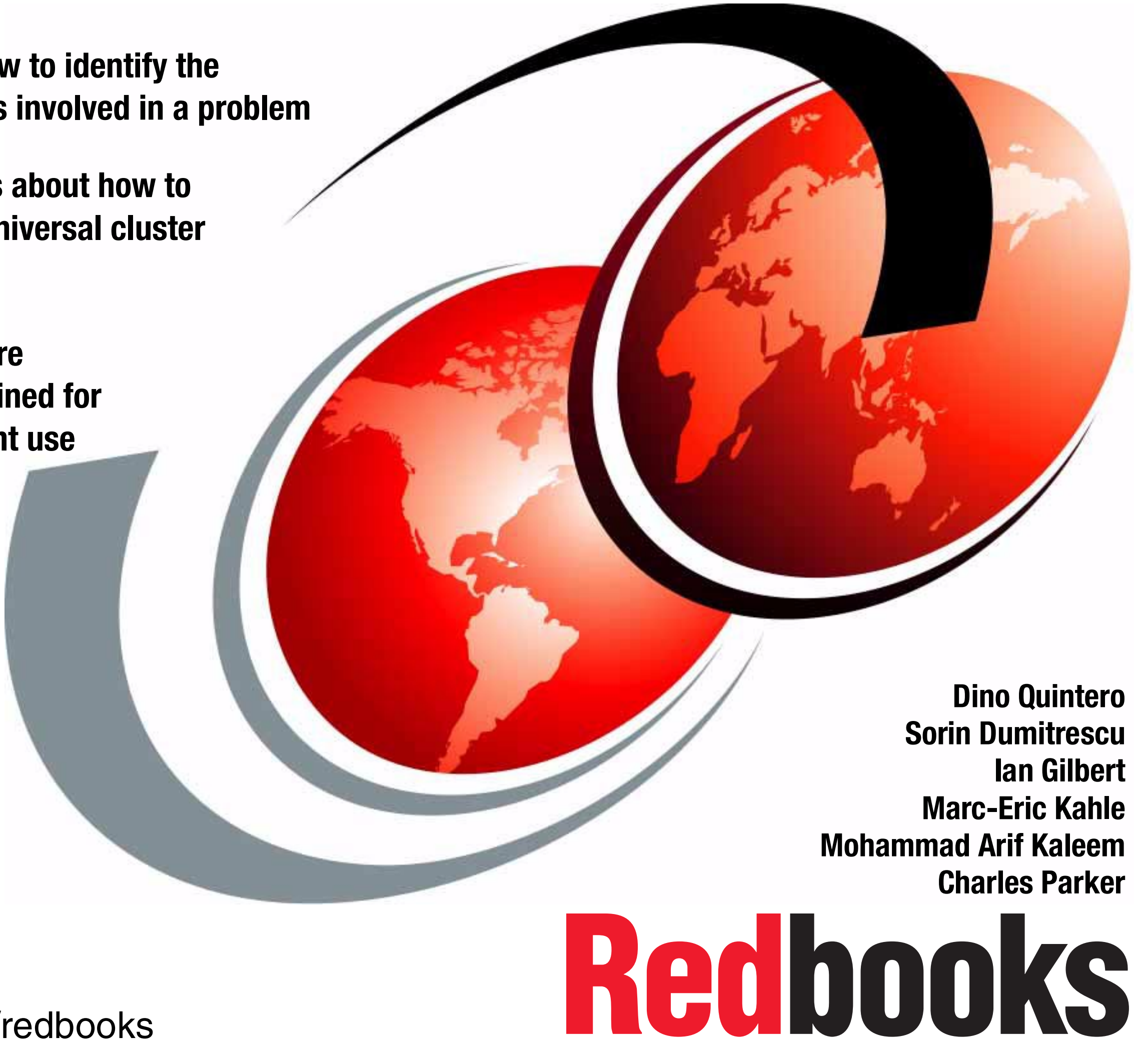
Universal Clustering Problem Determination Guide

Explains how to identify the components involved in a problem

Offers ideas about how to approach universal cluster problems

Chapters are self-cointained for independent use



Dino Quintero
Sorin Dumitrescu
Ian Gilbert
Marc-Eric Kahle
Mohammad Arif Kaleem
Charles Parker

Redbooks

ibm.com/redbooks

International Technical Support Organization

Universal Clustering Problem Determination Guide

October 2001

SG24-6602-00

Take Note! Before using this information and the product it supports, be sure to read the general information in “Special notices” on page 297.

First Edition (October 2001)

This edition applies to Version 3, Release 2 of the IBM Parallel Support Programs for AIX (PSSP) Licensed Program (product number 5765-D51), and AIX Version 4 Release 3 Licensed Program (product number 5765-C34)

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. JN9B Building 003 Internal Zip 2834
11400 Burnet Road
Austin, Texas 78758-3493

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

Tables

Preface

Problem determination and problem solving on the RS/6000 SP and clusters can be difficult because the malfunction may imply the movement of several components within AIX and PSSP. Problem determination on the RS/6000 universal cluster is part of the daily tasks of a system administrator. Although the RS/6000 universal cluster is a very stable platform, the manner in which distributed and parallel environments are intermixed may cause certain problems that cannot be easily solved by someone with only AIX experience.

This redbook gives a comprehensive explanation of certain RS/6000 SP and cluster components and provides the reader with tools and procedures that can be used for problem isolation and problem solving in a cluster environment.

This redbook is oriented to RS/6000 SP and cluster professionals who install, configure, and administer universal cluster systems. Several procedures are outlined and tested, along with the explanation for the causes of common SP and cluster problems.

This universal clustering guide is the perfect companion for the RS/6000 SP and clusters product manuals when you need to identify and solve system problems.

The redbook does not guarantee success, but it certainly takes you one step closer. This redbook describes each cluster component and helps you understand how they are related to one another. It gives a comprehensive explanation of processes occurring within the SP system and provides an approach to diagnosis and problem solving.

The team that wrote this redbook

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

Dino Quintero is a project leader at the International Technical Support Organization (ITSO), Poughkeepsie Center. He has over nine years experience in the Information Technology field. He holds BS and MS degrees in Computer Science Marist College. Before joining the ITSO, he worked as a Performance Analyst for the Enterprise Systems Group, and as a Disaster Recovery Architect for IBM Global Services. He has been with IBM since 1996. His areas of expertise include enterprise backup and recovery, disaster recovery planning and implementation, and RS/6000 clustering technologies. He is also a Microsoft Certified Systems Engineer. Currently, he focuses on RS/6000 Cluster Technology by writing redbooks and teaching IBM classes worldwide.

Sorin Dumitrescu is a Senior Technical Consultant for Mutual Computer Consulting in New York City. He has 15 years of experience in the IT field, the last 10 years with UNIX. He holds a master's degree in Electronic Engineering from the Polytechnic University of Bucharest, Romania. His area of expertise include AIX, SP, HACMP, SAN and disaster recovery solutions and he is an IBM Certified Advanced Technical Expert in AIX/SP/HACMP. He was responsible with RS/6000 software services in IBM Romania and with a large SP implementation for a leading investment bank in New York.

Ian Gilbert is a Advisory IT Specialist working in the IBM Software Support Center in Sydney, Australia. He is the lead SP support there with 6 years of experience supporting the RS/6000 SP. He has worked with RS/6000 systems and AIX since they were first introduced, both in the field, as a hardware engineer, and currently in software support. He has worked at IBM since 1986. His areas of expertise and interest include AIX, SP, TCP/IP, HACMP, and Linux.

Marc-Eric Kahle is a RS/6000 Hardware Support specialist at the IBM ITS Central Region Back Office in Boeblingen, Germany. He has nine years of experience in the RS/6000 and AIX fields. He has worked at IBM Germany for 14 years. His areas of expertise include RS/6000 Hardware, including the SP, and he is also a AIX certified specialist. He also participated on the redbook RS/6000 SP System Tuning update.

Mohammad Arif Kaleem is an Advisory Sales Specialist working at the Saudi Business Machines (SBM), Saudi Arabia. He is providing support to the largest SP site in Middle East. Arif holds an MBA degree with major in MIS from IBA, University of Karachi, Pakistan. He has a combined 12 years working experience with SBM and IBM Pakistan. His areas of expertise include; AIX, SP, HACMP, TSM and TCP/IP networking. In the past, Arif has been part of several other projects at ITSO, including; AIXV4 announcement, GPFS for HSM application and Inside the RS/6000 SP.

Charles Parker is the President of Blue Neuron Technology in Milford, CT USA. He has 10 years of experience in the RS/6000 field. He holds a Masters degree in Neuroscience from Florida State University. His areas of expertise include RS/6000 SP, HACMP, Disaster Recovery, and SHARK ESS. He has been an IBM Certified Advanced Technical Expert since 1998. He has been part of several large RS/6000 SP project throughout the United States.

