (MB)	178 [ 2%]	178 [ 2%]	178 [ 2%]
Available Memory (MB)	6227 [ 76%]	6227 [ 76%]	6227 [ 76%]

Active Memory Expansion Modeled Statistics :

Modeled Expanded Memory Size : 8.00 GB  
 Achievable Compression ratio :2.13

Expansion Factor	Modeled True Memory Size	Modeled Memory Gain	CPU Usage Estimate
1.00	8.00 GB	0.00 KB [ 0%]	0.00 [ 0%]
1.07	7.50 GB	512.00 MB [ 7%]	0.00 [ 0%]
1.15	7.00 GB	1.00 GB [ 14%]	0.00 [ 0%]
1.19	6.75 GB	1.25 GB [ 19%]	0.00 [ 0%]
1.28	6.25 GB	1.75 GB [ 28%]	0.00 [ 0%]
1.34	6.00 GB	2.00 GB [ 33%]	0.00 [ 0%]
1.40	5.75 GB	2.25 GB [ 39%]	0.00 [ 0%]

Active Memory Expansion Recommendation:  
 -----  
 The recommended AME configuration for this workload is to configure the LPAR with a memory size of 5.75 GB and to configure a memory expansion factor of 1.40. This will result in a memory gain of 39%. With this configuration, the estimated CPU usage due to AME is approximately 0.00

physical processors, and the estimated overall peak CPU resource required for the LPAR is 0.02 physical processors.

NOTE: amepat's recommendations are based on the workload's utilization level during the monitored period. If there is a change in the workload's utilization level or a change in workload itself, amepat should be run again.

The modeled Active Memory Expansion CPU usage reported by amepat is just an estimate. The actual CPU usage used for Active Memory Expansion may be lower or higher depending on the workload.

---

To use the **amepat** recording mode to generate a recording file and report, you would enter the command shown in Example 6-10 (this will start recording for a duration of 60 minutes).

**Note:** This will invoke the tool as a background process.

#### Example 6-10 Starting amepat in recording mode

---

```
# amepat -R /tmp/myrecord_amepat 60
Continuing Recording through background process...

# ps -ef | grep amepat
  root   5898374 12976300   0 11:14:36 pts/0   0:00 grep amepat
  root   20119654      1   0 10:42:14 pts/0   0:21 amepat -R /tmp/myrecord_amepat 60

# ls -ltr /tmp/myrecord_amepat
total 208
-rw-r--r--  1 root   system    22706 Aug 31 11:13 myrecord_amepat
```

---

In Example 6-11 the **amepat** command will generate a report, for workload planning purposes, using a previously generated recording file:

#### Example 6-11 Generating an amepat report using an existing recording file

---

```
# amepat -P /tmp/myrecord_amepat

Command Invoked           : amepat -P /tmp/myrecord_amepat
Date/Time of invocation   : Mon Aug 30 18:59:25 EDT 2010
Total Monitored time     : 1 hrs 3 mins 23 secs
Total Samples Collected  : 9

System Configuration:
-----
Partition Name           : 75021p01
Processor Implementation Mode : POWER7
Number Of Logical CPUs   : 16
Processor Entitled Capacity : 1.00
Processor Max. Capacity  : 4.00
True Memory              : 8.00 GB
SMT Threads              : 4
Shared Processor Mode    : Enabled-Uncapped
Active Memory Sharing    : Disabled
Active Memory Expansion  : Enabled
Target Expanded Memory Size : 16.00 GB
Target Memory Expansion factor : 2.00
```

System Resource Statistics:	Average	Min	Max
CPU Util (Phys. Processors)	0.01 [ 0%]	0.01 [ 0%]	0.01 [ 0%]
Virtual Memory Size (MB)	1653 [ 10%]	1653 [ 10%]	1656 [ 10%]
True Memory In-Use (MB)	1856 [ 23%]	1856 [ 23%]	1859 [ 23%]
Pinned Memory (MB)	1362 [ 17%]	1362 [ 17%]	1362 [ 17%]
File Cache Size (MB)	163 [ 2%]	163 [ 2%]	163 [ 2%]
Available Memory (MB)	14542 [ 89%]	14541 [ 89%]	14543 [ 89%]

AME Statistics:	Average	Min	Max
AME CPU Usage (Phy. Proc Units)	0.00 [ 0%]	0.00 [ 0%]	0.00 [ 0%]
Compressed Memory (MB)	0 [ 0%]	0 [ 0%]	0 [ 0%]
Compression Ratio	N/A		

Active Memory Expansion Modeled Statistics :

Modeled Expanded Memory Size : 16.00 GB  
Achievable Compression ratio :0.00

Active Memory Expansion Recommendation:

The amount of compressible memory for this workload is small. Only 1.78% of the current memory size is compressible. This tool analyzes compressible memory in order to make recommendations. Due to the small amount of compressible memory, this tool cannot make a recommendation for the current workload.

This small amount of compressible memory is likely due to the large amount of free memory. 38.66% of memory is free (unused). This may indicate the load was very light when this tool was run. If so, please increase the load and run this tool again.

Example 6-12 generates a report for workload planning, with the modeled memory expansion factors ranging between 2 to 4 with a 0.5 delta factor.

#### *Example 6-12 Modeled expansion factor report from a recorded file*

```
# amepat -e 2.0:4.0:0.5 -P /tmp/myrecord_amepat
```

```
Command Invoked          : amepat -e 2.0:4.0:0.5 -P /tmp/myrecord_amepat
Date/Time of invocation  : Mon Aug 30 18:59:25 EDT 2010
Total Monitored time     : 1 hrs 3 mins 23 secs
Total Samples Collected : 9
```

System Configuration:

```
-----
Partition Name           : 75021p01
Processor Implementation Mode : POWER7
Number Of Logical CPUs   : 16
Processor Entitled Capacity : 1.00
Processor Max. Capacity  : 4.00
True Memory              : 8.00 GB
SMT Threads              : 4
Shared Processor Mode    : Enabled-Uncapped
Active Memory Sharing    : Disabled
Active Memory Expansion  : Enabled
Target Expanded Memory Size : 16.00 GB
Target Memory Expansion factor : 2.00
```

System Resource Statistics:	Average	Min	Max
CPU Util (Phys. Processors)	0.01 [ 0%]	0.01 [ 0%]	0.01 [ 0%]



Virtual Memory Size (MB)	1653 [ 10%]	1653 [ 10%]	1656 [ 10%]
True Memory In-Use (MB)	1856 [ 23%]	1856 [ 23%]	1859 [ 23%]
Pinned Memory (MB)	1362 [ 17%]	1362 [ 17%]	1362 [ 17%]
File Cache Size (MB)	163 [ 2%]	163 [ 2%]	163 [ 2%]
Available Memory (MB)	14542 [ 89%]	14541 [ 89%]	14543 [ 89%]

AME Statistics:	Average	Min	Max
AME CPU Usage (Phy. Proc Units)	0.00 [ 0%]	0.00 [ 0%]	0.00 [ 0%]
Compressed Memory (MB)	0 [ 0%]	0 [ 0%]	0 [ 0%]
Compression Ratio	N/A		

Active Memory Expansion Modeled Statistics :

Modeled Expanded Memory Size : 16.00 GB  
Achievable Compression ratio :0.00

Active Memory Expansion Recommendation:

The amount of compressible memory for this workload is small. Only 1.78% of the current memory size is compressible. This tool analyzes compressible memory in order to make recommendations. Due to the small amount of compressible memory, this tool cannot make a recommendation for the current workload.

This small amount of compressible memory is likely due to the large amount of free memory. 38.66% of memory is free (unused). This may indicate the load was very light when this tool was run. If so, please increase the load and run this tool again.

To generate an AME monitoring only report from a previously recorded file, you would enter the command shown in Example 6-13.

### Example 6-13 AME monitoring report from a recorded file

```
# amepat -N -P /tmp/myrecord_amepat
WARNING: Running in no modeling mode.
```

```
Command Invoked      : amepat -N -P /tmp/myrecord_amepat
Date/Time of invocation : Mon Aug 30 18:59:25 EDT 2010
Total Monitored time   : 1 hrs 3 mins 23 secs
Total Samples Collected : 9
```

System Configuration:

```
-----
Partition Name       : 75021p01
Processor Implementation Mode : POWER7
Number Of Logical CPUs : 16
Processor Entitled Capacity : 1.00
Processor Max. Capacity : 4.00
True Memory          : 8.00 GB
SMT Threads          : 4
Shared Processor Mode : Enabled-Uncapped
Active Memory Sharing : Disabled
Active Memory Expansion : Enabled
Target Expanded Memory Size : 16.00 GB
Target Memory Expansion factor : 2.00
```

System Resource Statistics:	Average	Min	Max
CPU Util (Phys. Processors)	0.01 [ 0%]	0.01 [ 0%]	0.01 [ 0%]
Virtual Memory Size (MB)	1653 [ 10%]	1653 [ 10%]	1656 [ 10%]

True Memory In-Use (MB)	1856 [ 23%]	1856 [ 23%]	1859 [ 23%]
Pinned Memory (MB)	1362 [ 17%]	1362 [ 17%]	1362 [ 17%]
File Cache Size (MB)	163 [ 2%]	163 [ 2%]	163 [ 2%]
Available Memory (MB)	14542 [ 89%]	14541 [ 89%]	14543 [ 89%]
AME Statistics:	Average	Min	Max
-----	-----	-----	-----
AME CPU Usage (Phy. Proc Units)	0.00 [ 0%]	0.00 [ 0%]	0.00 [ 0%]
Compressed Memory (MB)	0 [ 0%]	0 [ 0%]	0 [ 0%]
Compression Ratio	N/A		

---

Example 6-14 will disable the Workload Planning capability and only monitor system utilization for 5 minutes.

*Example 6-14 Disable workload planning and only monitor system utilization*

---

```
# amepat -N 5
WARNING: Running in no modeling mode.

Command Invoked           : amepat -N 5

Date/Time of invocation   : Tue Aug 31 11:20:41 EDT 2010
Total Monitored time     : 6 mins 0 secs
Total Samples Collected  : 3

System Configuration:
-----
Partition Name           : 75021p01
Processor Implementation Mode : POWER7
Number Of Logical CPUs   : 16
Processor Entitled Capacity : 1.00
Processor Max. Capacity  : 4.00
True Memory              : 8.00 GB
SMT Threads              : 4
Shared Processor Mode    : Enabled-Uncapped
Active Memory Sharing    : Disabled
Active Memory Expansion  : Enabled
Target Expanded Memory Size : 16.00 GB
Target Memory Expansion factor : 2.00

System Resource Statistics:
-----
Average           Min           Max
-----
CPU Util (Phys. Processors)  0.01 [ 0%]  0.01 [ 0%]  0.01 [ 0%]
Virtual Memory Size (MB)     1759 [ 11%] 1759 [ 11%] 1759 [ 11%]
True Memory In-Use (MB)      1656 [ 20%] 1654 [ 20%] 1657 [ 20%]
Pinned Memory (MB)          1461 [ 18%] 1461 [ 18%] 1461 [ 18%]
File Cache Size (MB)         9 [ 0%]     9 [ 0%]    10 [ 0%]
Available Memory (MB)       14092 [ 86%] 14092 [ 86%] 14094 [ 86%]

AME Statistics:
-----
Average           Min           Max
-----
AME CPU Usage (Phy. Proc Units)  0.00 [ 0%]  0.00 [ 0%]  0.00 [ 0%]
Compressed Memory (MB)           220 [ 1%]  220 [ 1%]  221 [ 1%]
Compression Ratio                 2.27      2.27      2.28
Deficit Memory Size (MB)          550 [ 3%]  550 [ 3%]  550 [ 3%]
```

---

## 6.1.2 Enhanced AIX performance monitoring tools for AME

Several AIX performance tools can be used to monitor AME statistics and gather information relating to AME. The following AIX tools have been enhanced to support AME monitoring and reporting:

- ▶ `vmstat`
- ▶ `lparstat`
- ▶ `topas`
- ▶ `topas_nmon`
- ▶ `svmon`

The additional options for each tool are summarized in Table 6-6.

*Table 6-6 AIX performance tool enhancements for AME*

Tool	Option	Description
<code>vmstat</code>	<code>-c</code>	Provides compression, decompression, and deficit statistics.
<code>lparstat</code>	<code>-c</code>	Provides an indication of the CPU utilization for AME compression and decompression activity. Also provides memory deficit information.
<code>svmon</code>	<code>-O summary=ame</code>	Provides a summary view of memory usage broken down into compressed and uncompressed pages.
<code>topas</code>		The default <code>topas</code> screen displays the memory compression statistics when it is run in the AME environment.

The `vmstat` command can be used with the `-c` flag to display AME statistics, as shown in Example 6-15.

*Example 6-15 Using `vmstat` to display AME statistics*

```
# vmstat -wc 1 5
System configuration: lcpu=16 mem=16384MB tmem=8192MB ent=1.00 mmode=dedicated-E
kthr          memory          page          faults          cpu
-----
```

r	b	avm	fre	csz	cfr	dxm	ci	co	pi	po	in	sy	cs	us	sy	id	wa	pc	ec
51	0	1287384	2854257	35650	13550	61379	0	0	0	0	3	470	1712	99	0	0	0	3.99	399.4
53	0	1287384	2854264	35650	13567	61379	30	0	0	0	2	45	1721	99	0	0	0	3.99	399.2
51	0	1287384	2854264	35650	13567	61379	0	0	0	0	1	40	1712	99	0	0	0	3.99	399.2
0	0	1287384	2854264	35650	13567	61379	0	0	0	0	3	45	1713	99	0	0	0	3.99	399.2
51	0	1287384	2854264	35650	13567	61379	0	0	0	0	2	38	1716	99	0	0	0	3.99	399.2

In the output from Example 6-15 on page 221, the following memory compression statistics are provided:

- ▶ Expanded memory size (mem) of the LPAR is 16384 MB.
- ▶ True memory size (tmem) is 8192 MB.
- ▶ The memory mode (mmode) of the LPAR is AME enabled, Dedicated-Expanded.
- ▶ Compressed Pool size (csz) is 35650 4K pages.
- ▶ Amount of free memory (cfr) in the compressed pool is 13567 4K pages.
- ▶ Size of the expanded memory deficit (dxm) is 61379 4K pages.
- ▶ Number of compression operations or page-outs to the compressed pool per second (co) is 0.
- ▶ Number of decompression operations or page-ins from the compressed pool per second (ci) is 0.

The **lparstat** command can be used, with the **-c** flag, to display AME statistics, as shown in Example 6-16.

*Example 6-16 Using lparstat to display AME statistics*

```
# lparstat -c 1 5
```

```
System configuration: type=Shared mode=Uncapped mmode=Ded-E smt=4 lcpu=16
mem=16384MB tmem=8192MB psize=16 ent=1.00
```

%user	%sys	%wait	%idle	physc	%entc	lbusy	vcsw	phint	%xcpu	xphysc	dxm
91.9	8.1	0.0	0.0	3.99	399.3	100.0	1600	1	9.7	0.3861	2417
89.1	10.9	0.0	0.0	3.99	398.7	100.0	1585	0	15.0	0.5965	2418
85.5	14.5	0.0	0.0	3.99	399.2	100.0	1599	4	16.9	0.6736	2418
87.6	12.4	0.0	0.0	3.99	399.2	100.0	1600	16	16.7	0.6664	2418
82.7	17.3	0.0	0.0	3.99	399.4	100.0	1615	12	17.3	0.6908	742

In the output above, the following memory compression statistics are provided:

- ▶ Memory mode (mmode) of the LPAR is AME enabled, Dedicated-Expanded.
- ▶ Expanded memory size (mem) of the LPAR is 16384 MB.
- ▶ True memory size (tmem) of the LPAR is 8192 MB.
- ▶ Percentage of CPU utilized for AME activity (%xcpu) is 17.3.

- Size of expanded memory deficit (dxm) in megabytes is 742.

Example 6-17 displays output from the `lparstat -i` showing configuration information relating to LPAR memory mode and AME settings.

*Example 6-17 Using lparstat to view AME configuration details*

---

```
# lparstat -i | grep -i memory | grep -i ex
Memory Mode                : Dedicated-Expanded
Target Memory Expansion Factor : 2.00
Target Memory Expansion Size  : 16384 MB
```

---

The LPARs memory mode is Dedicated-Expanded, the target memory expansion factor is 2.0 and the target memory expansion size is 16384 MB.

The main panel of the `topas` command has been modified to display AME compression statistics. The data is displayed under an additional sub-section called **AME**, as shown in Example 6-18.

*Example 6-18 Additional topas sub-section for AME.*

---

```
Topas Monitor for host:750_2_LP01
Tue Aug 31 11:04:22 2010 Interval:FROZEN

CPU      User% Kern% Wait% Idle%  Physc  Entc%
Total    0.0   0.7   0.0  99.3   0.01  1.26

Network  BPS   I-Pkts  O-Pkts  B-In  B-Out
Total    462.0  0.50   1.00   46.00 416.0

Disk     Busy%   BPS    TPS  B-Read  B-Writ
Total    0.0     0     0     0       0

FileSystem      BPS    TPS  B-Read  B-Writ
Total           336.0  0.50  336.0  0

Name      PID  CPU%  PgSp  Owner
vmmd      393228 0.3  188K  root
xmgc      851994 0.2  60.0K root
topas     18939976 0.1  2.35M root
getty     6160394 0.0  580K  root
java     6095084 0.0  48.8M pconsole
gil       1966140 0.0  124K  root
sshd     6619376 0.0  1.18M root
clcomd   5767348 0.0  1.75M root
java     5177386 0.0  73.7M root
rpc.lock 5243052 0.0  204K  root
rmcd     5832906 0.0  6.54M root
netm     1900602 0.0  60.0K root
cmemd    655380 0.0  180K  root
lrud     262152 0.0  92.0K root
topasrec 5701812 0.0  1.07M root
amepat   20119654 0.0  328K  root
syncd   2949304 0.0  604K  root
random   3670206 0.0  60.0K root
j2pg     2424912 0.0  1.17M root
lvmbb    2490464 0.0  60.0K root

EVENTS/QUEUES  FILE/TTY
Cswitch        210  Readch    361
Syscall        120  Writch    697
Reads          0  Rawin     0
Writes         0  Ttyout    335
Forks          0  Igets     0
Execs          0  Namei     1
Runqueue      0  Dirblk    0
Waitqueue     0.0

MEMORY
PAGING Real,MB 16384
Faults 0 % Comp 14
Steals 0 % Noncomp 0
PgspIn 0 % Client 0
PgspOut 0
PageIn 0 PAGING SPACE
PageOut 0 Size,MB 512
Sios 0 % Used 3
% Free 97

AME
TMEM,MB 8192 WPAR Activ 0
CMEM,MB 139.82 WPAR Total 1
EF[T/A] 2.00/1.04 Press: "h"-help
CI: 0.0 CO: 0.0 "q"-quit
```

---

In Example 6-18 on page 223, the following memory compression statistics are provided from the **topas** command:

- ▶ True memory size (TMEM,MB) of the LPAR is 8192 MB.
- ▶ Compressed pool size (CMEM,MB) is 139.82 MB.
- ▶ EF[T/A] - The Target Expansion Factor is 2.00 and the Achieved Expansion Factor is 1.04.
- ▶ Rate of compression (co) and decompressions (ci) per second are 0.0 and 0.0 pages respectively.

The **topas** command CEC view has been enhanced to report AME status across all of the LPARs on a server. The memory mode for an LPAR is displayed in the CEC view. The possible memory modes shown by the **topas -C** command are shown in Table 6-7.

Table 6-7 *topas -C* memory mode values for an LPAR

Value	Description
M	In shared memory mode (shared LPARs only) and AME is disabled
-	Not in shared memory mode and AME is disabled
E	In shared memory mode and AME is enabled
e	Not in shared memory mode and AME is enabled.

Example 6-19 provides output from the **topas -C** command for a system with six AME enabled LPARs:

Example 6-19 *topas* CEC view with AME enabled LPARs

Topas CEC Monitor		Interval: 10		Thu Sep 16 10:19:22 2010											
Partitions Memory (GB)		Processors													
Shr: 6	Mon:46.0	InUse:18.0	Shr:4.3	PSz: 16	Don: 0.0	Shr_PhysB 0.65									
Ded: 0	Avl: -		Ded: 0	APP: 15.3	St1: 0.0	Ded_PhysB 0.00									
Host	OS	Mod	Mem	InU	Lp	Us	Sy	Wa	Id	PhysB	Vcsw	Ent	%EntC	Phi	pmem
-----shared-----															
75021p03	A71	Ued	8.0	3.2	16	8	27	0	64	0.57	0	1.00	56.5	0	-
75021p01	A71	Ued	16	8.0	16	0	1	0	98	0.02	286	1.00	2.4	0	-
75021p06	A71	Ued	2.0	2.0	8	0	5	0	94	0.02	336	0.20	10.6	1	-
75021p05	A71	Ued	4.0	1.0	4	0	7	0	92	0.02	0	0.10	16.9	0	-
75021p04	A71	Ued	8.0	2.2	16	0	0	0	99	0.02	0	1.00	1.5	0	-
75021p02	A71	Ued	8.0	1.7	16	0	0	0	99	0.01	276	1.00	1.2	0	-

The second character under the mode column (Mod) for each LPAR is e, which indicates Active Memory Sharing is disabled and AME is enabled.

The `topas_nmon` command supports AME statistic reporting in the `nmon` recording file. The MEM tag will report the size of the compressed pool in MB, the size of true memory in MB, the expanded memory size in MB and the size of the uncompressed pool in MB. The MEMNEW tag will show the compressed pool percentage. The PAGE tag will display the compressed pool page-ins and the compressed pool page-outs.

The `svmon` command can provide a detailed view of AME usage on an LPAR, as shown in Example 6-20.

*Example 6-20 AME statistics displayed using the svmon command*

---

```
# svmon -G -0 summary=ame,pgsz=on,unit=MB
Unit: MB
-----
```

	size	inuse	free	pin	virtual	available	mmode
memory	16384.00	1725.00	14114.61	1453.91	1752.57	14107.11	Ded-E
ucomprsd	-	1497.54	-				
comprsd	-	227.46	-				
pg space	512.00	14.4					

	work	pers	clnt	other
pin	937.25	0	0	516.66
in use	1715.52	0	9.47	
ucomprsd	1488.07			
comprsd	227.46			

---

```
True Memory: 8192.00
-----
```

	CurSz	%Cur	TgtSz	%Tgt	MaxSz	%Max	CRatio
ucomprsd	8052.18	98.29	1531.84	18.70	-	-	-
comprsd	139.82	1.71	6660.16	81.30	6213.15	75.84	2.28

	txf	cxp	dxp	dxm
AME	2.00	1.93	0.07	549.83

---

The following memory compressions statistics are displayed from the `svmon` command in Example 6-20:

- ▶ Memory mode (mmode) of the LPAR is AME enabled, Dedicated-Expanded.
- ▶ Out of a total of 1725.00 MB in use, uncompressed pages (ucomprsd) constitute 1497.54 MB and compressed pages (comprsd) constitute the remaining 227.46 MB.
- ▶ Out of a total of 1715.52 MB of working pages in use, uncompressed pages (ucomprsd) constitute 1488.07 MB and compressed pages (comprsd) constitute 227.46 MB.
- ▶ Expanded memory size (memory) of the LPAR is 16384 MB.
- ▶ True memory size (True Memory) of the LPAR is 8192 MB.

- ▶ Current size of the uncompressed pool (ucomprsd CurSz) is 8052.18 MB (98.29% of the total true memory size of the LPAR, %Cur).
- ▶ Current size of the compressed pool (comprsd CurSz) is 139.82 MB (1.71% of the total true memory size of the LPAR, %Cur).
- ▶ The target size of the compressed pool (comprsd TgtSz) required to achieve the target memory expansion factor (txf) of 2.00 is 1531.84 MB (18.70% of the total true memory size of the LPAR, %Tgt).
- ▶ The size of the uncompressed pool (ucomprsd TgtSz) in that case becomes 6660.16 MB (81.30% of the total true memory size of the LPAR, %Tgt).
- ▶ The maximum size of the compressed pool (comprsd MaxSz) is 6213.15 MB (75.84% of the total true memory size of the LPAR, %Max).
- ▶ The current compression ratio (CRatio) is 2.28 and the current expansion factor (cxf) is 1.93.
- ▶ The amount of expanded memory deficit (dxm) is 549.83 MB and the deficit expansion factor (dxf) is 0.07

The `-O summary=longname` option provides a summary of memory compression details, from the `svmon` command, as shown in Example 6-21:

*Example 6-21 Viewing AME summary usage information with svmon*

```
# svmon -G -O summary=longname,unit=MB
Unit: MB
-----
Active Memory Expansion
-----
Size      Inuse      Free      DXMSz     UCMinuse   CMInuse    TMSz     TMFr      CPSz      CPFr      txf      cxf      CR
16384.00  1725.35   14114.02  550.07   1498.47   226.88    8192.00  6553.71  139.82   40.5     2.00    1.93   2.28
```

In the output, the following memory compression statistics are provided:

- ▶ Out of the total expanded memory size (Size) of 16384 MB, 1725.35 MB is in use (Inuse) and 14114.02 MB is free (Free). The deficit in expanded memory size (DXMSz) is 550.07 MB.
- ▶ Out of the total in use memory (Inuse) of 1725.35 MB, uncompressed pages (UCMinuse) constitute 1498.47 MB and the compressed pages (CMInuse) constitute the remaining 226.88 MB.
- ▶ Out of the true memory size (TMSz) of 8192 MB, only 6553.71 MB of True Free memory (TMFr) is available.
- ▶ Out of the compressed pool size (CPSz) of 139.82 MB, only 40.5 MB of free memory (CPFr) is available in the compressed pool.
- ▶ Whereas the target expansion factor (txf) is 2.00, the current expansion factor (cxf) achieved is 1.93.
- ▶ The compression ratio (CR) is 2.28.



## 6.2 Hot Files Detection and filemon

An enhancement to the **filemon** command allows for the detection of *hot* files within a file system. The introduction of flash storage or Solid-State Disk (SSD) has necessitated the need for a method to determine the most active files in a file system. These files can then be located on or relocated to the fastest storage available. The enhancement is available in AIX V7.1, AIX V6.1 with Technology Level 4 and AIX V5.3 with Technology Level 11.

For a file to be considered “hot” it must be one that is read from, or written to frequently, or read from, or written to in large chunks of data. The **filemon** command can assist in determining which files are *hot* and will produce a report highlighting which files are the best candidates for SSD storage.

Using the **-O hot** option with the **filemon** command, administrators can generate reports that will assist with the placement of data on SSDs. The reports contain statistics for I/O operations of hot files, logical volumes and physical volumes. This data guides an administrator in determining which files and/or logical volumes are the ideal candidates for migration to SSDs. The *hotness* of a file and/or logical volume is based on the number of read operations, average number of bytes read per read operation, number of read sequences and the average sequence length.

The report generated by the **filemon** command consists of the three main sections. The first section contains information relating to the system type, the **filemon** command and the **trace** command. The second section is a summary which displays the total number of read/write operations, the total time taken, the total data read/written and the CPU utilization. The third section contains the hot data reports. There are three hot data reports in this section:

- ▶ Hot Files Report
- ▶ Hot Logical Volumes Report
- ▶ Hot Physical Volumes Report

Table 6-8 describes the information collected in the Hot Files Report section.

Table 6-8 Hot Files Report Description

Column	Description
Name	The name of the file.

Column	Description
Size	The size of the file. The default unit is MB. The default unit is overridden by the unit specified by the -O unit option.
CAP_ACC	The capacity accessed. This is the unique data accessed in the file. The default unit is MB. The default unit is overridden by the unit specified by the -O unit option.
IOP/#	The number of I/O operations per unit of data accessed. The unit of data is taken from the -O unit option. The default is MB. Other units could be K for KB, M for MB, G for GB and T for TB. For example, 0.000/K, 0.256M, 256/G, 2560/T.
LV	The name of the logical volume where the file is located. If this information cannot be obtained, a "-" is reported.
#ROP	Total number of read operations for the file.
#WOP	Total number of write operations for the file.
B/ROP	The minimum, average and maximum number of bytes read per read operation.
B/WOP	The minimum, average and maximum number of bytes write per read operation.
RTIME	The minimum, average and maximum time taken per read operation in milliseconds.
WTIME	The minimum, average and maximum time taken per write operation in milliseconds.
SeqLen	The minimum, average and maximum length of read sequences.
#Seq	Number of read sequences. A sequence is a string of 4K pages that are read (paged in) consecutively. The number of read sequences is an indicator of the amount of sequential access.

Table 6-9 describes the information collected in the Hot Logical Volumes Report.

*Table 6-9 Hot Logical Volumes Report Description*

Column	Description
Name	The name of the logical volume.
Size	The size of the logical volume. The default unit is MB. The default unit is overridden by the unit specified by the -O unit option.

Column	Description
CAP_ACC	The capacity accessed. This is the unique data accessed in the logical volume. The default unit is MB. The default unit is overridden by the unit specified by the -O unit option.
IOP/#	The number of I/O operations per unit of data accessed. The unit of data is taken from the -O unit option. The default is MB. Other units could be K for KB, M for MB, G for GB and T for TB. For example, 0.000/K, 0.256M, 256/G, 2560/T.
#Files	Number of files accessed in this logical volume.
#ROP	Total number of read operations for the logical volume.
#WOP	Total number of write operations for the logical volume.
B/ROP	The minimum, average and maximum number of bytes read per read operation.
B/WOP	The minimum, average and maximum number of bytes written per write operation.
RTIME	The minimum, average and maximum time taken per read operation in milliseconds.
WTIME	The minimum, average and maximum time taken per write operation in milliseconds.
SeqLen	The minimum, average and maximum length of read sequences.
#Seq	Number of read sequences. A sequence is a string of 4K pages that are read (paged in) consecutively. The number of read sequences is an indicator of the amount of sequential access.

Table 6-10 describes the information collected in the Hot Physical Volumes Report.

*Table 6-10 Hot Physical Volumes Report Description*

Column	Description
Name	The name of the physical volume.
Size	The size of the physical volume. The default unit is MB. The default unit is overridden by the unit specified by -O unit option.
CAP_ACC	The capacity accessed. This is the unique data accessed for the physical volume. The default unit is MB. The default unit is overridden by the unit specified by -O unit option.

Column	Description
IOP/#	The number of I/O operations per unit of data accessed. The unit of data is taken from the -O unit option. The default is MB. Other units could be K for KB, M for MB, G for GB and T for TB. For example, 0.000/K, 0.256M, 256/G, 2560/T.
#ROP	Total number of read operations for the physical volume.
#WOP	Total number of write operations for the physical volume.
B/ROP	The minimum, average and maximum number of bytes read per read operation.
B/WOP	The minimum, average and maximum number of bytes written per write operation.
RTIME	The minimum, average and maximum time taken per read operation in milliseconds.
WTIME	The minimum, average and maximum time taken per write operation in milliseconds.
Seqlen	The minimum, average and maximum length for read sequences.
#Seq	Number of read sequences. A sequence is a string of 512-byte blocks that are read consecutively. The number of read sequences is an indicator of the amount of sequential access.

Each of the hot reports are also sorted by capacity accessed. The data contained in the hot reports can be customized by specifying different options to the -O hot flag, as shown in Table 6-11:

Table 6-11 *filemon -O hot flag options*

Flag	Description
-O hot=r	Generates reports based on read operations only.
-O hot=w	Generates reports based on write operations only.

If the administrator specifies the -O hot=r option, then only read operation based reports are generated. If the administrator specifies the -O hot=w option, then only write operation based reports are captured.

The use of the -O hot option with the **filemon** command is only supported in automated offline mode. If you attempt to run the command in real-time mode you will receive an error message, as shown in Example 6-22 on page 231:

*Example 6-22 filemon -O hot is not supported in real-time mode*


---

```
# filemon -O hot -o fmon.out
hot option not supported in realtime mode
Usage: filemon [-i file -n file] [-o file] [-d] [-v] [-u] [-O opt [-w] [-I count:interval]] [-P] [-T
num] [-@ [WparList | ALL]] [-r RootString [-A -x "<User Command>"]]
-i file:  offline filemon - open trace file
-n file:  offline filemon - open gensyms file
          **Use gensyms -F to get the gensyms file
-o file:  open output file (default is stdout)
-d:       deferred trace (until 'trcon')
-T num:   set trace kernel buf sz (default 32000 bytes)
-P:       pin monitor process in memory
-v:       verbose mode (print extra details)
-u:       print unnamed file activity via pid
-O opt:   other monitor-specific options
-@ wparlist|ALL:
          output one report per WPAR in the list
-@:       output additionnal WPAR information
-A:       Enable Automated Offline Mode
-x:       Provide the user command to execute in double quotes if you provide argument to the
command
-r:       Root String for trace and gennames filenames
-w:       prints the hotness report in wide format(Valid only with -O hot option)
-I count:interval :   Used to specify multiple snapshots of trace collection (Valid only
with -O hot option)

valid -O options: [[detailed,]lf[=num],vm[=num],lv[=num],pv[=num],pr[=num],th[=num],all[=num]] |
abbreviated | collated | hot[={r|w}]lf[=num],lv[=num],pv[=num],sz=num,unit={KB|MB|GB|TB}
lf[=num]:  monitor logical file I/O and display first num records where num > 0
vm[=num]:  monitor virtual memory I/O and display first num records where num > 0
lv[=num]:  monitor logical volume I/O and display first num records where num > 0
pv[=num]:  monitor physical volume I/O and display first num records where num > 0
pr[=num]:  display data process-wise and display first num records where num > 0
th[=num]:  display data thread-wise and display first num records where num > 0
all[=num]: short for lf,vm,lv,pv,pr,th and display first num records where num > 0
detailed:  display detailed information other than summary report
abbreviated: Abbreviated mode (transactions). Supported only in offline mode
collated:  Collated mode (transactions). Supported only in offline mode
hot[={r|w}]: Generates hotness report(Not supported in realtime mode)
sz=num:    specifies the size of data accessed to be reported in the hotness report(valid only
with -O hot and in automated offline mode.
          Unit for this value is specified through -O unit option. Default is MB.)
unit={KB|MB|GB|TB}: unit for CAP_ACC and Size values in hotness report and unit for value
specified by -O sz option
```

---

Example 6-23 starts the **filemon** command in automated offline mode with the **-A** and **-x** flags, captures hot file data with the **-O hot** flag, specifies that trace data is stored in **fmon** (**.trc** is appended to the file name automatically) with the **-r** flag and writes I/O activity to the **fmon.out** file with the **-o** flag. A user specified command is placed after **-x** flag. The trace is collected until this command completes its work. A typical example of a user command is **sleep 60**.

*Example 6-23 Generating filemon hot file report in automated offline mode*


---

```
# filemon -O hot,unit=MB -r fmon -o fmon.out -A -x "sleep 60"
```

---

The contents of the **fmon.out** file are displayed in the examples that follow. Only the first few lines of each section of the report are displayed, as the report

contains a large amount of data. However, the data shown provides an introduction to the typical detail that is reported.

Example 6-24 shows the information and summary sections of the report.

*Example 6-24 Information and summary sections of the hot file report*

---

```
Thu Sep  2 19:32:27 2010
System: AIX 7.1 Node: 75021p04 Machine: 00F61AB24C00

Filemon Command used: filemon -0 hot,unit=MB -A -x sleep 60 -r fmon -o fmon.out
Trace command used: /usr/bin/trace -ad -L 2031364915 -T 1000000 -j
00A,001,002,003,38F,005,006,139,465,102,10C,106,4B0,419,107,101,104,10D,15B,12E,130
,163,19C,154,3D3,137,1BA,1BE,1BC,10B,AB2,221,232,1C9,2A2,
2A1,222,228,45B,5D8,3C4,3B9,223, -o fmon.trc

Summary Section
-----
Total monitored time: 60.012 seconds
Cpu utilization: 5.4%
Cpu allocation: 100.0%
Total no. of files monitored: 11
Total No. of I/O Operations: 126 ( Read: 126, write: 0 )
Total bytes transferred: 0.427 MB( Read: 0.427 MB, write: 0.000 MB )
Total IOP per unit: 295/MB
Total time taken for I/O operations(in miliseconds): 0.338 ( Read: 0.338, write:
0.000 )
```

---

The Hot Files Report section is shown in Example 6-25.

*Example 6-25 Hot Files Report*

---

```
Hot Files Report
-----
```

NAME	Size	CAP_ACC	IOP/#	LV
B/ROP	B/WOP	RTIME	WTIME	
#ROP	#WOP	SeqLen	#Seq	
/unix	33.437M	0.141M	256/M	/dev/hd2
4096,4096,4096	0,0,0	0.002,0.003,0.008	0.000,0.000,0.000	
97	0	1,1,1	97	
/etc/security/user	0.011M	0.012M	256/M	/dev/hd4
4096,4096,4096	0,0,0	0.003,0.004,0.008	0.000,0.000,0.000	
5	0	1,1,1	5	
/etc/security/group	0.000M	0.012M	256/M	/dev/hd4
4096,4096,4096	0,0,0	0.001,0.003,0.004	0.000,0.000,0.000	
4	0	1,1,1	4	

---

The Hot Logical Volume Report is shown in Example 6-26.

*Example 6-26 Hot Logical Volume Report*

Hot Logical Volume Report

NAME	Size	CAP_ACC	IOP/#	#Files
B/ROP	B/WOP	RTIME	WTIME	
#ROP	#WOP	SeqLen	#Seq	
/dev/loglv00	64.000M	0.000M	256/M	0
0,0,0	8,8,8	0.000,0.000,0.000	0.362,0.362,0.362	
0	1	0,0,0	0	
/dev/hd8	64.000M	0.070M	256/M	0
0,0,0	8,8,8	0.000,0.000,0.000	3.596,11.490,99.599	
0	25	0,0,0	0	
/dev/hd4	1984.000M	154.812M	256/M	4
0,0,0	8,8,8	0.000,0.000,0.000	3.962,93.807,141.121	
0	21	0,0,0	0	

The Hot Physical Volume Report is shown in Example 6-27.

*Example 6-27 Hot Physical Volume Report*

Hot Physical Volume Report

NAME	Size	CAP_ACC	IOP/#	
B/ROP	B/WOP	RTIME	WTIME	
#ROP	#WOP	SeqLen	#Seq	
/dev/hdisk0	35840.000M	17442.406M	52/M	
0,0,0	8,40,512	0.000,0.000,0.000	1.176,6.358,28.029	
0	132	0,0,0	0	
/dev/hdisk1	51200.000M	11528.816M	256/M	
0,0,0	8,8,8	0.000,0.000,0.000	0.351,0.351,0.351	
0	1	0,0,0	0	

The Hot File Report, sorted by capacity accessed section is shown in Example 6-28:

*Example 6-28 Hot Files sorted by capacity accessed*

Hot Files Report(sorted by CAP\_ACC)

NAME	Size	CAP_ACC	IOP/#	LV
B/ROP	B/WOP	RTIME	WTIME	

#ROP	#WOP	SeqLen	#Seq
MYFILE3		100.000M 100.000M 1024/M	/dev/hd3
0,0,0	4096,1024,4096	0.000,0.000,0.000	0.010,0.006,159.054
0	102400	0,0,0	0
MYFILE2		100.000M 100.000M 1024/M	/dev/hd3
0,0,0	4096,1024,4096	0.000,0.000,0.000	0.010,0.016,888.224
0	102400	0,0,0	0
MYFILE1		100.000M 100.000M 1024/M	/dev/hd3
0,0,0	4096,1024,4096	0.000,0.000,0.000	0.009,0.012,341.280
0	102400	0,0,0	0

The Hot Logical Volume Report, sorted by capacity accessed section is displayed in Example 6-29.

#### Example 6-29 Hot Logical Volumes

Hot Logical Volume Report(sorted by CAP\_ACC)

NAME	Size	CAP_ACC	IOP/#	#Files
B/ROP	B/WOP	RTIME	WTIME	
#ROP	#WOP	SeqLen	#Seq	
/dev/hd2	1984.000M	1581.219M	256/M	3
0,0,0	8,8,8	0.000,0.000,0.000	11.756,42.800,81.619	
0	12	0,0,0	0	
/dev/hd3	4224.000M	459.812M	8/M	3
0,0,0	8,263,512	0.000,0.000,0.000	3.720,339.170,1359.117	
0	10364	0,0,0	0	
/dev/hd9var	384.000M	302.699M	256/M	2
0,0,0	8,8,8	0.000,0.000,0.000	3.935,50.324,103.397	
0	15	0,0,0	0	

The Hot Physical Volume Report, sorted by capacity accessed section is displayed in Example 6-30.

#### Example 6-30 Hot Physical Volumes

Hot Physical Volume Report(sorted by CAP\_ACC)

NAME	Size	CAP_ACC	IOP/#
B/ROP	B/WOP	RTIME	WTIME
#ROP	#WOP	SeqLen	#Seq



```

/dev/hdisk0          35840.000M    17998.020M    8/M
0,0,0                8,262,512      0.000,0.000,0.000  0.984,3.001,59.713
0                    10400          0,0,0         0
-----

```

The Hot Files Report, sorted by IOP/# is shown in Example 6-31.

*Example 6-31 Hot Files sorted by IOP*

Hot Files Report(sorted by IOP/#)

```

-----
NAME                Size      CAP_ACC  IOP/#  LV
B/ROP              B/WOP      RTIME    WTIME
#ROP              #WOP      SeqLen   #Seq
-----
/etc/objrepos/SWservAt.vc  0.016M  0.000M  52429/M /dev/hd4
40,20,40          0,0,0      0.002,0.001,0.003  0.000,0.000,0.000
4                  0          1,1,1         1
-----
/var/adm/cron/log      0.596M  0.000M  14075/M /dev/hd9var
0,0,0            39,74,110  0.000,0.000,0.000  0.009,0.015,0.021
0                  2          0,0,0         0
-----
/etc/objrepos/SWservAt  0.012M  0.000M  5269/M /dev/hd4
328,199,468      0,0,0      0.002,0.001,0.004  0.000,0.000,0.000
4                  0          1,1,1         1
-----

```

The Hot Logical Volume report, sorted by IOP/# is shown in Example 6-32.

*Example 6-32 Hot Logical Volumes sorted by IOP*

Hot Logical Volume Report(sorted by IOP/#)

```

-----
NAME                Size      CAP_ACC  IOP/#  #Files
B/ROP              B/WOP      RTIME    WTIME
#ROP              #WOP      SeqLen   #Seq
-----
/dev/fs1v00         128.000M  0.000M  256/M    0
0,0,0             8,8,8      0.000,0.000,0.000  59.731,59.731,59.731
0                  1          0,0,0         0
-----
/dev/fs1v01         64.000M  0.000M  256/M    0
0,0,0             8,8,8      0.000,0.000,0.000  3.854,3.854,3.854
0                  1          0,0,0         0
-----
/dev/fs1v02         128.000M  0.000M  256/M    0
0,0,0             8,8,8      0.000,0.000,0.000  4.108,4.108,4.108
0                  1          0,0,0         0
-----

```

The Hot Physical Volume Report, sorted by IOP/# is shown in Example 6-33.

*Example 6-33 Hot Physical Volumes sorted by IOP*

---

Hot Physical Volume Report(sorted by IOP/#)

---

NAME	Size	CAP_ACC	IOP/#
B/ROP	B/WOP	RTIME	WTIME
#ROP	#WOP	SeqLen	#Seq
/dev/hdisk0	35840.000M	17998.020M	8/M
0,0,0	8,262,512	0.000,0.000,0.000	0.984,3.001,59.713
0	10400	0,0,0	0

---

The Hot Files Report, sorted by #ROP is shown in Example 6-34.