Team member photo:

(Front, Left to Right) Marc-Eric Kahle, Charles Parker, Ian Gilbert

(Back, Left to Right) Mohammad Arif Kaleem, Sorin Dumitrescu, Dino Quintero (project leader)

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

International Technical Support Organization, Austin Center
Matthew Parente

IBM Poughkeepsie
Peter Chenevert
Bruno Bonetti
Hal Turner
Bob Demkowicz
Chris DeRobertis
Jim Daley
Dave Quenzler

Paul Sonenberg
Joanna Husta
Bernard King-Smith
Scott Greenlese
Dan O'Brien
David Wong
Lissa Valletta
Myung Bae
Bob Leddy
Felipe Knop

IBM France
Patrice Quet
Claude Cregut
Serge Dyzers
Philippe Demontoux

IBM Germany
Hans Mozes
Mathias Mueller

BLUENEURON Technology
Jim Eyhorn

Special notice

This publication is intended to help the RS/6000 SP and clusters technical community when troubleshooting software problems. The information in this publication is not intended to substitute the PSSP Diagnosis Guide, GA22-7350 or any other problem determination documentation provided by RS/6000 hardware, AIX software, or PSSP software. See the PUBLICATIONS section of the IBM Programming Announcement for RS/6000 information about what publications are considered to be product documentation.

IBM trademarks

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

AFS
AIX
AS/400
e (logo)®
Electronic Service Agent
IBM ®
IBM.COM
LoadLeveler
Micro Channel
Netfinity
Notes
PAL
PowerPC
pSeries
Redbooks
Redbooks Logo
RETAIN
RMF
RS/6000
Service Director
SP

Comments welcome

Your comments are important to us!

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

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

  ibm.com/redbooks

- Send your comments in an Internet note to:

  redbook@us.ibm.com

- Mail your comments to the address on page ii.

1

Problem determination essentials

This chapter will set the stage for rest of this book. Here, we highlighted important planning tasks that are often overlooked and lead to problems that can be avoided. Some pro-active actions have been introduced to minimize the frequency of problem occurrences.

In this chapter, we define the problem determination process and introduce the PD (problem determination) methodologies that can be used in conjunction with the available tools. The problem determination tools are discussed in subsequent chapters.

1.1 Problem determination

There are two levels of the problem determination process:

- Take pro-active actions to minimize the occurrence of problems.
- Define clear problem determination methodologies to address problems that do occur.

1.2 Proactive actions

Although this book primarily addresses the steps you need to take when the problems have indeed occurred, we would also like to highlight some areas that, if planned for properly, can minimize the frequency of problems.

Remember that your SP is a non-static entity. It keeps changing its state over time as you install/relocate new hardware, install new LPPs, upgrade the software level, etc. You are required to have clear change management policies to apply new changes to your system. If not planned properly, there is always a potential that the change will introduce some hardware/software incompatibility or co-existence issue and lead to a problem.

People commonly assume that planning is required only at the time of initial installation. In reality, you need to practice to check on the hardware/software compatibility every time you plan a change.

In the following sections, you will find basic areas that you should review at the time of initial installation and again when applying changes to your system. This practice will minimize the frequency of problems.

1.2.1 There is no substitute for good planning

The importance of good planning cannot be overstated. Typically small to medium sites do not pay due attention to site planing activities. As previously mentioned, good site plan, installation plan, and subsequently change management policies can avoid many potential problems.

The latest versions of installation planning guides are available at:

`http://www.rs6000.ibm.com/resource/aix_resource/sp_books/planning/index.htm`

We recommend that you download, read and comply with the following guides:

- *RS/6000 SP Planning Volume 1, Hardware and Physical Environment,* GA22-7280

- *RS/6000 SP: Planning Volume 2, Control Workstation and Software Environmen*t, GA22-7281

These guides provide detailed planning sheets for major installation steps. They must be filled out before the initial system installation and documented for future use.

1.2.2 Comply with the configuration rules

The SP platform offers three type of nodes based on their form factor: Thin, wide and high nodes. It is important that your configuration comply with the node placement rule for switch connectivity. Remember, not all node combinations are allowed in a switch environment. Pay particular attention to the switch configuration rules if you are relocating nodes on a existing SP.

For example, if you have a frame of eight thin nodes and four wide nodes, you cannot attach an additional node on the expansion frame, even though you still have four free ports available on your switch. This is because PSSP treats this frame as an all thin nodes frame and, therefore, four ports on the switch are wasted in this scenario.

> **Important:** This is only applicable on the SP Switch.

The configuration rules have been discussed in detail in the following books:

- *RS/6000 SP Planning Volume 1, Hardware and Physical Environment*, GA22-7280
- Chapter 9, “Configuration rules and topologies,” *RS/6000 SP Cluster: The Path to Universal Clustering*, SG24-5374-01

Next, we provide a quick overview to cross-check your node placement in a switch environment.

When a switch frame has homogenous nodes

If a switch frame has all thin, wide, or high nodes, the expansion frame can be added as shown in Figure 1-1 on page 4. The switch nodes’ number offset are the numerals shown in the node position.

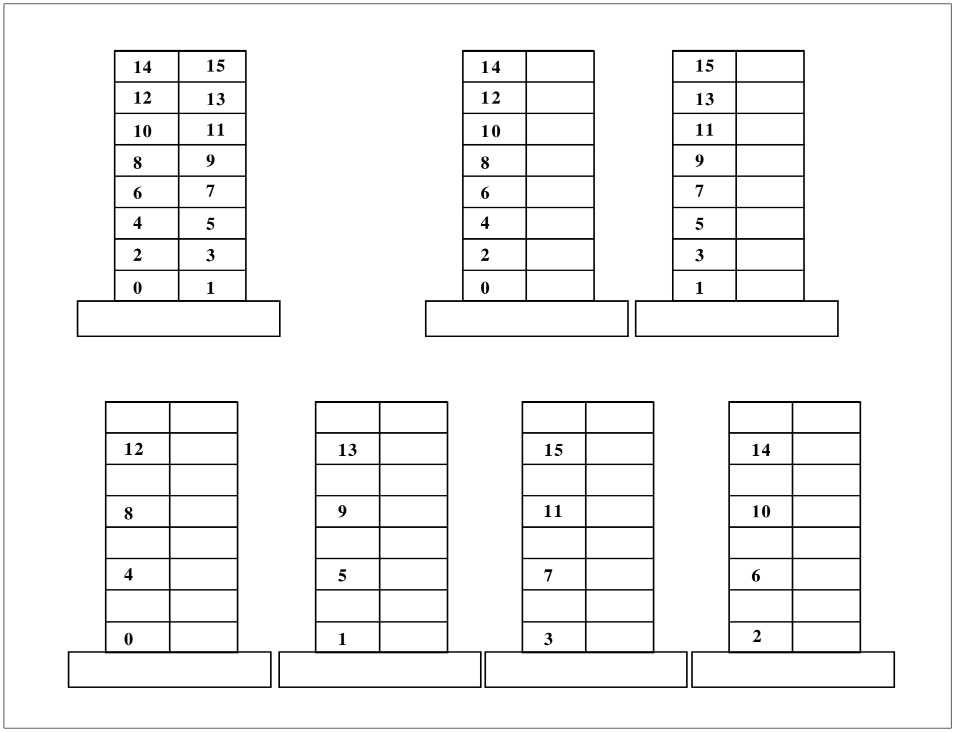


Figure 1-1 Switch example – the numerals are the switch node number offset

Switch frame has nodes of heterogeneous form factor

Some sites may contain nodes of different form factors shared on a common switch. Such situations require that the rule defined in Figure 1-1 be adhered to in adding expansion frames.

Figure 1-2 on page 5 provides an example of eight thin and eight high nodes mixed over a switch and utilizing all the available switch ports by using expansion frame. The switch node offsets are shown in the respective node location.