*Example 6-34 Hot Files sorted by #ROP*

---

Hot Files Report(sorted by #ROP)

---

NAME	Size	CAP_ACC	IOP/#	LV
B/ROP	B/WOP	RTIME	WTIME	
#ROP	#WOP	SeqLen	#Seq	
/unix	33.437M	0.141M	256/M	/dev/hd2
4096,4096,4096	0,0,0	0.002,0.003,0.008	0.000,0.000,0.000	
97	0	1,1,1	97	
/usr/lib/nls/msg/en_US/ksh.cat	0.006M	0.008M	4352/M	/dev/hd2
4096,241,4096	0,0,0	0.003,0.000,0.004	0.000,0.000,0.000	
68	0	1,2,2	2	
/etc/security/user	0.011M	0.012M	256/M	/dev/hd4
4096,4096,4096	0,0,0	0.003,0.004,0.008	0.000,0.000,0.000	
5	0	1,1,1	5	

---

The Hot Logical Volume Report, sorted by #ROP is shown in Example 6-35.

*Example 6-35 Hot Logical Volumes sorted by #ROP*

---

Hot Logical Volume Report(sorted by #ROP)

---

NAME	Size	CAP_ACC	IOP/#	#Files
B/ROP	B/WOP	RTIME	WTIME	
#ROP	#WOP	SeqLen	#Seq	
/dev/hd3	4224.000M	459.812M	8/M	3
0,0,0	8,263,512	0.000,0.000,0.000	3.720,339.170,1359.117	
0	10364	0,0,0	0	

```

-----
/dev/hd2          1984.000M    1581.219M    256/M        3
0,0,0           8,8,8          0.000,0.000,0.000  11.756,42.800,81.619
0                12             0,0,0        0
-----
/dev/hd9var      384.000M       302.699M     256/M        2
0,0,0           8,8,8          0.000,0.000,0.000  3.935,50.324,103.397
0                15             0,0,0        0
-----

```

The Hot Physical Volumes sorted by #ROP is shown in Example 6-36.

*Example 6-36 Hot Physical Volumes sorted by #ROP*

Hot Physical Volume Report(sorted by #ROP)

```

-----
NAME              Size          CAP_ACC      IOP/#
B/ROP            B/WOP          RTIME        WTIME
#ROP             #WOP           SeqLen       #Seq
-----
/dev/hdisk0      35840.000M    17998.020M   8/M
0,0,0           8,262,512     0.000,0.000,0.000  0.984,3.001,59.713
0                10400         0,0,0        0
-----

```

The Hot Files Report, sorted by #WOP is shown in Example 6-37.

*Example 6-37 Hot Files sorted by #WOP*

Hot Files Report(sorted by #WOP)

```

-----
NAME              Size          CAP_ACC      IOP/#      LV
B/ROP            B/WOP          RTIME        WTIME
#ROP             #WOP           SeqLen       #Seq
-----
1                100.000M     100.000M    1024/M     /dev/hd3
0,0,0           4096,1024,4096  0.000,0.000,0.000  0.009,0.012,341.280
0                102400        0,0,0        0
-----
2                100.000M     100.000M    1024/M     /dev/hd3
0,0,0           4096,1024,4096  0.000,0.000,0.000  0.010,0.016,888.224
0                102400        0,0,0        0
-----
3                100.000M     100.000M    1024/M     /dev/hd3
0,0,0           4096,1024,4096  0.000,0.000,0.000  0.010,0.006,159.054
0                102400        0,0,0        0
-----

```

The Hot Logical Volume Report, sorted by #WOP is shown in Example 6-38 on page 238.

*Example 6-38 Hot Logical Volumes sorted by #WOP*


---

Hot Logical Volume Report(sorted by #WOP)

---

NAME	Size	CAP_ACC	IOP/#	#Files
B/ROP	B/WOP	RTIME		WTIME
#ROP	#WOP	SeqLen		#Seq
/dev/hd3	4224.000M	459.812M	8/M	3
0,0,0	8,263,512	0.000,0.000,0.000		3.720,339.170,1359.117
0	10364	0,0,0		0
/dev/hd8	64.000M	0.090M	256/M	0
0,0,0	8,8,8	0.000,0.000,0.000		1.010,75.709,1046.734
0	61	0,0,0		0
/dev/hd4	192.000M	154.934M	256/M	12
0,0,0	8,8,8	0.000,0.000,0.000		1.907,27.166,74.692
0	16	0,0,0		0

---

The Hot Physical Volume Report, sorted by #WOP is shown in Example 6-39.

*Example 6-39 Hot Physical Volumes sorted by #WOP*


---

Hot Physical Volume Report(sorted by #WOP)

---

NAME	Size	CAP_ACC	IOP/#
B/ROP	B/WOP	RTIME	WTIME
#ROP	#WOP	SeqLen	#Seq
/dev/hdisk0	35840.000M	17998.020M	8/M
0,0,0	8,262,512	0.000,0.000,0.000	0.984,3.001,59.713
0	10400	0,0,0	0

---

The Hot Files Report, sorted by RTIME is shown in Example 6-40.

*Example 6-40 Hot Files sorted by RTIME*


---

Hot Files Report(sorted by RTIME)

---

NAME	Size	CAP_ACC	IOP/#	LV
B/ROP	B/WOP	RTIME		WTIME
#ROP	#WOP	SeqLen		#Seq
/etc/vfs	0.002M	0.008M	256/M	/dev/hd4
4096,4096,4096	0,0,0	0.002,0.006,0.010		0.000,0.000,0.000
2	0	2,2,2		1
/etc/security/user	0.011M	0.012M	256/M	/dev/hd4
4096,4096,4096	0,0,0	0.003,0.004,0.008		0.000,0.000,0.000

---

```

5                0                1,1,1                5
-----
/usr/lib/nls/msg/en_US/cmdtrace.cat 0.064M  0.004M  256/M  /dev/hd2
4096,4096,4096  0,0,0                0.004,0.004,0.004  0.000,0.000,0.000
2                0                1,1,1                2
-----

```

The Hot Logical Volume Report, sorted by RTIME is shown in Example 6-41.

*Example 6-41 Hot Logical Volumes sorted by RTIME*

Hot Logical Volume Report(sorted by RTIME)

```

-----
NAME                Size                CAP_ACC                IOP/#                #Files
B/ROP                B/WOP                RTIME                WTIME
#ROP                #WOP                SeqLen                #Seq
-----
/dev/fs1v02                128.000M                0.000M                256/M                0
0,0,0                8,8,8                0.000,0.000,0.000  4.108,4.108,4.108
0                1                0,0,0                0
-----
/dev/fs1v01                64.000M                0.000M                256/M                0
0,0,0                8,8,8                0.000,0.000,0.000  3.854,3.854,3.854
0                1                0,0,0                0
-----
/dev/fs1v00                128.000M                0.000M                256/M                0
0,0,0                8,8,8                0.000,0.000,0.000  59.731,59.731,59.731
0                1                0,0,0                0
-----

```

The Hot Physical Volume Report, sorted by RTIME is shown in Example 6-42.

*Example 6-42 Hot Physical Volumes sorted by RTIME*

Hot Physical Volume Report(sorted by RTIME)

```

-----
NAME                Size                CAP_ACC                IOP/#
B/ROP                B/WOP                RTIME                WTIME
#ROP                #WOP                SeqLen                #Seq
-----
/dev/hdisk0                35840.000M                17998.020M                8/M
0,0,0                8,262,512                0.000,0.000,0.000  0.984,3.001,59.713
0                10400                0,0,0                0
-----

```

The Hot Files Report, sorted by WTIME is shown in Example 6-43.

*Example 6-43 Hot Files sorted by WTIME*

Hot Files Report(sorted by WTIME)

```

-----
NAME                Size      CAP_ACC  IOP/#  LV
B/ROP              B/WOP      RTIME      WTIME
#ROP              #WOP      SeqLen     #Seq
-----
2                  100.000M  100.000M  1024/M  /dev/hd3
0,0,0             4096,1024,4096  0.000,0.000,0.000  0.010,0.016,888.224
0                  102400      0,0,0      0
-----
/var/adm/cron/log   0.596M    0.000M    14075/M  /dev/hd9var
0,0,0             39,74,110  0.000,0.000,0.000  0.009,0.015,0.021
0                  2           0,0,0      0
-----
1                  100.000M  100.000M  1024/M  /dev/hd3
0,0,0             4096,1024,4096  0.000,0.000,0.000  0.009,0.012,341.280
0                  102400      0,0,0      0
-----
3                  100.000M  100.000M  1024/M  /dev/hd3
0,0,0             4096,1024,4096  0.000,0.000,0.000  0.010,0.006,159.054
0                  102400      0,0,0      0
-----

```

The Hot Logical Volume Report, sorted by WTIME is shown in Example 6-44.

*Example 6-44 Hot Logical Volumes sorted by WTIME*

Hot Logical Volume Report(sorted by WTIME)

```

-----
NAME                Size      CAP_ACC  IOP/#  #Files
B/ROP              B/WOP      RTIME      WTIME
#ROP              #WOP      SeqLen     #Seq
-----
/dev/hd3            4224.000M  459.812M   8/M      3
0,0,0             8,263,512  0.000,0.000,0.000  3.720,339.170,1359.117
0                  10364      0,0,0      0
-----
/dev/hd8            64.000M    0.090M     256/M    0
0,0,0             8,8,8      0.000,0.000,0.000  1.010,75.709,1046.734
0                  61         0,0,0      0
-----
/dev/fs1v00         128.000M   0.000M     256/M    0
0,0,0             8,8,8      0.000,0.000,0.000  59.731,59.731,59.731
0                  1          0,0,0      0
-----

```

The Hot Physical Volume Report, sorted by WTIME is shown in Example 6-45 on page 241.

Example 6-45 Hot Physical Volumes sorted by WTIME

Hot Physical Volume Report(sorted by WTIME)			
NAME	Size	CAP_ACC	IOP/#
B/ROP	B/WOP	RTIME	WTIME
#ROP	#WOP	SeqLen	#Seq
/dev/hdisk0	35840.000M	17998.020M	8/M
0,0,0	8,262,512	0.000,0.000,0.000	0.984,3.001,59.713
0	10400	0,0,0	0

## 6.3 Memory affinity API enhancements

AIX 7.1 allows an application to request for a thread to have a *strict* attachment to an SRAD for purposes of memory affinity. The new form of attachment is similar to the current SRAD attachment APIs except that the thread will not be moved to a different SRAD for purposes of load balancing by the dispatcher.

The following is a comparison between new *strict* attachment API and the existing *advisory* attachment API.

- ▶ When a thread has an *advisory* SRAD attachment, the AIX thread dispatcher is free to ignore the attachment if the distribution of load across various SRADs justifies migration of the thread to another SRAD. The new *strict* attachment will override any load balancing efforts of the dispatcher.
- ▶ The current advisory SRAD attachment APIs allow SRAD attachments to R\_PROCESS, R\_THREAD, R\_SHM, R\_FILDES and R\_PROCMEM resourcetypes. The new strict SRAD attachment only allows SRAD attachment to R\_THREAD resource type. Any other use of strict SRAD attachment will result in an EINVAL error code.
- ▶ The pthread\_attr\_setsrad\_np API is modified to accept a new *flag* parameter that will indicate whether the srad attachment is *strict* or *advisory*.

The following is a list of functionality that is not changed from advisory SRAD attachments. They are mentioned here for completeness.

- ▶ If a strict attachment is sought for an SRAD that has only folded processors at the time of the attachment request, the request is processed normally. The threads are placed temporarily on the node global run queue. The expectation is that folding is a temporary situation and the threads will get run time when the processors are unfolded.

- ▶ Unauthorized applications can make *strict* SRAD attachments. root authority or CAP\_NUMA\_ATTACH capability is not a requirement. This is the same behavior as in advisory SRAD attachment APIs.
- ▶ If a *strict* attachment is attempted to an SRAD that has only exclusive CPUs, then the attachment will succeed and the thread will be marked as permanently borrowed. This is the same behavior as in advisory SRAD attachment APIs.
- ▶ DR CPU remove operation will ignore *strict* SRAD attachments when calculating CPU costs which DRM uses to pick the CPU to remove. This is the same behavior as in advisory SRAD attachment APIs.
- ▶ Advisory attachments are ignored in the event of a DR operation requiring all threads to be migrated off a processor. This holds true for *strict* attachments as well.
- ▶ When a request for an advisory SRAD attachment conflicts with an existing RSET attachment, the SRAD attachment is still processed if there is at least one processor in the intersection between the SRAD and the RSET. This holds true for *strict* SRAD attachments.
- ▶ When an advisory attachment is sought for a thread that already has a previous attachment, the older attachment is overridden by the new one. This behavior is maintained when seeking a *strict* attachment as well.

### 6.3.1 API enhancements

#### **ra\_attach, ra\_fork, ra\_exec**

A new flag *R\_STRICT\_SRAD* is added to the *flags* parameter of the *ra\_attach*, *ra\_fork* and *ra\_exec* APIs.

*R\_STRICT\_SRAD* flag indicates a thread is attached to an SRAD in a *strict* manner. It will run within the same SRAD, unaffected by load balancing operations. It will be re-homed to a different SRAD only if a DR operation removes all processors from the current SRAD. It is important to note that when strict SRAD attachments are used, the application must cater for the possibility of uneven load across SRADs.

**Note:** *ra\_detach* removes all SRAD attachments, *strict* is used to detach an existing SRAD attachment, any attachment *strict* or *advisory* will be removed



## 6.3.2 pthread attribute API

There are two existing pthread APIs to set/get an SRAD in the pthread attributes, namely *pthread\_attr\_setsrad\_np* and *pthread\_attr\_getsrad\_np*. These are modified to have a *flags* parameter that will indicate if the SRAD attachment is strict.

### **pthread\_attr\_setsrad\_np**

Syntax:

```
int pthread_attr_setsrad_np (attr, srad, flags)
```

```
pthread_attr_t *attr;
```

```
sradid_t srad;
```

```
int flags;
```

Description:

The *flags* parameter indicates whether the SRAD attachment is strict or advisory.

Parameters:

flags: Setting *R\_STRICT\_SRAD* indicates that the srad is a strictly preferred one.

### **pthread\_attr\_getsrad\_np**

Syntax:

```
int pthread_attr_getsrad_np (attr, sradp, flagsp)
```

```
pthread_attr_t *attr;
```

```
sradid_t *sradp;
```

```
int *flagsp;
```

Description:

The *flagsp* parameter returns *R\_STRICT\_SRAD* if the SRAD attachment, if any, is strict.

Parameters:

flagsp: Set to *R\_STRICT\_SRAD* if SRAD attachment is strict, NULL otherwise.

## 6.4 iostat command enhancement

Debugging I/O performance and hang issues is a time consuming and iterative process. To help with the analysis of I/O issues the **iostat** command has been enhanced in AIX 6.1 TL6 and in AIX 7.1. With this enhancement useful data can be captured to help identify and correct the problem quicker.

The enhancement to the **iostat** command leverages the bufx capabilities in AIX to produce an end to end I/O metrics report. It is called the Block I/O Device Utilization Report.

The Block I/O Device Utilization report provides statistics per I/O device. The report helps you in analyzing the I/O statistics at VMM or file system, and disk layers of I/O stack. The report also helps you in analyzing the performance of the I/O stack.

A new flag, **-b**, is available for the **iostat** command that will display block I/O device utilization statistics.

Example 6-46 shows an example of the command output when this new flag is used.

### *Example 6-46 Example of the new iostat output*

---

```
# iostat -b 5

System configuration: lcpu=2 drives=3 vdisks=3
Block Devices :7
device          reads writes   bread  bwrite  rserv  wserv  rerr  werr
hdisk0          0.00  0.00   0.000  0.000   0.00   0.00  0.00  0.00
hd8             0.00  0.00   0.000  0.000   0.00   0.00  0.00  0.00
hd4             0.00  0.00   0.000  0.000   0.00   0.00  0.00  0.00
hd9var         0.00  0.00   0.000  0.000   0.00   0.00  0.00  0.00
hd2            0.00  0.00   0.000  0.000   0.00   0.00  0.00  0.00
hd3            0.00  0.00   0.000  0.000   0.00   0.00  0.00  0.00
hd10opt        0.00  0.00   0.000  0.000   0.00   0.00  0.00  0.00
```

---

The meaning of the columns is as follows:

<b>device</b>	Indicates the device name
<b>reads</b>	Indicates the number of read requests over the monitoring interval.
<b>writes</b>	Indicates the number of write requests over the monitoring interval.

<b>bread</b>	Indicates the number of bytes read over the monitoring interval.
<b>bwrite</b>	Indicates the number of bytes written over the monitoring interval.
<b>rserv</b>	Indicates the read service time per read over the monitoring interval. The default unit of measure is milliseconds.
<b>wserv</b>	Indicates the write service time per write over the monitoring interval. The default unit of measure is milliseconds.
<b>rerr</b>	Indicates the number of read errors over the monitoring interval.
<b>werr</b>	Indicates the number of write errors over the monitoring interval.

The **raso** command is used to turn the statistic collection on and off. Example 6-47 shows how to use the **raso** command to turn on the statistic collection that the **iostat** command uses.

*Example 6-47 Using the raso command to turn on statistic collection*

---

```
# raso -o biostat=1
Setting biostat to 1
#
```

---

The **raso -L** command will show the current status of statistic collection. Example 6-48 shows the output of the **raso -L** command.

*Example 6-48 Using raso -L command to see if statistic collection is on*

---

```
# raso -L
NAME                                CUR  DEF  BOOT  MIN  MAX  UNIT          TYPE
DEPENDENCIES
-----
biostat                             1    0    0    0    1    boolean      D
-----
kern_heap_noexec                    0    0    0    0    1    boolean      B
-----
kernel_noexec                       1    1    1    0    1    boolean      B
-----
mbuf_heap_noexec                     0    0    0    0    1    boolean      B
-----
mtrc_commonbufsize                  1209 1209 1209  1    16320 4KB pages    D
  mtrc_enabled
  mtrc_rarebufsize
-----
mtrc_enabled                         1    1    1    0    1    boolean      B
-----
mtrc_rarebufsize                     62   62   62   1    15173 4KB pages    D
  mtrc_enabled
  mtrc_commonbufsize
```

tprof_cyc_mult	1	1	1	1	100	numeric	D
tprof_evt_mult	1	1	1	1	10000	numeric	D
tprof_evt_system	0	0	0	0	1	boolean	D
tprof_inst_threshold	1000	1000	1000	1	2G-1	numeric	D

n/a means parameter not supported by the current platform or kernel

Parameter types:

S = Static: cannot be changed  
 D = Dynamic: can be freely changed  
 B = Bosboot: can only be changed using bosboot and reboot  
 R = Reboot: can only be changed during reboot  
 C = Connect: changes are only effective for future socket connections  
 M = Mount: changes are only effective for future mountings  
 I = Incremental: can only be incremented  
 d = deprecated: deprecated and cannot be changed

Value conventions:

K = Kilo: 2<sup>10</sup>      G = Giga: 2<sup>30</sup>      P = Peta: 2<sup>50</sup>  
 M = Mega: 2<sup>20</sup>      T = Tera: 2<sup>40</sup>      E = Exa: 2<sup>60</sup>

#

**Note:** The biostat tuning parameter is dynamic. It does not require a reboot to take effect.

Turning on the statistic collection uses a little more memory but does not have a CPU utilization impact.



# Networking

AIX v7.1 provides many enhancements in the networking area. Described in this chapter, they include:

- ▶ 7.1, “Enhancement to IEEE 802.3ad link aggregation” on page 248
- ▶ 7.2, “Removal of BIND 8 application code” on page 258
- ▶ 7.3, “Network Time Protocol version 4” on page 259
- ▶ 7.4, “Reliable Datagram Sockets (RDS) v3 for RDMA support” on page 263

## 7.1 Enhancement to IEEE 802.3ad link aggregation

This section will discuss the enhancement to the Ethernet link aggregation in AIX V7.1.

This feature first became available in AIX V7.1 and is included in AIX 6.1 TL 06.

### 7.1.1 EtherChannel and Link Aggregation in AIX

EtherChannel and IEEE 802.3ad Link Aggregation are network port aggregation technologies that allow multiple Ethernet adapters to be teamed to form a single pseudo Ethernet device. This teaming of multiple Ethernet adapters to form a single pseudo Ethernet device is known as aggregation.

Conceptually, IEEE 802.3ad Link Aggregation works the same as EtherChannel.

Advantages of using IEEE 802.3ad Link Aggregation over EtherChannel are that IEEE 802.3ad Link Aggregation can create the link aggregations in the switch automatically, and that it allows you to use switches that support the IEEE 802.3ad standard but do not support EtherChannel.

**Note:** When using IEEE 802.3ad Link Aggregation ensure that your Ethernet switch hardware supports the IEEE 802.3ad standard.

With the release of AIX V7.1 and AIX V6.1 TL06, configuring an AIX Ethernet interface to use the 802.3ad mode requires that the Ethernet switch ports also be configured in IEEE 802.3ad mode.

### 7.1.2 IEEE 802.3ad Link Aggregation functionality

The IEEE 802.3ad Link Aggregation protocol, also known as Link Aggregation Control Protocol (LACP), relies on LACP Data Units (LACPDU) to control the status of link aggregation between two parties, the actor and the partner.

The actor is the IEEE 802.3ad Link Aggregation and the partner is the Ethernet switch port.

The Link Aggregation Control Protocol Data Unit (LACPDU) contains the information about the actor and the actor's view of its partner. Each port in the aggregation acts as an actor and a partner. LACPDU is exchanged at the rate specified by the actor. All ports under the link aggregation are required to participate in LACP activity.

Both the actor and the partner monitor LACPDU in order to ensure that communication is correctly established and that they have the correct view of the other's capability.

The aggregated link is considered to be non-operational when there is a disagreement between an actor and its partner. When an aggregation is considered non-operational then the that port will not used to transfer data packets. A port will only be used to transfer data packets if both the actor and the partner have exchanged LACPDU and they agree with each other's view.

### 7.1.3 AIX v7.1 enhancement to IEEE 802.3ad Link Aggregation

Prior to AIX v7.1, the AIX implementation of the IEEE 802.3ad protocol did not wait for the LACP exchange to complete before using the port for data transmission.

This could result in packet loss if the LACP partner which may typically be an Ethernet switch, relies on LACP exchange to complete before it uses the port for data transmission. This could result in significant packet loss if the delay between the link status up and the LACP exchange complete is large.

AIX v7.1 includes an enhancement to the LACP implementation to allow ports to exchange LACPDU and agree upon each other's state before they are ready for data transmission.

This enhancement is particularly useful when using stacked Ethernet switches.

Without this enhancement to the AIX implementation of IEEE 802.3ad, Stacked Ethernet switches may experience delays between the time that an Ethernet port is activated and an LACPDU transmit occurs when integrating or reintegrating an Ethernet switch into the stacked Ethernet switch configuration.

**Important:** In previous versions of AIX, the implementation of the IEEE 802.3ad protocol did not require Ethernet switch ports to be configured to use 802.3ad protocol.

AIX V7.1 and AIX V6.1 TL06 require the corresponding Ethernet switch ports be configured in IEEE 802.3ad mode when the AIX Ethernet interface is operating in the 802.3ad mode.

When planning to upgrade or migrate to AIX V7.1 or AIX V6.1 TL06, ensure that any Ethernet switch ports in use by an AIX 802.3ad Link Aggregation are configured to support the 802.3ad protocol.

When operating in IEEE 802.3ad mode, the enhanced support allows for up to three LACPDU to be missed within the interval value. Once three LACPDU are missed within the interval value, AIX will not use the link for data transmission until such time as a new LACPDU is received.

The interval durations are displayed in Table 7-1.

Table 7-1 The LACP interval duration

Type of interval	Interval duration
short interval	3 seconds
long interval	90 seconds

In the following examples we will show an IEEE 802.3ad Link Aggregation change from an operational to non-operational state, then revert to operational status due to a hardware cabling issue.

Our IEEE 802.3ad Link Aggregation pseudo Ethernet device is defined as ent6. The ent6 pseudo Ethernet device consists of the two logical Ethernet devices ent2 and ent4. Example 7-1 lists the **lsdev -Cc adapter** command output, displaying the ent6 pseudo Ethernet device.

**Note:** The **lsdev** command displays the ent6 pseudo Ethernet device as an EtherChannel / IEEE 802.3ad Link Aggregation. We will discuss later in the example how to determine whether the ent6 pseudo device is operating as an IEEE 802.3ad Link Aggregation.

Example 7-1 The **lsdev -Cc adapter** command

---

```
# lsdev -Cc adapter
ent0 Available Virtual I/O Ethernet Adapter (1-lan)
ent1 Available Virtual I/O Ethernet Adapter (1-lan)
ent2 Available 00-08 2-Port 10/100/1000 Base-TX PCI-X Adapter (14108902)
ent3 Available 00-09 2-Port 10/100/1000 Base-TX PCI-X Adapter (14108902)
ent4 Available 01-08 2-Port 10/100/1000 Base-TX PCI-X Adapter (14108902)
ent5 Available 01-09 2-Port 10/100/1000 Base-TX PCI-X Adapter (14108902)
ent6 Available EtherChannel / IEEE 802.3ad Link Aggregation
vsa0 Available LPAR Virtual Serial Adapter
vscsi0 Available Virtual SCSI Client Adapter
#
```

---

By using the **lsattr -E1** command, we can display the logical Ethernet devices that are a make up the ent6 pseudo Ethernet device.



The **lsattr -E1 ent6** command will also display in which mode the pseudo Ethernet device is operating. In we can see that the ent6 pseudo Ethernet device is made up of the ent2 and ent4 logical Ethernet devices. Additionally, the ent6 pseudo Ethernet device is operating in IEEE 802.3ad mode and the interval is long.

*Example 7-2 Displaying the logical Ethernet devices in the ent6 pseudo Ethernet device*

---

```
# lsattr -E1 ent6
adapter_names  ent2,ent4      EtherChannel Adapters                True
alt_addr       0x0000000000000000 Alternate EtherChannel Address         True
auto_recovery  yes                    Enable automatic recovery after failover True
backup_adapter NONE                  Adapter used when whole channel fails  True
hash_mode     default               Determines how outgoing adapter is chosen True
interval      long                  Determines interval value for IEEE 802.3ad mode True
mode          8023ad                EtherChannel mode of operation        True
netaddr       0                     Address to ping                        True
no loss_failover yes                Enable lossless failover after ping failure True
num_retries   3                     Times to retry ping before failing     True
retry_time    1                     Wait time (in seconds) between pings  True
use_alt_addr  no                    Enable Alternate EtherChannel Address  True
use_jumbo_frame no                  Enable Gigabit Ethernet Jumbo Frames   True
#
```

---

The ent2 and ent4 devices are each defined on port T1 of a one Gigabit Ethernet adapter in the AIX V7.1 partition.

Example 7-3 lists the physical hardware locations for the ent2 and ent4 logical Ethernet devices by using the **lsslot -c pci** and **lscfg -vl** commands.

*Example 7-3 The lsslot and lscfg commands display the physical Ethernet adapters*

---

```
# lsslot -c pci
# Slot          Description          Device(s)
U78A0.001.DNWHZS4-P1-C4 PCI-X capable, 64 bit, 266MHz slot ent2 ent3
U78A0.001.DNWHZS4-P1-C5 PCI-X capable, 64 bit, 266MHz slot ent4 ent5

# lscfg -vl ent2
ent2          U78A0.001.DNWHZS4-P1-C4-T1  2-Port 10/100/1000 Base-TX PCI-X
Adapter (14108902)

                2-Port 10/100/1000 Base-TX PCI-X Adapter:
                Part Number.....03N5297
                FRU Number.....03N5297
                EC Level.....H13845
                Manufacture ID.....YL1021
                Network Address.....00215E8A4072
                ROM Level.(alterable).....DV0210
                Hardware Location Code.....U78A0.001.DNWHZS4-P1-C4-T1

# lscfg -vl ent4
```

```
ent4          U78A0.001.DNWHZS4-P1-C5-T1 2-Port 10/100/1000 Base-TX PCI-X
Adapter (14108902)
```

```
2-Port 10/100/1000 Base-TX PCI-X Adapter:
Part Number.....03N5297
FRU Number.....03N5297
EC Level.....H13845
Manufacture ID.....YL1021
Network Address.....00215E8A41B6
ROM Level.(alterable).....DV0210
Hardware Location Code.....U78A0.001.DNWHZS4-P1-C5-T1
```

```
#
```

---

Example 7-4 shows the **entstat -d** command being used to display the status of the ent6 pseudo Ethernet device.

**Note:** Due to the large amount of output displayed by the **entstat -d** command, only the fields relevant to this example have been shown.

*Example 7-4 The entstat -d ent6 output - Link Aggregation operational*

---

```
# entstat -d ent6
```

```
-----
ETHERNET STATISTICS (ent6) :
Device Type: IEEE 802.3ad Link Aggregation
Hardware Address: 00:21:5e:8a:40:72
Elapsed Time: 0 days 21 hours 43 minutes 30 seconds
-----
ETHERNET STATISTICS (ent2) :
Device Type: 2-Port 10/100/1000 Base-TX PCI-X Adapter (14108902)
Hardware Address: 00:21:5e:8a:40:72
```

```
IEEE 802.3ad Port Statistics:
```

```
-----
Actor State:
LACP activity: Active
LACP timeout: Long
Aggregation: Aggregatable
Synchronization: IN_SYNC
Collecting: Enabled
Distributing: Enabled
Defaulted: False
Expired: False
```

```
Partner State:
  LACP activity: Active
  LACP timeout: Long
  Aggregation: Aggregatable
  Synchronization: IN_SYNC
  Collecting: Enabled
  Distributing: Enabled
  Defaulted: False
  Expired: False
```

```
-----
ETHERNET STATISTICS (ent4) :
Device Type: 2-Port 10/100/1000 Base-TX PCI-X Adapter (14108902)
Hardware Address: 00:21:5e:8a:40:72
```

```
IEEE 802.3ad Port Statistics:
-----
```

```
Actor State:
  LACP activity: Active
  LACP timeout: Long
  Aggregation: Aggregatable
  Synchronization: IN_SYNC
  Collecting: Enabled
  Distributing: Enabled
  Defaulted: False
  Expired: False
```

```
Partner State:
  LACP activity: Active
  LACP timeout: Long
  Aggregation: Aggregatable
  Synchronization: IN_SYNC
  Collecting: Enabled
  Distributing: Enabled
  Defaulted: False
  Expired: False
```

```
#
```

---

In Example 7-4 on page 252, the Actor State for both the ent2 and ent4 logical Ethernet devices shows the Distributing state as Enabled and the Expired state as False. The Synchronization state is IN\_SYNC.

Additionally, in Example 7-4 on page 252 the Partner State for both the ent2 and ent4 logical Ethernet devices shows the Distributing state as Enabled and the Expired state as False. The Synchronization state is IN\_SYNC.

This is the normal status mode for an operational IEEE 802.3a Link Aggregation.

The administrator is alerted of a connectivity issue by an error in the AIX error report. By using the **entstat -d** command the administrator discovers that the ent4 logical Ethernet device is no longer operational.

Example 7-5 lists the output from the **entstat -d** command. In this example, the Actor State and Partner State values for the ent4 logical Ethernet device status have changed. The ent2 logical Ethernet device status remains unchanged.

**Note:** Due to the large amount of output displayed by the **entstat -d** command, only the fields relevant to this example have been shown.

*Example 7-5 The entstat -d ent6 output - Link Aggregation non-operational*

```
# errpt
ECOBCCD4 0825110510 T H ent4          ETHERNET DOWN
A6DF45AA 0820181410 I O RMCdaemon      The daemon is started.
# entstat -d ent6
-----
ETHERNET STATISTICS (ent6) :
Device Type: IEEE 802.3ad Link Aggregation
Hardware Address: 00:21:5e:8a:40:72
Elapsed Time: 0 days 22 hours 12 minutes 19 seconds
-----
ETHERNET STATISTICS (ent2) :
Device Type: 2-Port 10/100/1000 Base-TX PCI-X Adapter (14108902)
Hardware Address: 00:21:5e:8a:40:72

IEEE 802.3ad Port Statistics:
-----

Actor State:
  LACP activity: Active
  LACP timeout: Long
  Aggregation: Aggregatable
  Synchronization: IN_SYNC
  Collecting: Enabled
  Distributing: Enabled
  Defaulted: False
```

Expired: **False**

Partner State:

LACP activity: Active  
LACP timeout: Long  
Aggregation: Aggregatable  
Synchronization: IN\_SYNC  
Collecting: Enabled  
Distributing: **Enabled**  
Defaulted: False  
Expired: **False**

-----  
ETHERNET STATISTICS (ent4) :

Device Type: 2-Port 10/100/1000 Base-TX PCI-X Adapter (14108902)  
Hardware Address: 00:21:5e:8a:40:72  
-----

IEEE 802.3ad Port Statistics:  
-----

Actor State:

LACP activity: Active  
LACP timeout: Long  
Aggregation: Aggregatable  
Synchronization: IN\_SYNC  
Collecting: Enabled  
Distributing: **Disabled**  
Defaulted: False  
Expired: **True**

Partner State:

LACP activity: Active  
LACP timeout: Long  
Aggregation: Aggregatable  
Synchronization: **OUT\_OF\_SYNC**  
Collecting: Enabled  
Distributing: Enabled  
Defaulted: False  
Expired: False

#  
-----

In Example 7-5 on page 254, the Actor State for the ent4 logical Ethernet device shows the Distributing state as Disabled and the Expired state as True. The Synchronization state is IN\_SYNC.

Additionally, in Example 7-5 on page 254 the Partner State for the ent4 logical Ethernet device shows the Distributing state as Enabled and the Expired state as False. The Synchronization state is OUT\_OF\_SYNC.

The ent2 logical Ethernet adapter status remains unchanged.

From this, the administrator can determine that the ent4 logical Ethernet adapter has disabled its LACPDU sending and has expired its state, as it has failed to receive three LACPDU response from the Ethernet switch port partner. In turn, the partner is now displayed as OUT\_OF\_SYNC, as the actor and partner are unable to agree upon their status.

Prior to the IEEE 802.3ad enhancement in AIX V7.1, the **entstat** output may not have reliably displayed the status for devices that do not report their *up/down* state, which could result in significant packet loss.

With the AIX v7.1 enhancement to IEEE 802.3ad Link Aggregation, the actor determines that the partner is not responding to three LACPDU packets and discontinues activity on that logical Ethernet adapter, until such time as it receives an LACPDU packet from the partner.

**Note:** In this example, the `interval` is set to `long` (90 seconds).

AIX v7.1 still supports *device up/down* status reporting, but if no *device down* status was reported, then the link status would be changed after 270 seconds.

The `interval` may be changed to `short`, which would reduce the link status change to 9 seconds, such changes should be tested to determine whether long or short interval is suitable for your specific environment.

It was determined that the loss of connectivity was due to a network change which resulted in the network cable connecting the ent4 logical Ethernet device to the Ethernet switch port being moved to another switch port that was not enabled. Once the cabling was reinstated, the administrator again checked ent6 pseudo Ethernet device with the **entstat -d** command.

**Note:** Due to the large amount of output displayed by the **entstat -d** command, only the fields relevant to this example have been shown.

*Example 7-6 The entstat -d ent6 output - Link Aggregation recovered and operational*

---

**# entstat -d ent6**

-----  
ETHERNET STATISTICS (ent6) :  
Device Type: IEEE 802.3ad Link Aggregation  
Hardware Address: 00:21:5e:8a:40:72  
Elapsed Time: 0 days 22 hours 33 minutes 50 seconds  
=====

ETHERNET STATISTICS (ent2) :  
Device Type: 2-Port 10/100/1000 Base-TX PCI-X Adapter (14108902)  
Hardware Address: 00:21:5e:8a:40:72

IEEE 802.3ad Port Statistics:  
-----

Actor State:

LACP activity: Active  
LACP timeout: Long  
Aggregation: Aggregatable  
Synchronization: IN\_SYNC  
Collecting: Enabled  
Distributing: Enabled  
Defaulted: False  
Expired: False

Partner State:

LACP activity: Active  
LACP timeout: Long  
Aggregation: Aggregatable  
Synchronization: IN\_SYNC  
Collecting: Enabled  
Distributing: Enabled  
Defaulted: False  
Expired: False

-----  
ETHERNET STATISTICS (ent4) :  
Device Type: 2-Port 10/100/1000 Base-TX PCI-X Adapter (14108902)  
Hardware Address: 00:21:5e:8a:40:72

IEEE 802.3ad Port Statistics:  
-----

Actor State:

```
LACP activity: Active
LACP timeout: Long
Aggregation: Aggregatable
Synchronization: IN_SYNC
Collecting: Enabled
Distributing: Enabled
Defaulted: False
Expired: False
```

Partner State:

```
LACP activity: Active
LACP timeout: Long
Aggregation: Aggregatable
Synchronization: IN_SYNC
Collecting: Enabled
Distributing: Enabled
Defaulted: False
Expired: False
```

#

---

In Example 7-6 on page 257 the Actor State for the ent4 logical Ethernet device once more shows the Distributing state as Enabled and the Expired state as False. The Synchronization state is IN\_SYNC.

Additionally, In Example 7-6 on page 257 the Partner State for the ent4 logical Ethernet device shows the Distributing state as Enabled and the Expired state as False. The Synchronization state is IN\_SYNC.

The ent2 logical Ethernet adapter status remains unchanged.

From this, the administrator can determine that the ent4 logical Ethernet adapter has received an LACPDU from the its Ethernet switch partner and enabled link state. The link state is now synchronized and the IEEE 802.3ad Link Aggregation again operating normally.

## 7.2 Removal of BIND 8 application code

Berkeley Internet Name Domain (BIND) is a widely used implementation of the Domain Name System (DNS) protocol. Since the general availability of AIX V6.1 Technology Level 2 in November 2008 AIX supports BIND 9 (version 9.4.1). In comparison to the previous version, BIND 8, the majority of the code was redesigned for BIND 9 to effectively exploit the underlying BIND architecture, to introduce many new features and in particularly to support the DNS Security



Extensions. The Internet System Consortium (ISC <http://www.isc.org>) maintains the BIND code and officially declared the end-of life for BIND 8 in August 2007. Ever since no code updates were implemented in BIND 8. Also, the ISC only provides support for security related issues to BIND version 9 or higher.

In consideration of the named facts AIX Version 7.1 only supports BIND version 9 and the BIND 8 application code has been removed from the AIX V7.1 code base and is no longer provided on the product media. However, the complete BIND 8 library code in `/usr/ccs/lib/libbind.a` is retained since many AIX applications are using the provided functionality.

As consequence of the BIND 8 application code removal the following application programs are no longer available with AIX 7:

- ▶ `/usr/sbin/named8`
- ▶ `/usr/sbin/named8-xfer`

On an AIX 7 system the symbolic links of the `named` daemon is defined to point to the BIND 9 application which provides the server function for the Domain Name Protocol:

```
# cd /usr/sbin
# ls -l named
lrwxrwxrwx 1 root system 16 Aug 19 21:23 named -> /usr/sbin/named9
```

In previous AIX releases `/usr/sbin/named-xfer` is linked to the `/usr/sbin/named8-xfer` BIND 8 binary but because there is no equivalent program in BIND 9 the symbolic link `/usr/sbin/named-xfer` does no longer exist on AIX 7 systems.

## 7.3 Network Time Protocol version 4

The Network Time Protocol (NTP) is an Internet protocol used to synchronize the clocks of computers to some time reference, usually the Coordinated Universal Time (UTC). NTP is an Internet standard protocol originally developed by Professor David L. Mills at the University of Delaware.

The NTP version 3 (NTPv3) internet draft standard is formalized in the Request for Comments (RFC) 1305 (Network Time Protocol (Version 3) Specification, Implementation and Analysis). NTP version 4 (NTPv4) is a significant revision of the NTP standard, and is the current development version. NTPv4 has not been formalized but is described in the proposed standard RFC 5905 (Network Time Protocol Version 4: Protocol and Algorithms Specification).

The NTP subnet operates with a hierarchy of levels, where each level is assigned a number called the stratum. Stratum 1 (primary) servers at the lowest level are directly synchronized to national time services. Stratum 2 (secondary) servers at the next higher level are synchronize to stratum 1 servers and so on. Normally, NTP clients and servers with a relatively small number of clients do not synchronize to public primary servers. There are several hundred public secondary servers operating at higher strata and are the preferred choice.

According to a 1999 survey<sup>1</sup> of the NTP network there were at least 175,000 hosts running NTP in the internet. Among these there were over 300 valid stratum 1 servers. In addition there were over 20,000 servers at stratum 2, and over 80,000 servers at stratum 3.

Beginning with AIX V7.1 and AIX V6.1 TL 6100-06 the AIX operating system supports NTP version 4 in addition to the older NTP version 3. The AIX NTPv4 implementation is based on the port of the ntp-4.2.4 version of the Internet Systems Consortium (ISC) code and is in full compliance with RFC 2030 (Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI).

Additional information about the Network Time Protocol project, the Internet Systems Consortium, and the Request for Comments can be found at:

- ▶ <http://www.ntp.org/>
- ▶ <http://www.isc.org/>
- ▶ <http://www.rfcs.org/>

As in previous AIX releases the NTPv3 code is included with the bos.net.tcp.client fileset which is provided on the AIX product media and installed by default. The new NTPv4 functionality is delivered through the ntp.rte and the ntp.man.en\_US filesets of the AIX Expansion Pack.

The ntp.rte fileset for the NTP runtime environment installs the following NTPv4 programs under the /usr/sbin/ntp4 directory:

<b>ntptrace4</b>	Perl script that traces a chain of NTP hosts back to their master time source.
<b>sntp4</b>	SNTP client which queries a NTP server and displays the offset time of the system clock with respect to the server clock.
<b>ntpq4</b>	Standard NTP query program.
<b>ntp-keygen4</b>	Command which generates public and private keys.
<b>ntpd4</b>	Special NTP query program.

---

<sup>1</sup> Source: *A Survey of the NTP Network*, found at <http://alumni.media.mit.edu/~nelson/research/ntp-survey99>

**ntpdate4** Sets the date and time using the NTPv4 protocol.  
**ntpd4** NTPv4 daemon.

System administrators can use the **ls1pp** command to get a full listing of the ntp.rte fileset content:

```
75011p01:sbin/ntp4> ls1pp -f ntp.rte
  Fileset          File
```

```
-----
Path: /usr/lib/objrepos
ntp.rte 6.1.6.0      /usr/lib/nls/msg/en_US/ntpdate4.cat
                  /usr/lib/nls/msg/en_US/ntp4.cat
                  /usr/sbin/ntp4/ntp4trace4
                  /usr/sbin/ntp4/sntp4
                  /usr/sbin/ntp4/ntp4q4
                  /usr/sbin/ntp4/ntp-keygen4
                  /usr/sbin/ntp4/ntpdc4
                  /usr/sbin/ntp4/ntpdate4
                  /usr/lib/nls/msg/en_US/ntpdc4.cat
                  /usr/lib/nls/msg/en_US/ntp4.cat
                  /usr/sbin/ntp4
                  /usr/lib/nls/msg/en_US/libntp4.cat
                  /usr/sbin/ntp4/ntpd4
```

The NTPv3 and NTPv4 binaries can coexist on an AIX system. The NTPv3 functionality is installed by default through the bos.net.tcp.client fileset and the commands are placed in the /usr/sbin/ntp3 subdirectory. During the installation process a set of default symbolic links are created in the /usr/sbin directory to map the NTP commands to the NTPv3 binaries. Consequently AIX points by default to NTPv3 binaries.

If the system administrator likes to use the NTPv4 services the default symbolic links have to be changed manually to point to the appropriate commands under the /usr/sbin/ntp4 directory after the NTPv4 code has been installed from the AIX Expansion Pack. Table 7-2 provides a list of the NTPv4 binaries, the NTPv3 binaries, and the default symbolic links on AIX.

Table 7-2 NTP binaries directory mapping on AIX

NTPv4 binaries in /usr/sbin/ntp4	NTPv3 binaries in /usr/sbin/ntp3	Default symbolic links to NTPv3 binaries from /usr/sbin directory
ntpd4	xntpd	/usr/sbin/xntpd --> /usr/sbin/ntp3/xntpd
ntpdate4	ntpdate	/usr/sbin/ntpdate --> /usr/sbin/ntp3/ntpdate

NTPv4 binaries in /usr/sbin/ntp4	NTPv3 binaries in /usr/sbin/ntp3	Default symbolic links to NTPv3 binaries from /usr/sbin directory
ntpd4	xntpd	/usr/sbin/ntpd --> /usr/sbin/ntp3/xntpd
ntpq4	ntpq	/usr/sbin/ntpq --> /usr/sbin/ntp3/ntpq
ntp-keygen4	Not available	/usr/sbin/ntp-keygen --> /usr/sbin/ntp4/ntp-keygen4
ntptrace4	ntptrace	/usr/sbin/ntptrace --> /usr/sbin/ntp3/ntptrace
sntp4	sntp	/usr/sbin/sntp --> /usr/sbin/ntp3/sntp

In comparison with the NTPv3 protocol the utilization of NTPv4 offers improved functionality, and many new features and refinements. A comprehensive list which summarizes the differences between the NTPv4 and the NTPv3 version is provided by the *NTP Version 4 Release Notes* which can be found at:

<http://www.eecis.udel.edu/~mills/ntp/html/release.html>

The following list is an extract of the release notes which gives an overview of the new features pertaining to AIX.

1. Support for the IPv6 addressing family. If the Basic Socket Interface Extensions for IPv6 (RFC 2553) is detected, support for the IPv6 address family is generated in addition to the default support for the IPv4 address family.
2. Most calculations are now done using 64-bit floating double format, rather than 64-bit fixed point format. The motivation for this is to reduce size, improve speed and avoid messy bounds checking.
3. The clock discipline algorithm has been redesigned to improve accuracy, reduce the impact of network jitter and allow increased in poll intervals to 36 hours with only moderate sacrifice in accuracy.
4. The clock selection algorithm has been redesigned to reduce “clockhopping” when the choice of servers changes frequently as the result of comparatively insignificant quality changes.
5. This release includes support for Autokey public-key cryptography, which is the preferred scheme for authenticating servers to clients. [...]
6. The OpenSSL cryptographic library has replaced the library formerly available from RSA Laboratories. All cryptographic routines except a version of the MD5 message digest routine have been removed from the base distribution.
7. NTPv4 includes three new server discovery schemes, which in most applications can avoid per-host configuration altogether. Two of these are

- based on IP multicast technology, while the remaining one is based on crafted DNS lookups. [...]
8. This release includes comprehensive packet rate management tools to help reduce the level of spurious network traffic and protect the busiest servers from overload. [...]
  9. This release includes support for the orphan mode, which replaces the local clock driver for most configurations. Orphan mode provides an automatic, subnet-wide synchronization feature with multiple sources. It can be used in isolated networks or in Internet subnets where the servers or Internet connection have failed. [...]
  10. There are two new burst mode features available where special conditions apply. One of these is enabled by the **iburst** keyword in the server configuration command. It is intended for cases where it is important to set the clock quickly when an association is first mobilized. The other is enabled by the **burst** keyword in the server configuration command. It is intended for cases where the network attachment requires an initial calling or training procedure. [...]
  11. The reference clock driver interface is smaller, more rational and more accurate.
  12. In all except a very few cases, all timing intervals are randomized, so that the tendency for NTPv3 to self-synchronize and bunch messages, especially with a large number of configured associations, is minimized.
  13. Several new options have been added for the **ntpd** command line. For the inveterate knob twiddlers several of the more important performance variables can be changed to fit actual or perceived special conditions. In particular, the **tos** and **tos** commands can be used to adjust thresholds, throw switches and change limits.
  14. The **ntpd** daemon can be operated in a one-time mode similar to **ntpdate**, which program is headed for retirement. [...]

## 7.4 Reliable Datagram Sockets (RDS) v3 for RDMA support

RDS (Reliable Datagram Sockets) is a transport-layer networking protocol that bypasses the TCP/IP stack. RDSv3 refers to version 3 of this protocol, which includes enhancements to support RDMA exploitation from applications communicating through RDS sockets. OpenFabrics Enterprise Distribution (OFED) is the name of the open-source Linux-based reference implementation of the OpenFabrics Alliance software stack (also known as OFA Linux software

stack). This is a networking software stack designed to support server and storage clustering and grid connectivity using RDMA-based InfiniBand and iWARP (R-NIC) fabrics. It is optimized for high performance using RDMA and transport-offload technologies available in high-speed networks adapters.

- ▶ The OFED RDSv3 implementation for AIX support communication between AIX machines only.
- ▶ Both the InfiniBand and loopback transports are supported in OFED RDSv3 for AIX.
- ▶ OFED RDSv3 is delivered as a 64-bit-only kernel extension. However, both 64-bit and 32-bit user-space applications are supported.
- ▶ OFED RDSv3 is supported only for the InfiniBand Dual Port 4X IB GALAXY-2 adapters and future InfiniBand adapters available for IBM Power Systems. This support will rely on the ability of the AIX InfiniBand software stack to support these adapters.
- ▶ Coexistence of RDSv3 nodes and AIX RDSv2 nodes in the same cluster will not be supported. The reason for this is that applications running on top of RDSv3 cannot communicate with applications running on top of AIX RDSv2, as the over-wire protocols of AIX RDSv2 and RDSv3 are not compatible.
- ▶ Currently, only IPv4 addresses are supported by RDSv3 on AIX due to IPv6 addresses not supported by the OFED RDSv3 protocol.



# 8

## Security, authentication, and authorization

This chapter is dedicated to the latest security topics as they apply to AIX V7.1. Topics include:

- ▶ 8.1, “Domain Role Based Access Control” on page 266
- ▶ 8.2, “Auditing enhancements” on page 321
- ▶ 8.3, “Propolice or Stack Smashing Protection” on page 328
- ▶ 8.4, “Security enhancements” on page 329
- ▶ 8.5, “Remote Statistic Interface (Rsi) client firewall support” on page 336
- ▶ 8.6, “AIX LDAP authentication enhancements” on page 337
- ▶ 8.7, “RealSecure Server Sensor” on page 338

## 8.1 Domain Role Based Access Control

The section will discuss domain Role Based Access Control (RBAC).

This feature first became available in AIX V7.1 and is included in AIX 6.1 TL 06.

Domain RBAC is an enhancement to Enhanced Role Based Access Control, introduced in AIX V6.1.

### 8.1.1 The traditional approach to AIX security

The traditional approach to privileged administration in the AIX operating system has relied on a single system administrator account, named the root user.

The root user account is the superuser, as the root user account has the authority to perform all privileged system administration on the AIX system.

Using the root user, the administrator could perform day to day activities including, but not limited to, adding user accounts, setting users passwords, removing files, and maintaining system log files.

Reliance on a single superuser for all aspects of system administration raises issues in regards to the separation of administrative duties.

The root user allows the administrator to have a single point of administration when managing the AIX operating system, but in turn allows an individual to have unrestricted access to the operating system and its resources. While this freedom could be a benefit in day to day administration it also has the potential to introduce security exposures.

While a single administrative account may be acceptable in certain business environments, some environments use multiple administrators, each with responsibility for performing different tasks.

Alternatively, in some environments, the superuser role is shared among two or more system administrators. This shared administrative approach may breach business audit guidelines in an environment that requires that all privileged system administration is attributable to a single individual.

Sharing administration functions may create issues from a security perspective.

With each administrator having access to the root user, there was no way to limit the operations that any given administrator could perform.



Since the root user is the most privileged user, the root user could perform operations and also be able to erase any audit log entries designed to keep track of these activities, thereby making the identification to an individual of the administrative actions impossible.

Additionally, if the access to the root user's password were compromised and an unauthorized individual accesses the root user, then that individual could cause significant damage to the systems' integrity.

Role Based Access Control offers the option to define roles to users to perform privileged commands based upon the user's needs.

## 8.1.2 Enhanced and Legacy Role Based Access Control

In this section we will discuss the differences between the two operating modes of RBAC available in AIX, Legacy mode and Enhanced mode.

The release of AIX V6.1 saw the introduction of an enhanced version of Role Based Access Control (RBAC), which added to the version of RBAC already available in AIX since V4.2.1.

To distinguish between the two versions, the following naming conventions are used:

Enhanced RBAC	The enhanced version of RBAC introduced in AIX V6.1
Legacy RBAC	The version of RBAC introduced in AIX V4.2.1

The following is a brief overview of Legacy RBAC and Enhanced RBAC

For more information on Role Based Access Control see *AIX V6 Advanced Security Features Introduction and Configuration*, SG24-7430.

<http://www.redbooks.ibm.com/abstracts/sg247430.html?Open>

### Legacy RBAC

Legacy RBAC was introduced in AIX V4.2.1. The AIX security infrastructure began to provide the administrator with the ability to allow a user account other than the root user to perform certain privileged system administration tasks.

Legacy RBAC often requires that the command being controlled by an authorization have *setuid* to the root user in order for an authorized invoker to have the proper privileges to accomplish the operation.

The Legacy RBAC implementation introduced a pre-defined set of authorizations that can be used to determine access to administrative commands and could be expanded by the administrator.

Legacy RBAC includes a framework of administrative commands and interfaces to create roles, assign authorizations to roles, and assign roles to users.

The functionality of Legacy RBAC was limited as:

- ▶ The framework requires changes to commands/applications for them to be RBAC enabled.
- ▶ The predefined authorizations are not granular.
- ▶ Users often required membership in a certain group as well as having a role with a given authorization in order to execute a command.
- ▶ A true separation of duties is difficult to implement. If a user account is assigned multiple roles, then all assigned roles are always active. There is no method to activate only a single role without activating all roles assigned to a user.
- ▶ The least privilege principle is not adopted in the operating system. Privileged commands must typically be *setuid* to the root user.

## Enhanced RBAC

Beginning with AIX V6.1, Enhanced RBAC provides administrators with a method to delegate roles and responsibilities among one or more general user accounts.

These general user accounts may then perform tasks that would traditionally be performed by the root user or through the use of *setuid* or *setgid*.

The Enhanced RBAC integration options use granular privileges and authorizations and give the administrator the ability to configure any command on the system as a privileged command.

Enhanced RBAC allows the administrator to provide for a customized set of authorizations, roles, privileged commands, devices, and files through the Enhanced RBAC security database.

The Enhanced RBAC security database may reside either in the local file system or be managed remotely through LDAP.

Enhanced RBAC consists of the following security database files:

- ▶ Authorization Database
- ▶ Role Database

- ▶ Privileged Command Database
- ▶ Privileged Device Database
- ▶ Privileged File Database

Enhanced RBAC includes a granular set of system-defined authorizations and allows an administrator to create additional user-defined authorizations as necessary.

Both Legacy RBAC and Enhanced RBAC are supported on AIX V7.1.

Enhanced RBAC is enabled by default in AIX V7.1, but will not be active until the administrator configures the Enhanced RBAC functions.

Role Based Access Control may be configured to operate in either Legacy or Enhanced mode.

There is no specific install package in AIX V7.1 for Legacy or Enhanced mode RBAC as the majority of the Enhanced RBAC commands are included in the `bos.rte.security` file set.

While Legacy RBAC is supported in AIX v7.1, administrators are encouraged to use Enhanced RBAC over Legacy RBAC.

Enhanced RBAC offers more granular control of authorizations and reduces the reliance upon *setuid* programs.

### 8.1.3 Domain Role Based Access Control

As discussed earlier, Enhanced RBAC provides administrators with a method to delegate roles and responsibilities to a non-root user, but Enhanced RBAC cannot provide the administrator with a mechanism to further limit those authorized users to specific system resources.

As an example, Enhanced RBAC could be used to authorize a non-root user to use the `crfs` command to extend the size of a JFS2 file system. After authorizing the non-root user, Enhanced RBAC could not limit the authorized non-root user to using the `crfs` command to extend only an individual or selected file systems.

Domain RBAC introduces into Role Based Access Control the domain, a feature which allows the administrator to further restrict an authorized user to a specific resource.

With the introduction of Enhanced RBAC in AIX V6.1 the administrator was offered a granular approach to managing roles and responsibilities.

With the introduction of Domain RBAC, the granularity is further extended to allow finer control over resources.

Domain RBAC requires that Enhanced RBAC be enabled. Domain RBAC will not operate within the Legacy RBAC framework.

**Note:** Unless noted, further references to RBAC will refer to Enhanced RBAC, as domain RBAC does not operate under Legacy RBAC.

Example 8-1 shows the `lsattr` command being used to determine whether Enhanced RBAC is enabled on an AIX V7.1 partition. The `enhanced_RBAC true` attribute shows that enhanced RBAC is enabled.

*Example 8-1 Using the `lsattr` command to display the `enhanced_RBAC` status*

---

```
# oslevel -s
7100-00-00-0000
# lsattr -El sys0 -a enhanced_RBAC
enhanced_RBAC true Enhanced RBAC Mode True
#
```

---

The `enhanced_RBAC` attribute may be enabled or disabled with the `chdev` command. If enhanced RBAC is not enabled on your partition it may be enabled by using the `chdev` command to change the `sys0` device.

Example 8-2 shows the `chdev` command being used to change the `enhanced_RBAC` attribute from `false` to `true`

**Note:** Changing the `enhanced_RBAC` attribute will require a reboot of AIX for the change to take effect.

*Example 8-2 Using the `chdev` command to enable the `enhanced_RBAC` attribute*

---

```
# lsattr -El sys0 -a enhanced_RBAC
enhanced_RBAC false Enhanced RBAC Mode True
# chdev -l sys0 -a enhanced_RBAC=true
sys0 changed
# lsattr -El sys0 -a enhanced_RBAC
enhanced_RBAC true Enhanced RBAC Mode True
# shutdown -Fr
```

```
SHUTDOWN PROGRAM
Thu Sep 16 11:00:50 EDT 2010
Stopping The LWI Nonstop Profile...
```

Stopped The LWI Nonstop Profile.  
0513-044 The sshd Subsystem was requested to stop.

Wait for 'Rebooting...' before stopping.  
Error reporting has stopped.

---

At the time of publication, Domain RBAC functionality is not available on Workload Partition (WPAR).

## Domain RBAC definitions

Domain RBAC introduces new concepts into the RBAC security framework.

Subject	A <i>subject</i> is defined as an entity which requires access to another entity. A subject is an initiator of an action. An example of a subject would be a process accessing a file. When the process access the file, the process is considered a subject. A user account may also be a subject when the user account has been granted association with a domain.
Object	An <i>object</i> is an entity which holds information that can be accessed by another entity. The object is typically accessed by a <i>subject</i> and is typically the target of the action. The object may be thought of as the entity on which the action is being performed. As an example, when a process 2001 tries to access another process, 2011 to send a signal then process 2001 is the subject and process 2011 is the object.
Domain	A <i>domain</i> is defined as a category to which an entity may belong. When an entity belongs to a domain, access control to the entity is governed by a rule set which is known as a <i>property</i> . An entity could belong to more than one domain at a time. Each domain has a unique numeric domain identifier. A maximum of 1024 domains are allowed, with the highest possible value of the domain identifier allowed as the number 1024. A user account may belong to a domain. When a user account belongs to a domain, the user account can be described as having an association with a domain.
Property	A property is the rule set that determines whether a subject is granted access to an object.
Conflict Set	A <i>conflict set</i> is a domain object attribute that restricts access to a domain based upon the existing domain(s)

access that an entity may already have defined. This will be further explained when discussing the `setsecattr` command, later in the section.

### Security Flags

A *security flag* is a domain object attribute that may restrict access to an object based upon the `FSF_DOM_ANY` or `FSF_DOM_ALL` attribute. When the `secflags` attribute is set to `FSF_DOM_ANY` a *subject* may access the *object* when it is associated with any of the domains specified in the `domains` attribute. When the `secflags` attribute is `FSF_DOM_ALL` a *subject* may access the *object* only when it is associated with all of the domains specified in the attribute. The default `secflags` value is `FSF_DOM_ALL`. If no `secflags` attribute value is specified, then the default value of `FSF_DOM_ALL` will be used.

In Example 8-3 we see the `ps` command being used to display the process identifier assigned to a the `vi` command. The `vi` command is being used by the root user to edit a file named `/tmp/myfile`.

*Example 8-3 Using the ps command to identify the process editing /tmp/myfile*

---

```
# cd /tmp
# pwd
/tmp
# ls -ltra myfile
-rw-r--r-- 1 root      system      15 Sep 02 11:58 myfile
# ps -ef|grep myfile
  root 6226020 6488264  0 11:59:42 pts/1  0:00 vi myfile
# ps -ft 6226020
  UID  PID  PPID  C  STIME  TTY  TIME  CMD
  root 6226020 6488264  0 11:59:42 pts/1  0:00 vi myfile
#
```

---

In Example 8-3 we see an example of the *subject* and the *object*.

- ▶ The *subject* is process id 6226020 which is a process that is executing the `vi` command to edit the file named `/tmp/myfile`
- ▶ The *object* is the file named `/tmp/myfile`

## 8.1.4 Domain RBAC command structure

Domain RBAC introduces four new commands into the RBAC framework.

These are the `mkdom`, `lsdom`, `chdom` and `rmdom` commands.

## The **mkdom** command

The **mkdom** command creates a new RBAC domain.

The syntax of the **mkdom** command is:

```
mkdom [ Attribute = Value ...] Name
```

The **mkdom** command creates a new domain in the domain database. The domain attributes can be set during the domain creation phase by using the `Attribute = Value` parameter.

The domain database is located in the `/etc/security/domains` file.

The **mkdom** command has the following requirements:

- ▶ The system must be operating in the Enhanced Role Based Access Control (RBAC) mode
- ▶ Modifications made to the domain database are not available for use until updated into the Kernel Security Tables (KST) with the **setkst** command.
- ▶ The **mkdom** command is a privileged command. Users of this command must have activated a role with the `aix.security.domains.create` authorization or be the root user.

Example 8-4 shows the **mkdom** command being used from the root user to create a new domain named `Network` with a domain identifier (Domain ID) of `22`:

*Example 8-4 Using the **mkdom** command to create the domain `Network` with a Domain ID of `22`*

---

```
# mkdom id=22 Network
# lsdom Network
Network id=22
#
```

---

**Note:** The **mkdom** command will not return with text output when a domain is successfully created. The **lsdom** command was used in Example 8-4 to display that the **mkdom** command did successfully create the `Network` domain. The **lsdom** command is introduced next.

The **mkdom** command contains character usage restrictions. For a full listing of these character restrictions see the **mkdom** command reference.

## The **lsdom** command

The **lsdom** command displays the domain attributes of an RBAC domain.

The domain database is located in the `/etc/security/domains` file.

The syntax of the `lsdom` command is:

```
lsdom [ -C ] [ -f ] [ -a Attr [Attr]... ] { ALL | Name [ , Name ] ... }
```

The `lsdom` command lists the attributes of either all domains or specific domains.

The `lsdom` command will list all domain attributes. To view selected attributes, use the `lsdom -a` command option.

The `lsdom` command can list the domain attributes in the following formats:

- ▶ List domain attributes in one line with the attribute information displayed as `Attribute = Value`, each separated by a blank space. This is the default list option.
- ▶ To list the domain attributes in stanza format, use the `lsdom -f` command flag.
- ▶ To list the information as colon-separated records, use the `lsdom -C` command flag

The `lsdom` command has the following domain name specification available:

ALL	Indicates that all domains will be listed, including the domain attributes
Name	Indicates the name of the domain which will have the attributes listed. This may be multiple domain names, comma separated.

The `lsdom` command has the following requirements:

- ▶ The system must be operating in the Enhanced Role Based Access Control (RBAC) mode
- ▶ The `lsdom` command is a privileged command. Users of this command must have activated a role with the `aix.security.domains.list` authorization or be the root user.

Example 8-5 shows the `lsdom -f` command being used by the root user to display the DBA and HR domains in stanza format:

*Example 8-5 Using the `lsdom` command `-f` to display the DBA and HR domains in stanza format*

---

```
# lsdom -f DBA,HR
```

```
DBA:
```

```
    id=1
```

```
HR:
```



```
id=2
```

```
#
```

---

## The **chdom** command

The **chdom** command modifies attributes of an existing RBAC domain.

The syntax of the **chdom** command is:

```
chdom Attribute = Value ... Name
```

If the specified attribute or attribute value is invalid, the **chdom** command does not modify the domain.

The **chdom** command has the following requirements:

- ▶ The system must be operating in Enhanced Role Based Access Control (RBAC) mode
- ▶ Modifications made to the domain database are not available for use until updated into the Kernel Security Tables with the **setkst** command
- ▶ The **chdom** command is a privileged command. Users of this command must have activated a role with the *aix.security.dom.change* authorization or be the root user

Example 8-6 shows the **chdom** command being used by the root user to change the ID of the Network domain from 22 to 20. The Network domain was created in Example 8-4 on page 273 and has not yet been used and is not associated with any entities.

*Example 8-6 Using the chdom command to change the ID attribute of the Network domain*

---

```
# lsdom -f Network
```

```
Network:
```

```
id=22
```

```
# chdom id=20 Network
```

```
# lsdom -f Network
```

```
Network:
```

```
id=20
```

```
#
```

---

**Note:** Modification of the ID attribute of a domain can affect the security aspects of the system, as processes and files might be using the current value of the ID.

It is recommended to modify the ID of a domain only if the domain has not been used, else the security aspects of the system could be adversely effected.

## The `rmdom` command

The `rmdom` command removes an RBAC domain.

The syntax of the `rmdom` command is:

```
rmdom Name
```

The `rmdom` command removes the domain that is identified by the Name parameter.

The `rmdom` command only removes the existing domains from the domain database.

A domain that is referenced by the domain object database cannot be removed until you remove the references to the domain.

The `rmdom` command has the following requirements:

- ▶ The system must be operating in Enhanced Role Based Access Control (RBAC) mode
- ▶ Modifications made to the domain database are not available for use until updated into the Kernel Security Tables with the `setkst` command
- ▶ The `rmdom` command is a privileged command. Users of this command must have activated a role with the `aix.security.dom.remove` authorization or be the root user

Example 8-7 shows the `rmdom` command being used by the root user to remove the Network domain. The Network domain has not yet been used and is not with any entities.

By using the `lssecattr -o ALL` command we can see that there are no domain objects referenced by the Network domain, so the Network domain may be removed.

*Example 8-7 Using the `rmdom` command to remove the Network domain*

---

```
# lsdom -f Network
Network:
```

id=22

```
# lssecattr -o ALL
/home/dba/privatefiles domains=DBA conflictsets=HR objtype=file
secflags=FSF_DOM_ANY
# rmdom Network
# lsdom -f Network
3004-733 Role "Network" does not exist.
# lsdom ALL
DBA id=1
HR id=2
#
```

**Note:** If a user account belonged to the Network domain, the user account would still see the domains=Network attribute listed from the **lsuser** output. This domains=Network attribute value can be removed with the **chuser** command.

In addition to the **mkdom**, **lsdom**, **chdom** and **rmdom** commands, domain RBAC introduces enhanced functionality to the existing commands in Table 8-1:

Table 8-1 Domain RBAC enhancements to existing commands

Command	Description	New Functionality
<b>setsecattr</b>	Add or modify the domain attributes for objects	-o
<b>lssecattr</b>	Display the domain attributes for objects	-o
<b>rmsecattr</b>	Remove domain object definitions	-o
<b>setkst</b>	Reads the security databases and loads the information from the databases into the kernel security tables	The option to download the domain and the domain object databases
<b>lsuser</b>	List user attributes	The attribute domain is added for users
<b>lssec</b>	List user attributes	The attribute domain is added for users
<b>chuser</b>	Change user attributes	The attribute domain is added for users

Command	Description	New Functionality
chsec	Change user attributes	The attribute domain is added for users

The Domain RBAC enhanced functionality to the commands in Table 8-1 on page 277 is further explained in the following examples.

### The setsecattr command

The **setsecattr** command includes the **-o** flag. The **setsecattr** command is used to add and modify domain attributes for objects. An example of the **setsecattr** command is shown in Example 8-8:

*Example 8-8 The setsecattr -o command*

---

```
# setsecattr -o domains=DBA conflictsets=HR objtype=file \
secflags=FSF_DOM_ANY /home/dba/privatefiles
#
```

---

As discussed earlier, domain RBAC introduces the *conflict set* and *security flag* object attributes into the RBAC framework.

The *conflict set* attribute can deny access to an object based upon existing domain association. When used, the `conflictsets` attribute would be set to a domain name other than the domain defined in the `domains` attribute.

In Example 8-8 the `conflictsets` attribute is defined as HR and the `domains` attribute as DBA. Both HR and DBA are names of domains defined in the RBAC security database.

Using the `conflictsets` attribute in this manner will restrict access to the `/home/dba/privatefiles` object by entities that have an association with the HR domain, regardless of whether these entities have membership to the DBA domain.

Example 8-9 shows the **lssecattr** and the **ls -ltra** command being used to display the attributes of the file named `/home/dba/privatefiles`.

*Example 8-9 Using the lssecattr and ls -ltra command to display the file named /home/dba/privatefiles*

---

```
# cd /home/dba
# lssecattr -o privatefiles
/home/dba/privatefiles domains=DBA conflictsets=HR \
objtype=file secflags=FSF_DOM_ANY
# ls -ltra /home/dba/privatefiles
```

```

-rw-r--r--    1 dba      staff          33 Sep 03 11:18 privatefiles
# lssec -f /etc/security/user -s dba -a domains
dba domains=DBA
# lssecattr -o /home/dba/example111
"/home/dba/example111" does not exist in the domained object database.
#

```

---

From the output in Example 8-9 on page 278 we can determine that:

- ▶ The **lssecattr** command shows that the file named `/home/dba/privatefiles` is defined as a domain RBAC object. If the file was not defined as a domain RBAC object then the output returned would be similar to the response from the **lssecattr -o /home/dba/example111** command which returned `"/home/dba/example111" does not exist in the domained object database.`
- ▶ The **lssecattr** command shows that the `domains` attribute is defined as the DBA domain and the `conflictsets` attribute is defined as the HR domain.
- ▶ The **lssecattr** command shows the `secflags=FSF_DOM_ANY`. In this example the `FSF_DOM_ANY` does not offer any further restriction because the domain RBAC object `/home/dba/privatefiles` is defined with only a single domain.
- ▶ The **ls -ltr** command shows that the `dba` user account has read and write access to the file named `/home/dba/privatefiles` through Discretionary Access Control (DAC).
- ▶ The **lssec** command shows that the `dba` user account has been granted association to the DBA domain but has not been granted association to the HR domain, as only the DBA domain is returned in the `domains=DBA` listing.

By using the combination of `conflictsets` and `domains` in Example 8-9 on page 278 the `dba` user account would be able to access the file named `/home/dba/privatefiles`.

If the `dba` user account was to be granted association to the HR domain, then the `dba` user account would no longer be able to access the file named `/home/dba/privatefiles` because the HR domain is defined as a *conflict set* to the domain RBAC object `/home/dba/privatefiles`.

The access to the file named `/home/dba/privatefiles` would be refused even though the `dba` user has read and write access to the file via DAC.

The `secflags=FSF_DOM_ANY` attribute sets the behavior of the `domains` attribute of the object. In Example 8-9 on page 278 the object `/home/dba/privatefiles` is defined with only the DBA domain.

If the object `/home/dba/privatefiles` had been defined to multiple domains, and the `secflags` attribute been set as `FSF_DOM_ALL`, then the `dba` user account would have to be associated with all domains defined in the `domains` attribute for the `/home/dba/privatefiles` object, else access to the `/home/dba/privatefiles` would be denied.

### The `lssecattr` command

The `lssecattr` command now includes the `-o` flag. The `lssecattr` command is used to display the domain attributes for *objects*. An example of the `lssecattr` command is shown in Example 8-10:

*Example 8-10 The `lssecattr -o` command*

---

```
# lssecattr -o /home/dba/privatefiles
/home/dba/privatefiles domains=DBA conflictsets=HR objtype=file \
secflags=FSF_DOM_ANY
#
```

---

### The `rmsecattr` command

The `rmsecattr` command now includes the `-o` flag. The `rmsecattr` command is used to remove domain object definitions from the RBAC security database. An example of the `rmsecattr` command is shown in Example 8-11:

*Example 8-11 The `rmsecattr -o` command*

---

```
# rmsecattr -o /home/dba/privatefiles
#
```

---

### The `setkst` command

The `setkst` command is used to read the security database and load the security databases into the kernel security tables (KST).

The `setkst` command includes the option to load the domain and the domain object database.

The domain and domain object database are located in the `/etc/security` directory in the following files:

The <code>domains</code> file	The domain security database. To update the domain security database into the KST, use the <code>setkst -t dom</code> command
The <code>domobj</code> file	The domain object security database. To update the domain object security database into the KST, use the <code>setkst -t domobj</code> command

An example of the **setkst** command is shown in Example 8-12:

*Example 8-12 The setkst -t command updating the domain into the KST*

---

```
# setkst -t dom
Successfully updated the Kernel Domains Table.
#
```

---

**Note:** Changes made to the RBAC database are not activated into the Kernel Security Table (KST) until such time as the **setkst** command is executed.

## The lskst command

The **lskst** command lists the entries in the Kernel Security Tables (KST).

The **lskst** command includes the option to list the domain and the domain object database.

An example of the **lskst** command is show in Example 8-13

*Example 8-13 Listing the kernel security tables with the lskst -t command*

---

```
# lskst -t domobj
/home/dba/privatefiles objtype=FILE domains=DBA \
conflictsets=HR secflags=FSF_DOM_ANY
#
```

---

## The lsuser command

The **lsuser** command includes the option to display the domains to which a user has association. An example of the **lsuser** command is shown in Example 8-14:

*Example 8-14 The lsuser -a command - display a user domain access*

---

```
# lsuser -a domains dba
dba domains=DBA
#
```

---

## The lssec command

As with the **lsuser** command, the **lssec** command includes the option to display the domains to which a user has an association. An example of the **lssec** command is shown in Example 8-15:

*Example 8-15 The lssec -f command - display a user domain access*

---

```
# lssec -f /etc/security/user -s dba -a domains
dba domains=DBA
```

#  

---

## The chuser command

The **chuser** command includes the option to change the domains to which a user has an association. An example of the **chuser** command is shown in Example 8-16:

*Example 8-16 The chuser command - change a user domain association*

---

```
# lsuser -a domains dba
dba domains=DBA
# chuser domains=HR dba
# lsuser -a domains dba
dba domains=HR
#
```

---

To remove all domains to which a user has an association, the **chuser** command can be used without any domain attribute, as shown in Example 8-17:

*Example 8-17 The chuser command - remove all domain association from a user*

---

```
# lsuser -a domains dba
dba domains=HR
# chuser domains= dba
# lsuser -a domains dba
dba
# lssec -f /etc/security/user -s dba -a domains
dba domains=
#
```

---

Example 8-17 shows the different output returned by the **lssec -f** command and the **lsuser -a** command.

## The chsec command

As with the **chuser** command, the **chsec** command includes the option to change the domains to which a user has an association. An example of the **chsec** command is shown in Example 8-18:

*Example 8-18 The chsec command - adding DBA domain access to the dba user*

---

```
# lssec -f /etc/security/user -s dba -a domains
dba domains=
# chsec -f /etc/security/user -s dba -a domains=DBA
# lssec -f /etc/security/user -s dba -a domains
```



```
dba domains=DBA
#
```

---

## 8.1.5 LDAP support in Domain RBAC

The Enhanced RBAC security database may reside either in the local file system or be managed remotely through LDAP.

At the time of publication the domain RBAC databases must reside locally in the `/etc/security` directory.

When upgrading an LPAR that is using RBAC with LDAP authentication, the LDAP authentication will remain operational. Any domain RBAC definitions will reside locally in the `/etc/security` directory.

The `/etc/nscontrol.conf` file contains the location and lookup order for the RBAC security database.

Example 8-19 shows the RBAC security database stanza output of the `/etc/nscontrol.conf` file.

The `secorder` attribute describes the location of the security database file. It is possible to store the Enhanced RBAC security database files either in the `/etc/security` directory or on an LDAP server or a combination of the `/etc/security` directory and on an LDAP server.

Domain RBAC security database files are only stored in the `/etc/security` directory, so they will not have a stanza in the `/etc/nscontrol.conf` file.

The options for the `secorder` attribute are:

files	The database file is located in the <code>/etc/security</code> directory. This is the default location.
LDAP	The database file is located on an LDAP server
LDAP,files	The database file is located on the LDAP server and the <code>/etc/security</code> directory. The lookup order is LDAP first, followed by the <code>/etc/security</code> directory
files,LDAP	The database file is located in the <code>/etc/security</code> directory and the LDAP server. The lookup order is the <code>/etc/security</code> directory first, followed by the LDAP server

*Example 8-19 The `/etc/nscontrol.conf` file*

---

```
# more /etc/nscontrol.conf
# IBM_PROLOG_BEGIN_TAG
```

```

# This is an automatically generated prolog.
#
output omitted .....
#
authorizations:
    recorder = files

roles:
    recorder = files

privcmds:
    recorder = files

privdevs:
    recorder = files

privfiles:
    recorder = files
#

```

---

Example 8-19 on page 283 shows that the five files in the Enhanced RBAC security database are stored in the /etc/security directory and LDAP is not being used for RBAC on this server.

## 8.1.6 Scenarios

This section will introduce four scenarios to describe the usage of the new features available in domain RBAC.

The four scenarios consist of:

- |                  |   |
|------------------|---|
| Device scenario  | Using domain RBAC to control privileged command execution on logical volume devices.          |
| File scenario    | Two scenarios. Using domain RBAC to restrict user access and to remove user access to a file. |
| Network scenario | Use domain RBAC to restrict privileged access to a network interface.                         |

These four scenarios show examples of how domain RBAC may be used to provide additional functionality to the AIX security framework.

The AIX partition used in the scenario:

- ▶ Has AIX V7.1 installed

- ▶ Is operating in Enhanced\_RBAC mode
- ▶ Has no additional or customized RBAC roles or authorizations defined
- ▶ Has no previous domain RBAC customizing defined

**Note:** At the time of publication, Domain RBAC may be managed through the command line only. Domain RBAC support is not included in the System Management Interface Tool (SMIT).

## Device scenario

Domain RBAC allows the administrator to define devices as domain RBAC objects.

In this scenario, logical volume devices will be defined as domain RBAC objects.

The AIX V7.1 LPAR consists of two volume groups, rootvg and appsvg.

The appsvg contains application data, which is supported by the application support team by using the appuser user account.

The application support team have requested the ability to add/modify and delete the four file systems used by the application.

The application file systems reside exclusively in a volume group named appsvg.

The systems administrator will grant the application support team the ability to add/modify/delete the four application file systems in the appsvg volume group, but restrict add/modify/delete access to all other file systems on the LPAR.

Enhanced RBAC allows the systems administrator to grant the application support team the privileges to add/modify/delete the four file systems without having to grant access to the root user.

Enhanced RBAC does not allow the systems administrator to restrict access to only those four file systems needed by the application support team.

Domain RBAC will allow such a granular separation of devices and allow the systems administrator to allow add/modify/delete access to only the four application file systems and restrict add/modify/delete access to the remaining file systems.

The system administrator identifies that the application support team will require access to the following AIX privileged commands.

<b>crfs</b>	create a new file system
<b>chfs</b>	modify an existing file system

<b>rmfs</b>	remove an existing file system
<b>mount</b>	mount a file systems
<b>unmount</b>	unmount a file system

With the privileged commands identified, the administrator will define an RBAC role to allow the application support team to perform these five privileged commands.

Unless noted otherwise, all commands in the scenario will be run as the root user.

AIX includes pre-defined RBAC roles, one of which is the FSAdmin role. The FSAdmin role includes commands that may be used to manage file systems and could be used in this situation.

In this scenario the administrator will create a new RBAC role, named `apps_fs_manage`, using the `mkrole` command.

The benefits in creating the `apps_fs_manage` role are:

- ▶ This will introduce an example of using the `mkrole` command used in Enhanced RBAC
- ▶ The `apps_fs_manage` role will include only a subset of the privileged commands included in the FSAdmin role. This complies with the Least Privilege Principal

Before using the `mkrole` command to create the `apps_fs_manage` role, the administrator must determine the access authorizations required by each of the commands that will be included in the `apps_fs_manage` role.

The `lssecattr` command is used to determine the access authorizations.

Example 8-20 shows the `lssecattr` command being used to determine the access authorizations of each of the five privileged commands that will be included in the `apps_fs_manage` role.

*Example 8-20 Using the `lssecattr` command to identify command authorizations*

---

```
# id
uid=0(root) gid=0(system)
groups=2(bin),3(sys),7(security),8(cron),10(audit),11(lp)
# lssecattr -c -a accessauths /usr/sbin/crfs
/usr/sbin/crfs accessauths=aix.fs.manage.create
# lssecattr -c -a accessauths /usr/sbin/chfs
/usr/sbin/chfs accessauths=aix.fs.manage.change
# lssecattr -c -a accessauths /usr/sbin/rmfs
```

```

/usr/sbin/rmfs accessauths=aix.fs.manage.remove
# lssecattr -c -a accessauths /usr/sbin/mount
/usr/sbin/mount accessauths=aix.fs.manage.mount
# lssecattr -c -a accessauths /usr/sbin/umount
/usr/sbin/umount accessauths=aix.fs.manage.unmount
#

```

---

Example 8-20 on page 286 shows that the privileged commands require the following access authorizations:

<b>crfs</b> command	Requires the access authorization <code>aix.fs.manage.create</code>
<b>chfs</b> command	Requires the access authorization <code>aix.fs.manage.change</code>
<b>rmfs</b> command	Requires the access authorization <code>aix.fs.manage.remove</code>
<b>mount</b> command	Requires the access authorization <code>aix.fs.manage.mount</code>
<b>unmount</b> command	Requires the access authorization <code>aix.fs.manage.unmount</code>

At this stage, the administrator has identified the privileged commands required by the application support team, decided on the name of the RBAC role to be created and determined the access authorizations required for the five privileged commands.

The administrator may now create the `apps_fs_manage` RBAC role by using the **mkrole** command.

Example 8-21 shows the **mkrole** command being used to create the RBAC role named `apps_fs_manage`:

*Example 8-21 Using the **mkrole** command - create the `apps_fs_manage` role*

---

```

# id
uid=0(root) gid=0(system)
groups=2(bin),3(sys),7(security),8(cron),10(audit),11(lp)
# mkrole authorizations=aix.fs.manage.create,aix.fs.manage.change,/
aix.fs.manage.remove,/aix.fs.manage.mount,aix.fs.manage.unmount/
dfltsmsg='Manage apps filesystems' apps_fs_manage
# lsrole apps_fs_manage
apps_fs_manage authorizations=aix.fs.manage.create,aix.fs.manage.change,/
aix.fs.manage.remove,aix.fs.manage.mount,aix.fs.manage.unmount rolist=
groups= visibility=1 screens=* dfltsmsg=Manage apps filesystems msgcat=
auth_mode=INVOKER id=11
#

```

---

**Note:** The **smitty mkrole** fastpath may also be used to create an RBAC role. Due to the length of the authorization definitions, using the **smitty mkrole** fastpath may be convenient when multiple access authorizations are included in a role

Once the **apps\_fs\_manage** role has been created, the role must be updated into the Kernel Security Tables (KST) with the **setkst** command. The role will not be available for use until the **setkst** command updates the changes into the KST.

In Example 8-22 we see the **lsrole** command being used to list the **apps\_fs\_manage** role.

The **lsrole** command output shows that the **apps\_fs\_manage** role exists in the RBAC database, but when the **swrole** command is used to switch to the role, the role switching is not allowed.

This is because the **apps\_fs\_manage** role has not been updated into the KST.

The administrator can verify this by using the **lskst** command.

The **lskst** command will list the KST, whereas the **lsrole** command will list the contents of the RBAC security database in the `/etc/security` directory.

Example 8-22 shows the usage of the **lsrole**, **swrole** and **lskst** commands:

*Example 8-22 Using the lsrole, the swrole and the lstkst commands*

---

```
# lsrole apps_fs_manage
apps_fs_manage authorizations=aix.fs.manage.create,aix.fs.manage.change,/
aix.fs.manage.remove,aix.fs.manage.mount,aix.fs.manage.unmount rolist=
groups= visibility=1 screens=* dfltmsg=Manage apps filesystems msgcat=
auth_mode=INVOKER id=11
# swrole apps_fs_manage
swrole: 1420-050 apps_fs_manage is not a valid role.
# lstkst -t role apps_fs_manage
3004-733 Role "apps_fs_manage" does not exist.
#
```

---

In Example 8-23 on page 289 we use the **setkst** command to update the KST with the changes made to the RBAC security database.

The **setkst** command may be run without any options or with the **setkst -t** option.

The **setkst -t** command allows the KST to be updated with only a selected RBAC database table or tables.

Example 8-23 shows the **setkst -t** command being used to update the KST with only the RBAC role database information.

*Example 8-23 The setkst -t command - updating the role database into the KST*

---

```
# lskst -t role apps_fs_manage
3004-733 Role "apps_fs_manage" does not exist.
# setkst -t role
Successfully updated the Kernel Role Table.
# lskst -t role -f apps_fs_manage
apps_fs_manage:

authorizations=aix.fs.manage.change,aix.fs.manage.create,aix.fs.manage.mount,/
aix.fs.manage.remove,aix.fs.manage.unmount
    rolist=
    groups=
    visibility=1
    screens=*
    dfltmsg=Manage apps filesystems
    msgcat=
    auth_mode=INVOKER
    id=11

#
```

---

After updating the KST, the appuser account must be associated with the apps\_fs\_manage role.

It is recommended to first use the **lsuser** command to display whether any roles have previously been associated with the appuser account.

In this case, the appuser account has no role associations defined, as can be seen from the **lsuser** command output in Example 8-24.

If the appuser account had existing roles associated, the existing roles would need to be included in the **chuser** command along with the new apps\_fs\_manage role.

The **chuser** command is used in Example 8-24 to associate the appuser account with the apps\_fs\_manage role.

*Example 8-24 The lsuser and chuser commands - assigning the apps\_fs\_manage role to the appuser account with the chuser command*

---

```
# lsuser -a roles appuser
appuser roles=
# chuser roles=apps_fs_manage appuser
# lsuser -a roles appuser
```

```
appuser roles=apps_fs_manage
#
```

---

At this stage, the administrator has completed the steps required to grant the appuser account the ability to perform the **crfs**, **chfs**, **rmfs**, **mount** and **umount** commands. Even though these privileged commands could normally only be executed by the root user, the RBAC framework allows a non privileged user to execute these commands, once the appropriate access authorizations and role(s) have been created and associated.

To demonstrate this, the appuser account will perform the **chfs** and **umount** commands.

The Example 8-25 shows the appuser account login and uses the **rolelist** command to display to which RBAC roles it has an association with and whether the role is effective.

A role that is active on the user account is known as the effective role.

*Example 8-25 Using the **rolelist -a** and **rolelist -e** commands*

---

```
$ id
uid=301(appuser) gid=202(appgroup) groups=1(staff)
$ rolelist -a
apps_fs_manage  aix.fs.manage.change
                  aix.fs.manage.create
                  aix.fs.manage.mount
                  aix.fs.manage.remove
                  aix.fs.manage.umount
$ rolelist -e
rolelist: 1420-062 There is no active role set.
$
```

---

From the **rolelist -a** and **rolelist -e** output in the user can determine that the appuser has been associated with the **apps\_fs\_manage** role, but the role is not currently the effective role.

The user will use the **swrole** command to switch to the **apps\_fs\_manage** role.

Once the **swrole** command is used to switch to the **apps\_fs\_manage** role, the role will become the effective role, allowing the appuser account to perform the privileged commands defined in the **apps\_fs\_manage** role.

Example 8-26 on page 291 shows the appuser account using the **swrole** command to switch to the **apps\_fs\_manage** role.



*Example 8-26 The appuser account using the swrole command to switch to the apps\_fs\_manage role*

---

```
$ ps
  PID   TTY   TIME CMD
 7995462 pts/0  0:00 -ksh
 9633860 pts/0  0:00 ps
$ swrole apps_fs_manage
appuser's Password:
$ rolelist -e
apps_fs_manage  Manage apps filesystems
$ ps
  PID   TTY   TIME CMD
 7995462 pts/0  0:00 -ksh
 9044098 pts/0  0:00 ps
 9240642 pts/0  0:00 ksh
$
```

---

**Note:** The **swrole** command will require authentication with the user's password credentials.

The **swrole** command initiates a new shell, which can be seen with the new PID 940642, displayed in the **ps** command output.