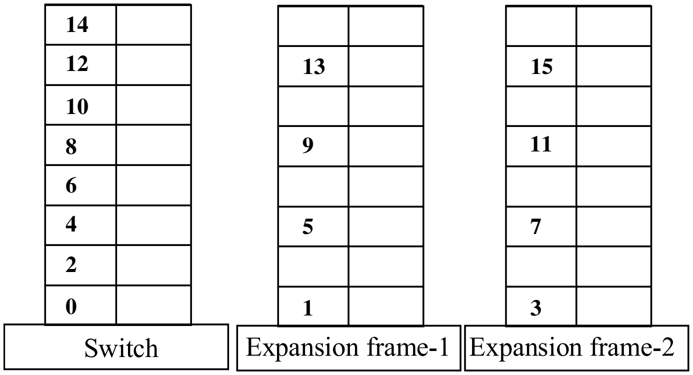
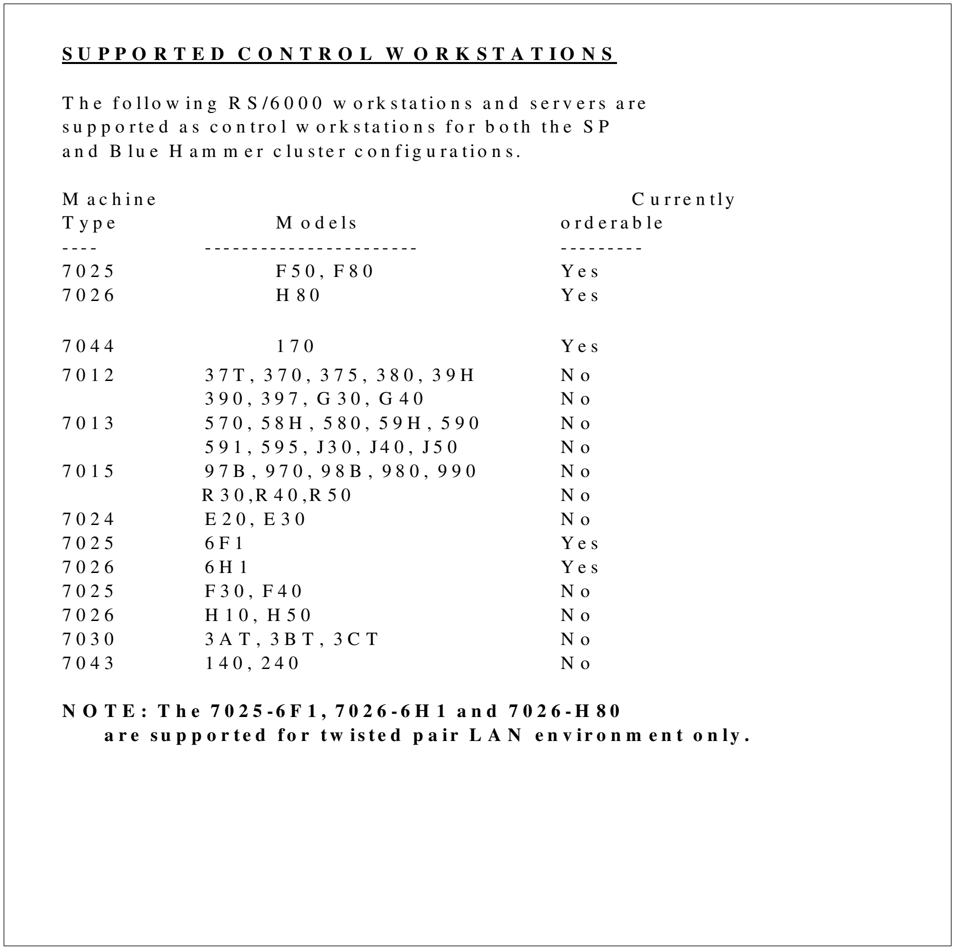


Figure 1-2 Nodes with heterogeneous form factor

1.2.3 Use a supported control workstation

The SP control workstation provides a single point of system management. However, it is important to note that not all RS/6000 or pSeries servers are supported to work as a control workstation. A list of the latest supported control workstations is presented in Figure 1-3 on page 6. Refer to the latest edition of *RS/6000 SP Planning Volume 1, Hardware and Physical Environment*, GA22-7280 for the most recent list.

Attention: pSeries 620 model 6F1, pSeries 660 model 6H1 and RS/6000 Model H80 are only supported on a twisted pair ethernet.



```
SUPPORTED CONTROL WORKSTATIONS

The following RS/6000 workstations and servers are
supported as control workstations for both the SP
and Blue Hammer cluster configurations.

Machine                                     Currently
Type          Models                        orderable
----     ----------------------             ---------
7025          F50, F80                      Yes
7026          H80                           Yes

7044          170                           Yes
7012     37T, 370, 375, 380, 39H            No
         390, 397, G30, G40                 No
7013     570, 58H, 580, 59H, 590            No
         591, 595, J30, J40, J50            No
7015     97B, 970, 98B, 980, 990            No
         R30,R40,R50                        No
7024     E20, E30                           No
7025     6F1                                Yes
7026     6H1                                Yes
7025     F30, F40                           No
7026     H10, H50                           No
7030     3AT, 3BT, 3CT                      No
7043     140, 240                           No

NOTE: The 7025-6F1, 7026-6H1 and 7026-H80
    are supported for twisted pair LAN environment only.
```

Figure 1-3   List of supported control workstations as of June 2001

1.2.4 Software levels and dependencies

The software compatibility matrix is often referenced and complied during the installation and planning phase. However, when a site upgrades software to a newer version, software compatibility issues are often overlooked and can cause problems in the operation of the SP.

Software compatibility and dependencies must be crossed-checked every time you plan to upgrade to a new version. Upgrading the level of one software component may also require an upgrade to other existing software components on your SP.

We recommend that you install the latest PTFs on your system. This can potentially avoid a problem that is fixed in the latest available PTFs. Table 1-1 on page 7 lists the IBM LPPs supported for different AIX and PSSP releases.

Note: If you are running multiple levels of AIX and PSSP on your SP, ensure that the control workstation is running the highest level of AIX and PSSP.

Table 1-1 IBM LPP's per supported PSSP and AIX release

PSSP and AIX	IBM LPP's
PSSP3.2 AIX 4.3.3	► LoadLeveler 2.2, 2.1 ► PE 3.1 ► ESSL 3.1 ► PESSL 2.1 ► GPFS 1.3, 1.2, 1.1 ► HACMP/ES and HACMP 4.3 ► HACMP with HAGEO or GeoRM 2.1
PSSP 3.1.1. AIX 4.3.3	► LoadLeveler 2.2, 2.1 ► PE 2.4 ► ESSL 3.1 ► PESSL 2.1 ► GPFS 1.2, 1.1 ► HACMP/ES and HACMP 4.3
PSSP 2.4 AIX 4.2.1 or 4.3.3	► LoadLeveler 2.2, 1.3 ► PE 2.3 ► PESSL 2.1 ► GPFS 1.1 ► RVSD 2.1.1 ► HACMP/ES and HACMP 4.2

Note: Prior to PSSP 3.1, IBM Recoverable Virtual Shared Disk was a separate LPP. The High Availability Control WorkStation and the Performance Toolbox Parallel Extensions components were priced features. They are now optional components that you receive with PSSP.

1.2.5 Verify the AIX and PSSP level for your hardware

Some customers, typically medium to large sized businesses, have multiple SP systems installed at their site and may relocate hardware (such as an adapter or the entire node) from one SP to another. However, this increases the chances of hardware and software incompatibility. A node working fine on one SP system may not work on the other, as the second system might not be running the required level of software.

To avoid this type of problem, always cross-check compatibility issues every time you add new hardware to your SP. If you are running an old PSSP and AIX version, you may need to apply new PTFs, or sometimes even need to completely upgrade the software.

Table 1-2 shows AIX and PSSP support for different node types.

Table 1-2 AIX and PSSP support for different node types

AIX/PSSP	120 MHZ Thin	135 MHZ Wide	160 MHZ Thin	200 MHZ High	332 MHZ Thin	332 MHZ Wide	POWER 3 Thin	POWER 3 Wide	POWER 3 High
AIX 4.2.1 PSSP 2.4	S	S	S	S	S	S	N	N	N
AIX 4.3.1 PSSP 2.4	S	S	S	S	S	S	N	N	N
AIX 4.3.2 PSSP 3.3	A	A	A	A	A	A	A	A	N
AIX 4.3.3 PSSP3.3.1	A	A	A	A	A	A	A	A	A
AIX 4.3.3 PSSP 3.2	A	A	A	A	A	A	A	A	A
AIX 4.3.3 PSSP3.3.1	A	A	A	A	A	A	A	A	A
AIX 4.3.3 PSSP 3.2	A	A	A	A	A	A	A	A	A

A = Available; factory- and field-orderable

S = Supported

N = Not supported

Note:

- AIX V4.2.1 with PSSP V2.4 is required for low cost models with high nodes. The low-cost models include tall and short frames with 8-port switch, and short frames without a switch.
- A single 332 MHz thin node in a drawer is supported only by AIX V4.3.2 with PSSP V3.1.
- 375 MHZ POWER3 SMP Nodes (features 2056 and 2057) are not supported on AIX 4.3.2 with PSSP 3.1.

1.2.6 Migration and coexistence of AIX and PSSP levels

Existing SP users must be fully aware of available migration paths when they upgrade to a newer versions of AIX and PSSP. Only certain migration paths are possible. Table 1-3 lists the currently supported migration path for AIX and PSSP.

Table 1-3 Supported migration paths for AIX and PSSP

From	To
PSSP 2.4 and AIX 4.2.1 or 4.3.3	PSSP 3.2 and AIX 4.3.3
PSSP 3.1.1 and AIX 4.3.3	PSSP 3.2 and AIX 4.3.3

Coexistence of different levels of software running in a single SP must be verified. Older versions of AIX and PSSP had software co-existence restrictions. Table 1-4, lists the supported combination of AIX and PSSP.

Table 1-4 Level of PSSP and AIX supported in a mixed system partition

Product	AIX 4.1.5	AIX 4.2.1	AIX 4.3.3
PSSP 3.2	N	N	S
PSSP 3.1.1	N	N	S
PSSP 2.4	N	S	S

S = Supported

N = Not supported

1.3 Problem determination methodologies

The keys to the problem determination process are understanding the symptoms and identifying the causes of symptoms. However, this is easier said than done. Particularly in a distributed environment, such as an SP or a cluster, the root cause of the problem may not be local and can be difficult to locate.

There are six basic steps required to identify and resolve a problem in a smart, systematic way:

Defining the problem

The first and most important step is to correctly define the problem. Find out the problem symptoms and a complete description of the error messages generated by the system. Also check whether the problem can be regenerated and study the actions that led up to the problem.

A good starting point is to refer to the *Parallel System Support Programs for AIX Diagnosis Guide, Version 3 Release 2*, GA22-7350-02 and *Parallel System Support Programs for AIX Messages Reference, Version 3 Release 2*, GA22-7352.

Isolating the problem

Different problems may have the same symptoms, but different causes. Determine the origin of the problem. Do some basic checking to roughly identify the cause. This may allow you to determine whether the problem is general to the entire system or is isolated to a well-defined component. Take a *global to local* approach where, in a step by step process, you can narrow down the problem causes and eventually pin-point the root cause.

For example, think of a problem where you cannot mount a directory on a node that was exported from the control workstation. Follow the steps below as a guideline to isolate the problem:

- Check to see if the **mount** command syntax is correct.
- Are you mounting to a valid directory?
- Verify that the directory exported has the correct access permission for the node.
- Are there any security issues with Kerberos and *acl* files?
- Do you have the same problems on all nodes or it is specific to this node only?

The important concept is to approach the problem in a systematic way so that you can isolate it to a single component.

Check to see if the problem has occurred before

If your site keeps documentation of problem history, then checking this documentation would be of help in this step. It is easy to address a problem effectively if you have dealt with a similar problem before. This is particularly true for site-specific problems, where certain combinations of hardware, software, and procedure triggers a problem.

Generate action plan - use all resources

While troubleshooting a problem, you need to adopt an intuitive approach; try to utilize all the resources and tools available to you. Before proceeding to the actual problem, cross check that all the hardware and software prerequisites have been met.There is no substitute for experience; the more you know about your site, the better it is. Never overlook the bigger picture of your environment. The major hindrance in the problem determination process is having a narrow view of the problem and limited knowledge of the overall environment.

AIX and PSSP provide a number of tools that will simplify the problem determination process. These tools are described in Chapter 2, “Problem determination tools” on page 13. If all tools fail, you need to be aware of the procedure to escalate the problems to your local IBM center for resolution.

Take corrective measures

Once the problem is defined and the root cause is identified, corrective steps are required. Often the problem is with some specific procedure that could be corrected, or a defect is found that requires the application of PTFs or a software or hardware upgrade.

Document the problem for the future

This area is generally overlooked, but it proves significantly important when expediting the resolution of future problems.

Typically, you should record the following information;

- Problem description
  - Problem symptoms
  - Description of error message or error codes
- Action or procedure that led up to the problem
- Applications running at that time
- Type of hardware used
- Software level including version, release, and service level
- Result of any corrective action, especially the one that failed and caused problems on the system

- Actual problem resolution with comments
- Number assigned to the problem, if you contacted IBM support

2

Problem determination tools

In this chapter, we introduce a wide range of tools available with AIX and PSSP.

First, the AIX and PSSP log management is discussed in detail. The purpose of these logs and their relevance with respect to the associated problem is discussed in detail. The commands and files related to AIX, BSD and SP-specific logs are also introduced, wherever necessary. The AIX error notification facility is explained with an example to help you write your own notification procedures.

Tools such as diag, trace, dump, invscout, and service agent are also covered to facilitate the problem determination process.

The SP problem management subsystem (PMAN) is introduced to help you manage problems on the SP. Key components of PMAN including pmand, pmanrmd, pmandef and sp_configd are explained in good detail. An example is provided to help you write your own event monitor.

Last, but not least, SP perspectives are discussed with respect to problem determination. We introduce event perspective and hardware perspective and elaborate them with a real-life example to demonstrate their usability.

2.1 PD tools

When you encounter a problem on your SP, the logical starting point is to review the error message and survey the relevant error logs.

In this section, we will discuss the various tools available for problem determination on the SP. These tools are provided to help you diagnose and fix the problems that can occur on the RS/6000 SP.

To learn about the available tools, we recommend you refer to, along with this redbook, the following books:

- *Parallel System Support Programs for AIX Diagnosis Guide, Version 3 Release 2*, GA22-7350
- *Parallel System Support Programs for AIX Messages Reference, Version 3 Release 2*, GA22-7352

The following problem determination tools available on the RS/6000 SP will be discussed in detail in the next sections.

- Log management facility
  - AIX error loggin facility
  - BSD error logs (syslogd)
  - PSSP specific error logs
- AIX trace facility
- AIX system dump facility
- The **diag** command
- Service Director
- Electronic service agent
- The **invscout** command
- Problem management subsystem
  - pmand
  - pmanrmd
  - pmandef
  - sp_configd
- SP Perspective
  - Hardware Perspective
  - Event Perspective

- Other command line tools

2.1.1 Error logs

The error logs provide a good starting point for the problem determination process. Useful details, such as error type, time stamp, error description, type and nature of the problem can be extracted from the logs. You can then create an action plan to troubleshoot the problem.

The RS/6000 SP uses the AIX and the BSD error log to report errors. IBM-specific software components generally use the AIX facility. BSD error logging is needed to support the public domain software included in PSSP, such as NTP and file collection. AIX includes the BSD error logging facility as a standard feature.

Beside errors and information being logged into the AIX error log, PSSP subsystems write to their own log files. Most of the SP log files are located in the /var/adm/SPlogs directory.

Appendix A, “SP Logs” on page 269 provides a list of all logs managed by PSSP.

2.1.2 Log management

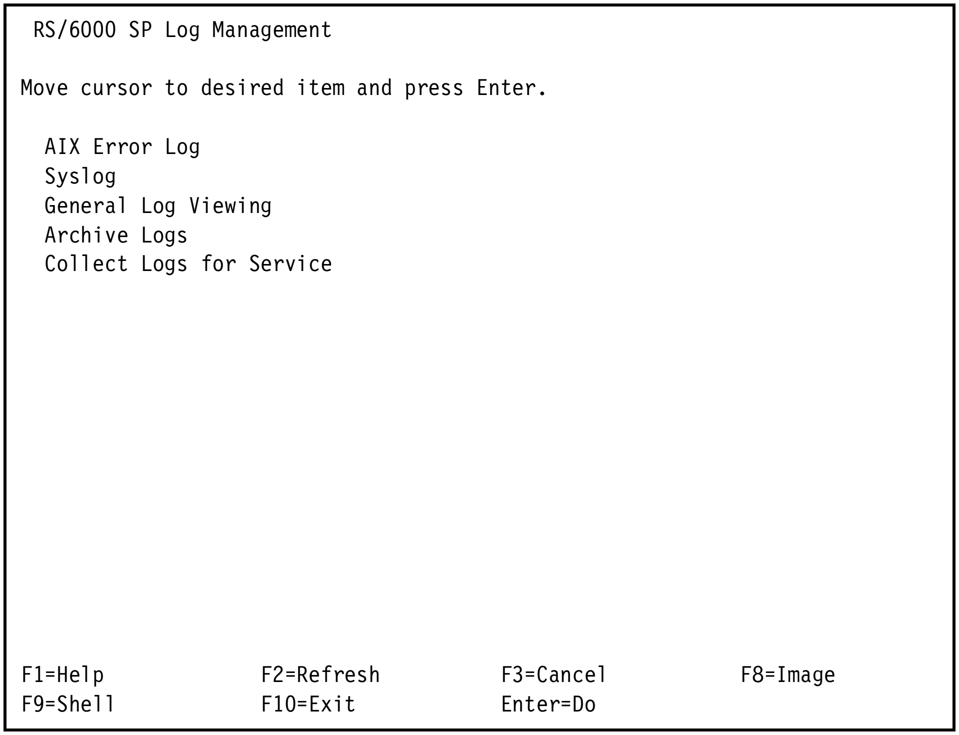
The SP nodes and the control workstation (CWS) are also independent AIX systems where you can locally run the AIX specific error log commands, such as **errpt** and **errclear**. However, a mechanism is required to manage a variety of error logs that are spread over several nodes, from a central location, regardless if they are AIX-, BSD-, or PSSP-specific logs.