The appuser account may now execute the privileged commands in the apps\_fs\_manage role.

In Example 8-27 the appuser account will use the **chfs** command to add 1 GB to the /apps04 file system.

*Example 8-27 The appuser account using the chfs command to add 1 GB to the /apps04 file system*

---

```
$ id
uid=301(appuser) gid=202(appgroup) groups=1(staff)
$ df -g /apps04
Filesystem      GB blocks      Free %Used      Iused %Iused Mounted on
/dev/appslv_04    1.25         0.18  86%          15      1% /apps04
$ chfs -a size=+1G /apps04
Filesystem size changed to 4718592
$ df -g /apps04
Filesystem      GB blocks      Free %Used      Iused %Iused Mounted on
/dev/appslv_04    2.25         1.18  48%          15      1% /apps04
$
```

---

The appuser was successful in using the **chfs** command to add 1 GB to the /apps04 file system.

The RBAC role is allowing the appuser account to execute the **chfs** command. This is the expected operation of the RBAC role.

In Example 8-28 the appuser account will use the **umount** command to unmount the /apps01 file system.

*Example 8-28 The appuser account using the umount command to unmount the /apps01 file system*

---

```

$ df -g /apps01
Filesystem      GB blocks      Free %Used      Iused %Iused Mounted on
/dev/apps1v_01  1.25          0.18  86%           15    1% /apps01
$ umount /apps01
$ df -g /apps01
Filesystem      GB blocks      Free %Used      Iused %Iused Mounted on
/dev/hd4        0.19          0.01  95%          9845  77% /
$ ls1v apps1v_01
LOGICAL VOLUME:      apps1v_01          VOLUME GROUP:      appsvg
LV IDENTIFIER:      00f61aa600004c000000012aee536a63.1 PERMISSION:
read/write
VG STATE:           active/complete    LV STATE:          closed/syncd
TYPE:               jfs2               WRITE VERIFY:      off
MAX LPs:            512                PP SIZE:           64
megabyte(s)
COPIES:             1                  SCHED POLICY:      parallel
LPs:                36                 PPs:               36
STALE PPs:          0                  BB POLICY:         relocatable
INTER-POLICY:       minimum            RELOCATABLE:       yes
INTRA-POLICY:       middle              UPPER BOUND:       32
MOUNT POINT:        /apps01            LABEL:             /apps01
MIRROR WRITE CONSISTENCY: on/ACTIVE
EACH LP COPY ON A SEPARATE PV ?: yes
Serialize IO ?:     NO
$

```

---

In Example 8-28, the appuser was successfully able to use the **umount** command to **umount** the /apps01 file system. By using the **df** command and the **ls1v** command, we can determine that the /apps01 file system has been unmounted.

The RBAC role is allowing the appuser account to execute the **umount** command. This is the expected operation of the RBAC role.

By using RBAC, the administrator has been able to grant the appuser account access to selected privileged commands. This has satisfied the request requirements of the application support team, as the appuser may now manage the four file systems in the appsvg.

Prior to domain RBAC, there was no RBAC functionality to allow the administrator to grant a user privileged access to only selected devices. For example, if privileged access was granted to the **chfs** command then the privilege could be used to change the attributes of all file systems.

This meant that there was no way to prevent a user granted privileged access to the **chfs** command from accessing or modifying file systems to which they may not be authorized to access or administer.

The /backup file system was not a file system to which the appuser account requires privileged access, but because the appuser account has been granted privileged access to the **chfs** command, the administrator is unable to use Enhanced RBAC to limit the file systems that the appuser may modify.

In Example 8-29 we see the appuser account using the **chfs** command add 1 GB to the /backup file system.

*Example 8-29 The appuser account using the chfs command to change the /backup file system*

---

```
$ id
uid=301(appuser) gid=202(appgroup) groups=1(staff)
$ df -g /backup
Filesystem      GB blocks      Free %Used      Iused %Iused Mounted on
/dev/backup_lv  1.25          1.15    8%           5      1% /backup
$ chfs -a size=+1G /backup
Filesystem size changed to 4718592
$ df -g /backup
Filesystem      GB blocks      Free %Used      Iused %Iused Mounted on
/dev/backup_lv  2.25          2.15    5%           5      1% /backup
$
```

---

The appuser account was able to modify the /backup file system because the apps\_fs\_manage role includes the access authorization for the **chfs** command.

The RBAC role is functioning correctly, but does not offer the functionality to limit the **chfs** command execution to only selected file systems.

Domain RBAC introduces the domain into Role Based Access Control.

The domain allows the administrator to further granualize the privileged command execution by limiting access to system resources to which a user may be granted privileged command execution.

The administrator will now use domain RBAC to:

1. Create two RBAC domains
2. Create multiple domain RBAC objects
3. Update the Kernel Security Tables (KST)
4. Associate the RBAC domain to the appuser account
5. Attempt to use the **ch1v** command to change the /apps04 and /backup file systems

Firstly, the administrator will create two RBAC domains:

applvDom	This domain will be used to reference the /apps01, /apps02, /apps03 and /apps04 file systems
privlvDom	This domain will be used to restrict access to the file systems to which the appuser may access

**Note:** RBAC domains names do have to be in mixed case. Mixed case has been used in this scenario as an example.

Example 8-30 shows the **mkdom** command being used by the root user to create the applvDom and privlvDom domains.

*Example 8-30 The mkdom command - creating the applvDom and privlvDom domains*

---

```
# id
uid=0(root) gid=0(system)
groups=2(bin),3(sys),7(security),8(cron),10(audit),11(lp)
# mkdom applvDom
# lsdom applvDom
applvDom id=1
# mkdom privlvDom
# lsdom privlvDom
privlvDom id=2
#
```

---

The next step is to define the file systems as domain RBAC objects.

The **setsecattr** command is used to define domain RBAC *objects*. In this scenario the administrator wishes to grant privileged access to four file systems

and restrict privileged access to the remaining file systems. To do this the administrator will need to define each file system as a domain RBAC *object*.

The administrator ensures that all file systems on the server are mounted then uses the **df** command to check the logical volume and file system names.

*Example 8-31 The df -kP output - file systems on the AIX V7.1 LPAR*

---

```
# df -kP
Filesystem      1024-blocks      Used Available Capacity Mounted on
/dev/hd4         196608         186300      10308      95% /
/dev/hd2         2031616        1806452      225164      89% /usr
/dev/hd9var       393216         335268       57948       86% /var
/dev/hd3         131072          2184       128888       2% /tmp
/dev/hd1         65536           428       65108       1% /home
/dev/hd11admin   131072          380       130692       1% /admin
/proc            -              -            -            - /proc
/dev/hd10opt     393216         179492      213724      46% /opt
/dev/livedump    262144          368       261776       1% /var/adm/ras/livedump
/dev/backup_lv   2359296        102272      2257024       5% /backup
/dev/apps1v_01   1310720        1117912      192808      86% /apps01
/dev/apps1v_02   1310720        1117912      192808      86% /apps02
/dev/apps1v_03   1310720        1117912      192808      86% /apps03
/dev/apps1v_04   2359296        1118072      1241224      48% /apps04
#
```

---

The administrator now uses the **setsecattr** command to define each of the four application file systems as domain RBAC objects.

Example 8-32 shows the **setsecattr** command being used by the root user to define the domain RBAC objects for the four appsvg file systems.

**Note:** When defining a file system object in domain RBAC, the logical volume device name will be used for the domain *object*.

*Example 8-32 Using the setsecattr command to define the four application file systems as domain RBAC objects*

---

```
# id
uid=0(root) gid=0(system) groups=2(bin),3(sys),7(security),8(cron),10(audit),11(lp)
# setsecattr -o domains=applvDom objtype=device secflags=FSF_DOM_ANY /dev/apps1v_01
# setsecattr -o domains=applvDom objtype=device secflags=FSF_DOM_ANY /dev/apps1v_02
# setsecattr -o domains=applvDom objtype=device secflags=FSF_DOM_ANY /dev/apps1v_03
# setsecattr -o domains=applvDom objtype=device secflags=FSF_DOM_ANY /dev/apps1v_04
# lssecattr -o /dev/apps1v_01
/dev/apps1v_01 domains=applvDom objtype=device secflags=FSF_DOM_ANY
# lssecattr -o /dev/apps1v_02
/dev/apps1v_02 domains=applvDom objtype=device secflags=FSF_DOM_ANY
# lssecattr -o /dev/apps1v_03
```

```

/dev/appslv_03 domains=applvDom objtype=device secflags=FSF_DOM_ANY
# lssecattr -o /dev/appslv_04
/dev/appslv_04 domains=applvDom objtype=device secflags=FSF_DOM_ANY
#

```

---

In Example 8-32 on page 295 the following attributes were defined

Domain	The <code>domains</code> attribute is the domain to which the domain RBAC <i>object</i> will be associated
Object Type	This is the type of domain RBAC object. The <code>objtype=device</code> is used for a logical volume
Security Flags	When the <code>secflags</code> attribute is set to <code>FSF_DOM_ANY</code> a <i>subject</i> may access the <i>object</i> when it contains any of the domains specified in the <i>domains</i> attribute
Device Name	This is the full path name to the logical volume corresponding to the file system. As an example, <code>/dev/appslv_01</code> is the logical volume corresponding to the <code>/apps01</code> file system

**Note:** In domain RBAC, all *objects* with an `objtype=device` must specify the full path name to the device, starting with the `/dev` name.

As an example, the `rootvg` volume group device would be specified to domain RBAC as `objtype=/dev/rootvg`.

The administrator will now use the `setsecattr` command to define the remaining file systems as domain RBAC *objects*.

Example 8-33 shows the `setsecattr` command being used by the root user to define the domain RBAC *objects* for the remaining file systems

*Example 8-33 Using the `setsecattr` command to define the remaining file systems as domain RBAC objects*

---

```

# id
uid=0(root) gid=0(system)
groups=2(bin),3(sys),7(security),8(cron),10(audit),11(lp)
# setsecattr -o domains=privlvDom conflictsets=applvDom \
objtype=device secflags=FSF_DOM_ANY /dev/hd4
# setsecattr -o domains=privlvDom conflictsets=applvDom \
objtype=device secflags=FSF_DOM_ANY /dev/hd2
# setsecattr -o domains=privlvDom conflictsets=applvDom \
objtype=device secflags=FSF_DOM_ANY /dev/hd9var
# setsecattr -o domains=privlvDom conflictsets=applvDom \
objtype=device secflags=FSF_DOM_ANY /dev/hd3

```

```
# setsecattr -o domains=privlvDom conflictsets=applvDom \
objtype=device secflags=FSF_DOM_ANY /dev/hd1
# setsecattr -o domains=privlvDom conflictsets=applvDom \
objtype=device secflags=FSF_DOM_ANY /dev/hd11admin
# setsecattr -o domains=privlvDom conflictsets=applvDom \
objtype=device secflags=FSF_DOM_ANY /dev/proc
# setsecattr -o domains=privlvDom conflictsets=applvDom \
objtype=device secflags=FSF_DOM_ANY /dev/hd10opt
# setsecattr -o domains=privlvDom conflictsets=applvDom \
objtype=device secflags=FSF_DOM_ANY /dev/livedump
# setsecattr -o domains=privlvDom conflictsets=applvDom \
objtype=device secflags=FSF_DOM_ANY /dev/backup_lv
# lssecattr -o /dev/hd4
/dev/hd4 domains=privlvDom conflictsets=applvDom objtype=device \
secflags=FSF_DOM_ANY
#
```

---

In Example 8-33 on page 296 the following attributes were defined:

Domain	The domains attribute is the domain to which the domain RBAC <i>object</i> will be associated
Conflict Set	This is an optional attribute. By defining the conflictsets=applvDom, this <i>object</i> will not be accessible if the entity has an existing association to the applvDom domain.
Object Type	This is the type of domain RBAC <i>object</i> . The objtype=device is used for a logical volume
Security Flags	When the secflags attribute is set to FSF_DOM_ANY a <i>subject</i> may access the <i>object</i> when it contains any of the domains specified in the domains attribute
Device Name	This is the full path name to the logical volume corresponding to the file system. As an example, /dev/hd2 is the logical volume corresponding to the /usr file system

The administrator will now use the **setkst** command to update the KST with the changes made with the **setsecattr** and **mkdom** commands.

Example 8-34 shows the **setkst** command being executed from the root user:

*Example 8-34 Using the setkst command to update the KST*

---

```
# id
```

```
uid=0(root) gid=0(system)
groups=2(bin),3(sys),7(security),8(cron),10(audit),11(lp)
# setkst
Successfully updated the Kernel Authorization Table.
Successfully updated the Kernel Role Table.
Successfully updated the Kernel Command Table.
Successfully updated the Kernel Device Table.
Successfully updated the Kernel Object Domain Table.
Successfully updated the Kernel Domains Table.
#
```

---

The administrator will now use the **chuser** command to associate the appuser account with the applvDom domain.

Example 8-35 shows the **chuser** command being executed by the root user:

*Example 8-35 Using the chuser command to associate the appuser account with the applvDom domain*

---

```
# lsuser -a domains appuser
appuser
# chuser domains=applvDom appuser
# lsuser -a domains appuser
appuser domains=applvDom
#
```

---

The administrator has now completed the domain RBAC configuration. The four application file systems have been defined as domain RBAC *objects* and the appuser has been associated with the applvDom domain.

The administrator has also defined the remaining file systems as domain RBAC *objects*. This will restrict privileged access to users only associated with the privlvDom domain, and added a conflict set to the applvDom domain.

The conflict set will ensure that if the appuser account were to be granted an association to the privlvDom domain, the file system objects could not be modified with the privileged commands, as the privlvDom and applvDom domains are in conflict.

In Example 8-36 the appuser account uses the **swrole** command to switch to the apps\_fs\_manage role.

*Example 8-36 The appuser account uses the swrole command to switch to the apps\_fs\_manage role*

---

```
$ id
```



```
uid=301(appuser) gid=202(appgroup) groups=1(staff)
$ rolelist -a
apps_fs_manage  aix.fs.manage.change
                 aix.fs.manage.create
                 aix.fs.manage.mount
                 aix.fs.manage.remove
                 aix.fs.manage.unmount
$ swrole apps_fs_manage
appuser's Password:
$
```

---

The appuser account may now use the privileged commands in the apps\_fs\_manage role.

In Example 8-37 the appuser uses the **chfs** command to increase the size of the /apps01 file system by 1 GB. This command will successfully complete because the /dev/appslv\_01 device was defined as a domain RBAC *object* to which the appuser has been granted an association through the applvDom domain.

Example 8-37 shows the appuser account using the **chfs** command to add 1 GB to the /apps01 file system:

*Example 8-37 The appuser account using the chfs command to add 1 GB to the /apps01 file system*

---

```
$ df -g /apps01
Filesystem  GB blocks   Free %Used   Iused %Iused Mounted on
/dev/appslv_01  1.25    0.18  86%      15    1% /apps01
$ chfs -a size=+1G /apps01
Filesystem size changed to 4718592
$ df -g /apps01
Filesystem  GB blocks   Free %Used   Iused %Iused Mounted on
/dev/appslv_01  2.25    1.18  48%      15    1% /apps01
$
```

---

In Example 8-37 we see that the **chfs** command has been successful.

Next, the appuser uses the **chfs** command to increase the size of the /backup file system by 1 GB.

Example 8-38 on page 300 shows the appuser account attempting to use the **chfs** command to add 1 GB to the /backup file system:

*Example 8-38 The appuser account attempting to use the chfs command to add 1 GB to the /backup file system*

---

```
$ df -g /backup
Filesystem      GB blocks      Free %Used      Iused %Iused Mounted on
/dev/backup_lv  2.25           2.15  5%            5      1% /backup
$ chfs -a size=+1G /backup
/dev/backup_lv: Operation not permitted.
$ df -g /backup
Filesystem      GB blocks      Free %Used      Iused %Iused Mounted on
/dev/backup_lv  2.25           2.15  5%            5      1% /backup
$
```

---

In Example 8-38, the **chfs** command was not successful.

The **chfs** command was not successful because the `/dev/backup_lv` device was defined as a domain RBAC object but the `appuser` account has not been granted association to the `privlvDom` domain.

Domain RBAC has restricted the `appuser` account using the **chfs** command to change the `/backup` file system because the `appuser` account has no association with the `privlvDom` domain.

Even though the `appuser` account has used the **swrole** command to switch to the `apps_fs_manage` role, the privileged **chfs** command is unsuccessful because domain RBAC has denied the `appuser` account access based on the domain object attributes of the `/backup_lv` *object* and the domain association of the `appuser` account.

By using this methodology, domain RBAC has restricted the `appuser` to managing only the file systems to which it has direct responsibility, and excluded privileged access to the remaining file systems on the LPAR.

In Example 8-39 on page 301 the `appuser` account changes directory to the `/tmp` file system and uses the **touch appuser\_tmp\_file** command to show that the `appuser` account may still access the `/tmp` file system, but may not execute privileged commands, even though the `apps_fs_manage` role is effective.

In Example 8-39 on page 301, the `appuser` account may also run the **whoami** command which is located in the `/usr/bin` directory in the `/usr` file system.

The `/usr` file system was also defined as a domain RBAC *object*, but is still accessible from the `appuser` and other user accounts, though the `appuser` account may not perform privileged operations on the `/usr` file system as shown when the `appuser` account attempts to execute the **chfs -a freeze=30 /usr** command.

*Example 8-39 The appuser account using the touch and whoami commands*

---

```
$ id
uid=301(appuser) gid=202(appgroup) groups=1(staff)
$ roletlist -e
apps_fs_manage Manage apps filesystems
$ cd /tmp
$ touch appuser_tmp_file
$ ls -ltra appuser_tmp_file
-rw-r--r--  1 appuser  appgroup          0 Sep 13 19:44 appuser_tmp_file
$ whoami
appuser
$ chfs -a freeze=30 /usr
/dev/hd2: Operation not permitted.
$
```

---

The appuser and other user accounts may still access the domained file systems, such as the /tmp and /usr file systems as general users, but the privileged commands available to the appuser account in the apps\_fs\_manage role may not be used on file systems other than the /apps01, /apps02, /apps03 and /apps04 file systems.

### File scenario - Restrict access

In a default installation of AIX, some files may be installed with DAC permissions that allow the files to be read by non privileged users. Though the files may only be modified by the root user, these files may contain information that the administrator may not wish to be readable by all users.

By using domain RBAC, the administrator can restrict file access to only those user accounts that are deemed to require access.

In this scenario the administrator has been requested to limit read access of the /etc/hosts file to only the netuser user account. This can be accomplished by using domain RBAC.

In this scenario we have:

- ▶ An AIX V7.1 partition with enhanced RBAC enabled
- ▶ A non privileged user named netuser
- ▶ A non privileged user named appuser

In Example 8-40 on page 302, the user netuser account uses the **head -15** command to view the first 15 lines of the /etc/hosts file.

The **ls -ltra** command output shows that the DAC permissions allow any user account to view the /etc/hosts file.

*Example 8-40 The netuser account - using the head -15 command to view the first 15 lines of the /etc/hosts file*

---

```
$ id
uid=302(netuser) gid=204(netgroup) groups=1(staff)
$ ls -ltra /etc/hosts
-rw-rw-r--  1 root    system      2052 Aug 22 20:35 /etc/hosts
$ head -15 /etc/hosts
# IBM_PROLOG_BEGIN_TAG
# This is an automatically generated prolog.
#
# bos61D src/bos/usr/sbin/netstart/hosts 1.2
#
# Licensed Materials - Property of IBM
#
# COPYRIGHT International Business Machines Corp. 1985,1989
# All Rights Reserved
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
#@(#)47      1.2  src/bos/usr/sbin/netstart/hosts, cmdnet, bos61D,
d2007_49A2 10/1/07 13:57:52
# IBM_PROLOG_END_TAG
$
```

---

In Example 8-41, the user appuser uses the **head-15** command to view the first 15 lines of the **/etc/hosts** file. Again, the **ls-ltra** command output shows that the DAC permissions allow any user account to view the **/etc/hosts** file.

*Example 8-41 The appuser account - using the head -15 command to view the first 15 lines of the /etc/hosts file*

---

```
$ id
uid=301(appuser) gid=202(appgroup) groups=1(staff)
$ ls -ltra /etc/hosts
-rw-rw-r--  1 root    system      2052 Aug 22 20:35 /etc/hosts
$ head -15 /etc/hosts
# IBM_PROLOG_BEGIN_TAG
# This is an automatically generated prolog.
#
# bos61D src/bos/usr/sbin/netstart/hosts 1.2
#
# Licensed Materials - Property of IBM
#
# COPYRIGHT International Business Machines Corp. 1985,1989
# All Rights Reserved
#
# US Government Users Restricted Rights - Use, duplication or
```

```
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
# @(#)47      1.2  src/bos/usr/sbin/netstart/hosts, cmdnet, bos61D,
d2007_49A2 10/1/07 13:57:52
# IBM_PROLOG_END_TAG
$
```

---

Both the netuser and appuser accounts are able to view the `/etc/hosts` file, due to the DAC of the `/etc/hosts` file.

By creating an RBAC domain and defining the `/etc/hosts` file as a domain RBAC *object*, access to the `/etc/hosts` file may be restricted, based upon the user account's association with the RBAC domain.

In Example 8-42, the root user logs in and uses the `mkdom` command to create an RBAC domain named `privDom`. The `privDom` domain has a domain ID of 3, which has been automatically system generated as the administrator did not include a domain ID in the the `mkdom` command.

*Example 8-42 Using the `mkdom` command to create the `privDom` domain*

---

```
# id
uid=0(root) gid=0(system)
groups=2(bin),3(sys),7(security),8(cron),10(audit),11(lp)
# mkdom privDom
# lsdom privDom
privDom id=3
#
```

---

From the root user, the administrator next defines the `/etc/hosts` file as a domain RBAC *object*.

In Example 8-43, the administrator uses the `setsecattr` command to define the `/etc/hosts` file as a domain RBAC object and assign the RBAC domain as `privDom`. The `objtype` attribute is set as the `type file`.

*Example 8-43 Using the `setsecattr` command to define the `/etc/hosts` file as a domain RBAC object*

---

```
# setsecattr -o domains=privDom objtype=file secflags=FSF_DOM_ANY /etc/hosts
# lssecattr -o /etc/hosts
/etc/hosts domains=privDom objtype=file secflags=FSF_DOM_ANY
#
```

---

For these changes to be available for use, the root user must updated the KST with the `setkst` command.

Example 8-44 shows the `lskst -t` command being used to list the KST prior to the `setkst` command being run.

Once the `setkst` command is run, the `privDom` domain and `/etc/hosts` file are both updated into the KST and are available for use.

*Example 8-44 Updating the KST with the setkst command*

---

```
# lskst -t dom privDom
Domain "privDom" does not exist.
# lskst -t domobj /etc/hosts
Domain object "/etc/hosts" does not exist.
# setkst
Successfully updated the Kernel Authorization Table.
Successfully updated the Kernel Role Table.
Successfully updated the Kernel Command Table.
Successfully updated the Kernel Device Table.
Successfully updated the Kernel Object Domain Table.
Successfully updated the Kernel Domains Table.
# lskst -t dom privDom
privDom id=4
# lskst -t domobj /etc/hosts
/etc/hosts objtype=FILE domains=privDom \
conflictsets= secflags=FSF_DOM_ANY
#
```

---

At this stage, the `/etc/hosts` file has been defined as domain RBAC *object* and the KST updated.

The `/etc/hosts` file will now operate as a domain RBAC *object* and restrict access to any user accounts that have not been associated with the `privDom` domain.

This can be tested by attempting to access the `/etc/hosts` file from the `netuser` and `appuser` accounts.

**Note:** The root user is automatically a member of all RBAC domains so does not require any special access to the `privDom` domain

Example 8-45 and 8-46 shows the `netuser` account using the `head -15` command to read the `/etc/hosts` file.

*Example 8-45 The netuser account using the head -15 command to access the /etc/hosts file*

---

```
$ id
```

```
uid=302(netuser) gid=204(netgroup) groups=1(staff)
$ ls -ltra /etc/hosts
-rw-rw-r--  1 root    system      2052 Aug 22 20:35 /etc/hosts
$ head -15 /etc/hosts
/etc/hosts: Operation not permitted.
$
```

---

*Example 8-46* The appuser account using the head -15 command to access the /etc/hosts file

---

```
$ id
uid=301(appuser) gid=202(appgroup) groups=1(staff)
$ ls -ltra /etc/hosts
-rw-rw-r--  1 root    system      2052 Aug 22 20:35 /etc/hosts
$ head -15 /etc/hosts
/etc/hosts: Operation not permitted.
$
```

---

The netuser and appuser accounts are no longer able access the /etc/hosts file, even though the /etc/hosts file DAC allows for read access by any user. This is because the /etc/hosts file is now a domain RBAC object and access is dependant on the privDom domain association.

In Example 8-47, the administrator associates the netuser account with the privDom domain by using the **chuser** command from the root user.

*Example 8-47* Using the chuser command to grant the netuser account association to the privDom domain

---

```
# lsuser -a domains netuser
netuser
# chuser domains=privDom netuser
# lsuser -a domains netuser
netuser domains=privDom
#
```

---

Now that the netuser account has been associated with the privDom domain, the netuser account may again access the /etc/hosts file.

**Note:** Due to the **chuser** attribute change, the netuser account must log out and login for the domain=privDom association to take effect.

In Example 8-48 on page 306 we see the netuser account using the **head -15** command to access the /etc/hosts file.

*Example 8-48 The netuser account using the head -15 command to access the /etc/hosts file*

---

```
$ id
uid=302(netuser) gid=204(netgroup) groups=1(staff)
$ ls -ltra /etc/hosts
-rw-rw-r--  1 root    system      2052 Aug 22 20:35 /etc/hosts
$ head -15 /etc/hosts
# IBM_PROLOG_BEGIN_TAG
# This is an automatically generated prolog.
#
# bos61D src/bos/usr/sbin/netstart/hosts 1.2
#
# Licensed Materials - Property of IBM
#
# COPYRIGHT International Business Machines Corp. 1985,1989
# All Rights Reserved
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
# @(#)47      1.2  src/bos/usr/sbin/netstart/hosts, cmdnet, bos61D,
d2007_49A2 10/1/07 13:57:52
# IBM_PROLOG_END_TAG
$
```

---

The netuser account is now able to access the /etc/hosts file.

Associating the netuser account with the privDom domain has allowed the netuser account to access the *object* and list the contents of the /etc/hosts file with the **head -15** command.

Domain RBAC will still honor the DAC for the file object, so the netuser account will have only read access to the /etc/host file. Domain RBAC does not automatically grant write access to the file, but does allow the administrator to restrict the access to the /etc/hosts file without having to change the DAC file permission bits.

The appuser account will remain unable to access the /etc/hosts file because it has not been associated with the privDom domain.

Example 8-49 shows the appuser account attempting to access the /etc/hosts file by using the **head -15** command

*Example 8-49 The appuser account using the head -15 command to access the /etc/hosts file*

---

```
$ id
```



```
uid=301(appuser) gid=202(appgroup) groups=1(staff)
$ ls -ltra /etc/hosts
-rw-rw-r--  1 root    system      2052 Aug 22 20:35 /etc/hosts
$ head -15 /etc/hosts
/etc/hosts: Operation not permitted.
$
```

---

The appuser account is denied access to the /etc/hosts file because it does not have the association with the pri vDom domain.

The administrator has successfully completed the request as the /etc/hosts file is now restricted to access by only the netuser account.

More than one user can be associated with a domain, so were more users to require access to the /etc/hosts file, the administrator need only use the **chuser** command to grant those users association with the pri vDom domain.

The root user is automatically considered a member of all domains, so the root user remains able to access the /etc/hosts file.

**Note:** When restricting access to files consider the impact to existing AIX commands and functions.

As an example, restricting access to the /etc/passwd file would result in non privileged users being no longer able to successfully execute the **passwd** command to set their own passwords.

## File scenario - Remove access

In this scenario we will discuss how domain RBAC can be used to remove access to files or non privileged users.

In a default installation of AIX, some files may be installed with DAC permissions that allow the files to be read by non privileged users. Though the files may only be modified by the root user, these files may contain information that the administrator may not wish to readable by all users.

By using domain RBAC, the administrator can remove file access to user accounts that are deemed to not require access to such files.

In this scenario the administrator has chosen to remove read access to the /etc/ssh/sshd\_config. This can be accomplished by using domain RBAC.

In this scenario we have:

- ▶ An AIX V7.1 partition with enhanced RBAC enabled

- A non privileged user named appuser

In Example 8-50 we see the user appuser using the **head-15** command to view the first 15 lines of the `/etc/ssh/sshd_config` file.

We can see from the **ls -ltr** command output that the DAC permissions allow any user account to view the `/etc/ssh/sshd_config` file.

*Example 8-50 The appuser account - using the head -15 command to view the first 15 lines of the /etc/ssh/sshd\_config file*

---

```
$ id
uid=301(appuser) gid=202(appgroup) groups=1(staff)
$ ls -ltr /etc/ssh/sshd_config
-rw-r--r--  1 root      system          3173 Aug 19 23:29 /etc/ssh/sshd_config
$ head -15 /etc/ssh/sshd_config
#
#OpenBSD: sshd_config,v 1.81 2009/10/08 14:03:41 markus Exp $

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options change a
# default value.

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
$
```

---

As shown in Example 8-50, the `/etc/ssh/sshd_config` file has DAC permissions that allow all users on the LPAR to read the file.

By creating an RBAC domain and defining the `/etc/ssh/sshd_config` file as a domain RBAC *object*, the administrator may restrict access to the `/etc/ssh/sshd_config` to only user accounts with membership to the RBAC domain.

By not associating the RBAC domain to any user accounts, the RBAC object will not be accessible to any user accounts other than the root user.

In Example 8-51 on page 309, the administrator uses the root user to create an RBAC domain named `lockDom`. The `lockDom` domain has a domain ID of 4, which has been automatically system generated as no domain ID was specified with the **mkdom** command.

*Example 8-51 Using the mkdom command to create the lockDom domain*

---

```
# id
uid=0(root) gid=0(system)
groups=2(bin),3(sys),7(security),8(cron),10(audit),11(lp)
# mkdom lockDom
# lsdom lockDom
lockDom id=4
#
```

---

The administrator next uses the `setsecatr` command to define the `/etc/ssh/sshd_config` file as a domain RBAC object.

In Example 8-52, the root user executes the `setsecatr` command to define the `/etc/ssh/sshd_config` file as a domain RBAC object and set the RBAC domain as `lockDom`.

*Example 8-52 Using the setsecatr command to define the /etc/ssh/sshd\_config file as a domain RBAC object*

---

```
# id
uid=0(root) gid=0(system)
groups=2(bin),3(sys),7(security),8(cron),10(audit),11(lp)
# setsecatr -o domains=lockDom objtype=file \
seclags=FSF_DOM_ANY /etc/ssh/sshd_config
# lssecatr -o /etc/ssh/sshd_config
/etc/ssh/sshd_config domains=lockDom objtype=file seclags=FSF_DOM_ANY
#
```

---

The `/etc/ssh/sshd_config` file has now been defined as a domain RBAC *object*.

To update the RBAC database change into the KST, the administrator uses the `setkst` command.

Example 8-53 shows the root user running the `lskst` command to list the contents of the KST. The root user then updates the KST by running the `setkst` command.

*Example 8-53 Using the setkst command to update the KST and the lskst command to list the KST*

---

```
# lskst -t dom lockDom
Domain "lockDom" does not exist.
# lskst -t domobj /etc/ssh/sshd_config
Domain object "/etc/ssh/sshd_config" does not exist.
# setkst
Successfully updated the Kernel Authorization Table.
```

---

```

Successfully updated the Kernel Role Table.
Successfully updated the Kernel Command Table.
Successfully updated the Kernel Device Table.
Successfully updated the Kernel Object Domain Table.
Successfully updated the Kernel Domains Table.
# lskst -t dom lockDom
lockDom id=4
# lskst -t domobj /etc/ssh/sshd_config
/etc/ssh/sshd_config objtype=FILE domains=lockDom conflictsets=
secflags=FSF_DOM_ANY
#

```

---

At this stage, the `/etc/ssh/sshd_config` file is now defined as a domain RBAC *object* and the KST updated. Access to the `/etc/ssh/sshd_config` file is now restricted to the root user and any user accounts that are associated with the lockDom domain.

Because no user accounts have an association with the lockDom domain, the `/etc/ssh/sshd_config` file is now only accessible by the root user.

Example 8-54 shows the appuser account attempting to access the `/etc/ssh/sshd_config` file with the **head**, **more**, **cat**, **pg** and **vi** commands:

*Example 8-54 Using the head, more, cat, pg and vi commands to attempt access to the /etc/ssh/sshd\_config file*

---

```

$ id
uid=301(appuser) gid=202(appgroup) groups=1(staff)
$ head -15 /etc/ssh/sshd_config
/etc/ssh/sshd_config: Operation not permitted.
$ more /etc/ssh/sshd_config
/etc/ssh/sshd_config: Operation not permitted.
$ cat /etc/ssh/sshd_config
cat: 0652-050 Cannot open /etc/ssh/sshd_config.
$ pg /etc/ssh/sshd_config
/etc/ssh/sshd_config: Operation not permitted.
$ vi /etc/ssh/sshd_config
~
...
...
~
"/etc/ssh/sshd_config" Operation not permitted.
$

```

---

The appuser account is not able to access the `/etc/ssh/sshd_config` file.

The only user able to access the `/etc/ssh/sshd_config` file is the root user.

If the `appuser` account were to be associated with the `lockDom` domain then the `appuser` account would again be able to access the `/etc/ssh/sshd_config` file, based on the file DAC permission.

The benefits in using domain RBAC to restrict file access include:

File modification	There is no requirement to modify the file DAC settings, including ownership and bit permissions
Quick to reinstate	Reinstating the file access does not require the administrator to modify file DAC. The administrator can generally reinstate the file access by removing the <i>object</i> from the domain RBAC and updating the KST
Granular control	The administrator may still grant access to the file <i>object</i> by associating user accounts with the RBAC domain, if required for temporary or long term access

**Note:** When removing access to files consider the impact to existing AIX commands and functions.

As an example, removing access to the `/etc/security/passwd` file would result in non privileged users no longer being able to successfully execute the `passwd` command to set their own passwords.

## Network scenario

In this scenario, domain RBAC will be used to restrict privileged access to an Ethernet network interface.

In domain RBAC, network objects may be either of two object types:

<code>netint</code>	This object type is a network interface. As an example, the <code>en0</code> Ethernet interface would be an object type of <code>netint</code>
<code>netport</code>	This object type is a network port. As an example, the TCP port 22 would be an object type of <code>netport</code>

By using domain RBAC, the administrator can restrict a subject from performing privileged commands upon a `netint` or `netport` object.

In this scenario, the AIX V7.1 LPAR has two Ethernet network interfaces configured.

The administrator will use domain RBAC to:

- ▶ allow the `netuser` account to use the `ifconfig` command on the `en2` Ethernet interface

- ▶ restrict the appuser account from using the **ifconfig** command on the en0 Ethernet interface.

Unless noted otherwise, all commands in the scenario will be run as the root user.

The administrator first uses the **lssecattr** command to determine which access authorizations the **ifconfig** command requires.

Example 8-55 shows the root user using the **lssecattr** command to display the access authorizations required by the **ifconfig** command:

*Example 8-55 Using the lssecattr command from the root user to list the access authorizations for the ifconfig command*

---

```
# lssecattr -c -a accessauths /usr/sbin/ifconfig
/usr/sbin/ifconfig accessauths=aix.network.config.tcpip
#
```

---

The **ifconfig** command requires the `aix.network.config.tcpip` access authorization.

The administrator will now use the **authrpt** command to determine whether there is an existing role that contains the necessary access authorizations required for the executing the **ifconfig** command. The **authrpt -r** command will limit the output displayed to only the roles associated with an authorization.

Example 8-56 shows **authrpt -r** command being used to report on the `aix.network.config.tcpip` authorization.

*Example 8-56 Using the authrpt command from the root user to determine role association with the aix.network.config.tcpip authorization*

---

```
# authrpt -r aix.network.config.tcpip
authorization:
aix.network.config.tcpip
roles:

#
```

---

The `roles:` field in Example 8-56 has no value returned, which shows that there is no existing role associated with the `aix.network.config.tcpip` authorization. The administrator must use the **mkrole** command to create a role and associate the `aix.network.config.tcpip` authorization to the role.

Example 8-57 on page 313 shows the administrator using the **mkrole** command to create the `netifconf` role and include the `aix.network.config.tcpip`

authorization as the `accessauths` attribute. The administrator then updates the KST with the `setkst` command.

*Example 8-57 Using the `mkrole` command from the root user to create the `netifconf` role and associate with the `aix.network.config.tcpip` authorization*

---

```
# mkrole authorizations=aix.network.config.tcpip \
dflmsg="Manage net interface" netifconf
# lsrole netifconf
netifconf authorizations=aix.network.config.tcpip roletlist= \
groups= visibility=1 screens=* dflmsg=Manage net interface \
msgcat= auth_mode=INVOKER id=19
# setkst
Successfully updated the Kernel Authorization Table.
Successfully updated the Kernel Role Table.
Successfully updated the Kernel Command Table.
Successfully updated the Kernel Device Table.
Successfully updated the Kernel Object Domain Table.
Successfully updated the Kernel Domains Table.
#
```

---

The administrator will next use the `lsuser` command to display the existing roles, if any, that the `netuser` command may have associated to it. The administrator will then associate the `netuser` with the `netifconf` role, including any existing roles in the `chuser` command.

Example 8-58 shows the `chuser` command being used to associate the `netuser` account with the `netifconf` role. The `lsuser` command showed that the `netuser` did not have any existing roles.

*Example 8-58 Using the `chuser` command from the root user to associate the `netuser` account with the `netifconf` role*

---

```
# lsuser -a roles netuser
netuser roles=
# chuser roles=netifconf netuser
# lsuser -a roles netuser
netuser roles=netifconf
#
```

---

At this stage, the `netuser` account has been associated with the `netifconf` role and may execute the `ifconfig` privileged command.

The administrator may verify this by using the `authrpt` command and the `rolerpt` command.

Example 8-59 shows the **authrpt** command being used to report the `aix.network.config.tcpip` authorization association with the `netifconf` role

Example 8-59 also shows the **rolerpt** command being used to report the `netifconf` role has an association with the `netuser` account.

*Example 8-59 The root user using the `authrpt` and `rolerpt` commands*

---

```
# authrpt -r aix.network.config.tcpip
authorization:
aix.network.config.tcpip
roles:
netifconf
# rolerpt -u netifconf
role:
netifconf
users:
netuser
#
```

---

The administrator will now use domain RBAC to restrict the authority of the `netuser` account's usage of the **ifconfig** command so that the **ifconfig** command will only execute successfully when used upon the `en2` Ethernet interface.

The administrator will use domain RBAC to:

1. Creating two RBAC domains
2. Create two domain RBAC objects
3. Update the Kernel Security Tables (KST)
4. Associate the RBAC domain to the `netuser` account
5. Attempt to use the **ifconfig** command to change the status of the `en0` and `en2` Ethernet interfaces

In Example 8-60 the administrator uses the **ifconfig -a** command to display the network interfaces. The `en0` and `en2` Ethernet interfaces are both active, shown by the `UP` status.

*Example 8-60 The root user using the `ifconfig -a` command to display the network interface status*

---

```
# ifconfig -a
en0:
flags=1e080863,480<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GR
OUPRT,64BIT,CHECKSUM_OFFLOAD(ACTIVE),CHAIN>
```



```

        inet 192.168.101.12 netmask 0xfffffc00 broadcast
192.168.103.255
        tcp_sendspace 262144 tcp_recvspace 262144 rfc1323 1
en2:
flags=5e080867,c0<UP,BROADCAST,DEBUG,NOTRAILERS,RUNNING,SIMPLEX,MULTICA
ST,GROUPRT,64BIT,CHECKSUM_OFFLOAD(ACTIVE),PSEG,LARGESEND,CHAIN>
        inet 10.10.100.2 netmask 0xffffffff broadcast 10.10.100.255
        tcp_sendspace 131072 tcp_recvspace 65536 rfc1323 0
lo0:
flags=e08084b,c0<UP,BROADCAST,LOOPBACK,RUNNING,SIMPLEX,MULTICAST,GROUPR
T,64BIT,LARGESEND,CHAIN>
        inet 127.0.0.1 netmask 0xff000000 broadcast 127.255.255.255
        inet6 ::1%1/0
        tcp_sendspace 131072 tcp_recvspace 131072 rfc1323 1
#

```

---

After verifying the names of the Ethernet network interfaces in Example 8-60 on page 314, the administrator will now begin the domain RBAC configuration.

in Example 8-61 the root user is used to create the netDom and privNetDom RBAC domains:

*Example 8-61 The root user using the mkdomb command to create the netDom and the privNetDom RBAC domains*

```

# mkdomb netDom
# lsdom netDom
netDom id=5
# mkdomb privNetDom
# lsdom privNetDom
privNetDom id=6
#

```

---

Next, in Example 8-62 the administrator uses the **setsecattr** command to define the en2 and en0 Ethernet network interfaces as domain RBAC objects. The **setkst** command is then run to update the KST.

*Example 8-62 The setsecattr command being used by the root user to define the en0 and en2 domain RBAC objects*

```

# setsecattr -o domains=netDom objtype=netint secflags=FSF_DOM_ANY en2
# setsecattr -o domains=privNetDom conflictsets=netDom \
objtype=netint secflags=FSF_DOM_ANY en0
# lssecattr -o en2
en2 domains=netDom objtype=netint secflags=FSF_DOM_ANY
# lssecattr -o en0

```

```

en0 domains=privNetDom conflictsets=netDom objtype=netint
secflags=FSF_DOM_ANY
# setkst
Successfully updated the Kernel Authorization Table.
Successfully updated the Kernel Role Table.
Successfully updated the Kernel Command Table.
Successfully updated the Kernel Device Table.
Successfully updated the Kernel Object Domain Table.
Successfully updated the Kernel Domains Table.
#

```

---

In Example 8-62 on page 315 the administrator has included the `conflictsets=netDom` attribute when defining the `en0` object. This means that if an entity were granted association with the `privNetDom` and the `netDom`, the entity would not be granted authorization to perform actions on the `en0` object, as the `privNetDom` and `netDom` domains are in conflict.

**Note:** The root user has an automatic association to all domains and objects. The root user does not honor the `conflictsets` attribute as the root user must remain able to access all domain RBAC objects.

The `netuser` next has its domain association extended to include the `netDom` domain. The `netuser` account is already associated with the `privDom` domain from a previous scenario. The `privDom` domain association is included in the `chuser` command, else access to the `privDom` domain would be removed.

Example 8-63 shows the `chuser` command being used to associate the `netuser` account with the `netDom` domain.

**Note:** The `privDom` domain will not be used in this scenario and should not be confused with the `privNetDom` domain, which is used in this scenario.

*Example 8-63 Using the `chuser` command to associate the `netuser` account with the `netDom` domain*

---

```

# lsuser -a domains netuser
netuser domains=privDom
# chuser domains=privDom,netDom netuser
# lsuser -a domains netuser
netuser domains=privDom,netDom
#

```

---

The administrator has now completed the domain RBAC configuration tasks.

The netuser account is now used to test the use of the **ifconfig** command and the domain RBAC configuration.

In Example 8-64 the netuser logs into the AIX V7.1 LPAR and uses the **swrole** command to switch to the netifconf role. The **rolelist -e** command shows that the netifconf role becomes the active role.