In order to run AIX errol log and BSD syslog parallel (SP-specific) commands to generate parallel log reports, proper authorization is required. The error log management functions are provided in the ssp.sysman install option of the IBM Parallel System Support Programs for AIX.

The SP error log management consists of:

- SMIT panel interfaces
- A set of external commands
  - **splm** for general log viewing, archiving, and service collection
  - **psyslrpt** for generating reports of BSD syslog log files
  - **psyslclr** for trimming BSD syslog log files
  - **penotify** for creating, removing, or displaying Error Notification Objects
- Sysctl-based server functions

The log management primary access is available through the **smit splogmgt** command. Figure 2-1 provides the output of the smit log management screen.



```
 RS/6000 SP Log Management

Move cursor to desired item and press Enter.

  AIX Error Log
  Syslog
  General Log Viewing
  Archive Logs
  Collect Logs for Service











F1=Help             F2=Refresh          F3=Cancel           F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 2-1 SMIT splogmgt command

Log management functions are built upon the sysctl facility, which uses SP security services. Configuring the log management is different when using DCE or Kerberos V4.

Configuring log management with DCE authentication

When using DCE authentication, a log management use can obtain authorization by using the **dce_login** command with a DCE principal that is a member of the sysctl-logmgt group. The user can also be authorized by some other entry added to the DCE ACL in the /etc/logmgt.acl file by an SP security administrator who is a member of the spsec-admin group.

Configuring log management with Kerberos V4 authentication

When using Kerberos V4 authentication, the user needs to issue the **k4init** command to be identified to the SP authentication services. The user can now generate parallel AIX Error Log and BSD syslog reports and view any logs. All other log management commands additionally require that the user be defined as a principal in the /etc/logmgt.acl file. All users defined in this file must also be placed in the authentication database as a principal (PSSP Kerberos V4 or AFS).

> **Note:** Note that the majority of log management activities represent administrative tasks normally requiring root authority and that a user defined in the /etc/logmgt.acl file will execute commands as the root user.

Example 2-1 provides a sample of the /etc/logmgt.acl file:

Example 2-1 /etc/logmt.acl

```
#acl#
# This sample acl file for log management commands contains
# a commented line for a principal
#_PRINCIPAL root.admin@HPSSL.KGN.IBM.COM
# for trimming SPdaemon.log by cleanup.logs.ws
_PRINCIPAL rcmd.k7s
_PRINCIPAL root.admin@MSC.ITSO.IBM.COM
```

Configuring sysctl for log management

The log management server functions executed by Sysctl are located in /usr/lpp/ssp/sysctl/bin/logmgt.cmds. During system installation, an include statement for this file is added to the default Sysctl configuration file /etc/sysctl.conf. If you use an alternate Sysctl configuration file, you must update the file with a statement to include the logmgt.cmds file. In addition, you must restart the sysctld daemon to pick up this change.

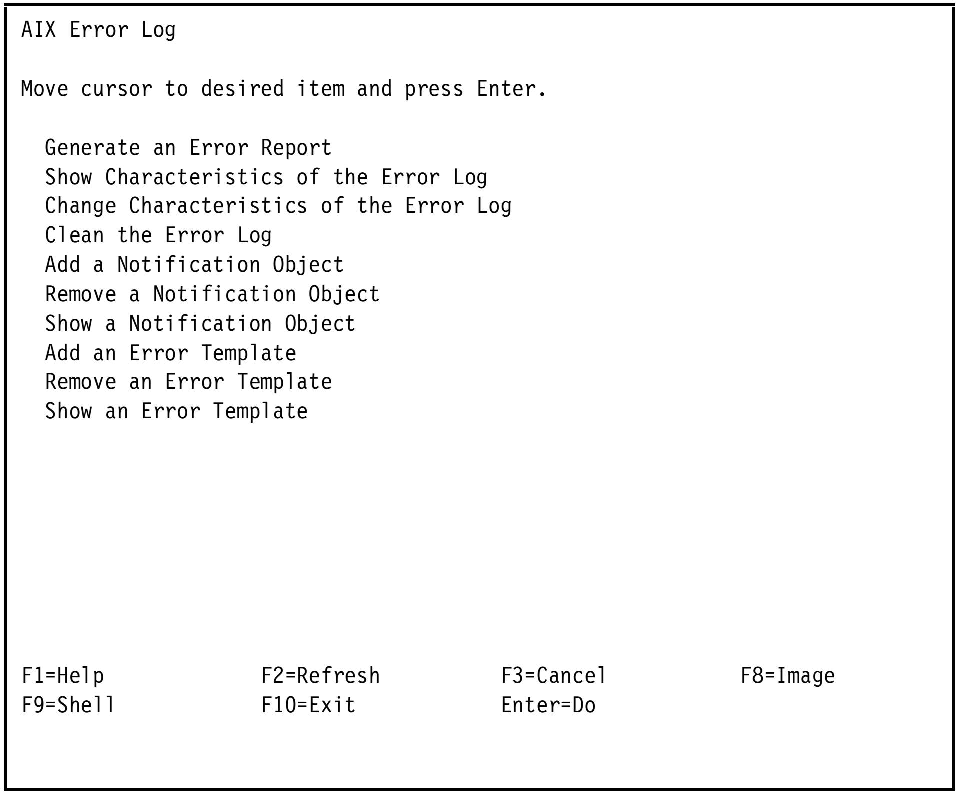
2.1.3 AIX error logging facility

All problem determination tools available through standard AIX are also available for SP nodes and the Control Workstations (CWS). However, being a distributed, clustered environment, new parallel external commands and SMIT panels have been added to facilitate the AIX log management on SP.

AIX provides facilities and tools for error logging, system trace, and system dump. Most of these facilities are included in the bos.rte fileset within AIX and, therefore, are installed on every node and CWS automatically. However, some additional tools are included in an optionally installable package, bos.sysmgt.serv_aid, which should be installed on your nodes and CWS.

The AIX error logging facility records hardware and software failures or informational messages in the /var/adm/ras/errlog. By analyzing this log, you would know what went wrong, when, and possibly why. While executing the AIX log management commands, you can specify whether the command should be executed on all nodes or a particular node. The default is the local node. Figure 2-2 provides the SMIT menu which is resulted from **`smit sperrlog`** command.

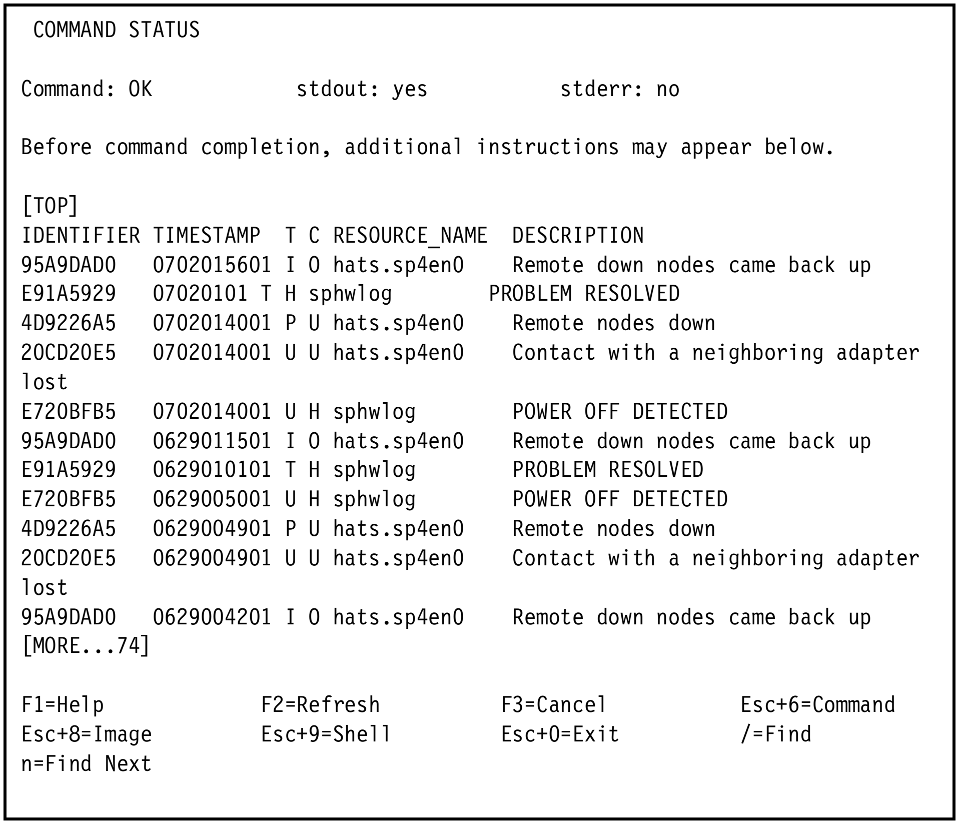
From this screen, apart from generating the AIX error report, a variety of other tasks can also be performed.



```
AIX Error Log

Move cursor to desired item and press Enter.

  Generate an Error Report
  Show Characteristics of the Error Log
  Change Characteristics of the Error Log
  Clean the Error Log
  Add a Notification Object
  Remove a Notification Object
  Show a Notification Object
  Add an Error Template
  Remove an Error Template
  Show an Error Template









F1=Help             F2=Refresh          F3=Cancel           F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 2-2   smit sperrlog

If you choose Generate an Error Report option, the SMIT will present several other options to filter the error log output. Figure 2-3 on page 19 provides an example of the AIX error log generated on the CWS using the default options.



```
 COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

[TOP]
IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
95A9DAD0   0702015601 I O hats.sp4en0    Remote down nodes came back up
E91A5929   07020101 T H sphwlog        PROBLEM RESOLVED
4D9226A5   0702014001 P U hats.sp4en0    Remote nodes down
20CD20E5   0702014001 U U hats.sp4en0    Contact with a neighboring adapter
lost
E720BFB5   0702014001 U H sphwlog        POWER OFF DETECTED
95A9DAD0   0629011501 I O hats.sp4en0    Remote down nodes came back up
E91A5929   0629010101 T H sphwlog        PROBLEM RESOLVED
E720BFB5   0629005001 U H sphwlog        POWER OFF DETECTED
4D9226A5   0629004901 P U hats.sp4en0    Remote nodes down
20CD20E5   0629004901 U U hats.sp4en0    Contact with a neighboring adapter
lost
95A9DAD0   0629004201 I O hats.sp4en0    Remote down nodes came back up
[MORE...74]

F1=Help             F2=Refresh          F3=Cancel           Esc+6=Command
Esc+8=Image         Esc+9=Shell         Esc+0=Exit          /=Find
n=Find Next
```

Figure 2-3 AIX errorlog

The following is a brief explanation of the title headers that appear in the summary or detailed output of the AIX error log:

- IDENTFIER is the error ID.

  The **`errpt -t`** command will display all possible types of errors that can be logged by the AIX logging facility.

- TIMESTAMP format is mmddhhmmyy where:
  - mm represent the month
  - dd is the day
  - hh is hour on 24-hour clock
  - mm represent the minute
  - yy is for the year

The **errpt -s mmddhhmmyy -e mmddhhmmyy** command will display all errors logged during the starting and ending time specified.

- T represents the type of error. The possible values are:
  - PERM for permanent errors
  - INFO for information
  - TEMP for temporary errors
  - UNKN for unknown error types
  - PEND for imminent errors
  - PERF for performance

  The command **errpt -T PERM** will display all the errors of a permanent type in the AIX error log.

  Note that, in the summary output of the AIX error log, you will only see one letter for the error type. For example, P for permanent error (instead of PERM*)*.
- C represent the error class. The possible values are:
  - H for hardware
  - S for software
  - U for undetermined
- RESOURCE NAME describes the subsystem name.
- DESCRIPTION provides a brief description of the error available in the error log template.

A more comprehensive and detailed output could be generated by using the **errpt -a** command on a selected node (or use the **smit perrpt** command and select the detailed output option). This command typically generates a long list, displaying details about all the logged errors. To focus on a specific error, issue the command **errpt -a -j 95A9DAD0**, where 95A9DAD0 is the error identifier. Example 2-2 provides detail output of the previous selected error.

Example 2-2 Detail output of selected error

```
---------------------------------------------------------------------------
LABEL:          TS_NODEUP_ST
IDENTIFIER:     95A9DAD0

Date/Time:       Mon Jul  2 01:56:02 EDT
Sequence Number: 1625
Machine Id:      000354784C00
Node Id:         sp4en0
Class:           O
Type:            INFO
```

```
Resource Name:   hats.sp4en0

Description
Remote down nodes came back up

Probable Causes
Remote nodes powered on
Remote nodes re-booted
Previous network problems corrected

User Causes
Remote nodes powered on by user
Remote nodes rebooted by user
:Remote nodes rebooted by user

        Recommended Actions
        Verify that remote nodes are up

Detail Data
DETECTING MODULE
rsct,connect.C,           1.60,1580
ERROR ID
6EfReJ0mn.Ev.79B0AG.e.1...................
REFERENCE CODE
6Z0WYB/AZ.Ev.Wrs0AG.e.1...................
File containing up node numbers and associated REFERENCE CODE
/var/adm/ffdc/dumps/hats.23750.20010702.015602
```

The Detail Data section of the detail output error log provides a DETECTING MODULE string that identifies the software component, module name, module level, and the defective line. This information can help you isolate the software module that might have caused the problem.

The errdemon daemon keeps the log file updated based on information and errors logged by subsystems through the errlog facility, or through the errsave facility, if they are running at kernel level. In either case, the errdemon daemon adds the entries in the error log on a first-come-first-served basis.

2.1.4 Managing the AIX error log

You can continue to use standard AIX error log commands by supplement them with the **dsh** command in a cluster environment. To take a snapshot of the error log from all the nodes and centralize them on CWS, execute the following command:

```
#dsh -a errpt -s 0215133001 >/var/adm/ras/ALL.0215133001.log
```

This command will accumulate error logs of all nodes on the CWS, starting from FEB 15 2001 at 13:30 hours. The format of the -s flag is mmddhhmmyy (month, day, hour, minute, and year). The name and location of this file is arbitrary.

You can invoke *smit* fastpath parallel commands to manage AIX error logs on selected nodes or on the CWS using the SP log management feature.

> **Attention:** The fastpath invocation to generate an error report menu is: **`smit perrpt`**

Trimming the AIX error log

By using the following command, you can trim or completely clean the entries in the error logs on selected nodes.

```
smit perrclear
```

Show characteristics of AIX error Log

The fastpath invocation to show the characteristics menu of the error log is:

```
smit perrdemon_shw
```

Change characteristics of AIX error log

The fastpath invocation to change characteristics for the error log is:

```
smit perrdemon_chg
```

> **Important:** Keep at least 4 MB for the error log.

2.1.5 AIX error notification

PSSP provides facilities for log monitoring and error notification. This differs from the standard AIX notification in the sense that, although it uses the AIX notification method, it also provides a global view of your system. You can, for example, create a monitor for your AIX error log on all your nodes at once with a single command or with a few clicks of mouse.

Error notification objects are ODM objects held in the class errnotify. They are used by the AIX error notification facility to invoke methods upon occurrence of an error event. Fields in the errnotify class are matched with the fields in the error template for selection. If an error is logged matching the selection criteria defined in a notification object, the method associated with that object is invoked.

You can add, remove, and show notification objects in parallel on the SP system.

- The fastpath invocation to Add a Notification Object menu is:

  smit padd_en

- The fastpath invocation to Remove a Notification Object menu is:

  smit prem_en

- The fastpath invocation to Show a Notification Object menu is:

  smit pshw_en

From the command line, use **penotify -f show** to add, remove, or show error notification objects in parallel on the SP.

Important: No parallel log management command, including **penotify** will be executed without proper authorization. You need to add your PRINCIPLE in the/etc/logmgt.acl file, as shown in Example 2-1 on page 17.

Chapter 4, "Using the AIX Error Log Notification Facility" of the *PSSP: Diagnosis Guide*, GA22-7350, provides good examples of setting up the notification methods.

Error notification example

You can setup error notification to inform you when a specific error occurred or, more intuitively, you can get an alert even before the error become a real problem. This can be done if you monitor the error type PEND for all pending errors.

It is a good idea to trap errors both on nodes and on the CWS. To do so, you can add an appropriate error notification object in the ODM. The following list defines an error notification that will send an e-mail to root when an error type PEND occurs:

1. Set up a work directory to hold the script that runs when the error occurs:

   ```
   #mkdir /<work_directory>/objects
   #mkdir /<work_directory>/methods
   ```

2. Create a script that runs when the error occurs, as in Example 2-3 on page 24.

Tip: Do not forget to provide the scripts with running privileges (+x).

Example 2-3 Script that runs when an error occurs

```
#cat /<work_directory>/methods/errnot.sh
~
#!/bin/ksh
######################################################################
#Run errpt to get the fully expanded error report for the error
#that was just written and redirect to a unique tempfile with the PID
#of this script.
######################################################################
errpt -a -l $1 >/tmp/tempfile.$$
####################################################################
#Mail the fully expanded error report to root@controlworkstation
#This could be anywhere in the network.
#root@controlworkstation is the user and hostname that the
#administrator wants to be notified at.
#####################################################################
mail root </tmp/tempfile.$$
```

3. Create a new errornotify object in the ODM.

   Make a script with the new entries as shown in Example 2-4.