*Example 8-64 The netuser account uses the swrole command to switch to the netifconf role*

---

```
$ id
uid=302(netuser) gid=204(netgroup) groups=1(staff)
$ rolelist -a
netifconf      aix.network.config.tcPIP
$ swrole netifconf
netuser's Password:
$ rolelist -e
netifconf      Manage net interface
$
```

---

In Example 8-65 the netuser account uses the **ifconfig** command to display the status of the en2 Ethernet interface, showing that the status is UP. The **ping** command is used to confirm the UP status and has 0 % packet loss.

The netuser account then uses the **ifconfig en2 down** command to inactivate the en2 interface. The **ifconfig** command no longer displays the UP status and the **ping** command returns 100 % packet loss.

The netuser account has successfully used the **ifconfig** command to deactivate the en2 Ethernet interface.

*Example 8-65 The netuser account using the ifconfig command to deactivate the en2 Ethernet interface*

---

```
$ ifconfig en2
en2:
flags=5e080867,c0<UP,BROADCAST,DEBUG,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GROUPRT,64BIT,CHECKSUM_OFFLOAD(ACTIVE),PSEG,LARGESEND,CHAIN>
    inet 10.10.100.2 netmask 0xffffffff broadcast 10.10.100.255
    tcp_sendspace 131072 tcp_recvspace 65536 rfc1323 0
$ ping -c2 -w 2 10.10.100.5
PING 10.10.100.5: (10.10.100.5): 56 data bytes
64 bytes from 10.10.100.5: icmp_seq=0 ttl=64 time=1 ms
64 bytes from 10.10.100.5: icmp_seq=1 ttl=64 time=0 ms
```

```

----10.10.100.5 PING Statistics----
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max = 0/0/1 ms
$ ifconfig en2 down
$ ifconfig en2
en2:
flags=5e080866,c0<BROADCAST,DEBUG,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,
GROUPRT,64BIT,CHECKSUM_OFFLOAD(ACTIVE),PSEG,LARGESEND,CHAIN>
    inet 10.10.100.2 netmask 0xffffffff broadcast 10.10.100.255
    tcp_sendspace 131072 tcp_recvspace 65536 rfc1323 0
$ ping -c2 -w 2 10.10.100.5
PING 10.10.100.5: (10.10.100.5): 56 data bytes
0821-069 ping: sendto: The network is not currently available.
ping: wrote 10.10.100.5 64 chars, ret=-1
0821-069 ping: sendto: The network is not currently available.
ping: wrote 10.10.100.5 64 chars, ret=-1

----10.10.100.5 PING Statistics----
2 packets transmitted, 0 packets received, 100% packet loss
$

```

---

In Example 8-66, the netuser account then uses the **ifconfig en2 up** command to reactivate the en2 interface. The **ifconfig** command displays the UP status and the **ping** command returns 0 % packet loss.

The netuser account has successfully used the **ifconfig** command to activate the en2 Ethernet interface.

*Example 8-66 The netuser account using the ifconfig command to activate the en2 Ethernet interface*

```

$ ifconfig en2 up
$ ifconfig en2
en2:
flags=5e080867,c0<UP,BROADCAST,DEBUG,NOTRAILERS,RUNNING,SIMPLEX,MULTICA
ST,GROUPRT,64BIT,CHECKSUM_OFFLOAD(ACTIVE),PSEG,LARGESEND,CHAIN>
    inet 10.10.100.2 netmask 0xffffffff broadcast 10.10.100.255
    tcp_sendspace 131072 tcp_recvspace 65536 rfc1323 0
$ ping -c2 -w 2 10.10.100.5
PING 10.10.100.5: (10.10.100.5): 56 data bytes
64 bytes from 10.10.100.5: icmp_seq=0 ttl=64 time=0 ms
64 bytes from 10.10.100.5: icmp_seq=1 ttl=64 time=0 ms

----10.10.100.5 PING Statistics----
2 packets transmitted, 2 packets received, 0% packet loss

```

```
round-trip min/avg/max = 0/0/0 ms
$
```

---

By using RBAC the netuser account has been able to successfully use the **ifconfig** command to activate and deactivate the en2 Ethernet interface.

In Example 8-67, domain RBAC is used to restrict the netuser account from using the **ifconfig** command to change the status en0 interface. When the netuser account uses the **ifconfig en0 down** command, the **ifconfig** command is not successful.

*Example 8-67 The netuser account is unsuccessful in using the ifconfig command to inactivate the en0 Ethernet interface*

---

```
$ id
uid=302(netuser) gid=204(netgroup) groups=1(staff)
$ rolelist -e
netifconf      Manage net interface
$ ifconfig en0
en0:
flags=1e080863,480<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GR
OUPRT,64BIT,CHECKSUM_OFFLOAD(ACTIVE),CHAIN>
    inet 192.168.101.12 netmask 0xfffffc00 broadcast
    192.168.103.255
    tcp_sendspace 262144 tcp_recvspace 262144 rfc1323 1
$ ping -c2 -w 2 192.168.101.11
PING 192.168.101.11: (192.168.101.11): 56 data bytes
64 bytes from 192.168.101.11: icmp_seq=0 ttl=255 time=0 ms
64 bytes from 192.168.101.11: icmp_seq=1 ttl=255 time=0 ms

----192.168.101.11 PING Statistics----
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max = 0/0/0 ms
$ ifconfig en0 down
0821-555 ioctl (SIOCIFATTACH).: The file access permissions do not
allow the specified action.
$ ifconfig en0
en0:
flags=1e080863,480<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GR
OUPRT,64BIT,CHECKSUM_OFFLOAD(ACTIVE),CHAIN>
    inet 192.168.101.12 netmask 0xfffffc00 broadcast
    192.168.103.255
    tcp_sendspace 262144 tcp_recvspace 262144 rfc1323 1
$ ping -c2 -w 2 192.168.101.11
PING 192.168.101.11: (192.168.101.11): 56 data bytes
```

```
64 bytes from 192.168.101.11: icmp_seq=0 ttl=255 time=0 ms
64 bytes from 192.168.101.11: icmp_seq=1 ttl=255 time=0 ms
```

```
----192.168.101.11 PING Statistics----
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max = 0/0/0 ms
$
```

---

Example 8-67 on page 319 shows the netuser account using the **ifconfig** command to display the status of the en0 Ethernet interface, showing that the status is UP. The **ping** command is used to confirm the UP status and has 0 % packet loss.

The netuser account then uses the **ifconfig en0 down** command to inactivate the en0 interface.

Because the netuser account has no association with the privNetDom domain, the **ifconfig** command returns the message:

```
0821-555 ioctl (SIOCIFATTACH).: The file access permissions do not allow
the specified action.
```

The **ifconfig** command is not successful and the status of the en0 Ethernet interface remains UP.

By using this methodology, domain RBAC has restricted the netuser account to using the **ifconfig** command to manage only the en2 network interface, and excluded privileged access to the en0 network interface.

In Example 8-62 on page 315 the administrator chose to use the **setsecattr** command with the optional **conflictsets=netDom** attribute. The **conflictsets=netDom** attribute can be used to further increase the security layer within the domain RBAC security framework

Because the en0 object defines the domain attribute as privNetDom and the conflict set attribute is defined as netDom, the en0 object association will not be granted to an entity if the entity has associations to both the privNetDom and netDom domains.

In Example 8-68, the **chuser** command is used to add the privNetDom association with the netuser account. The existing association with the privDom and netDom domains are included in the **chuser** command.

*Example 8-68 The chuser command used to add the privNetDom association to the netuser account*

---

```
# chuser domains=privDom,netDom,privNetDom netuser
```

```
# lsuser -a roles netuser
netuser roles=netifconf
#
```

---

Because the **chuser** command was used to add grant the netuser account an association with the `privDom`, `netDom` and `privNetDom` domains and the `en0` object includes the conflict set between the `privNetDom` and the `netDom` domain, the netuser account will not be granted access to the `en0` object.

Example 8-69 shows the netuser account attempting to use the **ifconfig** command to deactivate the `en2` and `en0` Ethernet interfaces.

As in Example 8-65 on page 317, the **ifconfig en2 down** command is successful, because the netuser account has the `netifconf` role active and the domain RBAC configuration has been configured to allow for the operation of the **ifconfig** command on the `en2` object.

Example 8-69, the **ifconfig en0 down** command is not successful, because the `conflictsets=netDom` attribute does not allow the netuser account access to the `en0` device.

*Example 8-69 The netuser account using the ifconfig command to deactivate the en0 interface - the conflict set does not allow access to the en0 domain RBAC object*

---

```
$ id
uid=302(netuser) gid=204(netgroup) groups=1(staff)
$ rolelist -a
netifconf      aix.network.config.tcpip
$ swrole netifconf
netuser's Password:
$ ifconfig en2 down
$ ifconfig en0 down
0821-555 ioctl (SIOCIFATTACH).: The file access permissions do not
allow the specified action.
$
```

---

## 8.2 Auditing enhancements

The following sections contain a list of the enhancements for auditing.

## 8.2.1 Auditing with full pathnames

The AIX audit subsystem allows auditing of objects with full path names for certain events, such as FILE\_Open, FILE\_Read and FILE\_Write. This helps to achieve security compliance and gives complete information about the file that is being audited.

An option is provided to the **audit** command to enable auditing with full pathnames.

```
audit { on [ panic | fullpath ] | off | query | start | shutdown }{-@ wparname ...}
```

Likewise, the **audit** subroutine can also be used to enable full path auditing.

Example 8-70 shows how to enable or disable auditing with full pathnames.

*Example 8-70 Configuring auditing with full pathnames.*

---

```
# audit query
auditing off
bin processing off
audit events:
    none

audit objects:
    none

# audit start

# audit off
auditing disabled

# audit on fullpath
auditing enabled

# cat newfile1

# auditpr -v < /audit/trail |grep newfile1
    flags: 67109633 mode: 644 fd: 3 filename /tmp/newfile1
    flags: 67108864 mode: 0 fd: 3 filename /tmp/newfile1
    file descriptor = 3 filename = /tmp/newfile1

# audit query
auditing on[fullpath]
audit bin manager is process 7143522
audit events:
```



```

    general -
    FS_Mkdir,FILE_Unlink,FILE_Rename,FS_Chdir,USER_SU,PASSWORD_Change,FILE_
    Link,FS_Chroot,PORT_Locked,PORT_Change,FS_Rmdir
    .....
    .....

```

---

## 8.2.2 Auditing support for Trusted Execution

Trusted Execution (TE) offers functionalities that are used to verify the integrity of the system and implement advanced security policies, which together can be used to enhance the trust level of the complete system. The functionalities offered can be grouped into the following:

- ▶ Managing Trusted Signature Database
- ▶ Auditing integrity of the Trusted Signature Database
- ▶ Configuring Security Policies

New auditing events have been added to record security relevant information which can be analyzed to detect potential and actual violations of the system security policy.

Table 8-2 lists the audit events which have been added to audit Trusted Execution events.

*Table 8-2 audit event list*

Event	Description
TEAdd_Stnz	This event is logged whenever a new stanza is being added to the /etc/security/tsd/tsd.dat (tsd.dat) database.
TEDel_Stnz	This event is logged whenever a stanza is deleted from the tsd.dat database.
TESwitch_algo	This event is logged when a hashing algorithm is changed for a command present in the tsd.dat database.
TEQuery_Stnz	This event is logged when tsd.dat database is queried.

Event	Description
TE_Policies	This event is logged when modifying TE policies using <b>trustchk</b> command. The various TE policies are listed below together with the possible values they can take: <ul style="list-style-type: none"> <li>▶ TE ON/OFF</li> <li>▶ CHKEEXEC ON/OFF</li> <li>▶ CHKSHLIB ON/OFF</li> <li>▶ CHKSCRIPT ON/OFF</li> <li>▶ CHKKERNEXT ON/OFF</li> <li>▶ STOP_UNTRUSTD ON/OFF/TROJAN</li> <li>▶ STOP_ON_CHKFAIL ON/OFF</li> <li>▶ LOCK_KERN_POLICIES ON/OFF</li> <li>▶ TSD_FILES_LOCK ON/OFF</li> <li>▶ TEP ON/OFF</li> <li>▶ TLP ON/OFF</li> </ul>
TE_VerifyAttr	This event is logged when the user attribute verification fails.
TE_Untrusted	Reports non trusted files when they are executed
TE_FileWrite	Reports files which get opened in write mode
TSDTPolicy_Fail	Reports setting/setting of the Trusted Execution policy
TE_PermChk	Reports when Owner/Group/Mode checks fail in the kernel
TE_HashComp	Reports when crypto hash comparison fails in the kernel

### 8.2.3 Recycling Audit trail files

Audit related parameters are configured in the `/etc/security/audit/config` file. When the size of files `/audit/bin1` or `/audit/bin2` reaches the `binsize` parameter (defined in config file) it is written to `/audit/trail` file. The size of trail file is in turn limited by the size of the `/` filesystem. When the file system free space reaches the `freespace` (defined in config file) value, it will start logging the error message in `syslog`. However, incase there is no space in `/` file system, then auditing will stop without affecting the functionality of running system and error will be logged in `syslog`.

To overcome this difficulty, tunable parameters have been provided in `/etc/security/audit/config` file:

<code>backupsiz</code>	A backup of the trail file is taken when the size of trail file reaches this value. The existing trail file will be truncated. Size should be specified in units of 512-byte blocks.
<code>backuppath</code>	A valid full directory path, where backup of the trail file needs to be taken.

In the `/etc/security/audit/bincmds`, `auditcat` command will be invoked in the following ways:

```
auditcat -p -s $backupsizesize -d $backuppatherpath -o $trail $bin
```

or

```
auditcat -p -s <size value> -d <path value> -o $trail $bin
```

In the first case, it will replace the value of `$backupsizesize` and `$backuppatherpath` from values mentioned in `/etc/security/audit/config` file. In the later case it will take the actual values as specified at the command line.

Backup trail file name will be in the following format:

```
trail.YYYYMMDDThhmmss.<random number>
```

Example 8-71 shows configuration of recycling of audit trail files.

*Example 8-71 Recycling of audit trail files.*

---

```
# grep bincmds /etc/security/audit/config
      cmds = /etc/security/audit/bincmds

# cat /etc/security/audit/bincmds
/usr/sbin/auditcat -p -s 16 -d /tmp/audit -o $trail $bin

# audit start

# pwd
/tmp/audit

# ls
trail.20100826T025603.73142
```

---

**Note:** If incase copy of the trail file to newpath fails due to lack of space or any other reason, it will take the backup of trail file in the `/audit` file system (or in current file system if it is different from `/audit`, defined in config file). However, if `/audit` is full then it will not take the backup of the trail file and the legacy behavior will prevail i.e auditing will stop and error will be logged to `syslog`.

The `auditmerge` command is used to merge binary audit trails. This is especially useful if there are audit trails from several systems that need to be combined. The `auditmerge` command takes the names of the trails on the command line and sends the merged binary trail to standard output. Example 8-72 on page 326

shows use of **auditmerge** and **auditpr** commands to read the audit records from the trail files.

*Example 8-72 Merging audit trail files*

---

```
auditmerge trail.system1 trail.system2 | auditpr -v -hhe1rRtpc
```

---

## 8.2.4 Role based auditing

Auditing has been enhanced to audit events on per role basis. This capability will provide administrator with more flexibility to monitor the system based on roles.

In role based auditing, auditing events are assigned to roles which are in turn assigned to users. This can be considered equivalent to assigning the audit events for all the users having those roles. Auditing events are triggered for all users who are having the role configured for auditing.

As an example, audit events EventA and EventB are assigned to role Role1. The users User1, User2 and User3 have been assigned the role Role1. When auditing is started, events EventA and EventB will be audited for all the three users: User1, User2 and User3. Figure 8-1 represents Role based auditing.

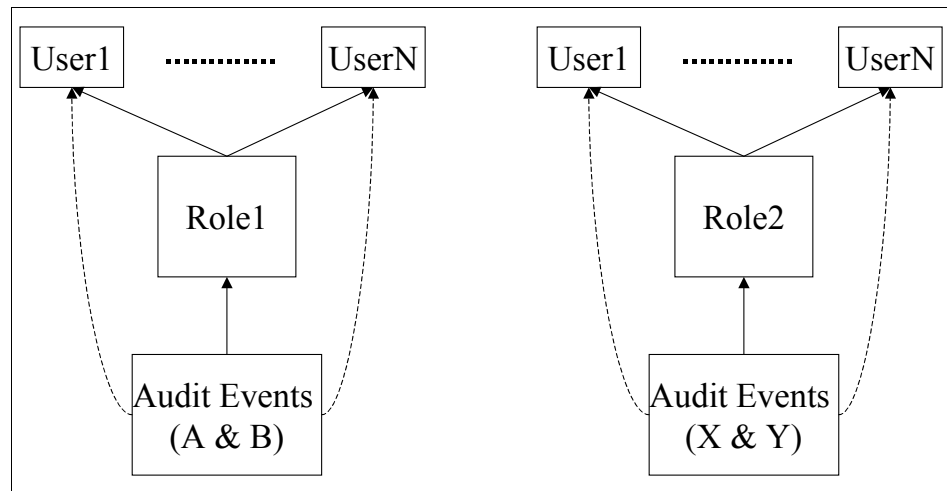


Figure 8-1 Illustration of Role based auditing

Example 8-73 shows the usage of role based auditing.

*Example 8-73*

---

```
# mkrole auditclasses=files roleA
```

---

```

# setkst
Successfully updated the Kernel Authorization Table.
Successfully updated the Kernel Role Table.
Successfully updated the Kernel Command Table.
Successfully updated the Kernel Device Table.
Successfully updated the Kernel Object Domain Table.
Successfully updated the Kernel Domains Table.

# mkuser roles=roleA default_roles=roleA userA

# passwd userA
Changing password for "userA"
userA's New password:
Enter the new password again:

# audit start

# login userA
userA's Password:
[compat]: 3004-610 You are required to change your password.
Please choose a new one.
userA's New password:
Enter the new password again:
*****
*                                                                 *
*                                                                 *
* Welcome to AIX Version 7.1!                                     *
*                                                                 *
*                                                                 *
* Please see the README file in /usr/lpp/bos for information pertinent to *
* this release of the AIX Operating System.                       *
*                                                                 *
*                                                                 *
*****

$ rolelist -e
roleA
$ exit

.....
.....
# id
uid=0(root) gid=0(system)
groups=2(bin),3(sys),7(security),8(cron),10(audit),11(lp)

# auditpr -v </audit/trail |grep userA
userA

```

```

FILE_Open      userA  OK      Thu Aug 26 02:11:02 2010 tsm
Global
FILE_Read      userA  OK      Thu Aug 26 02:11:02 2010 tsm
Global
FILE_Close     userA  OK      Thu Aug 26 02:11:02 2010 tsm
Global
....
....

```

---

## 8.2.5 Object auditing for NFS mounted files

All of the operations carried on the auditable objects residing on the NFS mounted file systems, are logged on the client, provided that there are no operations on those objects by the NFS server or by the other NFS clients or fullpath auditing is enabled on the client. If fullpath auditing is not enabled and if the file is modified by the server or by other clients, the consecutive auditing might be undefined. This behavior is corrected by restarting audit on client.

To illustrate, in the context of Network File System (NFS), if an inode is re-assigned to another file in server side, client will not be aware of it. Hence, it will keep track of the wrong files.

As a solution, if a file system is mounted on multiple clients, we recommend to audit the operations on the server to get the exact log of the events or enable the fullpath auditing on the client.

```
# audit on fullpath
```

By enabling full path auditing,

- ▶ If the file say xyz is deleted on the server and recreated with same name (with same or different inode) then the client will continue auditing it.
- ▶ If the file say xyz is deleted on the server and recreated with same inode (but with different name) then client will not audit it.

## 8.3 Propolice or Stack Smashing Protection

Stack Smashing Protection is supported on AIX since AIX 6.1 TL4 and using XLC compiler version 11. This feature can be used to minimize the risk of security vulnerabilities such as buffer overflows in AIX.

On AIX 7.1, most of the setuid programs are shipped with this feature enabled automatically and no explicit configuration is required.

For more information regarding the compiler option `-qstackprotect`, refer to the IBM XLC compiler version 11 documentation.

In Example 8-74, when the test program is compiled with `-qstackprotect` option on XLC v11 compiler and executed on AIX 6.1 TL6 or 7.1 system, buffer overflow will be detected resulting in termination of the process as shown below:

*Example 8-74 Propolice or Stack Smashing Protection*

---

```
# cat test.c
char largebuffer[34];

main()
{
    char buffer[31];

    memcpy(buffer, largebuffer, 34);
}

# ./test
*** stack smashing detected ***: program terminated
IOT/Abort trap(coredump)
```

---

**Note:** Propolice may not detect all buffer overruns. Its main goal is to prevent buffer overruns from overwriting the stack in a way that could lead to execution of malicious code. So as long as other local variables are overwritten, Propolice may not trigger.

## 8.4 Security enhancements

### 8.4.1 ODM directory permissions

The Object Data Manager (ODM) is a data manager used for storing system configuration information. On AIX, the directories and files that make up the ODM are owned by root and are part of the system group. Both owner and group have write permissions. The group write permission opens a security hole by allowing any user in the system group the ability to create and modify files. This puts the system at risk from corruption and the potential to give unauthorized access to system users.

This security vulnerability is resolved by removing the group write permissions on these two directories:

```
/etc/objrepos
```

```
/etc/lib/objrepos
```

## 8.4.2 Configurable NGROUPS\_MAX

The current hardcoded value for the maximum number of groups a user can be part of is 128. On AIX 7.1, this limit has been increased to 2048 (NGROUPS\_MAX). A new kernel parameter *ngroups\_allowed* is introduced, which can be tuned in the range of:  $128 \geq ngroups\_allowed \leq NGROUPS\_MAX$ .

The default will be 128. This tunable will allow administrators to configure the maximum number of groups users can be members of. NGROUPS\_MAX is the max value that the tunable can be set to.

The `lsattr` command shows the current *ngroups\_allowed* value. The `chdev` command is used to modify the value. The `smitty chgsys` fastpath can also be used to modify this parameter. Programatically, the `sys_parm` subroutine with the `SYSP_V_NGROUPS_ALLOWED` parameter can be used to retrieve the *ngroups\_allowed* value.

Example 8-75 shows configuring *ngroups\_allowed* parameter.

*Example 8-75 Modifying ngroups\_allowed*

---

```
# lsattr -El sys0 |grep ngroups_allowed
ngroups_allowed 128          Number of Groups Allowed
True

# chdev -l sys0 -a ngroups_allowed=2048
sys0 changed
```

---

**Note:** The system must be rebooted in order for the changes to take effect.

## 8.4.3 Kerberos client kadmind\_timeout option

When using authentication other than the KRB5 load module such as Single Sign On (SSO), there can be long delays when the kadmind server is down. This is because there are multiple kadmind connect calls for each Kerberos task, which causes multiple tcp timeouts.



To solve this problem, a new option has been introduced in the `/usr/lib/security/methods.cfg` for the KRB5 load module, `kadmind_timeout=<seconds>`. The `kadmind_timeout` option specifies the amount of time for the KRB5 load module to wait before attempting a `kadmind` connect call after a previous timeout. If `kadmind_timeout` time has not elapsed since the last timeout, then the KRB5 load module will not attempt to contact the down server. Therefore, there will only be one timeout within the `kadmind_timeout` time frame. The `KADMIND_TIMEOUT_FILE` will be used to notify all processes that there was a previous timeout. Whenever a process successfully connects to the `kadmind` server, the `KADMIND_TIMEOUT_FILE` will be deleted.

Example 8-76 shows a sample configuration from the `/usr/lib/security/methods.cfg` file.

*Example 8-76 Kerberos client kadmind\_timeout option*

---

```

/usr/lib/security/methods.cfg:

KRB5:
    program = /usr/lib/security/KRB5
    program_64 = /usr/lib/security/KRB5_64
    options = kadmind_timeout=300

KRB5files
    options = db=BUILTIN,auth=KRB5

```

---

#### 8.4.4 KRB5A load module removal

KRB5 load module handles both KRB5 and KRB5A Kerberos environments. Hence the KRB5A load module has been removed on AIX 7.1.

#### 8.4.5 Chpasswd support for LDAP

The `chpasswd` command administers users' passwords. The root user can supply or change users' passwords specified through standard input. The `chpasswd` command has been enhanced to set Lightweight Directory Access Protocol (LDAP) user passwords in an `ldap_auth` environment by specifying `-R LDAP` and not specifying the `-e` flag for encrypted format. If you specify the `-e` option for the encrypted format, the `chpasswd` command-encrypted format and LDAP server-encrypted format must match.

## 8.4.6 AIX password policy enhancements

The following are the major password policy enhancements.

### Restricting user name or regular expression in the password

The AIX password policy has been strengthened such that passwords are not allowed to contain user names or regular expressions.

User name can be disallowed in the password by adding an entry with the key word '\$USER' in the dictionary files. This key word, '\$USER' cannot be part of any word or regular expression of the entries in dictionary files.

As an example, if root user has the entry \$USER in the dictionary file say dicfile, then the root cannot have the following passwords: root, root123, abcRoot, aRooTb, etc.

Example 8-77 shows how the password can be strengthened to *not to* contain any user names.

#### *Example 8-77 Disallowing user names in passwords*

---

```
# chsec -f /etc/security/user -s default -a dictionlist=/usr/share/dict/words
# tail /usr/share/dict/words
zoom
Zorn
Zoroaster
Zoroastrian
zounds
z's
zucchini
Zurich
zygote
$USER

$ id
uid=205(tester) gid=1(staff)
$ passwd
Changing password for "tester"
tester's Old password:
tester's New password: (the password entered is "tester")
3004-335 Passwords must not match words in the dictionary.
tester's New password:
Enter the new password again:
```

---

Passwords can be further strengthened by disallowing regular expressions. This is achieved by including the regular expression in the dictionary file. To

differentiate between a word and a regular expression in the dictionary file, a regular expression will be indicated with '\*' as first character.

For example, if administrator wishes to disallow any password beginning with "pas", then he can mention the following entry in dictionary file:

```
*pas*
```

The first \* will be used to indicate a regular expression entry and remaining part will be the regular expression i.e. pas\*. Example 8-78 shows the complete procedure.

*Example 8-78 Disallowing regular expressions in passwords*

---

```
# tail /usr/share/dict/words
Zorn
Zoroaster
Zoroastrian
zounds
z's
zucchini
Zurich
zygote
$USER
*pas*

$ id
uid=205(tester) gid=1(staff)
$ passwd
Changing password for "tester"
tester's Old password:
tester's New password: (the password entered is "passw0rd")
3004-335 Passwords must not match words in the dictionary.
tester's New password:
Enter the new password again:
```

---

## Enforcing restrictions on the passwords

Passwords can be strengthened to force users to set passwords to contain following character elements:

- ▶ Uppercase Letters: A, B, C ... Z
- ▶ Lowercase Letters: a, b, c .. z
- ▶ Numbers: 0, 1, 2, ... 9
- ▶ Special Characters: ~!@#%&\*( )- \_ = + [ ] { } | \ ; : " ' , . < > ? / <space>

The following security attributes are used in this regard:

minloweralpha	Defines the minimum number of lower case alphabetic characters that must be in a new password. The value is a decimal integer string. The default is a value of 0, indicating no minimum number. The allowed range is from 0 to PW_PASSLEN.
minupperalpha	Defines the minimum number of upper case alphabetic characters that must be in a new password. The value is a decimal integer string. The default is a value of 0, indicating no minimum number. The allowed range is from 0 to PW_PASSLEN.
mindigit	Defines the minimum number of digits that must be in a new password. The value is a decimal integer string. The default is a value of 0, indicating no minimum number. The allowed range is from 0 to PW_PASSLEN.
minspecialchar	Defines the minimum number of special characters that must be in a new password. The value is a decimal integer string. The default is a value of 0, indicating no minimum number. The allowed range is from 0 to PW_PASSLEN.

The following rules are applied on these attributes, while setting the password:

- ▶ Rule 1
  - If minloweralpha > minalpha then minloweralpha=minalpha
  - If minupperalpha > minalpha then minupperalpha=minalpha
  - If minlowercase + minuppercase > minalpha then minuppercase=minalpha – minlowercase

Table 8-3 gives an example scenario for Rule 1:

*Table 8-3 Example scenario for Rule 1s*

Value set for the attributes in the /etc/security/user file			Effective value while setting the password per Rule 1		
minupperalpha	minloweralpha	minalpha	minupperalpha	minloweralpha	minalpha
2	3	7	2	3	2
8	5	7	2	5	0
5	6	7	1	6	0

- ▶ Rule 2
  - If mindigit > minother then mindigit=minother

- If minspecialchar > minother then minspecialchar=minother
- If minspecialchar + mindigit >minother then minspecialchar = minother – mindigit

Table 8-4 gives an example scenario for Rule 2:

Table 8-4 Example scenario for Rule 2

Value set for the attributes in the /etc/security/user file			Effective value while setting the password per Rule 2		
minspecialchar	mindigit	minother	minspecialchar	mindigit	minother
2	3	7	2	3	2
8	5	7	2	5	0
5	6	7	1	6	0

**Note: minother** Defines the minimum number of non-alphabetic characters in a password. The default is 0. The allowed range is from 0 to PW\_PASSLEN.

Example 8-79 shows the usage of **minloweralpha** security attribute.

Example 8-79 Usage of minloweralpha security attribute

```
# chsec -f /etc/security/user -s default -a minloweralpha=5

# grep minloweralpha /etc/security/user
* minloweralpha Defines the minimum number of lower case alphabetic characters
*   Note: If the value of minloweralpha or minupperalpha attribute is
*   attribute. If 'minloweralpha + minupperalpha' is greater than
*   'minalpha - minloweralpha'.
*   minloweralpha = 5
# chsec -f /etc/security/user -s default -a minalpha=8

# grep minalpha /etc/security/user
* minalpha      Defines the minimum number of alphabetic characters in a
*   greater than minalpha, then that attribute is reduce to minalpha
*   minalpha, then minupperalpha is reduce to
*   'minalpha - minloweralpha'.
*   'minalpha + minother', whichever is greater. 'minalpha + minother'
*   should never be greater than PW_PASSLEN. If 'minalpha + minother'
*   'PW_PASSLEN - minalpha'.
*   minalpha = 8
Changing password for "tester"
tester's Old password:
tester's New password: (the password entered is "comp")
```

3004-602 The required password characteristics are:

- a maximum of 8 repeated characters.
- a minimum of 8 alphabetic characters.
- a minimum of 5 lower case alphabetic characters.
- a minimum of 0 digits.

```
3004-603 Your password must have:
      a minimum of 8 alphabetic characters.
      a minimum of 5 lower case alphabetic characters.
tester's New password:
Enter the new password again:
$
```

---

## 8.5 Remote Statistic Interface (Rsi) client firewall support

In Rsi communication between xmservd/xmtpas and consumers, normally a random port was used by consumers. To force the consumers to open ports within the specified range, a new configuration line is introduced in AIX v7.1 and AIX 6.1 TL06. This new configuration enhancement is specified in Rsi.hosts file. Rsi agent first attempts to locate the Rsi.hosts file in the \$HOME directory. If the file is not found, attempt is made to locate the Rsi.hosts file in /etc/perf directory followed by a search in /usr/lpp/perfmgr directory.

If an Rsi.hosts file is located, specified range of ports are opened including the starting and the ending ports. If the Rsi.hosts file cannot be located in these directories or if the portrange is specified incorrectly, the Rsi communication will make use of random ports.

User can specify the port range in the Rsi.hosts file as shown below

```
portrange <start_port> <end_port>
```

As an example:

```
portrange 3001 3003
```

Once the Rsi agent is started, it makes use of the ports in the specified range. In the above example, Rsi agent will use 3001 or 3002 or 3003. In this example, Rsi agent can only listen on three ports (3001, 3002 and 3003). Subsequent Rsi communication will fail.

## 8.6 AIX LDAP authentication enhancements

AIX LDAP authentication has been enhanced with the following new features

### 8.6.1 Case sensitive LDAP user names

The LDAP uid and cn attributes are used to store user account name and group account name. Both uid and the cn attributes are defined as directory string and were case insensitive. Starting AIX 6.1 TL06 and AIX 7.1, both uid and cn can be case sensitive by enabling `caseExactAccountName` configuration parameter in `/etc/security/ldap/ldap.cfg` file. Table 8-5 provides a list of the `caseExactAccountName` values.

Table 8-5 The `caseExactAccountName` values

Name	Value	Comments
caseExactAccountName	no (Default)	case insensitive behavior
	yes	exact case match

### 8.6.2 LDAP alias support

This feature allows AIX users to login with an alias name defined in the LDAP directory entry. As an example, if a LDAP directory entry looks like the one shown in the following with an alias name `usr1`:

```
dn:uid=user1,ou=people,cn=aixdata
uid:user1
uid:usr1
objectclass:posixaccount
```

AIX LDAP authentication recognizes both uid's `user1` and `usr1`. If a command `lsuser` is run for user name `user1` or `usr1` it displays the same information as they are alias. Previously, LDAP authentication only recognized uid `user1`.

### 8.6.3 LDAP caching enhancement

AIX LDAP `secldapclntd` client daemon caches user and group entries retrieved from LDAP server. AIX 6.1 TL06 and AIX 7.1 offers the ability to control the caching mechanism through a new attribute called `TO_BE_CACHED`. This change translates into having an additional column in the existing mapping files located in `/etc/security/ldap` directory. All attributes in the LDAP mapping files

have a value of *yes* for TO\_BE\_CACHED new field by default. Administrators can selectively set an attribute to *no* to disable the caching of that attribute.

Table 8-6 provides a list of TO\_BE\_CACHED attribute values.

Table 8-6 TO\_BE\_CACHED valid attribute values

Name	Value	Comments
TO_BE_CACHED	no	LDAP client sends query directly to LDAP server
	yes(Default)	LDAP client checks it's cache before sending the query to LDAP Server

## 8.6.4 Other LDAP enhancements

The following are additional LDAP enhancements:

- ▶ AIX LDAP supports Windows® 2008 Active Directory (AD) and Active Directory application mode (ADAM).
- ▶ The `1s1dap` command lists users, groups, NIS entities (hosts, networks, protocols, services, rpc, netgroup), automount maps, and RBAC entries (authorizations, roles, privileged commands and devices). This command is extended to cover advance accounting.
- ▶ AIX LDAP module is a full functional module covering both authentication and identification. It can not be used as a authentication-only module as some customers have requested. This functionality is enhanced to have the same module support as a full functional module or an authentication only module.

## 8.7 RealSecure Server Sensor

Multi-layered prevention technology in IBM RealSecure Server Sensor for AIX guards against threats from internal and external attacks.

Refer to the below URL for further details about this product:

<http://www.ibm.com/systems/power/software/aix/security/solutions/iss.html>





# 9

## Installation, backup, and recovery

The following AIX 7.1 topics are covered in this chapter:

- ▶ 9.1, “AIX V7.1 minimum system requirements” on page 340
- ▶ 9.2, “Loopback device support in NIM” on page 346
- ▶ 9.3, “Bootlist command path enhancement” on page 348
- ▶ 9.4, “NIM thin server 2.0” on page 350
- ▶ 9.5, “Activation Engine for VDI customization ” on page 355
- ▶ 9.6, “SUMA and Electronic Customer Care integration” on page 361
- ▶ 9.7, “Network Time Protocol version 4” on page 367

## 9.1 AIX V7.1 minimum system requirements

This section discusses the minimum and recommended system requirements needed to install and run AIX V7.1.

### 9.1.1 Required hardware

Only 64-bit Common Hardware Reference Platform (CHRP) machines are supported with AIX V7.1. The following processors are supported:

- ▶ PowerPC® 970
- ▶ POWER4
- ▶ POWER5
- ▶ POWER6
- ▶ POWER7.

To determine the processor type on an AIX system you can run the **prtconf** command, as show in Example 9-1.

*Example 9-1 Using prtconf to determine the processor type of a Power system*

---

```
# prtconf | grep 'Processor Type'  
Processor Type: PowerPC_POWER7
```

---

**Note:** The RS64, POWER3™, and 604 processors, 32-bit kernel, 32-bit kernel extensions and 32-bit device drivers are not supported.

#### Minimum firmware levels

It is recommended that you update your systems to the latest firmware level before migrating to AIX V7.1. Please refer to the AIX V7.1 Release Notes for information relating to minimum system firmware levels required for AIX V7.1.

[http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.ntl/releasenotes\\_kickoff.htm](http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.ntl/releasenotes_kickoff.htm)

For the latest Power system firmware updates, please refer to the following website:

<http://www14.software.ibm.com/webapp/set2/firmware/gjsn>

#### Memory requirements

The minimum memory requirement for AIX V7.1 is 512 MB.

The current minimum memory requirements for AIX V7.1 vary based on the configuration of a system. It may be possible to configure a smaller amount of memory for a system with a very small number of devices or small maximum memory configuration.

The minimum memory requirement for AIX V7.1 may increase as the maximum memory configuration or the number of devices scales upward.

### Paging space requirements

For all *new* and *complete overwrite* installations, AIX V7.1 creates a 512 MB paging space device named `/dev/hd6`.

### Disk requirements

A minimum of 5 GB of physical disk space is required for a default installation of AIX V7.1. This includes all devices, the Graphics bundle, and the System Management Client bundle. Table 9-1 provides information relating to disk space usage with a default installation of AIX V7.1.

Table 9-1 Disk space requirements for AIX V7.1

Location	Allocated (Used)
/	196 MB (181 MB)
/usr	1936 MB (1751 MB)
/var	380 MB (264 MB)
/tmp	128 MB (2 MB)
/admin	128 MB (1 MB)
/opt	384 MB (176 MB)
/var/adm/ras/livedump	256 MB (1 MB)

**Note:** If the /tmp file system has less than 64 MB, it is increased to 64 MB during a migration installation so that the AIX V7.1 boot image can be created successfully at the end of the migration.

Starting with AIX V6.1 Technology Level 5, the boot logical volume is required to be 24 MB in size.

The pre\_migration script will check if the logical volume is the correct size. The script is located on your AIX V7.1 installation media or it can also be located in an AIX V7.1 NIM SPOT.

If necessary, the boot logical volume, hd5, size will be increased. The logical partitions must be contiguous and within the first 4 GB of the disk. If the system does not have enough free space, a message will be displayed stating there is insufficient space to extend the hd5 boot logical volume.

To install AIX V7.1, you must boot the system from the product media. The product media can be physical installation media such as DVD or it can be a NIM resource. For further information and instructions on installing AIX V7.1, please refer to the AIX Installation and Migration Guide, SC23-6722, in the AIX Information Center.

[http://publib.boulder.ibm.com/infocenter/aix/v7r1/topic/com.ibm.aix.install/doc/insgdrf/insgdrf\\_pdf.pdf](http://publib.boulder.ibm.com/infocenter/aix/v7r1/topic/com.ibm.aix.install/doc/insgdrf/insgdrf_pdf.pdf)

## AIX edition selection

It is now possible to select the edition of the AIX operating system during the base operating system (BOS) installation.

AIX V7.1 is available in three different editions:

<b>Express</b>	This edition is the default selection. It is suitable for low end Power systems for consolidating small workloads onto larger servers.
<b>Standard</b>	This edition is suitable for most workloads. It allows for vertical scalability up to 256 cores/1024 threads.
<b>Enterprise</b>	This edition includes the same features as the Standard edition but with enhanced enterprise management capabilities. IBM Systems Directory Enterprise Edition and the Workload Partitions Manager™ for AIX are included. Systems Director Enterprise Edition also includes IBM Systems Director, Active Energy Manager, VMControl, IBM Tivoli® Monitoring and Tivoli Application Dependency Discovery Manager (TADDMM).

Some of the differences between the AIX V7.1 editions are shown in Table 9-2 .

Table 9-2 AIX edition and features

AIX V7.1 Feature	Express	Standard	Enterprise
Vertical Scalability	4 cores, 8 GB per core	256 cores, 1024 Threads	256 cores, 1024 Threads
Cluster Aware AIX	Only with PowerHA	Yes	Yes
AIX Profile Manager (requires IBM Systems Director)	Management target only	Yes	Yes
AIX 5.2 Versioned WPAR support (requires the AIX 5.2 WPAR for AIX 7 product)	Yes	Yes	Yes
Full exploitation of POWER7 features	Yes	Yes	Yes
Workload Partition support	Yes	Yes	Yes
WPAR Manager and Systems Director Enterprise Edition	No	No	Yes

As shown in Example 9-2, the administrator can change the AIX edition installed by selecting 5 Select Edition from the BOS installation menu.

Example 9-2 Selecting the AIX edition during as BOS installation

---

Installation and Settings

Either type 0 and press Enter to install with current settings, or type the number of the setting you want to change and press Enter.

- ```

1 System Settings:
  Method of Installation.....New and Complete Overwrite
  Disk Where You Want to Install.....hdisk0

2 Primary Language Environment Settings (AFTER Install):
  Cultural Convention.....C (POSIX)
  Language.....C (POSIX)
  Keyboard.....C (POSIX)

```

```

3 Security Model.....Default
4 More Options (Software install options)
5 Select Edition.....express
>>> 0 Install with the settings listed above.

```

```

88 Help ? | -----
99 Previous Menu | WARNING: Base Operating System Installation will
                | destroy or impair recovery of ALL data on the
                | destination disk hdisk0.
>>> Choice [0]:

```

Possible selections are *express*, *standard*, and *enterprise*. The default value is *express*. The edition value can also be set during non-prompted NIM installations by using the `INSTALL_EDITION` field in the `control_flow` stanza of the `bosinst_data` NIM resource. The AIX edition can be modified after BOS installation using the **chedition** command, as shown in Example 9-3.

---

*Example 9-3 The chedition command flags and options*

---

```

# chedition
Usage chedition: List current edition on the system
    chedition -l

Usage chedition: Change to express edition
    chedition -x [-d Device [-p]]

Usage chedition: Change to standard edition
    chedition -s [-d Device [-p]]

Usage chedition: Change to enterprise edition
    chedition -e [-d Device [-p]]

```

---

The edition selected defines the signature file that is copied to the `/usr/lpp/bos` directory. There are three signature files included in the `bos.rte` package. The files are located in `/usr/lpp/bos/editions`. These files are used by the IBM Tivoli License Manager (ITLM) to determine the current edition of an AIX system. When an edition is selected during installation (or modified post install), the corresponding signature file is copied to the `/usr/lpp/bos` directory.

For example, to change the edition from *express* to *enterprise* you would enter the command shown in Example 9-4. You will notice that the corresponding signature file changes after the new selection.

---

*Example 9-4 Modifying the AIX edition with the chedition command*

---

```

# chedition -l

```

**standard**

```
# ls -ltr /usr/lpp/bos | grep AIX
-r--r--r-- 1 root    system      50 May 25 15:25 AIXSTD0701.SYS2
# chedition -e
chedition: The edition of the system has been changed to enterprise.
# ls -ltr /usr/lpp/bos | grep AIX
-r--r--r-- 1 root    system      50 May 25 15:25 AIXENT0701.SYS2
# chedition -l
enterprise
```

---

For further usage information relating to the **chedition** command, please refer to the command reference section in the AIX Information Center.

<http://publib.boulder.ibm.com/infocenter/aix/v7r1/topic/com.ibm.aix.cmds/doc/aixcmds1/chedition.htm>

A SMIT interface to manage AIX editions is also available with the SMIT fastpath, **smit editions**.

For further information relating to managing AIX editions, please refer to the AIX V7.1 Information Center.

[http://publib.boulder.ibm.com/infocenter/aix/v7r1/topic/com.ibm.aix.install/doc/insgdrf/sw\\_aix\\_editions.htm](http://publib.boulder.ibm.com/infocenter/aix/v7r1/topic/com.ibm.aix.install/doc/insgdrf/sw_aix_editions.htm)

## IBM Systems Director Command Agent

AIX V7.1 includes the IBM Systems Director Common Agent as part of the default install options. It is included in the System Management Client Software bundle.

When AIX is restarted, the Director agent and its prerequisite processes are automatically enabled and started. If these services are not required on a system, please follow the instructions in the AIX V7.1 Release Notes to disable them.