> **Important:** Save the errornotify ODM objects with the command:
>
> `#cp /etc/objrepos/errnotify /etc/objrepos/errnotify.bkp`

Example 2-4 errornotify entries for PEND errors

```
#cat /<work_directory/objects/odmadd.txt
~
errnotify:
en_name =”errnot.PEND.obj ”
en_persistenceflg =1
en_type =”PEND ”
en_method =”/<work_directory/methods/errnot.sh $1 ”
errnotify:
en_name =”errnot.pend.obj ”
en_persistenceflg =1
en_type =”pend ”
en_method =”/<work_directory/methods/errnot.sh $1 ”
errnotify:
en_name =”errnot.Pend.obj ”
en_persistenceflg =1
en_type =”Pend ”
en_method =”/<work_directory/methods/errnot.sh $1 ”
```

Note: The variations of PEND are added because the use of an upper case format is not strictly adhered to by all AIX LPPs and vendors.

4. Add the entries to the odm database.

   #odmadd /<work_directory/objects/odmadd.txt

   An e-mail similar to Example 2-5 on page 25, will be sent to root whenever an error type PEND occurs.

Example 2-5 E-mail sent to root when an error occurs

```
---------------------------------------------------------------------------
LABEL:TS_NODEDOWN_EM
IDENTIFIER:4D9226A5
Date/Time:Fri Feb 16 18:37:37
Sequence Number:12232
Machine Id:000504936700
Node Id:sp6en0
Class:U
Type:PEND
Resource Name:hats.sp6en0
Resource Class:NONE
Resource Type:NONE
Location:NONE
VPD:
Description
Re ote nodes down
Probable Causes
Re ote nodes powered off
Re ote nodes crashed
Networking problems render remote nodes unreachable
Re ote nodes removed from configuration after refresh
Topology Services daemon on remote nodes stopped
User Causes
User powered off remote nodes
Re ote nodes removed from configuration after refresh
Recommended Actions
Confirm that this is desirable
Failure Causes
Re ote nodes crashed
Lost connection to remote nodes due to network problems
Re ote nodes hang
Recommended Actions
Get system dump from remote nodes.Re-boot remote nodes
Clear networking problems
Re-start Topology Services daemon on remote node
Contact IBM Service if problem persists
Detail Data
```

```
DETECTING MODULE
rsct,connect.C,1.57,1510
ERROR ID
.ZOWYB//bPXu.sib.UO.e.z...................
REFERENCE CODE
File containing down node numbers and associated REFERENCE CODE
/var/adm/ffdc/dumps/hats.23220.20010216.183737
```

Useful ODM commands

To view the recently created errnotify objects in the ODM database, enter:

```
#odmget -q “en_name =errnot.PEND.obj ” errnotify
#odmget -q “en_name =errnot.pend.obj ” errnotify
#odmget -q “en_name =errnot.Pend.obj ” errnotify
```

To delete the recently created errnotify objects, enter:

```
#odmdelete -o errnotify -q “en_name =errnot.PEND.obj ”
#odmdelete -o errnotify -q “en_name =errnot.pend.obj ”
#odmdelete -o errnotify -q “en_name =errnot.Pend.obj ”
```

To spread this process across the nodes, follow these steps:

1. Repeat step 1 with the **dsh -a** command:

```
#dsh -a mkdir /<work_directory>/objects
#dsh -a mkdir /<work_directory>/methods
```

2. Change the script to send an e-mail to root@MACN (CWS) so you have only one e-mail to look at. Copy the script from the MACN (CWS) to the nodes:

   ```
   #pcp -a /<work_directory>/methods/errnot.sh
   ```

3. Copy the script with the ODM entries to the nodes:

   ```
   #pcp -a /<work_directory>/objects/odmadd.txt
   ```

4. Repeat the **odmadd** to all nodes with the **dsh -a** command:

   #dsh -a odmadd /<work_directory/objects/odmadd.txt

Important:

Remember to save the errnotify ODM with the command:

```
#dsh -a cp -p /etc/objrepos/errnotify /etc/objrepos/errnotify.bkp
```

2.1.6 BSD error logging

The BSD syslog is a well known and widely implemented error logging facility. PSSP exploits the syslog mostly for information reporting as opposed to error reporting.

The syslogd daemon, which logs the errors, is configured with a file to determine the destination of the incoming messages. The default configuration file is /etc/syslog.conf. The BSD syslog errors are classified by the facility that is issuing the error, and by the error's priority value. Based on the priority values, entries in the configuration file determine the destination for each error message. Destinations can be a file, user ID, or the syslogd daemon on another machine. We suggest that error messages be logged locally rather than forwarded to a remote syslog because of the increased network traffic. File collections can be used to maintain consistent configuration of the syslog facility.

For a list of these facilities and priority values, refer to the syslogd in the *IBM AIX Commands Reference Volume 1*, SBOF-1877.

Format of the syslog file

The format of a syslog output file is:

```
MMM DD HH:MM:SS node_name resource[pid]: msg
```

Where:

- MMM DD HH:MM:SS is the timestamp (month day hour:minute:second).
- node_name is name of the node where the error has occurred.
- resource is the name of the failing system.
- pidis optionally logged process ID of the failing resource.
- msg is a free form error message.

Errors logged by SP components will contain in the message section, the following additional information pertaining to the logging resource:

LPP	LPP name.
Fn	filename.
SID	SID_level_of_the_file.
L#	Line number or function.

Useful commands

The SMIT fastpath command to access the Syslog general menu is:

```
smit spsyslog
```

The SMIT fastpath invocation to generate a Syslog report menu is:

```
smit spsyslrpt
```

From the command line: To report to local node all records logged by the FTP for all syslog log files on nodes in the current system partition:

```
psyslrpt -a -r ftp
```

From the command line: To report all records that were logged to files selected for the daemon and user facilities starting on March 3, and to report records to the local node from nodes host1 and host2.

```
psyslrpt -w host1,host2 -f user,daemon -s 03030000
```

> **Tip:** The size of syslog log files is not configurable and will continue to grow until manually trimmed. You must perform regular trimming of syslog file to avoid a file system full condition.

The fastpath invocation to trim syslog log files menu is:

```
smit spsyslclr
```

From the command line: To trim all records older than 30 days from the log file /var/adm/msgs on the local node.

```
psyslclr -y 30 -l /var/adm/msgs
```

From the command line: To trim all records from all log files found in the alternate syslog configuration file /etc/syslog.conf.

```
psyslclr -g /etc/syslog.conf -y 0
```

> **Attention:** If the syslogd daemon is stopped during the trimming process, use the -g option to restart it with either the default configuration file or an alternate file.

The **psyslclr** command can be added as a crontab entry for performing scheduled syslog trimming. On the control workstation, **psyslclr** is used to trim daemon facility messages older than six days. This is done in /usr/lpp/ssp/bin/cleanup.logs.ws, which runs from the control workstation's crontab file.

2.1.7 SP log files

Beside errors and information being logged into the AIX error log, most of the PSSP subsystems write to their own log files where, usually, you will find the information you need for problem isolation and problem determination.

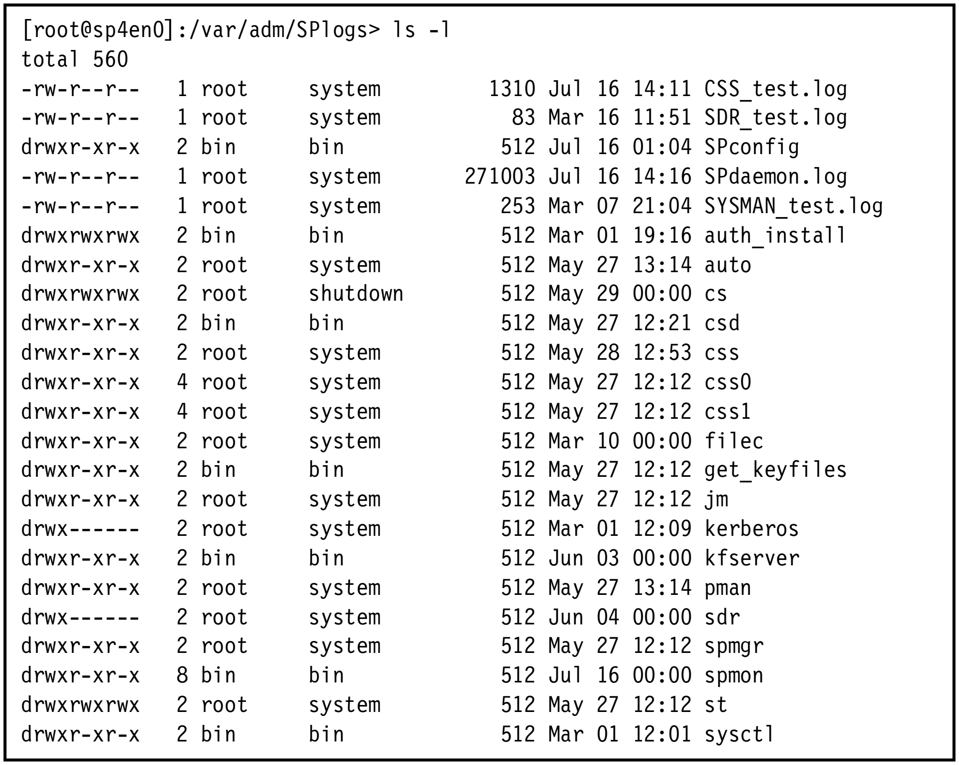
Although some components run only on the CWS (such as SDR daemon, host respond daemon, and switch admin daemon), others, such as the switch daemon, run only on the nodes. This needs to be taken into the consideration while searching for error logs. Appendix A, “SP Logs” on page 269 provides a complete list of SP logs and their locations.