Please refer to the AIX V7.1 Release Notes in the AIX Information Center for additional information relating to minimum system requirements.

[http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.ntl/releasenotes\\_kickoff.htm](http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.ntl/releasenotes_kickoff.htm)

## 9.2 Loopback device support in NIM

In addition to the Activation Engine, support for loopback devices will also be implemented in NIM. This support will allow a NIM administrator to use an ISO image, in place of the AIX installation media, as a source to create lpp\_source and spot resources.

This functionality will rely on the underlying AIX loopback device feature introduced in AIX 6.1 via the loopmount command. Loopback device support was implemented in AIX 6.1, allowing system administrators to mount ISO images locally onto a system in order to read/write them.

This functionality allows to limit requirement of using the physical AIX installation media to create lpp\_source and spot resources.

### 9.2.1 Support for loopback devices during the creation of lpp\_source and spot resources

On the AIX infocenter IBM site

<http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.kerneltechref/doc/ktechrf1/kgetssystemcfg.htm>

it is specified that you can define an lpp\_source in several ways. And one is that an ISO image containing installation images can be used to create an lpp\_source by specifying its absolute path name for the source attribute. For example,

```
nim -o define -t lpp_source -a server=master -a
location=/nim/lpp_source/lpp-71 -a source=/nim/dvd.71.v1.iso lpp-71
```

would define the lpp-71 lpp\_source at /nim/lpp\_source/lpp-71 on the master NIM server using the /nim/dvd.71.v1.iso ISO image.

If a user wanted to define a spot labeled "spot-71" at "/nim/spot/spot-71" on the "master" server using the "/nim/dvd.71.v1.iso" ISO image, then the following would be executed:

```
nim -o define -t spot -a server=master -a location=/nim/spot -a
source=/nim/dvd.71.v1.iso spot-71
```



## 9.2.2 Loopmount command

The **loopmount** command is the command used to associate an image file to a loopback device and optionally make an image file available as a file system via the loopback device. It is described in infocenter at

<http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.cmds/doc/aixcmds3/loopmount.htm>

A loopback device is A device that can be used as a block device to access files. It is described in infocenter at

[http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.baseadm/doc/baseadmndita/loopback\\_main.htm](http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.baseadm/doc/baseadmndita/loopback_main.htm)

The loopback file can contain an ISO image, a disk image, a file system, or a logical volume image. For example, by attaching a CD-ROM ISO image to a loopback device and mounting it, you can access the image the same way that you can access the CD-ROM device.

Use the loopmount command to create a loopback device, to bind a specified file to the loopback device, and to mount the loopback device. Use the loopumount command to unmount a previously mounted image file on a loopback device, and to remove the device. There is no limit on the number of loopback devices in AIX. A loopback device is never created by default; you must explicitly create the device. The block size of a loopback device is always 512 bytes.

### loopmount command restrictions

The following restrictions apply to a loopback device in AIX:

- ▶ The varyonvg command on a disk image is not supported.
- ▶ A CD ISO, and DVD UDF+ISO, and other CD/DVD images are only supported in read-only format.
- ▶ An image file can be associated with only one loopback device.
- ▶ Loopback devices are not supported in workload partitions.

### Support of loopmount command in NIM

In order to create an lpp\_source or spot resource from an ISO image, NIM must be able to mount ISO images using the loopmount executable.

NIM tries to mount the ISO image using

```
/usr/sbin/loopmount -i image_pathname -m mount_point_pathname -o "-V  
cdrfs -o ro
```

If the ISO image is already mounted the loopmount will return an error.

Since umount would unmount an ISO image nothing has changed,

Add ISO image documentation to the Define a Resource smitty menu  
(nim\_mkres fastpath)

## 9.3 Bootlist command path enhancement

Configuration Path commands such as **bootlist**, **lspath**, **chpath**, **rmpath** and **mkpath** have been enhanced with Multiple PATH I/O devices (MPIO) path manipulation. It means that you can now include the pathid of a device.

### 9.3.1 Bootlist device pathid specification

The **bootlist** command includes the specification of the device pathid.

The AIX V7.1 man page for that commands mention:

*Example 9-5 Bootlist man page pathid concerns*

---

#### Purpose

Displays information about paths to a device that is capable of multiPath I/O.

#### Syntax

```
bootlist [ { -m Mode } [ -r ] [ -o ] [ [ -i ] [ -V ] [ -F ] ] [ [
-f File ] [ Device [ Attr=Value ... ] ... ] ] ] [ -v ]
```

#### Description

.....

When you specify a path ID, identify the path ID of the target disk by using the pathid attribute. You can specify one or more path IDs with the pathid attribute by entering a comma-separated list of the required paths to be added to the boot list. When the bootlist command displays information with the -o flag, the pathid attribute is included for each disk that has an associated path ID.

#### Examples

11 To specify path ID 0 on disk hdisk0 for a normal boot operation, type:

```
bootlist -m normal hdisk0 pathid=0
```

12 To specify path ID 0 and path ID 2 on disk hdisk0 for a normal boot operation, type one of the following commands:

```
bootlist -m normal hdisk0 pathid=0,2
```

```
bootlist -m normal hdisk0 pathid=0 hdisk0 pathid=2
```

---

**Note:** As the pathid argument can be repeated, both syntax pathid=0,2 and pathid=0 pathid=2 are equivalent.

Order of pathid arguments is how bootlist will process the paths. For example, pathid=2,0,1 will be different than patid=0,1,2.

The **bootlist** command display option specify the pathid information

*Example 9-6 bootlist -m normal -o command output*

---

```
# bootlist -m normal -o
hdisk0 blv=hd5 pathid=0
```

---

## 9.3.2 Common new flag for pathid configuration commands

A new flag -i flag will print paths with the specified pathid specified as argument.

*Example 9-7 lspath, rmpath and mkpath command*

---

### **lspath Command**

#### **Purpose**

Displays information about paths to an MultiPath I/O (MPIO) capable device.

#### **Syntax**

```
lspath [ -F Format | -t ] [ -H ] [ -l Name ] [ -p Parent ] [ -s
Status] [ -w Connection ] [ -i PathID ]
```

...

-i PathID

Indicates the path ID associated with the path to be displayed.

---

### **rmpath Command**

#### **Purpose**

Removes from the system a path to an MPIO capable device.

#### **Syntax**

```
rmpath [ -l Name ] [ -p Parent ] [ -w Connection ] [ -i PathID ]
```

...

-i PathID

Indicates the path ID associated with the path to be removed and is used to uniquely identify a path.

---

### **mkpath Command**

**Purpose**

Adds to the system another path to an MPIO capable device.

**Syntax**

```
mkpath [ -l Name ] [ -p Parent ] [ -w Connection ] [ -i PathID]
```

...

**-i PathID**

Indicates the path ID associated with the path to be added and is used to uniquely identify a path. This flag cannot be used with the **-d** flag.

**Note:** `lspath` command get also a new flag **-t** which allows to print information using the pathid field

**-t** : Displays the path ID in addition to the current default output. The **-t** flag cannot be used with the **-F** or the **-A** flags.

```
# lspath -t
Enabled hdisk0 vscsi0 0
Enabled hdisk1 fscsi0 0
Enabled hdisk2 fscsi0 0
Enabled hdisk3 fscsi0 0
Enabled hdisk4 fscsi0 0
```

In case there is only one pathid, `lspath` and `lspath -i 0` get same output

```
# lspath
Enabled hdisk0 vscsi0
Enabled hdisk1 fscsi0
Enabled hdisk2 fscsi0
Enabled hdisk3 fscsi0
Enabled hdisk4 fscsi0
```

## 9.4 NIM thin server 2.0

The AIX Network Installation Manager (NIM) allows managing the installation of the Base Operating System (BOS) and any optional software on one or more machines.

The NIM environment includes a server machine called master and clients which receive resources from the server.

The Network Install component has provided several options for network security and firewall enhancements but in AIX 6.1 it didn't offer a method for encrypting nor securing network data on resource servers in the NIM environment. In AIX 7.1 the NIM service handler (nimsh) provides NIM users with a client configurable option for service authentication. Support of NFS V4 allows that capability.

NFS V4 support also permit support of IPv6 network. The NIM server has been updated to support IPv6 network.

An overview of the features and its implementation is described here after.

### 9.4.1 Functional enhancements

The service authentication mainly resides in the support of NFS V4 which provides information security in the following context:

- ▶ Identification: Creation and management of the identity of any users, hosts or services
- ▶ Authentication: Validation of the identity of users, hosts or service.
- ▶ Authorization: Control of the information and data that a user or entity can access.

Some security attributes have then been added to the NIM object database for the resource objects accessed through NFS V4.

The user may specify its NFS export requirements for each NIM resource object when it is created or when changing options. The NFS protocol options available are summarized in the following table:

*Table 9-3 NFS available options*

| option   | values (default bolded) |
|----------|-------------------------|
| version  | <b>v3</b> or v4         |
| security | <b>sys</b> or krb5      |

The Kerberos configuration specified with previous krb5 flag must be created by the user. Samples are available in /usr/samples/nim/krb5 and kerberos credentials are viewable using query commands so clients can verify their credentials.

**Note:** In order to propagate the Kerberos configuration to NIM clients, the credential must be valid for NFS access when strong security is enabled.

In the IPv6 network we can find two types of addresses:

- ▶ Link-local addresses prefixed by FE80::/16 which are used by hosts on the same physical network that is when there is only one hop of communication in between nodes.
- ▶ Global address which uniquely identify a host on any network

NIM supports installation of clients on IPv6 Networks. Thin Server IPv6 network clients are also supported.

To support IPv6, NIM commands and SMIT menus have been preserved but new objects have been added

*Table 9-4* New or modified NIM objects

| Object name      | Meaning                                                                                                                                                   |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| ent6             | Represents an ethernet IPv6 network<br>IPv6 clients must be a member of this network                                                                      |
| if1 new semantic | Third field of if1 must contain the client's link-local address instead of the MAC address like with<br>If1="v6net myclient.company.com fe80:23d7::663:4" |

**Note:** For IPv6 clients, BOOTP is not used but the boot image is downloaded directly through TFTP which requires specification of a boot image file name. The convention being used is that the boot image file name is simply the hostname used by the client

TFTP support is also available since the firmware has added new SMS menus for IPv6. See an example in 9.4.5, "IPv6 Boot Firmware syntax" on page 354

## 9.4.2 Limitations

As the security options rely on exporting options for machine, network and group objects in the NIM environment, the mount options must be consistent across NFS client access:

- ▶ The user cannot mix export options for a NFS mount specification
- ▶ Only one single version support for a file system
- ▶ The user is limited to exporting NIM spot resources with an NFS security option of sys.
- ▶ The user cannot define pseudo root mappings for NFS V4 exports. The NFS default of / will be used for accessing the NIM resources.

- ▶ The NFS options are only manageable from the NIM master. NIM clients can just do queries.
- ▶ The NFS attributes of the NFS protocol called `nfs_vers` and `nfs_sec` are what user gets when mounting resources or restricting access.

**Note:** The NFS server calls the `rpc.mountd` daemon to get the access rights of each clients, so the `rpc.mountd` daemon must be running on the server even if the server only exports file systems for NFS version 4 access.

- ▶ When master and client are on the same network, link-local addresses must be used.
- ▶ When master and client are on different networks, global addresses are used as normal.
- ▶ Gateway must ALWAYS be link-local
- ▶ NIM resources that are allocated to IPv6 clients must be exported using NFS4 with the option `-a nfs_vers=4`
- ▶ Only AIX 6.1 TL1 and greater can be installed over IPv6
- ▶ Only AIX 6.1 TL6 and greater thin servers can boot over IPv6
- ▶ Only AIX 6.1 and greater can be installed at the same time as other IPv6 clients
- ▶ Static IPv6 addresses are enforced so there is no DHCP support, no support for router discovery nor service discovery

### 9.4.3 NIM commands option for NFS setting on NIM master

On the NIM master, if SMIT panels would drive you to specify the NFS options, the `nim` command is able to enable NFS client communication options:

- ▶ To enable the global use of NFS reserved ports type:

```
# nim -o change -a nfs_reserved_port=yes master
```

- ▶ To disable global usage of NFS reserved ports type:

```
# nim -o change -a nfs_reserved_port=no master
```

- ▶ To enable port checking on the NIM master NFS server type:

```
# nfso -o portcheck=1
```

- ▶ To disable port checking on the NIM master NFS server.

```
# nfso -o portcheck=0
```

## 9.4.4 Simple Kerberos server setting on NIM master NFS server

In order to use kerberos security options for NFS you need to set a kerberos server. A sample is provided in

```
/usr/samples/nim/krb5/config_rpcsec_server.
```

To create a new system user based on the principal name and password provided just type:

```
/usr/samples/nim/krb5/config_rpcsec_server -p <password> -u <user principal name>.
```

If you want to delete the Kerberos V configuration information related to the Kerberos server and principals on the NIM master NFS server, just type the following command on the NIM master:

```
# /usr/sbin/unconfig.krb5
```

**Note:** As Kerberos is relying on time a mechanism should be invoked to automatically synchronize time through the network. The NIM server must run the AIX timed daemon or an NTP daemon.

## 9.4.5 IPv6 Boot Firmware syntax

The “boot” command has changed to support IPv6 and the new format:

```
> boot
/lhea@23c00300/ethernet@23e00200:ipv6,ciaddr=FE80::214:5EFF:FE51:D5,
giaddr=FE80::20D:60FF:FE4D:C1CE,siaddr=FE80::214:5EFF:FE51:D51,
filename=mylparwar.domain.com
```

## 9.4.6 /etc/exports file syntax

The syntax of a line in the /etc/exports file is:

```
directory -option[,option]
```

The directory is the full path name of the directory. Options can designate a simple flag such as ro or a list of host names. See the specific documentation of the /etc/exports file and the **exportfs** command for a complete list of options and their descriptions.

## 9.4.7 AIX Problem Determination Tools

Numerous files and commands can be used to investigate problems.



|                 |                                                                                                                                                                    |
|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>syslogd</b>  | NFS uses the syslog to write its error and debug information. Before carrying out any problem determination, the administrator should turn syslog logging on.      |
| <b>iptrace</b>  | To examine network traffic, the developer should create an iptrace.                                                                                                |
| <b>ipreport</b> | To decode an iptrace into a readable format, the developer should use ipreport and ensure that Kerberos packets are included in the log.                           |
| <b>rpcinfo</b>  | Used to check the status of remote procedural call servers.                                                                                                        |
| <b>fuser</b>    | Used to determine mount problems. fuser lists the process numbers of local processes that use the local or remote files specified by the command's file parameter. |
| <b>lsuf</b>     | Tool available at site <a href="http://www.bullfreeware.com">http://www.bullfreeware.com</a> for listing files opened by a process.                                |
| <b>nfs4cl</b>   | Allows display of NFS v4 statistics. The command can also be used to modify current NFS v4 properties.                                                             |
| <b>nfsstat</b>  | Displays information about NFS and RPC calls.                                                                                                                      |
| <b>errpt</b>    | Can be used to determine why a daemon is not starting or core dumping during its execution.                                                                        |

## 9.5 Activation Engine for VDI customization

This feature first became available in AIX 6.1 TL 06. and documentation is available in the infocenter under the topic Activation Engine.

The main purpose of the Activation Engine (AE) is to provide a toolkit that allows one image of an AIX instance to be deployed onto many target machines

Activation Engine (AE) is a script that runs at boot time and automatically configures the system with a set of defined system parameters. These parameters are placed in the virtual image template file on the optical media.

A generic system image, such as a VDI (Virtual Data Image) or mksysb, can be used to boot multiple clients using a virtual image template. So each of those target machines can have a completely different configurations including network addresses, custom file systems, and user accounts.

## 9.5.1 Step by step usage

Activation Engine use can be summarized with the following steps

1. Enable Activation Engine on AIX system.
2. Capture a VDI using current system as the source.
3. Create virtual image templates for any systems you wish to deploy to.
4. Place virtual image templates on optical drives of the systems you are deploying to.
5. Boot the target systems using the VDI.

### Enable Activation Engine on AIX system

AIX Activation Engine is available in it's own bos.ae installp package. Content of package is listed below. It provides the command **ae** itself as well as some sample scripts.

*Example 9-8* Content of ae package

---

```
# lsllpp -f bos.ae
  Fileset                File
-----
Path: /usr/lib/objrepos
  bos.ae 7.1.0.0        /usr/samples/ae/templates
                        /usr/samples/ae/scripts/ae_accounts
                        /opt/ibm/ae/dmtf_network_bringup
                        /opt/ibm/ae/ae
                        /usr/samples/ae
                        /opt/ibm
                        /opt/ibm/ae/ae_template.xsd
                        /usr/samples/ae/scripts
                        /usr/sbin/ae -> /opt/ibm/ae/ae
                        /usr/samples/ae/scripts/ae_filesystems
                        /opt/ibm/ae
                        /usr/samples/ae/templates/ae_template.xml
                        /usr/samples/ae/scripts/ae_network
                        /opt/ibm/ae/ae_template.dtd
```

---

The first step is to enable and configure AE on a target system. Enabling is done by running the **ae -o enable** command as shown in Example 9-9 on page 357 which will create an AE entry called aengine to /etc/inittab. That entry will be executed at boot time during init 2-9 steps.

*Example 9-9 .Enabling activation engine*

---

```
# ae -o enable
Activation Engine was successfully enabled.
Using template 'ae_template.xml' from first available optical media.
# grep engine /etc/inittab
aengine:23456789:wait:/usr/sbin/ae -o run ae_template.xml
```

---

The argument `ae_template.xml` is the name of the XML template which will be read from optical media at boot time. It is the default name but will be the template name specified as argument to the **ae -o enable** command. See the command syntax in Example 9-10

*Example 9-10 The Activation Engine command syntax*

---

```
# ae
USAGE: /usr/sbin/ae -o {enable | disable | status | check | run}

enable <template> - Enable the Activation Engine
disable - Disable the Activation Engine
status - Print current status of Activation Engine
check <template> - Validate a user created template against the
Activation Engine schema
run <template> - Execute the activation engine against a particular
template file
```

---

## Capture a VDI using current system as the source

The goal is to make an image of your current system. This is the image you will use to deploy to other systems. The target system must have the Activation Engine enabled so you can customize specific parameters at boot time. This capture step is usually performed using VM Control, which is one of the main consumers of AE.

It can also be done using usual `mksysb` command or NIM menus.

**Note:** Image creation must be performed after Activation Engine has been enabled.

## Create a virtual image template

Since each system gets its own network address, its custom users, and its filesystem you usually need to have a separate template file for each system you want to deploy to. It must be placed on optical media, which must be mountable by Activation Engine at boot time.

Configuration information is clearly separated in two type of files:

- ▶ The data are contained in a XML template files
- ▶ The code run as scripts that perform actions using the data extracted from XML template files.

The template file example `/usr/samples/ae/templates/ae_template.xml` listed below in Example 9-12 references `/user_template1.xml` as well as the scripts associated with the network,user and file systems sections as seen in that grep output shown in Example 9-11:

`/user_template1.xml` is the user created template file

*Example 9-11 Grep of script in user created template file.*

---

```
<!--<section name="network" script="ae_network">
<section name="accounts" script="ae_accounts">
<section name="filesystems" script="ae_filesystems">
```

---

These default scripts are available in `/usr/samples/ae/scripts`.

*Example 9-12 Sample script /usr/samples/ae/templates/ae\_template.xml*

---

```
# cat /usr/samples/ae/templates/ae_template.xml
<?xml version="1.0" encoding="UTF-8"?>
<template name="Sample Activation Engine template">
  <settings>
    <!-- log directory is created automatically if it doesn't exist
-->
    <logDirectory>/var/adm/ras/ae</logDirectory>
    <!-- / is assumed to be / dir of optical media -->
    <scriptsDirectory>/scripts</scriptsDirectory>
    <!-- Here we specify all user created templates that we want AE
to execute, in order. scripts are defined within -->
    <extensions>

<!--<extendedTemplate>/user_template1.xml</extendedTemplate>-->
    </extensions>
  </settings>
  <rules>
    <!-- the following section was commented to out prevent
accidental execution -->
    <!-- script paths are assumed to be relative to / directory of
optical media -->
    <!--<section name="network" script="ae_network">
      <ruleSet>
        <hostname>hostname.domain</hostname>
        <interface>en0</interface>
```

```

        <address>XX.XX.XX.XX</address>
        <mask>255.255.254.0</mask>
        <gateway>XX.XX.XX.0</gateway>
        <domain>hostname.domain</domain>
        <nameserver>XX.XX.XX.XX</nameserver>
        <start_daemons>yes</start_daemons>
    </ruleSet>
</section>
<section name="accounts" script="ae_accounts">
    <ruleSet>
        <username>username</username>
        <groups>sys</groups>
        <admin>>true</admin>
        <home>/home/apuzic</home>
    </ruleSet>
</section>
<section name="filesystems" script="ae_filesystems">
    <ruleSet>
        <mountpoint>/usr/testmount</mountpoint>
        <type>jfs2</type>
        <volume_group>rootvg</volume_group>
        <size>16M</size>
    </ruleSet>
</section>-->
</rules>
</template>

```

---

**Note:** A template can reference as many scripts as it wants, as long as all those scripts are present on the optical media.

## Creating AE scripts

Creating Activation Engine scripts is easier than creating the template files.

However scripts creation must follow the three distinct guidelines:

- ▶ The scripts must accept parameters defined in <ruleSets> tags of the template file that calls them. (See Example 9-12 on page 358).
- ▶ They must not pipe STDOUT or STDERR to any external files as Activation Engine is supposed to pipe both of those to the specified log files. This makes debugging and status tracking easier.
- ▶ The script must return 0 on a successful execution. Any other return code is interpreted as a failure.

**Note:** Each template can also link to other template files, which allows for further flexibility. For example, a user can create one template to customize all network parameters on the system, another to create new filesystems, and another to add new custom user accounts and groups. This allows for easier categorization of customized data. It also makes it easier to add new customized data to the image because the user can create a new template and have one of the existing templates point to the newly created file, which is a small change

## Checking virtual image templates

Running `ae -o check template_name` against your own template will check your XML file against the schema file and will tell you if there's any errors. It is suggested you do this before trying to use your template files, to make sure you're not trying to use the Activation Engine with an invalid template file in a production environment. A successful check is performed in Example 9-13.

*Example 9-13 Successful Activation Engine template file structure check*

```
# cp /usr/samples/ae/templates/ae_template.xml /
# ae -o check ae_template.xml
Template 'ae_template.xml' is valid AE template
# cp /usr/samples/ae/scripts/* /
```

**Note:** The `ae -o check` command will only check syntax of XML file but not the data content. It will not check the availability of the scripts files being referenced in that XML.

## Place virtual image templates

Once a valid XML template and a corresponding shell script has been created, the Activation Engine needs to be enabled on the target system.

By running `ae -o enable template_file` command we are telling AE to enable itself to run at next boot-up through an inittab entry. It will execute the processing of the XML template called `template_file`.

**Note:** we didn't have to specify any scripts to run. The scripts are all defined and referenced in the XML template file itself.

The template file has to be located in the root directory of the rootvg disc.

**Note:** Activation Engine checks all bootable optical media for virtual image templates and uses the first one found. So if you are booting a VDI on a system with two (or more) optical discs, and all discs have virtual image templates then AE will use the first template it finds on any of the mounted discs.

### Boot the target systems using the VDI

As Activation Engine is executed at boot time through the inittab entry, the scripts will be executed at that point and will only perform configurations limited to the boot stage. For example we cannot expect to install new filesets using AE.

**Note:** Activation Engine is only intended to be used for system configuration, not any modifications to the user space.

## 9.6 SUMA and Electronic Customer Care integration

In August 2004 AIX V5.3 introduced the Service Update Management Assistant (SUMA) tool which allows system administrators to automate the download of maintenance updates such as Maintenance Levels (MLs), Technology Levels (TLs) and Service Packs (SPs). In the AIX V5.3 and AIX V6.1 releases SUMA uses the undocumented *fixget* interface to initiating a standard multipart data HTTP POST transaction to the URL where the fix server's fixget script resides to retrieve AIX updates. The fix server's URL is configured through the `FIXSERVER_URL` parameter of the SUMA global configuration settings during the base configuration and can be viewed by the `suma -c` command. Example 9-14 shows the `suma -c` command output on an AIX V6.1 TL 6100-05 system after a SUMA base configuration has been performed.

*Example 9-14 SUMA default base configuration on AIX V6.1*

---

```
# suma -c
FIXSERVER_PROTOCOL=http
DOWNLOAD_PROTOCOL=ftp
DL_TIMEOUT_SEC=180
DL_RETRY=1
MAX_CONCURRENT_DOWNLOADS=5
HTTP_PROXY=
HTTPS_PROXY=
FTP_PROXY=
SCREEN_VERBOSE=LVL_INFO
NOTIFY_VERBOSE=LVL_INFO
LOGFILE_VERBOSE=LVL_VERBOSE
```

```
MAXLOGSIZE_MB=1
REMOVE_CONFLICTING_UPDATES=yes
REMOVE_DUP_BASE_LEVELS=yes
REMOVE_SUPERSEDE=yes
TMPDIR=/var/suma/tmp
FIXSERVER_URL=www14.software.ibm.com/webapp/set2/fixget
```

---

A usage message for the fixget script is given by the URL:

<http://www14.software.ibm.com/webapp/set2/fixget>

when entered in the address field of a web browser. Note that the fixget utility is not intended for direct customer use but is rather called internally by the SUMA tool.

Beginning with AIX V7.1 SUMA no longer uses fixget but utilizes the Electronic Customer Care (eCC) services to retrieve AIX updates.

IBM Electronic Customer Care services are strategically designed to offer a centralized access point to code updates for IBM Systems. Independent of a given platform similar terminology and application programming interfaces enable a standardized user interface with a consistent usage environment.

Currently eCC provides an update repository for instances like Power Systems Firmware, Hardware Management Console (HMC), IBM BladeCenter, Linux, IBM i and now also for AIX 7. The eCC Common Client's Java API is used as a common interface by all supported platforms to download the updates. In AIX V7.1 the eCC Common Client functionality is available through the bos.ecc\_client.rte fileset. The same fileset is also required to support the IBM Electronic Service Agent™ (ESA) and the Inventory Scout utility on AIX. This means that on AIX 7, SUMA, ESA, and the Inventory Scout are all consumers of the same eCC Common Client and share the eCC code, the libraries and the connectivity settings. However, each of the named utilities will run individually in a separate Java Virtual Machine.

## 9.6.1 SUMA installation on AIX 7

As in previous AIX releases the SUMA code is delivered through the bos.suma fileset. But on AIX 7 this fileset is not installed by default because as it is no longer included in the /usr/sys/inst.data/sys\_bundles/BOS.autoi file. In AIX 7 the bos.suma fileset is contained in the graphics software bundle (Graphics.bnd) and the system management software bundle (SystemMgmtClient.bnd). Both predefined system bundles are located in the /usr/sys/inst.data/sys\_bundles/ directory. The bos.suma fileset requires the installation of bos.ecc\_client.rte



fileset which in turn needs the support of Java 6 through the Java6.sdk fileset. Both, SUMA and eCC rely on the support of the Perl programming language. The following `ls1pp` command output shows the fileset dependencies of SUMA and eCC:

```
75011p01: /> ls1pp -p bos.suma bos.ecc_client.rte
  Fileset                Requisites
-----
Path: /usr/lib/objrepos
  bos.ecc_client.rte 7.1.0.0
                        *ifreq bos.rte 7.1.0.0
                        *prereq perl.rte 5.10.1.0
                        *prereq perl.libext 2.3.0.0
                        *prereq Java6.sdk 6.0.0.200
  bos.suma 7.1.0.0     *prereq bos.rte 7.1.0.0
                        *prereq bos.ecc_client.rte 7.1.0.0
                        *prereq perl.rte 5.8.2.0
                        *prereq perl.libext 2.1.0.0

Path: /etc/objrepos
  bos.ecc_client.rte 7.1.0.0
                        *ifreq bos.rte 7.1.0.0
                        *prereq perl.rte 5.10.1.0
                        *prereq perl.libext 2.3.0.0
                        *prereq Java6.sdk 6.0.0.200
  bos.suma 7.1.0.0     *prereq bos.rte 7.1.0.0
                        *prereq bos.ecc_client.rte 7.1.0.0
                        *prereq perl.rte 5.8.2.0
                        *prereq perl.libext 2.1.0.0
```

## 9.6.2 AIX 7 SUMA functional and configuration differences

The SUMA implementation in AIX V7.1 is governed by the following two guidelines:

1. IBM AIX operating system release and service strategy
2. Electronic Customer Care cross-platform service strategy for IBM Systems

The current AIX service strategy was introduced in 2007 and requires fixpacks like Technology Levels (TL) or Service Packs (SP) to be downloaded in a single entity. The download of individual fixes or filesets is no longer supported. SUMA in AIX 7 adheres to this service strategy and supports the following request type (RqType) values for the `suma` command only:

**ML** Request to download a specific maintenance or technology level.

<b>TL</b>	Request to download a specific technology level. The TL must be specified by the full name, for example 6100-03-00-0920 instead of 6100-03.
<b>SP</b>	Request to download a specific service pack. The SP must be specified by the full name, for example 6100-02-04-0920 instead of 6100-04-04.
<b>PTF</b>	Request to download a Program Temporary Fix (PTF). Only certain PTFs may be downloaded as an individual fileset. For example, PTFs containing bos.rte.install, bos.alt_disk_install.rte, or PTFs that come out in between service packs. Otherwise, the TL or SP must be downloaded.
<b>Latest</b>	Request to download the latest fixes. This RqType value returns the latest service pack of the TL specified in the FilterML field of the suma command. The FilterML field specifies a technology level to filter against; for example, 6100-03. If not specified, the value returned by oslevel -r on the local system will be used.

The following request type (RqType) values are obsolete and are no longer supported on AIX 7:

<b>APAR</b>	Request to download an APAR.
<b>Critical</b>	Request to download the latest critical fixes.
<b>Security</b>	Request to download the latest security fixes.
<b>Fileset</b>	Request to download a specific fileset.

Also, the field FilterSysFile which was once used to filter against the inventory of a running system is not supported on AIX 7 anymore.

The integration of SUMA and Electronic Customer Care has only been implemented on AIX 7 and not on any of the previous AIX releases. Nevertheless SUMA on AIX 7 can be used to download AIX V5.3 TL 5300-06 and newer updates. AIX V5.3 TL 5300-06 has been released in June 2007 and is the starting level of updates which are loaded into the eCC update repository.

The conversion of SUMA to use eCC instead of fixget has significant impact on the supported protocols utilized for fix server communication and to download updates. The following protocol specific characteristics and changes are related to the relevant SUMA configuration parameters:

**FIXSERVER\_PROTOCOL** The FIXSERVER\_PROTOCOL parameter specifies the protocol to be used for communication between the eCC Common Client and the eCC fix service provider as apart of the order request that SUMA will make to get the list of fixes. SUMA will utilize the Hypertext Transfer Protocol Secure

(HTTPS) protocol since it is the only supported protocol for communication between the eCC Common Client and the IBM fix service provider. The only allowed value for this configuration setting is https. The http setting of previous AIX releases is no longer supported.

**DOWNLOAD\_PROTOCOL**The `DOWNLOAD_PROTOCOL` parameter specifies the protocol to be used for communication by the eCC Common Client for a download request from SUMA. SUMA takes advantage of the secure and multi-threaded Download Director Protocol (DDP) if the Hypertext Transfer Protocol (HTTP) has been configured. The HTTP protocol is specified by default and is recommended as eCC protocol for downloading updates. The related value for this configuration setting is http. The `suma` command can be used to modify the default configuration to use the HTTP Secure (HTTPS) protocol for downloads. But the related https setting restricts the secure downloads to single-threaded operations. The ftp setting of previous AIX releases is no longer supported.

Example 9-15 shows the `suma -c` command output on an AIX V7.1 TL 7100-00 system after a SUMA base configuration has been performed.

*Example 9-15 SUMA default base configuration on AIX V7.1*

---

```
75011p01:/> suma -c
FIXSERVER_PROTOCOL=https
DOWNLOAD_PROTOCOL=http
DL_TIMEOUT_SEC=180
DL_RETRY=1
HTTP_PROXY=
HTTPS_PROXY=
SCREEN_VERBOSE=LVL_INFO
NOTIFY_VERBOSE=LVL_INFO
LOGFILE_VERBOSE=LVL_VERBOSE
MAXLOGSIZE_MB=1
REMOVE_CONFLICTING_UPDATES=yes
REMOVE_DUP_BASE_LEVELS=yes
REMOVE_SUPERSEDE=yes
TMPDIR=/var/suma/tmp
```

---

The SUMA related eCC specific base configuration properties are stored in the eccBase.properties file under the directory /var/suma/data. The initial version of the eccBase.properties file is installed as part of the bos.suma fileset. Example 9-16 shows the content of the eccBase.properties file after a SUMA default base configuration has been done on an AIX 7 system.

*Example 9-16 eccBase.properties file after SUMA default base configuration*

---

```
75011p01:/> cat /var/suma/data/eccBase.properties
## ecc version: 1.0504
#Thu Apr 08 09:02:56 CDT 2010
DOWNLOAD_READ_TIMEOUT=180
INVENTORY_COLLECTION_CONFIG_DIR=/var/suma/data
DOWNLOAD_RETRY_WAIT_TIME=1
TRACE_LEVEL=SEVERE
DOWNLOAD_SET_NEW_DATE=TRUE
AUDITLOG_MAXSIZE_MB=2
CONNECTIVITY_CONFIG_DIR=/var/ecc/data
PLATFORM_EXTENSION_CLASS=com.ibm.esa.ea.tx.ecc.PlatformExtensions
TRACE_FILTER=com.ibm.ecc
WS_TRACE_LEVEL=OFF
AUDITLOG_COUNT=2
TRACELOG_MAXSIZE_MB=4
DOWNLOAD_MAX_RETRIES=3
LOG_DIR=/var/suma/log
RETRY_COUNT=1
DOWNLOAD_MONITOR_INTERVAL=10000
REQUEST_TIMEOUT=600
```

---

The CONNECTIVITY\_CONFIG\_DIR variable in the eccBase.properties file points to the directory where the connectivity configuration information is stored in the eccConnect.properties file. An initial version of the eccConnect.properties file is installed as part of the bos.ecc\_client.rte fileset in the /var/ecc/data directory. The eccConnect.properties file connectivity configuration information is shared by SUMA, IBM Electronic Service Agent and the Inventory Scout. This file holds the proxy server information if required for the service communication.

The proxy configuration task is supported by the SMIT panels which are dedicated to set up an AIX service configuration. System administrators can use the **smit srv\_conn** fastpath to directly access the Create/Change Service Configuration menu. In this menu the Create/Change Primary Service Configuration selection will bring up the Create/Change Primary Service Configuration menu where the desired connection type can be configured. The following three alternatives are available for the connection type: Not configured, Direct Internet, and HTTP\_Proxy. For the connection type HTTP\_Proxy selection

you need to provide the IP address of the proxy server, the port number used, and an optional authentication user ID. Up to two additional service configurations (secondary, and tertiary) are supported to backup the primary connection in case of a failure. Note that the HTTP\_PROXY selection in SMIT supports both HTTP\_PROXY and HTTPS\_PROXY if the customer proxy server is configured to support both http and https.

## 9.7 Network Time Protocol version 4

The Network Time Protocol (NTP) is an Internet protocol used to synchronize the clocks of computers to some time reference, usually the Coordinated Universal Time (UTC). NTP is an Internet standard protocol originally developed by Professor David L. Mills at the University of Delaware.

The NTP version 3 (NTPv3) internet draft standard is formalized in the Request for Comments (RFC) 1305 (Network Time Protocol (Version 3) Specification, Implementation and Analysis). NTP version 4 (NTPv4) is a significant revision of the NTP standard, and is the current development version. NTPv4 has not been formalized but is described in the proposed standard RFC 5905 (Network Time Protocol Version 4: Protocol and Algorithms Specification).

The NTP subnet operates with a hierarchy of levels, where each level is assigned a number called the stratum. Stratum 1 (primary) servers at the lowest level are directly synchronized to national time services. Stratum 2 (secondary) servers at the next higher level are synchronize to stratum 1 servers and so on. Normally, NTP clients and servers with a relatively small number of clients do not synchronize to public primary servers. There are several hundred public secondary servers operating at higher strata and are the preferred choice.

According to a 1999 survey<sup>1</sup> of the NTP network there were at least 175,000 hosts running NTP in the internet. Among these there were over 300 valid stratum 1 servers. In addition there were over 20,000 servers at stratum 2, and over 80,000 servers at stratum 3.

Beginning with AIX V7.1 and AIX V6.1 TL 6100-06 the AIX operating system supports NTP version 4 in addition to the older NTP version 3. The AIX NTPv4 implementation is based on the port of the ntp-4.2.4 version of the Internet Systems Consortium (ISC) code and is in full compliance with RFC 2030 (Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI).

---

<sup>1</sup> Source: *A Survey of the NTP Network*, found at <http://alumni.media.mit.edu/~nelson/research/ntp-survey99>

Additional information about the Network Time Protocol project, the Internet Systems Consortium, and the Request for Comments can be found at:

- ▶ <http://www.ntp.org/>
- ▶ <http://www.isc.org/>
- ▶ <http://www.rfcs.org/>

As in previous AIX releases the NTPv3 code is included with the bos.net.tcp.client fileset which is provided on the AIX product media and installed by default. The new NTPv4 functionality is delivered through the ntp.rte and the ntp.man.en\_US filesets of the AIX Expansion Pack.

The ntp.rte fileset for the NTP runtime environment installs the following NTPv4 programs under the /usr/sbin/ntp4 directory:

<b>ntptrace4</b>	Perl script that traces a chain of NTP hosts back to their master time source.
<b>sntp4</b>	SNTP client which queries a NTP server and displays the offset time of the system clock with respect to the server clock.
<b>ntpq4</b>	Standard NTP query program.
<b>ntp-keygen4</b>	Command which generates public and private keys.
<b>ntpd4</b>	Special NTP query program.
<b>ntpdate4</b>	Sets the date and time using the NTPv4 protocol.
<b>ntpd4</b>	NTPv4 daemon.

System administrators can use the **ls1pp** command to get a full listing of the ntp.rte fileset content:

```
75011p01:sbin/ntp4> ls1pp -f ntp.rte
Fileset          File
```

```
-----
Path: /usr/lib/objrepos
ntp.rte 6.1.6.0  /usr/lib/nls/msg/en_US/ntpdate4.cat
                /usr/lib/nls/msg/en_US/ntpq4.cat
                /usr/sbin/ntp4/ntptrace4
                /usr/sbin/ntp4/sntp4
                /usr/sbin/ntp4/ntpq4
                /usr/sbin/ntp4/ntp-keygen4
                /usr/sbin/ntp4/ntpd4
                /usr/sbin/ntp4/ntpdate4
                /usr/lib/nls/msg/en_US/ntpd4.cat
                /usr/lib/nls/msg/en_US/ntpd4.cat
```

```

/usr/sbin/ntp4
/usr/lib/nls/msg/en_US/libntp4.cat
/usr/sbin/ntp4/ntpd4

```

The NTPv3 and NTPv4 binaries can coexist on an AIX system. The NTPv3 functionality is installed by default through the bos.net.tcp.client fileset and the commands are placed in the /usr/sbin/ntp3 subdirectory. During the installation process a set of default symbolic links are created in the /usr/sbin directory to map the NTP commands to the NTPv3 binaries. Consequently AIX points by default to NTPv3 binaries.

If the system administrator likes to use the NTPv4 services the default symbolic links have to be changed manually to point to the appropriate commands under the /usr/sbin/ntp4 directory after the NTPv4 code has been installed from the AIX Expansion Pack. Table 9-5 provides a list of the NTPv4 binaries, the NTPv3 binaries, and the default symbolic links on AIX.

Table 9-5 NTP binaries directory mapping on AIX

NTPv4 binaries in /usr/sbin/ntp4	NTPv3 binaries in /usr/sbin/ntp3	Default symbolic links to NTPv3 binaries from /usr/sbin directory
ntpd4	xntpd	/usr/sbin/xntpd --> /usr/sbin/ntp3/xntpd
ntpdate4	ntpdate	/usr/sbin/ntpdate --> /usr/sbin/ntp3/ntpdate
ntpdc4	xntpdc	/usr/sbin/ntpdc --> /usr/sbin/ntp3/xntpdc
ntpq4	ntpq	/usr/sbin/ntpq --> /usr/sbin/ntp3/ntpq
ntp-keygen4	Not available	/usr/sbin/ntp-keygen --> /usr/sbin/ntp4/ntp-keygen4
ntptrace4	ntptrace	/usr/sbin/ntptrace --> /usr/sbin/ntp3/ntptrace
sntp4	sntp	/usr/sbin/sntp --> /usr/sbin/ntp3/sntp

In comparison with the NTPv3 protocol the utilization of NTPv4 offers improved functionality, and many new features and refinements. A comprehensive list which summarizes the differences between the NTPv4 and the NTPv3 version is provided by the *NTP Version 4 Release Notes* which can be found at:

<http://www.eecis.udel.edu/~mills/ntp/html/release.html>

The following list is an extract of the release notes which gives an overview of the new features pertaining to AIX.

1. Support for the IPv6 addressing family. If the Basic Socket Interface Extensions for IPv6 (RFC 2553) is detected, support for the IPv6 address

- family is generated in addition to the default support for the IPv4 address family.
2. Most calculations are now done using 64-bit floating double format, rather than 64-bit fixed point format. The motivation for this is to reduce size, improve speed and avoid messy bounds checking.
  3. The clock discipline algorithm has been redesigned to improve accuracy, reduce the impact of network jitter and allow increased in poll intervals to 36 hours with only moderate sacrifice in accuracy.
  4. The clock selection algorithm has been redesigned to reduce “clockhopping” when the choice of servers changes frequently as the result of comparatively insignificant quality changes.
  5. This release includes support for Autokey public-key cryptography, which is the preferred scheme for authenticating servers to clients. [...]
  6. The OpenSSL cryptographic library has replaced the library formerly available from RSA Laboratories. All cryptographic routines except a version of the MD5 message digest routine have been removed from the base distribution.
  7. NTPv4 includes three new server discovery schemes, which in most applications can avoid per-host configuration altogether. Two of these are based on IP multicast technology, while the remaining one is based on crafted DNS lookups. [...]
  8. This release includes comprehensive packet rate management tools to help reduce the level of spurious network traffic and protect the busiest servers from overload. [...]
  9. This release includes support for the orphan mode, which replaces the local clock driver for most configurations. Orphan mode provides an automatic, subnet-wide synchronization feature with multiple sources. It can be used in isolated networks or in Internet subnets where the servers or Internet connection have failed. [...]
  10. There are two new burst mode features available where special conditions apply. One of these is enabled by the **iburst** keyword in the server configuration command. It is intended for cases where it is important to set the clock quickly when an association is first mobilized. The other is enabled by the **burst** keyword in the server configuration command. It is intended for cases where the network attachment requires an initial calling or training procedure. [...]
  11. The reference clock driver interface is smaller, more rational and more accurate.
  12. In all except a very few cases, all timing intervals are randomized, so that the tendency for NTPv3 to self-synchronize and bunch messages, especially with a large number of configured associations, is minimized.



13. Several new options have been added for the **ntpd** command line. For the inveterate knob twiddlers several of the more important performance variables can be changed to fit actual or perceived special conditions. In particular, the **tos** and **tos** commands can be used to adjust thresholds, throw switches and change limits.
14. The **ntpd** daemon can be operated in a one-time mode similar to **ntpdate**, which program is headed for retirement. [...]





# National language support

AIX Version 7.1 continues to extend the number of nations and regions supported under its national language support. In this chapter, details on the following locales (provided alphabetically) and facilities are provided:

- ▶ 10.1, “Unicode 5.2 support” on page 374
- ▶ 10.2, “Code set alias name support for iconv converters” on page 374
- ▶ 10.3, “NEC selected characters support in IBM-eucJP” on page 375

## 10.1 Unicode 5.2 support

As part of the continuous ongoing effort to adhere to the most recent industry standards, AIX V7.1 provides the necessary enhancements to the existing Unicode locales in order to bring them up to compliance with the latest version of the Unicode standard, which is Version 5.2, as published by the Unicode Consortium.

The Unicode is a standard character coding system for supporting the worldwide interchange, processing, and display of the written texts of the diverse languages used throughout the world. Since November 2007 AIX V6.1 supports Unicode 5.0 which defines standardized character positions for over 99,000 glyphs in total. More than 8,000 additional code points have been defined in Unicode 5.1 (1624 code points, April 2008) and Unicode 5.2 (6,648 code points, October 2009). AIX V7.1 provides the necessary infrastructure to handle, store and transfer all Unicode 5.2 characters.

For in-depth information about Unicode 5.2, visit the official Unicode home page at:

<http://www.unicode.org>

## 10.2 Code set alias name support for iconv converters

National Language Support (NLS) provides a base for internationalization in which data often can be changed from one code set to another. Support of several standard converters for this purpose is provided by AIX and the following conversion interfaces are offered by any AIX system:

**iconv command** Allows you to request a specific conversion by naming the FromCode and ToCode code sets.

**libiconv functions** Allows applications to request converters by name.

AIX can transfer, store and convert data in more than 130 different code sets. In order to meet market requirements and standards, the number of code sets has been increases dramatically by different vendors, organizations, and standard groups in the past decade. However, many code sets are maintained and named in different ways. This may raise code set alias name issues. A code set with specific encoding scheme can have two or more different code set names in different platforms or applications. For instance, ISO-8859-13 is an Internet Assigned Numbers Authority (IANA) registered code set for Estonian, a Baltic Latin language. The code set ISO-8859-13 is also named as IBM-921, CP921, ISO-IR-179, windows-28603, LATIN7, L7, 921, 8859\_13 and 28603 in different

platforms. For obvious interoperability reasons it is desirable to provide an alias name mapping function in the AIX /usr/lib/libiconv.a library to unambiguously identify code sets to the AIX converters.

AIX 7 introduces an AIX code set mapping mechanism in libiconv.a which holds more than 1300 code set alias names base on code set and alias names of different vendors, applications and open source groups. Mayor contributions are based on code sets related to the International Components for Unicode (ICU), Java, Linux, WebSphere®, and many others.

Using the new alias name mapping function, iconv can now easily map ISO-8859-13, CP921, ISO-IR-179, windows-28603, LATIN7, L7, 921, 8859\_13 or 28603 to IBM-921 (AIX default) and convert the data properly, for example. The code set alias name support for iconv converters is entirely transparent to the system and no initialization or configuration is required on behave of the system administrator.

## 10.3 NEC selected characters support in IBM-eucJP

There are 83 Japanese characters known as *NEC selected characters*. NEC selected characters refers to a proprietary encoding of Japanese characters historically established by the Nippon Electric Company (NEC) corporation. NEC selected characters have been supported by previous AIX releases through the IBM-943 and UTF-8 code sets.

For improved interoperability and configuration flexibility AIX V7.1 and the related AIX V6.1 TL 6100-06 release extend the NEC selected characters support to the IBM-eucJP code set used for the AIX ja\_JP local.

The corresponding AIX Japanese input method and the dictionary utilities were enhanced to accept NEC selected characters in the ja\_JP local and all IBM-eucJP code set related iconv converters were updated to handle the newly added characters.

Table 10-1 shows the local (language\_territory designation) and code set combinations of which all are now supporting NEC selected characters:

*Table 10-1 Locales and code sets supporting NEC selected character*

Local	Local code set	Full local name	Category
JA_JP	UTF-8	JA_JP.UTF-8	Unicode
ja_JP	IBM-eucJP	ja_JP.IBM-eucJP	Extended UNIX Code (EUC)

Local	Local code set	Full local name	Category
Ja_JP	IBM-943	Ja_JP.IBM-943	PC

Requirements and specifications for Japanese character sets can be found at the official web site of the Japanese Industrial Standards Committee:

<http://www.jisc.go.jp/>



# Hardware and graphics support

This chapter discusses the new hardware support and graphic topics new in AIX Version 7.1, arranged by the following topics:

- ▶ 11.1, “X11 Font Updates” on page 378
- ▶ 11.2, “AIX V7.1 storage device support” on page 387
- ▶ 11.3, “Hardware support” on page 392

## 11.1 X11 Font Updates

AIX V7.1 contains font updates for X11 and the Common Desktop Environment (CDE) to properly exploit the latest TrueType fonts. These fonts are licensed from the Monotype Imaging company (<http://www.monotypeimaging.com>).

Existing fonts and their X Logical Font Description (XLFD) family names have changed to match the names provided by Monotype. To preserve binary compatibility with prior releases of AIX, symbolic links have been provided to redirect the original file names to the new file names. Additionally, font aliases have been added to redirect the original XLFD names to the new names.

In Table 11-1, the original and new file and XLFD family names are shown.

Table 11-1 TrueType Fonts original and new XLFD family and file names

Original File Name	Original XLFD Family	New File Name	New XLFD Family	Width	Localization
tnrwt_j.ttf	TimesNewRomanWT	wt__j__b.ttf	wt serif j	proportional	Japanese
tnrwt_k.ttf	TimesNewRomanWT	wt__k__b.ttf	wt serif k	proportional	Korean
tnrwt_s.ttf	TimesNewRomanWT	wt__s__b.ttf	wt serif sc	proportional	Simplified Chinese
tnrwt_t.ttf	TimesNewRomanWT	wt__tt__b.ttf	wt serif tw	proportional	Traditional Chinese (Taiwan)
mtsans_j.ttf	SansWT	wts__j__b.ttf	wt sans j	proportional	Japanese
mtsans_k.ttf	SansWT	wts__k__b.ttf	wt sans k	proportional	Korean
mtsans_s.ttf	SansWT	wts__s__b.ttf	wt sans k	proportional	Simplified Chinese
mtsans_t.ttf	SansWT	wts__tt__b.ttf	wt sans tw	proportional	Traditional Chinese (Taiwan)
mtsansdj.ttf	SansMonoWT	wtsdj__b.ttf	wt sans duo j	monospaced	Japanese
mtsansdk.ttf	SansMonoWT	wtsdk__b.ttf	wt sans duo k	monospaced	Korean
mtsansds.ttf	SansMonoWT	wtsds__b.ttf	wt sans duo sc	monospaced	Simplified Chinese



Original File Name	Original XLFD Family	New File Name	New XLFD Family	Width	Localization
mtsansdt.ttf	SansMonoWT	wtsdt_b.ttf	wt sans duo tw	monospaced	Traditional Chinese (Taiwan)
MTSanXBA.ttf	SansMonoWTExtB	wtsdsxb_.ttf	wt sans duo extb	monospaced	Simplified Chinese

In Table 11-2, the corresponding fileset name, glyph list and CDE usage are shown for each new file name.

Table 11-2 Updated TrueType Font file names, fileset packages, glyph list and CDE usage

New File Name	Packaging Fileset	Glyph List	New CDE Usage
wt_j_b.ttf	X11.fnt.ucs.ttf	complete	none by default
wt_k_b.ttf	X11.fnt.ucs.ttf_KR	complete	none by default
wt_s_b.ttf	X11.fnt.ucs.ttf_CN	complete	"-dt-interface system-" font for "ucs2.india" encoding
wt_tt_b.ttf	X11.fnt.ucs.ttf_TW	complete	none by default
wts_j_b.ttf	X11.fnt.ucs.ttf	complete	none by default
wts_k_b.ttf	X11.fnt.ucs.ttf_KR	complete	none by default
wts_s_b.ttf	X11.fnt.ucs.ttf_CN	complete	none by default
wts_tt_b.ttf	X11.fnt.ucs.ttf_TW	complete	none by default
wtsdj_b.ttf	X11.fnt.ucs.ttf	complete	none by default
wtsdk_b.ttf	X11.fnt.ucs.ttf_KR	complete	none by default
wtsds_b.ttf	X11.fnt.ucs.ttf_CN	complete	"-dt-interface user-" font for "ucs2.india" encoding
wtsdt_b.ttf	X11.fnt.ucs.ttf_TW	complete	none by default
wtsdsxb_.ttf	X11.fnt.ucs.ttf_extb	Unicode Extension B	"-dt-interface user-" font for "unicode-2" encoding "-dt-interface system-" font for "unicode-2" encoding

Additional fonts for East Asian glyph and subset fonts and their new file and XLFD family names are shown in Table 11-3 on page 380.

Table 11-3 Additional East Asian XLFD family and file names

File Name	XLFD Family	Width	Localization	Glyph list
wt_j_eb.ttf	wt serif j ea	proportional	Japanese	East Asian subset
wt_k_eb.ttf	wt serif k ea	proportional	Korean	East Asian subset
wt_s_eb.ttf	wt serif sc ea	proportional	Simplified Chinese	East Asian subset
wt_tteb.ttf	wt serif tw ea	proportional	Traditional Chinese (Taiwan)	East Asian subset
wts_j_eb.ttf	wt sans j ea	proportional	Japanese	East Asian subset
wts_k_eb.ttf	wt sans k ea	proportional	Korean	East Asian subset
wts_s_eb.ttf	wt sans sc ea	proportional	Simplified Chinese	East Asian subset
wts_tteb.ttf	wt sans tw ea	proportional	Traditional Chinese (Taiwan)	East Asian subset
wtsdj_eb.ttf	wt sans duo j ea	monospaced	Japanese	East Asian subset
wtsdk_eb.ttf	wt sans duo k ea	monospaced	Korean	East Asian subset
wtsds_eb.ttf	wt sans duo sc ea	monospaced	Simplified Chinese	East Asian subset
wtsdtteb.ttf	wt sans duo tw ea	monospaced	Traditional Chinese (Taiwan)	East Asian subset

In Table 11-4, the corresponding East Asian subset font files names and CDE usage are shown for each additional font.

Table 11-4 East Asian subset font file names, files packages and CDE usage

File Name	Packaging Fileset Name	CDE Usage
wt_j_eb.ttf	X11.fnt.ucs.ttf	default "-dt-interface system-" font
wt_k_eb.ttf	X11.fnt.ucs.ttf_KR	"-dt-interface system-" font for "ucs2.cjk_korea" encoding
wt_s_eb.ttf	X11.fnt.ucs.ttf_CN	"-dt-interface system-" font for "ucs2.cjk_china" encoding
wt_tteb.ttf	X11.fnt.ucs.ttf_TW	"-dt-interface system-" font for "ucs2.cjk_taiwan" encoding
wts_j_eb.ttf	X11.fnt.ucs.ttf	none by default
wts_k_eb.ttf	X11.fnt.ucs.ttf_KR	none by default

File Name	Packaging Fileset Name	CDE Usage
wts_s_eb.ttf	X11.fnt.ucs.ttf_CN	none by default
wts_tteb.ttf	X11.fnt.ucs.ttf_TW	none by default
wtsdj_eb.ttf	X11.fnt.ucs.ttf	default "-dt-interface user-" font
wtsdk_eb.ttf	X11.fnt.ucs.ttf_KR	"-dt-interface user-" font for "ucs2.cjk_korea" encoding
wtsds_eb.ttf	X11.fnt.ucs.ttf_CN	"-dt-interface user-" font for "ucs2.cjk_china" encoding
wtsdtteb.ttf	X11.fnt.ucs.ttf_TW	"-dt-interface user-" font for "ucs2.cjk_taiwan" encoding

Table 11-5 lists the additional Middle Eastern localized glyph subset fonts available in AIX V7.1.

*Table 11-5 Middle Eastern glyph subset XLFD family and file names.*

File Name	XLFD Family	Width	Localization	Glyph List
wt_m____.ttf	wt serif me	proportional	Middle East	Middle East subset
wts_m____.ttf	wt sans me	proportional	Middle East	Middle East subset
wtsdm____.ttf	wt sans duo me	monospaced	Middle East	Middle East subset

The packaging fileset names and CDE usage for the additional Middle Eastern font file names are listed in Table 11-6

*Table 11-6 Middle Eastern font file names, fileset packages and CDE usage*

File Name	Packaging Fileset	CDE Usage
wt_m____.ttf	X11.fnt.ucs.ttf_ME	"-dt-interface system-" font for "ucs2.me" encoding X
wts_m____.ttf	X11.fnt.ucs.ttf_ME	none by default
wtsdm____.ttf	X11.fnt.ucs.ttf_ME	"-dt-interface user-" font for "ucs2.me" encoding

Table 11-7 lists the additional Hong Kong localized fonts.

*Table 11-7 Additional Hong Kong XLFD family and file names*

File Name	XLFD Family	Width	Localization	Glyph List
wt__th_b.ttf	wt serif hk	proportional	Traditional Chinese (Hong Kong)	complete

File Name	XLFD Family	Width	Localization	Glyph List
wt__theb.ttf	wt serif hk ea	proportional	Traditional Chinese (Hong Kong)	East Asian subset
wts_th_b.ttf	wt sans hk	proportional	Traditional Chinese (Hong Kong)	complete
wts_theb.ttf	wt sans hk ea	proportional	Traditional Chinese (Hong Kong)	East Asian subset
wtsdth_b.ttf	wt sans duo hk	monospaced	Traditional Chinese (Hong Kong)	complete
wtsdtheb.ttf	wt sans duo hk ea	monospaced	Traditional Chinese (Hong Kong)	East Asian subset

The packaging fileset names and CDE usage for the additional Hong Kong font file names are listed in Table 11-8

Table 11-8 Additional Hong Kong file names, fileset packages, and CDE usage.

File Name	Packaging Fileset	CDE Usage
wt__th_b.ttf	X11.fnt.ucs.ttf_HK	none by default
wt__theb.ttf	X11.fnt.ucs.ttf_HK	"-dt-interface system-" font for "ucs2.cjk_hongkong" encoding
wts_th_b.ttf	X11.fnt.ucs.ttf_HK	none by default
wts_theb.ttf	X11.fnt.ucs.ttf_HK	none by default
wtsdth_b.ttf	X11.fnt.ucs.ttf_HK	none by default
wtsdtheb.ttf	X11.fnt.ucs.ttf_HK	"-dt-interface user-" font for "ucs2.cjk_hongkong" encoding

The Windows glyph list (WGL) has been removed in AIX V7.1. This glyph is already a subset of other fonts. It is not necessary to provide fonts which contain only the WGL. Table 11-9 lists the file names that have been removed.

Table 11-9 Removed WGL file names and fileset packages

File Name	Packaging Fileset
mtsans_w.ttf	X11.fnt.ucs.ttf
mtsansdw.ttf	X11.fnt.ucs.ttf
tnrwt_w.ttf	X11.fnt.ucs.ttf

The X11 font library and rasterizer code have both been modified to enable the use of 4-byte font indexes (UCS-32 font indexes). This was required for the latest Unicode Extension B font (referenced by file name `wtsdsxb_.ttf`).

A consideration with Glyph subsets and the CDE. If one glyph in a font extends higher or lower than others, the font metrics will be affected such that a paragraph of text will appear to have excessive white space between each line.

To address this issue, the “-dt interface user-” and “-dt interface system-” font aliases used by CDE in Unicode locales will, by default, point to fonts containing a reduced set of glyphs. This reduced set does not contain the large glyphs causing increased line height.

To override this default and force the use of fonts containing the complete set of glyphs, add `/usr/lib/X11/fonts/TrueType/complete` to the front of your font path, so that the “-dt” font aliases in that directory are found before the ones in `/usr/lib/X11/fonts/TrueType`.

For example, if a user selects the `EN_US` locale at CDE login, but still needs to be able to display Indic characters, they can run the following command:

```
# xset +fp /usr/lib/X11/fonts/TrueType/complete
```

Note that an alternative would be for that user to have actually selected the `EN_IN` locale at CDE login instead of `EN_US`. Refer to the `/usr/lpp/X11/README` file for more information.

The following summarizes the modified, additional and associated requisite X11 filesets in AIX V7.1.

### **Modified filesets**

X11.fnt.ucs.ttf (AIXwindows Unicode TrueType Fonts).

X11.fnt.ucs.ttf\_CN (AIXwindows Unicode TrueType Fonts - CJK China).

X11.fnt.ucs.ttf\_KR (AIXwindows Unicode TrueType Fonts - CJK Korea).

X11.fnt.ucs.ttf\_TW (AIXwindows Unicode TrueType Fonts - CJK Taiwan).

X11.fnt.ucs.ttf\_extb (AIXwindows Unicode TrueType Fonts - Extension B).

### **Additional filesets**

X11.fnt.ucs.ttf\_ME (AIXwindows Unicode TrueType Fonts - Middle East).

X11.fnt.ucs.ttf\_HK (AIXwindows Unicode TrueType Fonts - CJK Hong Kong).

### Requisite filesets

Unicode Hong Kong locales (X11.loc.\*\_HK.base.rte) will require X11.fnt.ucs.ttf\_HK.

Unicode Arabic locales (X11.loc.AR\_\*.base.rte) will require X11.fnt.ucs.ttf\_ME.

Unicode Hebrew locales (X11.loc.HE\_\*.base.rte) will require X11.fnt.ucs.ttf\_ME.

Unicode Indic locales (X11.loc.\*\_IN.base.rte) will require X11.fnt.ucs.ttf\_CN.

X11.fnt.ucs.ttf\_ME will require X11.fnt.ucs.ttf\_CN.

Example 11-1 displays the corresponding `ls1pp` command output for the X11.fnt.ucs.ttf fileset names and their file contents in AIX V7.1:

#### *Example 11-1 X11 lpp package content*

```
# oslevel
7.1.0.0
# ls1pp -l X11.fnt.ucs.ttf
Fileset          Level State      Description
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf  7.1.0.0 COMMITTED  AIXwindows Unicode TrueType
                          Fonts
# ls1pp -f X11.fnt.ucs.ttf
Fileset          File
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf 7.1.0.0
                          /usr/lpp/X11/lib/X11/fonts/TrueType/wts_j_eb.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/courth.ttf
                          /usr/lpp/X11/lib/X11/fonts
                          /usr/lpp/X11/lib/X11/fonts/TrueType/wtsdj_b.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/mtsans_w.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wts_j_b.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.alias.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/mtsans_j.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wts_j_b.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/complete
                          /usr/lpp/X11/lib/X11/fonts/TrueType/wtsdj_eb.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/mtsansdw.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wtsdj_b.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/mtsansdj.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wtsdj_b.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/complete/fonts.alias.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/tnrwt_w.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wt_j_b.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/wt_j_b.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/tnrwt_j.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wt_j_b.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/helvth.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/timeth.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/wts_j_b.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType
```

```

        /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.scale.ttf
        /usr/lpp/X11/lib/X11/fonts/TrueType/wt__j_eb.ttf
# ls1pp -l X11.fnt.ucs.ttf_CN
Fileset                Level  State      Description
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_CN      7.1.0.0  COMMITTED  AIXwindows Unicode TrueType
                        Fonts - CJK China
# ls1pp -f X11.fnt.ucs.ttf_CN
Fileset                File
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_CN 7.1.0.0
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wtsds__b.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.scale.ttf_CN
                        /usr/lpp/X11/lib/X11/fonts/TrueType/mtsansds.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wtsds__b.ttf
                        /usr/lpp/X11/lib/X11/fonts
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wtsds_eb.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.alias.ttf_CN
                        /usr/lpp/X11/lib/X11/fonts/TrueType/complete/fonts.alias.ttf_CN
                        /usr/lpp/X11/lib/X11/fonts/TrueType/tnrwt_s.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wt__s__b.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/complete
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wt__s__b.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wts__s__b.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wt__s__eb.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wts__s__eb.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType
                        /usr/lpp/X11/lib/X11/fonts/TrueType/mtsans_s.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wts__s__b.ttf
# ls1pp -l X11.fnt.ucs.ttf_KR
Fileset                Level  State      Description
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_KR      7.1.0.0  COMMITTED  AIXwindows Unicode TrueType
                        Fonts - CJK Korea
# ls1pp -f X11.fnt.ucs.ttf_KR
Fileset                File
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_KR 7.1.0.0
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wt__k_eb.ttf
                        /usr/lpp/X11/lib/X11/fonts
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wts__k_eb.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.scale.ttf_KR
                        /usr/lpp/X11/lib/X11/fonts/TrueType/mtsans_k.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wts__k__b.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wtsdk__b.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.alias.ttf_KR
                        /usr/lpp/X11/lib/X11/fonts/TrueType/mtsansdk.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wtsdk__b.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/complete
                        /usr/lpp/X11/lib/X11/fonts/TrueType/complete/fonts.alias.ttf_KR
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wtsdk_eb.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/tnrwt_k.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wt__k__b.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wt__k__b.ttf
                        /usr/lpp/X11/lib/X11/fonts/TrueType
                        /usr/lpp/X11/lib/X11/fonts/TrueType/wts__k__b.ttf
# ls1pp -l X11.fnt.ucs.ttf_TW
Fileset                Level  State      Description
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_TW      7.1.0.0  COMMITTED  AIXwindows Unicode TrueType
                        Fonts - CJK Taiwan
# ls1pp -f X11.fnt.ucs.ttf_TW

```

```

Fileset          File
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_TW 7.1.0.0
    /usr/lpp/X11/lib/X11/fonts
    /usr/lpp/X11/lib/X11/fonts/TrueType/wtsdtt_b.ttf
    /usr/lpp/X11/lib/X11/fonts/TrueType/tnrwt_t.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wt__tt_b.ttf
    /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.scale.ttf_TW
    /usr/lpp/X11/lib/X11/fonts/TrueType/wtsdtt_eb.ttf
    /usr/lpp/X11/lib/X11/fonts/TrueType/complete
    /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.alias.ttf_TW
    /usr/lpp/X11/lib/X11/fonts/TrueType/complete/fonts.alias.ttf_TW
    /usr/lpp/X11/lib/X11/fonts/TrueType/wt__tt_b.ttf
    /usr/lpp/X11/lib/X11/fonts/TrueType/mtsans_t.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wts__tt_b.ttf
    /usr/lpp/X11/lib/X11/fonts/TrueType/wts__tt_b.ttf
    /usr/lpp/X11/lib/X11/fonts/TrueType
    /usr/lpp/X11/lib/X11/fonts/TrueType/wt__tteb.ttf
    /usr/lpp/X11/lib/X11/fonts/TrueType/mtsansdt.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wtsdtt_b.ttf
    /usr/lpp/X11/lib/X11/fonts/TrueType/wts__tteb.ttf
# lspp -f X11.fnt.ucs.ttf_ME
Fileset          File
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_ME 7.1.0.0
    /usr/lpp/X11/lib/X11/fonts/TrueType/wts_m__.ttf
    /usr/lpp/X11/lib/X11/fonts
    /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.scale.ttf_ME
    /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.alias.ttf_ME
    /usr/lpp/X11/lib/X11/fonts/TrueType/complete
    /usr/lpp/X11/lib/X11/fonts/TrueType/complete/fonts.alias.ttf_ME
    /usr/lpp/X11/lib/X11/fonts/TrueType/wtsdm__.ttf
    /usr/lpp/X11/lib/X11/fonts/TrueType
    /usr/lpp/X11/lib/X11/fonts/TrueType/wt__m__.ttf
# lspp -l X11.fnt.ucs.ttf_ME
Fileset          Level  State      Description
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_ME 7.1.0.0  COMMITTED  AIXwindows Unicode TrueType
                    Fonts - Middle East
# lspp -f X11.fnt.ucs.ttf_ME
Fileset          File
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_ME 7.1.0.0
    /usr/lpp/X11/lib/X11/fonts/TrueType/wts_m__.ttf
    /usr/lpp/X11/lib/X11/fonts
    /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.scale.ttf_ME
    /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.alias.ttf_ME
    /usr/lpp/X11/lib/X11/fonts/TrueType/complete
    /usr/lpp/X11/lib/X11/fonts/TrueType/complete/fonts.alias.ttf_ME
    /usr/lpp/X11/lib/X11/fonts/TrueType/wtsdm__.ttf
    /usr/lpp/X11/lib/X11/fonts/TrueType
    /usr/lpp/X11/lib/X11/fonts/TrueType/wt__m__.ttf
# lspp -l X11.fnt.ucs.ttf_HK
Fileset          Level  State      Description
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_HK 7.1.0.0  COMMITTED  AIXwindows Unicode TrueType
                    Fonts - CJK Hong Kong
# lspp -f X11.fnt.ucs.ttf_HK
Fileset          File
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_HK 7.1.0.0

```



```

/usr/lpp/X11/lib/X11/fonts/TrueType/wtsdth_b.ttf
/usr/lpp/X11/lib/X11/fonts
/usr/lpp/X11/lib/X11/fonts/TrueType/fonts.scale.ttf_HK
/usr/lpp/X11/lib/X11/fonts/TrueType/fonts.alias.ttf_HK
/usr/lpp/X11/lib/X11/fonts/TrueType/wtsdtheb.ttf
/usr/lpp/X11/lib/X11/fonts/TrueType/complete/fonts.alias.ttf_HK
/usr/lpp/X11/lib/X11/fonts/TrueType/complete
/usr/lpp/X11/lib/X11/fonts/TrueType/wt__th_b.ttf
/usr/lpp/X11/lib/X11/fonts/TrueType/wts__th_b.ttf
/usr/lpp/X11/lib/X11/fonts/TrueType/wt__theb.ttf
/usr/lpp/X11/lib/X11/fonts/TrueType
/usr/lpp/X11/lib/X11/fonts/TrueType/wts__theb.ttf

# ls|pp -l X11.fnt.ucs.ttf_extb
Fileset          Level  State      Description
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_extb 7.1.0.0 COMMITTED AIXwindows Unicode TrueType
                          Fonts - Extension B

# ls|pp -f X11.fnt.ucs.ttf_extb
Fileset          File
-----
Path: /usr/lib/objrepos
X11.fnt.ucs.ttf_extb 7.1.0.0
                          /usr/lpp/X11/lib/X11/fonts/TrueType/MTSanXBA.ttf ->
/usr/lpp/X11/lib/X11/fonts/TrueType/wtsdsxb_.ttf
                          /usr/lpp/X11/lib/X11/fonts
                          /usr/lpp/X11/lib/X11/fonts/TrueType/wtsdsxb_.ttf
                          /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.scale.extB
                          /usr/lpp/X11/lib/X11/fonts/TrueType/fonts.alias.extB
                          /usr/lpp/X11/lib/X11/fonts/TrueType

```

---

## 11.2 AIX V7.1 storage device support

AIX v7.1 includes expands the support for many IBM and vendor storage products.

The IBM System Storage® Interoperation Center (SSIC) provides a matrix for support listing operating system support for the various IBM and vendor storage products.

The System Storage Interoperation Center can be used to produce a matrix showing supported features and products by selecting search options including:

- ▶ Operating system
- ▶ Operating system technology level
- ▶ Connection protocol
- ▶ Host platform
- ▶ Storage product family

The System Storage Interoperation Centre can be found at the URL:

[http://www.ibm.com/systems/support/storage/config/ssic/displaysssearchwithoutjs.wss?start\\_over=yes](http://www.ibm.com/systems/support/storage/config/ssic/displaysssearchwithoutjs.wss?start_over=yes)

**Note:** At the time of publication, the SSIC was in the process of being updated to include support information for the AIX V7.1 release.

Figure 11-1 shows the System Storage Interoperation Centre.

Figure 11-1 The IBM System Storage Interoperation Centre

By making selections from the drop-down boxes, the SSIC may be used to determine which features and products are available and supported for AIX V7.1.

In Figure 11-2 on page 390 multiple features and products are selected, which restricts the display results to combinations of these features and products.

**Note:** The SSIC is updated regularly as feature and product offerings are added or removed. This search example was accurate at the time of publication but may change as features are added or removed.

IBM System Storage Interoperation Center (SSIC)

SSIC Education and Help

Please view the details of the interoperability configurations queried. This requires exporting the data, or clicking the Submit button at the bottom of the search interface, then clicking on the details link in the results table.

Revise Selected Criteria - click link below to change search query

(1) Operating System, (2) Product Family, (3) Host Platform, (4) Connection Protocol, (5) Product Model, (6) Server Model, (7) HBA Vendor, (8) Multipathing, (9) SAN Vendor, (10) HBA Model, (11) Product Version

New Search Configuration Results - 17

Product Family  
IBM System Storage Enterprise Tape  
IBM System Storage Entry Disk  
IBM System Storage LTO Ultrium Tape

Product Model  
DS8700  
DS8100DS8300

Product Version  
XIV Storage System (10.2)

Export Selected Product Version (xls)

Host Platform  
IBM System p  
IBM BladeCenter

Connection Protocol  
FC/CEE

HBA Vendor  
QLogic

SAN Vendor  
Brocade  
CISCO  
McDATA

Clustering  
IBM PowerHA 5.4.1  
IBM PowerHA 5.5  
Oracle RAC 11g  
Symantec Veritas Cluster Server 5.0

Storage Controller (SVC only)

Operating System  
IBM AIX 6.1 TL4  
IBM AIX 6.1  
IBM IFLS 1.0  
IBM OS/400 V4.5.3

Server model

HBA Model  
FC 1977  
FC 5716  
FC 5758

SAN Model  
IBM F08 (3534-F08)  
IBM F16 (2109-F16)  
IBM SAN 140M (2027-140)  
IBM SAN158-R (2005-R18)

Multipathing  
Symantec Veritas Volume Manager with DMP 5.0  
Symantec Veritas Volume Manager with DMP 5.1

Intercluster SAN Router (SVC only)

New Search Configuration Results - 17 Submit

ISV Applications  
ISV Solutions Resource Library: <http://www-03.ibm.com/systems/storage/solutions/isv/index.html>

Request for Price Quotations (RPQ)  
If a desired configuration is not available for selection in the above form, an RPQ should be submitted to IBM to request approval. To submit an RPQ, contact your local IBM Storage Specialist or Business Partner.

Legal Disclaimer  
The information provided in this document is provided "AS IS" without warranty of any kind, including any warranty of merchantability, fitness for a particular purpose, interoperability or compatibility. IBM does not provide service or support for the non-IBM products listed. For support issues regarding non-IBM products, please contact the manufacturer of the product directly. This information could include technical inaccuracies or typographical errors. IBM does not assume any liability for damages caused by such errors as this information is provided for the reader's convenience only.

Last accessed: Mon, 30 Aug 2010 10:34:27 Eastern Daylight Time, EDT

About IBM Privacy Contact Terms of use IBM Feeds Jobs

Figure 11-2 The IBM System Storage Interoperation Centre - search example

The product version output from the System Storage Interoperation Center may be exported into a .xls format spreadsheet.

Figure 11-3 on page 391 shows an example search with the Export Selected Product Version (xls) selection option identified. The Export Selected Product Version (xls) is shown highlighted.

Figure 11-3 The IBM System Storage Interoperation Centre - the export to .xls option

Using the System Storage Interoperation Centre can benefit system designers when determining which features are available when designing new hardware and software architecture.

The System Storage Interoperation Centre can also be used as an entry reference point by storage and system administrators to determine prerequisite hardware or software dependencies when planning for upgrades to existing environments.

The System Storage Interoperation Centre (SSIC) is not intended to replace such tools as the IBM System Planning Tool (SPT) for POWER® processor based systems or the IBM Fix Level Recommendation Tool (FLRT) For IBM POWER systems administrators. The SSIC should be used in conjunction with such tools as the SPT and FLRT, as well as any additional planing and architecture tools specific to your specific environment.

## 11.3 Hardware support

This chapter discusses the new hardware support and graphic topics new in AIX Version 7.1.

### 11.3.1 Hardware support

AIX V7.1 exclusively supports 64-bit Common Hardware Reference Platform (CHRP) machines with selected processors:

- ▶ PowerPC 970
- ▶ POWER4
- ▶ POWER5
- ▶ POWER6
- ▶ POWER7

The **prtconf** command can be used to determine the processor type of the managed system hardware platform.

Example 11-2 show the root user running the **prtconf** command:

*Example 11-2 prtconf command to determine the processor type of the system*

---

```
# whoami
root
# prtconf|grep 'Processor Type'
Processor Type: PowerPC_POWER7
#
```

---

The **prtconf** command run by LPAR in Example 11-2 shows that the processor type of the managed system hardware platform is POWER7.

To determine whether your managed system hardware platform may require Firmware updates or additional prerequisites in order to run AIX V7.1, refer to the AIX V7.1 Release Notes, found at:

[http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.ntl/releasenotes\\_kickoff.htm](http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.ntl/releasenotes_kickoff.htm)





# Abbreviations and acronyms

<b>ABI</b>	Application Binary Interface	<b>CD-ROM</b>	Compact Disk-Read Only Memory
<b>AC</b>	Alternating Current	<b>CDE</b>	Common Desktop Environment
<b>ACL</b>	Access Control List	<b>CEC</b>	Central Electronics Complex
<b>ACLs</b>	Access Control Lists	<b>CHRP</b>	Common Hardware Reference Platform
<b>AFPA</b>	Adaptive Fast Path Architecture	<b>CID</b>	Configuration ID
<b>AIO</b>	Asynchronous I/O	<b>CLDR</b>	Common Locale Data Repository
<b>AIX</b>	Advanced Interactive Executive	<b>CLI</b>	Command-Line Interface
<b>APAR</b>	Authorized Program Analysis Report	<b>CLVM</b>	Concurrent LVM
<b>API</b>	Application Programming Interface	<b>CLiC</b>	CryptoLight for C library
<b>ARP</b>	Address Resolution Protocol	<b>CMW</b>	Compartmented Mode Workstations
<b>ASMI</b>	Advanced System Management Interface	<b>CPU</b>	Central Processing Unit
<b>AltGr</b>	Alt-Graphic	<b>CRC</b>	Cyclic Redundancy Check
<b>Azeri</b>	Azerbaijan	<b>CSM</b>	Cluster Systems Management
<b>BFF</b>	Backup File Format	<b>CT</b>	Component Trace
<b>BIND</b>	Berkeley Internet Name Domain	<b>CUoD</b>	Capacity Upgrade on Demand
<b>BIST</b>	Built-In Self-Test	<b>DAC</b>	Discretionary Access Controls
<b>BLV</b>	Boot Logical Volume	<b>DCEM</b>	Distributed Command Execution Manager
<b>BOOTP</b>	Boot Protocol	<b>DCM</b>	Dual Chip Module
<b>BOS</b>	Base Operating System	<b>DES</b>	Data Encryption Standard
<b>BSD</b>	Berkeley Software Distribution	<b>DGD</b>	Dead Gateway Detection
<b>CA</b>	Certificate Authority	<b>DHCP</b>	Dynamic Host Configuration Protocol
<b>CAA</b>	Cluster Aware AIX	<b>DLPAR</b>	Dynamic LPAR
<b>CATE</b>	Certified Advanced Technical Expert	<b>DMA</b>	Direct Memory Access
<b>CD</b>	Compact Disk	<b>DNS</b>	Domain Name Server
<b>CD</b>	Component Dump facility		
<b>CD-R</b>	CD Recordable		

<b>DR</b>	Dynamic Reconfiguration	<b>HACMP™</b>	High Availability Cluster Multiprocessing
<b>DRM</b>	Dynamic Reconfiguration Manager	<b>HBA</b>	Host Bus Adapters
<b>DST</b>	Daylight Saving Time	<b>HMC</b>	Hardware Management Console
<b>DVD</b>	Digital Versatile Disk	<b>HPC</b>	High Performance Computing
<b>DoD</b>	Department of Defense	<b>HPM</b>	Hardware Performance Monitor
<b>EC</b>	EtherChannel	<b>HTML</b>	Hypertext Markup Language
<b>ECC</b>	Error Checking and Correcting	<b>HTTP</b>	Hypertext Transfer Protocol
<b>eCC</b>	Electronic Customer Care	<b>Hz</b>	Hertz
<b>EGID</b>	Effective Group ID	<b>I/O</b>	Input/Output
<b>EOF</b>	End of File	<b>IBM</b>	International Business Machines
<b>EPOW</b>	Environmental and Power Warning	<b>ICU</b>	International Components for Unicode
<b>EPS</b>	Effective Privilege Set	<b>ID</b>	Identification
<b>eRAS</b>	enterprise Reliability Availability Serviceability	<b>IDE</b>	Integrated Device Electronics
<b>ERRM</b>	Event Response Resource Manager	<b>IEEE</b>	Institute of Electrical and Electronics Engineers
<b>ESA</b>	Electronic Service Agent	<b>IETF</b>	Internet Engineering Task Force
<b>ESS</b>	Enterprise Storage Server®	<b>IGMP</b>	Internet Group Management Protocol
<b>EUC</b>	Extended UNIX Code	<b>IANA</b>	Internet Assigned Numbers Authority
<b>EUID</b>	Effective User ID	<b>IP</b>	Internetwork Protocol
<b>F/C</b>	Feature Code	<b>IPAT</b>	IP Address Takeover
<b>FC</b>	Fibre Channel	<b>IPL</b>	Initial Program Load
<b>FCAL</b>	Fibre Channel Arbitrated Loop	<b>IPMP</b>	IP Multipathing
<b>FDX</b>	Full Duplex	<b>IQN</b>	iSCSI Qualified Name
<b>FFDC</b>	First Failure Data Capture	<b>ISC</b>	Integrated Solutions Console
<b>FLOP</b>	Floating Point Operation	<b>ISSO</b>	Information System Security Officer
<b>FRU</b>	Field Replaceable Unit	<b>ISV</b>	Independent Software Vendor
<b>FTP</b>	File Transfer Protocol	<b>ITSO</b>	International Technical Support Organization
<b>GDPS®</b>	Geographically Dispersed Parallel Sysplex™	<b>IVM</b>	Integrated Virtualization Manager
<b>GID</b>	Group ID		
<b>GPFS</b>	General Parallel File System		
<b>GSS</b>	General Security Services		
<b>GUI</b>	Graphical User Interface		

<b>iWARP</b>	Internet Wide Area RDMA Protocol	<b>MIBs</b>	Management Information Bases
<b>J2</b>	JFS2	<b>ML</b>	Maintenance Level
<b>JFS</b>	Journaled File System	<b>MLS</b>	Multi Level Security
<b>KAT</b>	Kernel Authorization Table	<b>MP</b>	Multiprocessor
<b>KCT</b>	Kernel Command Table	<b>MPIO</b>	Multipath I/O
<b>KDT</b>	Kernel Device Table	<b>MPS</b>	Maximum Privilege Set
<b>KRT</b>	Kernel Role Table	<b>MTU</b>	Maximum Transmission Unit
<b>KST</b>	Kernel Security Table	<b>Mbps</b>	Megabits Per Second
<b>L1</b>	Level 1	<b>NDAF</b>	Network Data Administration Facility
<b>L2</b>	Level 2		
<b>L3</b>	Level 3	<b>NEC</b>	Nippon Electric Company
<b>LA</b>	Link Aggregation	<b>NFS</b>	Network File System
<b>LACP</b>	Link Aggregation Control Protocol	<b>NIB</b>	Network Interface Backup
<b>LAN</b>	Local Area Network	<b>NIH</b>	National Institute of Health
<b>LDAP</b>	Light Weight Directory Access Protocol	<b>NIM</b>	Network Installation Management
<b>LED</b>	Light Emitting Diode	<b>NIMOL</b>	NIM on Linux
<b>LFS</b>	Logical File System	<b>NIS</b>	Network Information Server
<b>LFT</b>	Low Function Terminal	<b>NLS</b>	National Language Support
<b>LMB</b>	Logical Memory Block	<b>NTP</b>	Network Time Protocol
<b>LPA</b>	Loadable Password Algorithm	<b>NVRAM</b>	Non-Volatile Random Access Memory
<b>LPAR</b>	Logical Partition	<b>ODM</b>	Object Data Manager
<b>LPP</b>	Licensed Program Product	<b>OFA</b>	OpenFabrics Alliance
<b>LPS</b>	Limiting Privilege Set	<b>OFED</b>	OpenFabrics Enterprise Distribution
<b>LRU</b>	Least Recently Used page replacement demon	<b>OSGi</b>	Open Services Gateway Initiative
<b>LUN</b>	Logical Unit Number	<b>OSPF</b>	Open Shortest Path First
<b>LUNs</b>	Logical Unit Numbers	<b>PCI</b>	Peripheral Component Interconnect
<b>LV</b>	Logical Volume	<b>PIC</b>	Pool Idle Count
<b>LVCB</b>	Logical Volume Control Block	<b>PID</b>	Process ID
<b>LVM</b>	Logical Volume Manager	<b>PIT</b>	Point-in-time
<b>LWI</b>	Light Weight Infrastructure	<b>PKI</b>	Public Key Infrastructure
<b>MAC</b>	Media Access Control	<b>PLM</b>	Partition Load Manager
<b>MBps</b>	Megabytes Per Second		
<b>MCM</b>	Multichip Module		

<b>PM</b>	Performance Monitor	<b>RNIC</b>	RDMA-capable Network Interface Controller
<b>POSIX</b>	Portable Operating System Interface	<b>RPC</b>	Remote Procedure Call
<b>POST</b>	Power-On Self-test	<b>RPL</b>	Remote Program Loader
<b>POWER</b>	Performance Optimization with Enhanced RISC (Architecture)	<b>RPM</b>	Red Hat Package Manager
<b>PPC</b>	Physical Processor Consumption	<b>RSA</b>	Rivet, Shamir, Adelman
<b>PPFC</b>	Physical Processor Fraction Consumed	<b>RSCT</b>	Reliable Scalable Cluster Technology
<b>PSPA</b>	Page Size Promotion Aggressiveness Factor	<b>RSH</b>	Remote Shell
<b>PTF</b>	Program Temporary Fix	<b>RTE</b>	Runtime Error
<b>PTX</b>	Performance Toolbox	<b>RTEC</b>	Runtime Error Checking
<b>PURR</b>	Processor Utilization Resource Register	<b>RUID</b>	Real User ID
<b>PV</b>	Physical Volume	<b>S</b>	System Scope
<b>PVID</b>	Physical Volume Identifier	<b>SA</b>	System Administrator
<b>PVID</b>	Port Virtual LAN Identifier	<b>SAN</b>	Storage Area Network
<b>QoS</b>	Quality of Service	<b>SAS</b>	Serial-Attached SCSI
<b>RAID</b>	Redundant Array of Independent Disks	<b>SCSI</b>	Small Computer System Interface
<b>RAM</b>	Random Access Memory	<b>SCTP</b>	Stream Control Transmission Protocol
<b>RAS</b>	Reliability, Availability, and Serviceability	<b>SDD</b>	Subsystem Device Driver
<b>RBAC</b>	Role Based Access Control	<b>SED</b>	Stack Execution Disable
<b>RCP</b>	Remote Copy	<b>SFDC</b>	Second Failure Data Capture
<b>RDAC</b>	Redundant Disk Array Controller	<b>SLs</b>	Sensitivity Labels
<b>RDMA</b>	Remote Direct Memory Access	<b>SMI</b>	Structure of Management Information
<b>RGID</b>	Real Group ID	<b>SMIT</b>	Systems Management Interface Tool
<b>RIO</b>	Remote I/O	<b>SMP</b>	Symmetric Multiprocessor
<b>RIP</b>	Routing Information Protocol	<b>SMS</b>	System Management Services
<b>RISC</b>	Reduced Instruction-Set Computer	<b>SMT</b>	Simultaneous Multi-threading
<b>RMC</b>	Resource Monitoring and Control	<b>SO</b>	System Operator
		<b>SP</b>	Service Processor
		<b>SPOT</b>	Shared Product Object Tree
		<b>SRC</b>	System Resource Controller
		<b>SRN</b>	Service Request Number

<b>SSA</b>	Serial Storage Architecture	<b>VPSS</b>	Variable Page Size Support
<b>SSH</b>	Secure Shell	<b>VRRP</b>	Virtual Router Redundancy Protocol
<b>SSL</b>	Secure Socket Layer	<b>VSD</b>	Virtual Shared Disk
<b>SUID</b>	Set User ID	<b>WED</b>	WebSphere Everyplace® Deployment V6.0
<b>SUMA</b>	Service Update Management Assistant	<b>WLM</b>	Workload Manager
<b>SVC</b>	SAN Virtualization Controller	<b>WPAR</b>	Workload Partitions
<b>TCB</b>	Trusted Computing Base	<b>WPS</b>	Workload Partition Privilege Set
<b>TCP/IP</b>	Transmission Control Protocol/Internet Protocol		
<b>TE</b>	Trusted Execution		
<b>TEP</b>	Trusted Execution Path		
<b>TLP</b>	Trusted Library Path		
<b>TLS</b>	Transport Layer Security		
<b>TSA</b>	Tivoli System Automation		
<b>TSD</b>	Trusted Signature Database		
<b>TTL</b>	Time-to-live		
<b>UCS</b>	Universal-Coded Character Set		
<b>UDF</b>	Universal Disk Format		
<b>UDID</b>	Universal Disk Identification		
<b>UFST</b>	Universal Font Scaling Technology		
<b>UID</b>	User ID		
<b>ULM</b>	User Loadable Module		
<b>UPS</b>	Used Privilege Set		
<b>VG</b>	Volume Group		
<b>VGDA</b>	Volume Group Descriptor Area		
<b>VGSA</b>	Volume Group Status Area		
<b>VIPA</b>	Virtual IP Address		
<b>VLAN</b>	Virtual Local Area Network		
<b>VMM</b>	Virtual Memory Manager		
<b>VP</b>	Virtual Processor		
<b>VPA</b>	Visual Performance Analyzer		
<b>VPD</b>	Vital Product Data		
<b>VPN</b>	Virtual Private Network		



# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

## IBM Redbooks

For information about ordering these publications, see “How to get Redbooks” on page 402. Note that some of the documents referenced here may be available in softcopy only.

- ▶ *AIX Version 4.2 Differences Guide*, SG24-4807
- ▶ *AIX Version 4.3 Differences Guide*, SG24-2014
- ▶ *AIX 5L Differences Guide Version 5.2 Edition*, SG24-5765
- ▶ *AIX 5L Differences Guide Version 5.3 Edition*, SG24-7463
- ▶ *AIX 5L Differences Guide Version 5.3 Addendum*, SG24-7414
- ▶ *IBM AIX Version 6.1 Differences Guide*, SG24-7559
- ▶ *Sun Solaris to IBM AIX 5L Migration: A Guide for System Administrators*, SG24-7245
- ▶ *AIX Reference for Sun Solaris Administrators*, SG24-6584
- ▶ *IBM AIX 5L Reference for HP-UX System Administrators*, SG24-6767
- ▶ *AIX V6 Advanced Security Features Introduction and Configuration*, SG24-7430
- ▶ *Tivoli Management Services Warehouse and Reporting*, SG24-7290
- ▶ *AIX Logical Volume Manager from A to Z: Introduction and Concepts*, SG24-5432
- ▶ *IBM System p5 Approaches to 24x7 Availability Including AIX 5L*, SG24-7196
- ▶ *Introduction to Workload Partition Management in IBM AIX Version 6.1*, SG24-7431
- ▶ *IBM Power 710 and 730 Technical Overview and Introduction*, REDP-4636
- ▶ *IBM Power 720 and 740 Technical Overview and Introduction*, REDP-4637
- ▶ *IBM Power 750 and 755 Technical Overview and Introduction*, REDP-4638
- ▶ *IBM Power 770 and 780 Technical Overview and Introduction*, REDP-4639

- ▶ *IBM Power 795 Technical Overview and Introduction*, REDP-4640

## Other publications

These publications are also relevant as further information sources:

- ▶ *Technical Reference: Kernel and Subsystems, Volume 1, SC23-6612*

## Online resources

These Web sites are also relevant as further information sources:

- ▶ My developerWorks Blogs, Chris's AIX blog:  
[https://www.ibm.com/developerworks/mydeveloperworks/blogs/cgaix/?lang=en\\_us](https://www.ibm.com/developerworks/mydeveloperworks/blogs/cgaix/?lang=en_us)
- ▶ My developerWorks: Blogs, AIXpert blog:  
[https://www.ibm.com/developerworks/mydeveloperworks/blogs/aixpert/?lang=en\\_us](https://www.ibm.com/developerworks/mydeveloperworks/blogs/aixpert/?lang=en_us)
- ▶ AIX 7.1 Information Center  
<http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp>

## How to get Redbooks

You can search for, view, or download Redbooks, Redpapers, Technotes, draft publications and Additional materials, as well as order hardcopy Redbooks publications, at this Web site:

[ibm.com/redbooks](http://ibm.com/redbooks)

## Help from IBM

IBM Support and downloads

[ibm.com/support](http://ibm.com/support)

IBM Global Services

[ibm.com/services](http://ibm.com/services)







# Index

## Symbols

\_\_system\_configuration 165  
 \_\_thread 10  
 /aha/fs/utilFs.monFactory 193  
 /audit/bin1 324  
 /audit/bin2 324  
 /etc/export 354  
 /etc/hosts 119  
 /etc/lib/objrepos 330  
 /etc/nscontrol.conf file 283  
 /etc/objrepos 330  
 /etc/objrepos/wboot  
     rootvg 54  
 /etc/security/audit/bincmds 325  
 /etc/security/audit/config 324  
 /etc/security/domains file 280  
 /etc/security/domobj file 280  
 /etc/security/ldap/ldap.cfg 337  
 /etc/wpars/wpar1.cf 63  
 /nre/opt 45, 50, 54  
 /nre/sbin 50, 54  
 /nre/usr 45, 50  
 /nre/usr, 54  
 /opt/mcr/bin/chkptwpar 82, 85  
 /usr/ccs/lib/libbind.a 259  
 /usr/include/sys/devinfo.h, LVM enhancement for  
 SSD 23  
 /usr/include/sys/kern\_socket.h header file 5  
 /usr/include/sys/systemcfg.h 165  
 /usr/lib/drivers/ahafs.ext 191  
 /usr/lib/libiconv.a library 375  
 /usr/lib/methods/wio 61  
 /usr/lib/security/methods.cfg 331  
 /usr/lpp/bos/editions, AIX edition selection 344  
 /usr/samples/ae/templates/ae\_template.xml 358  
 /usr/samples/nim/krb5 351  
 /usr/samples/nim/krb5/config\_rpcsec\_server 354  
 /usr/sbin/named8 program 259  
 /usr/sbin/named8-xfer program 259  
 /usr/sbin/named-xfer link 259  
 /usr/sys/inst.data/sys\_bundles/BOS.autoi 362  
 /var/adm/wpars/event.log example 82

## Numerics

1 TB segment 2

## A

accessxat 7  
 acesxat 7  
 Activation Engine 346  
     AE 355  
 Active Directory 338  
 Active Directory application mode 338  
 Active Memory Expansion (AME) 198  
 ADAM 338  
 advance accounting 338  
 advisory 241  
 ae command 357, 360  
 AE scripts 359  
 AF\_CLUSTER cluster socket family 118  
 AIX  
     Global> 36, 42, 58  
 AIX edition selection 342  
 AIX edition, enterprise 342  
 AIX edition, express 342  
 AIX edition, standard 342  
 AIX editions  
     enterprise 342  
     express 342  
     standard 342  
 AIX environment variables  
     MALLOCDDEBUG=log 18  
 AIX event infrastructure 189, 195  
     /aha/fs/utilFs.monFactory 193  
     /aha/fs/utilFs.monFactory/tmp.mon 194  
     /usr/lib/drivers/ahafs.ext 191  
     bos.ahafs 190  
     cDiskState cluster event producer 195  
     diskState cluster event producer 195  
     genkex 191  
     linkedCl cluster event producer 195  
     modDor event producer 195  
     modFile event producer 195  
     mon\_levent 192  
     monitor file 192  
     mount -v ahafs 191

networkAdapterState cluster event producer	API 139, 241
195	accessxat 7
nodeAddress cluster event producer 195	acessxat 7
nodeContact cluster event producer 195	chownxat 7
nodeList cluster event producer 195	faccessat 7
nodeState cluster event producer 195	fchmodat 7
pidProcessMon event producer 195	fchownat 7
processMon event producer 195	fexecve 7, 10
repDiskState cluster event producer 195	fstatat 7
select() completed 194	futimens 7, 10
utilFS event producer 195	isalnum_l 8
vgState cluster event producer 195	iscntrl_l 8
vmo event producer 195	isdigit_l 8
waitersFreePg event producer 195	isgraph_l 8
waitTmCPU event producer 195	islower_l 8
waitTmPgInOut event producer 195	isprint_l 8
AIX Runtime Expert catalog 168	ispunct_l 8
AIX Runtime Expert profile templates 167	isspace_l 8
alias 337	isupper_l 8
alias name mapping 375	isxdigit_l 8
ALLOCATED 39, 53, 64, 68, 80	kopenat 7
AME, AIX performance tools enhancement 221	linkat 7
AME, AIX support for Active Memory Expansion	mkdirat 7
198	mkfifoat 8
AME, enhanced AIX performance monitoring tools	mknodat 7
221	open 7, 9
AME, lparstat command 222	openat 7
AME, nmon command 225	openxat 7
AME, performance tools additional options 221	perfstat_cluster_list 140
AME, svmon command 225	perfstat_cluster_total 139
AME, topas command 223	perfstat_cpu_node 141
AME, topas_nmon command 225	perfstat_cpu_total_node 141
AME, vmstat command 221	perfstat_disk_node 141
amepat, Active Memory Expansion modeled statistics report 206	perfstat_disk_total_node 141
amepat, Active Memory Expansion statistics report	perfstat_diskadapter_node 141
205	perfstat_diskpath_node 141
amepat, AME monitoring only report 219	perfstat_logicalvolume_node 142
amepat, command 198	perfstat_memory_page_node 142
amepat, Command Information Section report 203	perfstat_memory_total_node 142
amepat, generate a recording file and report 217	perfstat_netbuffer_node 142
amepat, generate a workload planning report 218	perfstat_netinterface_node 142
amepat, recommendation report 207	perfstat_netinterface_total_node 142
amepat, recording mode 198	perfstat_pagingspace_node 143
amepat, reporting mode 198	perfstat_partition_total interface 139
amepat, System Configuration Section report 203	perfstat_partition_total_node 143
amepat, System Resource statistics report 204	perfstat_protocol_node 143
amepat, workload monitoring 198	perfstat_tape_node 143
amepat, workload planning 198	perfstat_tape_total_node 143
	perfstat_volume_group_node 143

pthread\_attr\_getsrad\_np 243  
 pthread\_attr\_setsrad\_np 241, 243  
 ra\_attach 242  
 ra\_exec 242  
 ra\_fork 242  
 readlinkat 7  
 renameat 7  
 stat64at 7  
 statx64at 7  
 statxat 7  
 symlinkat 7  
 ulinkat 7  
 utimensat 7, 10  
 utimes 9  
 application programming interface 139  
 apps\_fs\_manage role 286  
 artexdiff 170, 173–174  
 artexget 170, 172  
 artexget -V 176  
 artexlist 167, 170  
 artexmerge 170  
 artexset 170, 173–174  
 artexset -u 174  
 assembler 10  
 attribute  
     TO\_BE\_CACHED 337  
 audit API 322  
 audit command 322  
 audit events, trusted execution 323  
 audit roles 326  
 audit trail files 324  
 audit, audit subsystem, auditing events 322  
 auditcat command 325  
 auditmerge command 325  
 auditpr command 326  
 authentication 330  
     LDAP 337  
 Authorization Database  
     Enhanced RBAC 268  
 authprt command 313  
 AVAILABLE 68

## B

backuppah, audit trail file config parameter 324  
 backusize, audit trail file config parameter 324  
 Berkeley Internet Name Domain 258  
 binary compatibility 2  
 BIND 8 258

BIND 9 258  
 boot command 354  
 bootlist command 348  
     pathid attribute 348  
 bos.adt.include fileset. 5  
 bos.ae package 356  
 bos.ahafs 190  
 bos.ecc\_client.rte 362  
 bos.mp64 fileset 5  
 bos.suma 362  
 bos.wpars package 44, 46  
 bread, iostate output column 245  
 buffer overflows 328  
 bwrite, iostate output column 245

## C

CAA 117  
 CAP\_NUMA\_ATTACH 242  
 caseExactAccountName 337  
 cat command 310  
 cdat command 112  
 cfgmgr command 60–61  
 chcluster command 118  
 chdev command 60, 270, 330  
 chdom command 275  
 Checkpointable 82  
 chedition, command 344  
 chfs command 285  
 chownxat 7  
 chpasswd command 331  
 chpath command 348  
 chsec command 278  
 chuser command 277  
 chvg command, LVM enhancement for SSD 25  
 chwpar command 36, 40, 61, 81  
     kext=ALL 36  
 clcmd command 117–118  
 clDiskList cluster event producer 195  
 clDiskState cluster event producer 195  
 cluster 139  
 cluster aware AIX 117  
 cluster communication, network or storage interfaces 132  
 cluster data aggregation tool, FFDC 111  
 cluster disks 119  
 cluster multicast address 119  
 cluster network statistics 124  
 cluster specific events 131

- cluster storage interfaces 123
- cluster system architecture 130
- clustering 264
- clusters 117
- clusterwide
  - command distribution, clcmd command 117
  - communication, cluster socket family 118
  - event management, AIX event infrastructure 117
  - storage naming service 117
- code set 374
- code set mapping 375
- columns, iostat 244
- command
  - ae 357, 360
  - boot 354
  - bootlist 348
  - cfgmgr 60
  - chdev 60
  - chpath 348
  - chwpar 36, 81
  - errpt 83, 355
  - fuser 355
  - installp 44
  - ipreport 355
  - iptrace 355
  - loadkernext -l 41
  - loopmount 347
  - loopumount 347
  - lscfg 58, 60
  - lsdev 60
  - lsdev -X 75
  - lsdev -x 63
  - lsof 355
  - lspath 348–349
  - lsvg 59
  - lsvpd 60
  - lswpar 59, 85
  - lswpar -D 47
  - lswpar -M 47, 75
  - lswpar -t 47
  - lswpar -X 47
  - mkdev 60
  - mkpath 348–349
  - mkwpar 36, 44, 84
  - mkwpar -X local=yes/no 39
  - nfs4cl 355
  - nfsstat 355
  - nim 353
  - rmdev 60
  - rmpath 348–349
  - rpcinfo 355
  - startwpar 48
  - syslogd 355
  - trcrpt 83
  - varyoffvg 68
- commands
  - amepat 198
  - artexdiff 170
  - artexget 170
  - artexlist 167, 170
  - artexmerge 170
  - artexset 170
  - audit 322
  - auditcat 325
  - auditmerge 325
  - auditpr 326
  - authrpt 313
  - cat 310
  - chcluster 118
  - chdev 270, 330
  - chdom 275
  - chedition 344
  - chfs 285
  - chpasswd 331
  - chsec 278
  - chuser 277
  - chvg 25
  - clcmd 117–118
  - cpuextintr\_ctl 146
  - crfs 269, 285
  - crontab 116
  - dcat 112
  - dconsole 148, 150
  - dcp 152
  - dgetmacs 148–149
  - dkeyexch 148–149
  - dpasswd 148
  - dsh 153
  - enstat 252
  - extendvg 26
  - filemon 227
  - head 310
  - iostat 244
  - ksh93 188
  - lparstat 222
  - lsattr 270, 330
  - lscfg 251

lscluster 118  
 lsdom 273  
 lskst 288  
 lsldap 338  
 lspv 118  
 lsrole 288  
 lssec 277  
 lssecattr -o 276–277  
 lsslot 251  
 lsuser 277, 337  
 lsvg 24  
 migwpar 86  
 mkcluster 118  
 mkdom 273  
 mkvg 24  
 more 310  
 mount 286  
 nmon 225  
 perfstat 139  
 pg 310  
 ping 318  
 raso 245  
 rendev 180  
 replacepv 26  
 rmcluster 118  
 rmdom 276  
 rmfs 286  
 rmsecattr -o 277  
 rolerlist 290  
 rolerpt 313  
 setkst 277  
 setsecattr -o 276  
 skctl 110  
 svmon 225  
 swrole 288  
 sysdumpdev 102  
 topas 223  
 topas\_nmon 225  
 unmount 286  
 vi 310  
 vmo 186  
 vmstat 221  
 compatibility, binary compatibility 2  
 compiler options  
   -g 17  
   -qdbsfmt=dwarf 189  
   -qfunsect 11  
   -qtls 10  
   -qxflag=toctrel 11

compiler, XLC compiler v11 328  
 complex locks 13  
 Component Dump 137  
 Component Trace 137  
 conflict set  
   domain RBAC 271  
 core dump settings 12  
 CPU 146  
 cpuextintr\_ctl command 146  
 cpuextintr\_ctl system call 146  
 CPUs, 1024 CPU support 181  
 crfs command 269, 285  
 crontab command 116  
 CSM  
   Cluster Systems Management (CSM), removal  
   of 176  
   dsm.core package 179  
   removal of csm.core 177  
   removal of csm.dsh 177  
 CT SCTP component hierarchy 138  
 cctrl command 138

## D

daemon  
   rpc.mountd 353  
 dbx 16  
 dbx commad  
   print\_mangled 17  
 dbx commands  
   display 16  
   malloc 18  
   malloc allocation 18  
   malloc freespace 18  
 dbx environment variable  
   print\_mangled 17  
 dconsole 148, 150, 159  
 dconsole display modes 150  
 dcp 152  
 debug fill 11  
 debuggers  
   dbx 16  
 debugging information 188  
 debugging tools 188  
   DWARF 188  
 DEFINED 68  
 demangled 17  
 device  
   object type in domain RBAC 296

- device renaming 180
  - device, iostate output column 244
  - devices 180
    - sys0 270
  - devname 73, 78
  - devtype 73
  - dgetmacs 148–149, 154, 156
  - disabled read write locks 13
  - Discretionary Access Control (DAC)
    - Enhanced RBAC 279
  - disk, cluster disk 119
  - disk, repository disk 119
  - diskState cluster event producer 195
  - dispatcher 241
  - display 16
  - Distributed System Management 147
  - dkeyexch 148–149, 154
  - domain
    - domain RBAC 271
  - domain Enhanced RBAC 269
  - Domain Name System 258
  - domain RBAC 266, 272, 296
    - /etc/nscontrol.conf 283
    - /etc/security/domains 280
    - /etc/security/domobj 280
  - chdom 275
  - chfs 285
  - chsec 278
  - chuser 277
  - conflict set 271
  - crfs 285
  - domain 271
  - domain, root user membership 304
  - LDAP support 283
  - lsdom 273
  - lssec 277
  - lssecattr -o 276–277
  - lsuser 277
  - mkdom 273
  - mount 286
  - object 271
  - object, device 296
  - object, file 303
  - object, netint 311
  - object, netport 311
  - property 271
  - rmdom 276
  - rmfs 286
  - rmsecattr -o 277
  - scenarios 284
    - scenarios, device scenario 284
    - scenarios, file scenario 284
    - scenarios, network scenario 284
  - security flags 272
  - setkst 277
  - setsecattr -o 276
  - subject 271
  - unmount 286
  - DOWNLOAD\_PROTOCOL 365
  - dpasswd 148, 154
  - drw\_lock\_done kernel service 15
  - drw\_lock\_init kernel service 14
  - drw\_lock\_islocked kernel service 16
  - drw\_lock\_read kernel service 14
  - drw\_lock\_read\_to\_write kernel service 15
  - drw\_lock\_try\_read\_to\_write kernel service 15
  - drw\_lock\_try\_write kernel service 16
  - drw\_lock\_write kernel service 14
  - drw\_lock\_write\_to\_read kernel service 15
  - dsh 153
  - DSM and NIM 154
  - DWARF 188
- E**
- eCC 362
  - eCC Common Client 362
  - eccBase.properties 366
  - eccConnect.properties 366
  - Electronic Customer Care 362
  - Electronic Service Agent 362
  - enhanced korn shell 188
  - Enhanced RBAC 268
    - Authorization Database 268
    - authrpt 313
    - chdev command usage 270
    - Discretionary Access Control (DAC) 279
    - kernel security tables (KST) 273
    - lskst 288
    - lsrole 288
    - Privileged Command Database 269
    - Privileged Device Database 269
    - Privileged File Database 269
    - Role Database 268
    - rolelist 290
    - rolerpt 313
    - swrole 288
    - sys0 device 270



- system-defined authorizations 269
- user-defined authorizations 269
- Enhanced RBAC domain 269
- Enhanced RBAC mode 267
- Enhanced RBAC roles
  - apps\_fs\_manage 286
  - FSAdmin 286
- Enhanced RBAC security database
  - security database 268
- entstat -d command 252
- environment variable 11
- errctrl command 138
- errprt command 83, 355
- esid\_allocator 2
- ETHERNET DOWN 254
- event producer 195
- events, auditing events 322
- events, cluster events 131
- EXPORTED 67, 80
- extendvg command, LVM enhancement for SSD 26

## F

- fabric 264
- factssat 7
- fastpath
  - vwpar 57
- fchmodat 7
- fchownat 7
- fcp 61
- fcs0 58, 62, 80
- fexecve 7, 10
- FFDC 111
- fiber channel adapter 58
- fibre channel adapters 118
- fibre channel adapters, list of supported adapters 135
- File
  - fcntl.h 9
  - sys/stat.h 9
  - unistd.h 10
- file
  - /etc/security/ldap/ldap.cfg 337
  - libperfstat.a 140
  - libperfstat.h 141
  - object type in domain RBAC 303
- filemon command 227
- filemon, Hot File Report, sorted by capacity ac-

- cessed 233
- filemon, Hot Files Report 232
- filemon, Hot Logical Volume Report 233
- filemon, Hot Logical Volume Report, sorted by capacity 234
- filemon, Hot Physical Volume 233
- filemon, Hot Physical Volume Report, sorted by capacity 234
- files
  - /etc/nscontrol.conf 283
  - /etc/security/domains 280
  - /etc/security/domobj 280
  - /usr/bin/ksh93 188
- fill 11–12
- firmware
  - boot 354
- firmware-assisted dump 102
  - diskless servers 109
  - ISCSI device support 109
  - scratch area memory 106
- first failure data capture 111
- Fix Level Recommendation Tool (FLRT) 392
- fixget interface 361
- FIXSERVER\_PROTOCOL 364
- FSAdmin role 286
- FSF\_DOM\_ALL 272
  - domain RBAC security flag 272
- FSF\_DOM\_ANY 272, 296
  - domain RBAC security flag 272, 296
- fstatat 7
- full path auditing 322
- fuser command 355
- futimens 7, 10
- fw-assisted type of dump 103

## G

- g 17
- genkex 191
- genkex command 39, 41
- getsystemcfg() 165–166
- Global AIX instance 57
- global device view 117
- Global> 36, 42, 58
- graphics software bundle 362
- groups, user groups 330

## H

- HACMP clusters 117

hardware storage keys 110  
 head command 310  
 high availability 117, 139  
 hot file detection, filemon command 227  
 hot files detection, jfs2 28  
 HTTP\_Proxy 366  
 HTTPS\_PROXY 367

**I**

IBM Director 83  
 IBM Systems Director Common Agent 345  
 IBM Text-to-Speech (TTS)  
   removal from AIX Expansion Pack 180  
   Text-to-Speech, removal of 179  
   tts\_access.base 179  
   tts\_access.base.en\_US 179  
 IBM-943 code set 375  
 IBM-eucJP code set 375  
 iconv command 374  
 iconv converters 374–375  
 IEEE 802.3ad 248, 256  
 ifconfig command  
   commands ifconfig 311  
 importvg command 71  
 IN\_SYNC 253  
 InfiniBand 264  
 installp command 44  
 interrupts 146  
 Inventory Scout 362  
 iostat -b command 244  
 iostat output columns 244  
 ipreport command 355  
 iptrace command 355  
 IPv4 264  
 IPv6 264  
 IPv6 network 351  
 isalnum\_l 8  
 iscntrl\_l 8  
 isdigit\_l 8  
 isgraph\_l 8  
 islower\_l 8  
 isprint\_l 8  
 ispunct\_l 8  
 isspace\_l 8  
 isupper\_l 8  
 isxdigit\_l 8  
 iWARP 264

**J**

ja\_JP local 375  
 Japanese input method 375  
 Java6.sdk 363  
 jfs2, enhanced support for SSD 28  
 jfs2, HFD ioctl calls summary 29  
 jfs2, HFD sample code 32  
 jfs2, HFD\_\* ioctl calls 28  
 jfs2, Hot File Detection (HFD) 28  
 jfs2, Hot File Detection /usr/include/sys/hfd.h 28  
 jfs2, Hot Files Detection in 27

**K**

k\_cpuextintr\_ctl kernel service 146  
 kadmind\_timeout, Kerberos client option 331  
 kerberos 330  
 kern\_soaccept kernel service 5  
 kern\_sobind kernel service 5  
 kern\_soclose kernel service 6  
 kern\_soconnect kernel service 5  
 kern\_socreate kernel service 5  
 kern\_sogetopt kernel service 5  
 kern\_solisten kernel service 5  
 kern\_soreceive kernel service 6  
 kern\_soreserve kernel service 6  
 kern\_sosend kernel service 6  
 kern\_sosetopt kernel service 6  
 kern\_soshutdown kernel service 6  
 kernel 185  
 Kernel extension 41  
   ALLOCATED status 39  
   genkex command 39  
   loadkernext -q command 39  
 kernel extension 185, 264  
 kernel security tables (KST)  
   Enhanced RBAC 273  
 kernel service  
   kgetsystemcfg() 165  
 kernel sockets API 5  
 kext=ALL 36  
 kgetsystemcfg() 165  
 kopenat 7  
 krb5 351  
 KRB5 load module 330  
 ksh93 188

**L**

LACP Data Units (LACPDU) 248

- LACPDU
    - packet 256
  - LDAP 331, 337
    - /etc/security/ldap/ldap.cfg 337
    - alias 337
    - caseExactAccountName 337
    - TO\_BE\_CACHED 337
  - LDAP support in domain RBAC 283
  - Legacy RBAC 267
    - setuid 267
  - Legacy RBAC mode 267
  - libiconv functions 374
  - libperfstat.a 140
  - libperfstat.h 141
  - library function
    - getsystemcfg() 165
  - lightweight directory access protocol, LDAP 331
  - Link Aggregation Control Protocol (LACP) 248
  - linkat 7
  - linkedCl cluster event producer 195
  - loadkernelnext -l command 41
  - loadkernelnext -q command 39
  - locking, kernel memory locking 185
  - locks, complex locks 13
  - locks, interrupt safe locks 13
  - log 18
  - loopback 264
  - loopback devices 346
  - loopmount command 347
  - loopumount command 347
  - lpp\_source 346
  - LRU, Least Recently Used memory management 185
  - lsattr command 270, 330
  - lsattr -El command 250
  - lscfg command 58, 60, 70
  - lscfg -vl command 251
  - lscluster command 118
  - lsdev -Cc adapter command 250
  - lsdev command 60, 69, 77
  - lsdev -x command 63, 75
  - lsdom command 273
  - lskst command 288
  - lsldap 338
  - lsf command 355
  - lspath command 348–350
  - lspv command 67, 70–71, 75, 118
  - lsrole command 288
  - lssec command 277
  - lssecattr -o command 276–277
  - lsslot -c pci command 251
  - lsuser 337
  - lsuser command 277
  - lsvg command 59
  - lsvg command , PV RESTRICTION for SSD 24
  - lsvpd command 60
  - lswpar command 39, 85
  - lswpar -D command 47, 53
  - lswpar -M command 47, 59, 75
  - lswpar -t command 47
  - lswpar -X command 39, 47
    - ALLOCATED status 39
  - LVM enhanced support for solid-state disks 22
- ## M
- Malloc 11
  - malloc 18
    - debug fill 11
    - painted 11
  - malloc allocation 18
  - malloc freespace 18
  - MALLOCDEBUG 11–12
  - MALLOCDEBUG=fill
    - "abc" 12
    - pattern 11
  - MALLOCDEBUG=log 18
  - mangled 17
  - maxpin tunable 186
  - memory
    - painted 11
  - memory, kernel memory 186
  - message number 84
  - migwpar command, steps to migrate the WPAR 89
  - migwpar command, WPAR types that are not supported for migration 87
  - migwpar, command 86
  - migwpar, migrating a detached WPAR to AIX V7.1 97
  - mindigit password attribute 334
  - minimum disk requirements for AIX V7.1 341
  - minimum firmware levels for AIX V7.1 340
  - minimum memory requirement for AIX V7.1 340
  - minimum system requirements for AIX V7.1 340
  - minloweralpha password attribute 334
  - minspecialchar password attribute 334
  - minupperalpha password attribute 334
  - MISSING 64

- mkcluster command 118
  - mkdev command 60
  - mkdirat 7
  - mkdom command 273
  - mkfloat 8
  - mknodat 7
  - mkpath command 348–349
  - mksysb 355
  - mksysb command 43
  - mkvg command 71
  - mkvg command, LVM enhancement for SSD 24
  - mkwpar command 36, 44–45, 51, 58, 73–74, 84
    - devname 73, 78
    - devtype 73
    - rootvg=yes 74
    - xfactor=n 44
  - mkwpar -X local=yeslno 39
  - mobility 81
  - modDir event producer 195
  - modFile event producer 195
  - mon\_1event 192
  - more command 310
  - mount command 286
  - mount -v ahafs 191
  - MPIO
    - see Multiple PATH I/O 348
  - MPIO Other DS4K Array Dis 77
  - MPIO Other DS4K Array Disk 66
  - multicast address 119
  - Multiple PATH I/O
    - devices 348
    - lspath command 349
    - mkpath command 349
    - rmpath command 349
- N**
- named daemon 259
  - national language support 373
  - NEC selected characters 375
  - netint
    - object type in domain RBAC 311
  - netport
    - object type in domain RBAC 311
  - Network Installation Manager 154, 350
  - network port aggregation technologies 248
  - Network Time Protocol 259, 367
  - networkAdapterState cluster event producer 195
  - NFS objects auditing 328
  - NFS V4 351
    - Authentication 351
    - Authorization 351
    - Identification 351
  - nfs\_reserved\_port 353
  - nfs\_sec 353
  - nfs\_vers 353
  - nfs4cl command 355
  - nfsd 353
  - nfsd command
    - portcheck 353
  - nfsstat command 355
  - ngroups\_allowed, kernel parameter 330
  - NGROUPS\_MAX 330
  - NIM 350
    - boot 352
    - clients 350
    - loopback devices 346
    - loopmount command 347
    - loopumount command 347
    - lpp\_source 346
    - master 350
    - NFS security 351
    - NFS version 351
    - nim -o define 346
    - spot resources 346
    - TFTP 352
  - nim command 353
  - NIM fastpath
    - nim\_mkres 348
  - nim -o define 346
  - NIM service handler 351
  - nim\_mkres fastpath 348
  - nimsh 351
  - node info file 152
  - NODE interfaces 140
  - node list 154
  - node performance 139
  - nodeAddress cluster event producer 195
  - nodeContact cluster event producer 195
  - nodeList cluster event producer 195
  - nodeState cluster event producer 195
  - NTP 354
    - ntp.rte fileset 260, 368
    - ntpd4 daemon 261, 368
    - ntpdate4 command 261, 368
    - ntpd4 program 260, 368
    - ntp-keygen4 command 260, 368
    - ntp4 program 260, 368

ntprtrace4 script 260, 368

## O

O\_DIRECTORY 9

O\_SEARCH 9

object

domain RBAC 271

object auditing 322, 328

object data manager, ODM 329

octal 12

ODM 329

OFED 263

open 7

O\_DIRECTORY 9

O\_SEARCH 9

Open Group Base Specifications 7

openat 7

OpenFabrics Enterprise Distribution 263

openxat 7

OUT\_OF\_SYNC 256

## P

package

bos.ae 356

bos.wpars 44

vwpar.52 44

wio.common 44

packages

csm.core 177

csm.dsh 177

dsm.core 179

page faults 185

paging space requirements for AIX V7.1 341

painted 11

passwords, enforcing restrictions 332

pathid attribute 348

pathname 10

pattern 11

performance

I/O stack 244

performance monitoring 139

performance statistics 139

performance, kernel memory pinning 185

perfstat 139

perfstat library 139

perfstat\_cluster\_list 140

PERFSTAT\_CLUSTER\_STATS 140

perfstat\_cluster\_total 139

perfstat\_config 140

perfstat\_cpu\_node 141

perfstat\_cpu\_total\_node 141

PERFSTAT\_DISABLE 140

perfstat\_disk\_node 141

perfstat\_disk\_total\_node 141

perfstat\_diskadapter\_node 141

perfstat\_diskpath\_node 141

PERFSTAT\_ENABLE 140

perfstat\_logicalvolume\_node 142

perfstat\_memory\_page\_node 142

perfstat\_memory\_total\_node 142

perfstat\_netbuffer\_node 142

perfstat\_netinterface\_node 142

perfstat\_netinterface\_total\_node 142

perfstat\_pagingspace\_node 143

perfstat\_partition\_total interface 139

perfstat\_partition\_total\_node 143

perfstat\_protocol\_node 143

perfstat\_tape\_node 143

perfstat\_tape\_total\_node 143

perfstat\_volumegroup\_node 143

per-thread 7

pg command 310

pidProcessMon event producer 195

ping command 318

pinning, kernel memory pinning 185

POE, Parallel Operation Environment 146

portcheck 353

powerHA 117

print\_mangled 17

Privileged Command Database

Enhanced RBAC 269

Privileged Device Database

Enhanced RBAC 269

Privileged File Database

Enhanced RBAC 269

proc\_getattr API 12

proc\_setattr API 12

processMon event producer 195

processors 146

processors, 1024 CPU support 181

property

domain RBAC 271

propolice 328

pthread\_attr\_getsrads\_np 243

pthread\_attr\_setsrad\_np 241

**Q**

-qdbfmt=dwarf 189  
 -qfuncsect 11  
 -qtls 10  
 -qxflag 11

**R**

R\_STRICT\_SRAD 242  
 ra\_attach 242  
 ra\_exec 242  
 ra\_fork 242  
 RAS 83  
 RAS component framework 137  
 RAS storage keys 110  
 raso -L command 245  
 RBAC 10, 338  
   modes 267  
   modes,Enhanced 268  
   modes,Legacy 267  
   role based auditing 326  
 RDMA 263  
 RDS 263  
 readlinkat 7  
 reads, iostate output column 244  
 real secure server sensor, security attacks 338  
 Redbooks Web site 402  
   Contact us xxviii  
 Reliability, Availability, and Serviceability 83  
 Reliable Datagram Sockets 263  
 reliable scalable cluster technology 117  
 Remote Statistic Interface 336  
 renameat 7  
 renaming devices 180  
 rendev command 180  
 repDiskState cluster event producer 195  
 replacepv command, LVM enhancement for SSD 26  
 repository disk 119  
 rerr, iostate output column 245  
 RFC 2030 (SNTPv4) 260, 367  
 RFC 5905 (NTPv4) 259, 367  
 rmcluster command 118  
 rmdev command 60, 63–64  
 rmdom command 276  
 rmfs command 286  
 rmpath command 348–349  
 rmsecattr -o command 277  
 rmwpar command 84

role based auditing 326  
 Role Database  
   Enhanced RBAC 268  
 rolist command 290  
 rolerpt command 313  
 root user  
   domain membership in domain RBAC 304  
   Role Based Access Control 266  
 rootvg WPAR 41, 57, 74, 82  
   SAN support 57  
 rootvg=yes 74, 78  
 rpc.mountd daemon 353  
 rpcinfo command 355  
 RSCT 117  
 rserv, iostate output column 245  
 RSET 242  
 Rsi 336  
 RTEC SCTP component hierarchy 138  
 Runtime Error Checking 137

**S**

SAN 118  
 SAN support 57  
 SAS adapter cluster communication 118  
 scenarios  
   domain RBAC 284  
 schedo event producer 195  
 scheduling data collections, FFDC 116  
 SCTP event label 138  
 SCTP\_ERR event label 138  
 sctp.sctp\_err eRAS sub-component 138  
 sctpctrl load command 138  
 secldapclntd 337  
 security flags 272, 296  
   domain RBAC 272  
 security policy, trusted execution 323  
 security vulnerabilities 328  
 serial-attached SCSI 118  
 service strategy 363  
 Service Update Management Assistant 361  
 setkst command 277  
 setsecattr 277  
 setsecattr -o command 276  
 setuid  
   Legacy RBAC 267  
 Shared Memory Regions 2  
 shm\_1tb\_shared 2  
 shm\_1tb\_unshared 2

- skctl command 110
  - SMIT
    - vwpar fastpath 57
  - sntp4 program 260, 368
  - spot resources 346
  - SRAD
    - advisory 241
    - R\_STRICT\_SRAD 242
    - strict 241
  - srv\_conn 366
  - SSD disk, configuring on an AIX system 23
  - ssh command 49
  - SSIC
    - exported into a .xls format 390
  - stack smashing protection 328
  - stackprotect, compiler option 329
  - startwpar command 48, 59, 76
  - stat64at 7
  - statx64at 7
  - statxat 7
  - stealing, page stealing 186
  - storage attached network 118
  - storage interfaces, cluster 123
  - storage keys 110
  - Stream Control Transmission Protocol 137
  - strict attachment 241
  - storage class
    - \_\_thread 10
  - struct timespec 8
  - subject
    - domain RBAC 271
  - SUMA 361
  - suma command 361
  - SUMA global configuration settings 361
  - swrole command 288
  - symlinkat 7
  - synchronisation state
    - IN\_SYNC 253
    - OUT\_OF\_SYNC 256
  - sys\_parm API 330
  - sys/stat.h 9
  - sys0 device 270
  - sysdumpdev command 108
    - full memory dump options 103
  - sysdumpdev -l command 102
  - syslog, auditing error messages 324
  - syslogd command 355
  - system dump
    - type of dump 102
  - system management software bundle 362
  - System Planning Tool (SPT) 392
  - System Storage Interoperation Centre (SSIC) 387
  - system-defined authorizations
    - Enhanced RBAC 269
- T**
- telnet command 49
  - TFTP 352
  - Thread Local Storage 10
  - TLS 10
  - TO\_BE\_CACHED 337
  - TOCREL 11
  - traditional type of dump 102
  - trail file recycling 324
  - trcrpt command 83
  - trusted execution 323
  - Trusted Kernel Extension 36
  - trusted signature database, trusted execution 323
  - tunables
    - esid\_allocator 2
    - shm\_1tb\_shared 2
    - shm\_1tb\_unshared 2
  - type of dump
    - fw-assisted 103
    - traditional 102
- U**
- ulinkat 7
  - Unicode 5.2 374
  - unistd.h 10
  - unmount command 286
  - unset 11
  - user-defined authorizations
    - Enhanced RBAC 269
  - UTF-8 code sets 375
  - utilFs 195
  - utimensat 7, 10
  - utimes 9
- V**
- varyoffvg command 68
  - varyonvg command 68
  - VDI 355
  - Versioned Workload Partitions 41
  - Versioned WPAR 41
    - /nre/opt 45

/nre/usr 45  
 vgState cluster event producer 195  
 vi command 310  
 VIOS-based VSCSI disks 57  
 Virtual Data Image 355  
 virtual image template 357  
 vmm\_klock\_mode tunable 186  
 VMM, Virtual Memory Management 186  
 vmo event producer 195  
 vscsi 61  
 VSCSI disks 41  
 vulnerabilities 328  
 vwpar.52 44  
 vwpar.52 package 44  
 vwpar.52.rte package 46  
 vwpar.sysmgt package 57

## W

waitersFreePg event producer 195  
 waitTmCPU event producer 195  
 waitTmPgInOut event producer 195  
 werr, iostate output column 245  
 wio 57  
 wio.common package 44, 46  
 wio0 50  
 WPAR  
   /etc/objrepos/wboot 54  
   cfgmgr command 60  
   chdev command 60  
   chwpar 36  
   lscfg 58  
   lscfg command 60  
   lsdev 60  
   lsdev -x 63  
   lsvg command 59  
   lsvpd command 60  
   lswpar command 39, 47  
     59  
   mkdev command 60  
   mkswpar -X command 39  
   mksysb command 43  
   mkwpar command 36  
   rmdev command 60  
   rootvg 41, 57  
   ssh 49  
   startwpar 48, 59  
   telnet 49  
   Trusted Kernel Extension 36

Versioned WPAR 41  
 VIOS disks 41  
 WPAR I/O Subsystem  
   wio0 50  
 WPAR I/O subsystem 60  
 WPAR Migration to AIX Version 7.1 86  
 WPAR mobility 83  
 wpar.52 package 55  
 writes, iostate output column 244  
 wserv, iostate output column 245

## X

X11 font updates 378  
   Common Desktop Environment (CDE) 378  
   TrueType fonts 378  
 xfactor=n 44



To determine the spine width of a book, you divide the paper PPI into the number of pages in the book. An example is a 250 page book using Plainfield opaque 50# smooth which has a PPI of 526. Divided 250 by 526 which equals a spine width of .4752". In this case, you would use the .5" spine. Now select the Spine width for the book and hide the others: **Special>Conditional Text>Show/Hide>SpineSize(-->Hide:)->Set** . Move the changed Conditional text settings to all files in your book by opening the book file with the spine:fm still open and **File>Import>Formats** the Conditional Text Settings (ONLY!) to the book files.  
Draft Document for Review September 17, 2010 5:34 pm

**7910spine.fm 419**



**Redbooks**

# IBM AIX Version 7.1 Differences Guide

(1.5" spine)  
1.5" <-> 1.998"  
789 <-> 1051 pages



**Redbooks**

# IBM AIX Version 7.1 Differences Guide

(1.0" spine)  
0.875" <-> 1.498"  
460 <-> 788 pages



**Redbooks**

# IBM AIX Version 7.1 Differences Guide

(0.5" spine)  
0.475" <-> 0.875"  
250 <-> 459 pages



**Redbooks**

# IBM AIX Version 7.1 Differences Guide

(0.2" spine)  
0.17" <-> 0.473"  
90 <-> 249 pages

(0.1" spine)  
0.1" <-> 0.169"  
53 <-> 89 pages

To determine the spine width of a book, you divide the paper PPI into the number of pages in the book. An example is a 250 page book using Plainfield opaque 50# smooth which has a PPI of 526. Divided 250 by 526 which equals a spine width of .4752". In this case, you would use the .5" spine. Now select the Spine width for the book and hide the others: **Special>Conditional Text>Show/Hide>SpineSize(->Hide)->Set** . Move the changed Conditional text settings to all files in your book by opening the book file with the spine:fm still open and **File>Import>Formats** the Conditional Text Settings (ONLY!) to the book files.  
Draft Document for Review September 17, 2010 5:34 pm

**7910spine.fm 420**



**Redbooks**

# IBM AIX Version 7.1 Differences Guide

(2.5" spine)  
2.5" <-> mnn.n"  
1315 <-> mnn pages



**Redbooks**

# IBM AIX Version 7.1 Differences Guide

(2.0" spine)  
2.0" <-> 2,498"  
1052 <-> 1314 pages



# IBM AIX Version 7.1 Differences Guide



**AIX - The industrial strength UNIX operating system**

**AIX Version 7.1 Standard Edition enhancements explained**

**An expert's guide to the new release**

This IBM® Redbooks® publication focuses on the differences introduced in IBM AIX® Version 7.1 Standard Edition when compared to AIX Version 6.1. It is intended to help system administrators, developers, and users understand these enhancements and evaluate potential benefits in their own environments.

AIX Version 7.1 introduces many new features, including the following.

- ▶ Role Based Access Control
- ▶ Support for up to 254 partitions on the Power 795
- ▶ Workload Partition Enhancements
- ▶ Topas performance tool enhancements
- ▶ Terabyte segment support
- ▶ Cluster Aware AIX functionality

There are many other new features available with AIX Version 7.1, and you can explore them all in this publication.

For clients who are not familiar with the enhancements of AIX through Version 5.3, a companion publication, AIX Version 6.1 Differences Guide, SG24-7559 is available,

## INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

### BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

**For more information:**  
[ibm.com/redbooks](http://ibm.com/redbooks)