PSSP log files are located in the /var/adm/SPlogs directory whereas RSCT log files are located in the /var/ha/log directory. In addition there are few miscellaneous logs files that can be found in the /tmp and /var/sysman/log/* directory.

Important: Considering that the /var file system is holding the majority of the log files, it is important to monitor the size of this file system. Refer to the *IBM RS/6000 SP: Planning, Volume 2, Control Workstation and Software Environment*, GA22-7281 for file system disk space requirements.

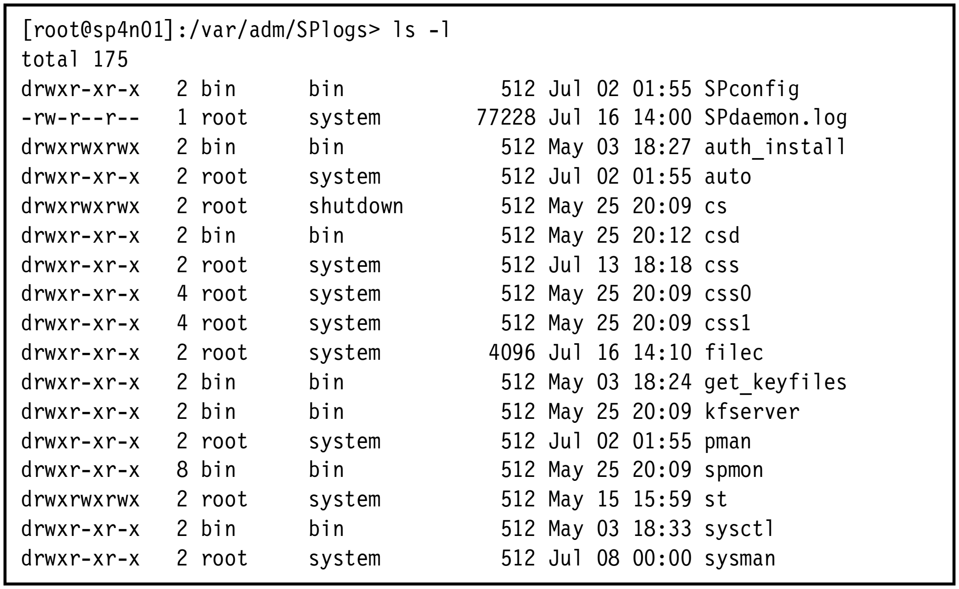
2.1.8 The /var/adm/SPlogs directory

This is the prime location for important SP logs. The contents of /var/adm/SPlogs directory on CWS is listed in Figure 2-4 on page 30.



```
[root@sp4en0]:/var/adm/SPlogs> ls -l
total 560
-rw-r--r--   1 root     system        1310 Jul 16 14:11 CSS_test.log
-rw-r--r--   1 root     system          83 Mar 16 11:51 SDR_test.log
drwxr-xr-x   2 bin      bin            512 Jul 16 01:04 SPconfig
-rw-r--r--   1 root     system      271003 Jul 16 14:16 SPdaemon.log
-rw-r--r--   1 root     system         253 Mar 07 21:04 SYSMAN_test.log
drwxrwxrwx   2 bin      bin            512 Mar 01 19:16 auth_install
drwxr-xr-x   2 root     system         512 May 27 13:14 auto
drwxrwxrwx   2 root     shutdown       512 May 29 00:00 cs
drwxr-xr-x   2 bin      bin            512 May 27 12:21 csd
drwxr-xr-x   2 root     system         512 May 28 12:53 css
drwxr-xr-x   4 root     system         512 May 27 12:12 css0
drwxr-xr-x   4 root     system         512 May 27 12:12 css1
drwxr-xr-x   2 root     system         512 Mar 10 00:00 filec
drwxr-xr-x   2 bin      bin            512 May 27 12:12 get_keyfiles
drwxr-xr-x   2 root     system         512 May 27 12:12 jm
drwx------   2 root     system         512 Mar 01 12:09 kerberos
drwxr-xr-x   2 bin      bin            512 Jun 03 00:00 kfserver
drwxr-xr-x   2 root     system         512 May 27 13:14 pman
drwx------   2 root     system         512 Jun 04 00:00 sdr
drwxr-xr-x   2 root     system         512 May 27 12:12 spmgr
drwxr-xr-x   8 bin      bin            512 Jul 16 00:00 spmon
drwxrwxrwx   2 root     system         512 May 27 12:12 st
drwxr-xr-x   2 bin      bin            512 Mar 01 12:01 sysctl
```

Figure 2-4 /var/adm/SPlogs - directory view on CWS



```
[root@sp4n01]:/var/adm/SPlogs> ls -l
total 175
drwxr-xr-x   2 bin      bin            512 Jul 02 01:55 SPconfig
-rw-r--r--   1 root     system       77228 Jul 16 14:00 SPdaemon.log
drwxrwxrwx   2 bin      bin            512 May 03 18:27 auth_install
drwxr-xr-x   2 root     system         512 Jul 02 01:55 auto
drwxrwxrwx   2 root     shutdown       512 May 25 20:09 cs
drwxr-xr-x   2 bin      bin            512 May 25 20:12 csd
drwxr-xr-x   2 root     system         512 Jul 13 18:18 css
drwxr-xr-x   4 root     system         512 May 25 20:09 css0
drwxr-xr-x   4 root     system         512 May 25 20:09 css1
drwxr-xr-x   2 root     system        4096 Jul 16 14:10 filec
drwxr-xr-x   2 bin      bin            512 May 03 18:24 get_keyfiles
drwxr-xr-x   2 bin      bin            512 May 25 20:09 kfserver
drwxr-xr-x   2 root     system         512 Jul 02 01:55 pman
drwxr-xr-x   8 bin      bin            512 May 25 20:09 spmon
drwxrwxrwx   2 root     system         512 May 15 15:59 st
drwxr-xr-x   2 bin      bin            512 May 03 18:33 sysctl
drwxr-xr-x   2 root     system         512 Jul 08 00:00 sysman
```

Figure 2-5 /var/adm/SPlogs - directory view on of a node

The structure of /var/adm/SPlogs directory on a node is presented in Figure 2-5.

We now review the subdirectories in /var/adm/SPlogs and discuss the differences between the logs available on the nodes and on the CWS.

SPconfig

The SPconfig subdirectory has information on VPD data. On the CWS, you have information from all the nodes. The information in this directory is created using the **`/usr/lpp/ssp/install/bin/save_config`** command. The **`save_config`** command, when run on a node, creates files locally and then copies them to the CWS. When executed on CWS, it collects information from the /var/adm/SPlogs/SPconfig directory of all nodes and copies them locally in the/var/adm/SPlogs/SPconfig directory of CWS. It is therefore recommended to run **`save_config`** first on nodes and then on the CWS.

The following is a brief explanation of the log files present in this directory.

nn.lpp.levels

The file name is pre-fixed by the node number (nn=0 for CWS). It is the output of **`lslpp`** command for the ssp.basic and ssp.css component. It only provides the base level information, even if you have a different modification applied. We recommend that you run the **`lslpp -l`** command when requiring the levels of PSSP.

nn.lscfg

This file contains the output of the **lscfg** command from the node and CWS. Where nn is the node number suffix.

nn.umlc

This file is a result of is a combination of the **lsvpd** and **/usr/lpp/diagnostics/bin/umlc** commands. All files are prefixed with node number. In a node, you see only its own files.

auth_install

The auth_install subdirectory has one file named *log*. This file has log information from the **spauthconfig** and **updauthfiles** commands. **spauthconfig** is invoked from /etc/rc.sp, so it runs on every reboot. The **updauthfiles** command is called upon from within the **spauthconfig** command. When DCE authentication method is used, the **rm_spsec** command is logged in the log file.

auto

The auto subdirectory contains a file called auto.log, which has information from the automount daemon.

cs

The *cs* subdirectory has information on the **cstartup** and **cshutdown** commands. The log files have the following format:

```
cshut.<timestam>.<pid>
cstart.<timestamp>.<pid>
```

This directory is available only on the CWS.

csd

The csd subdirectory contains information about IBM Recoverable Virtual Shared Disk (RVSD) subsystem. There are two important files in this directory:

1. vsd.log: Summary of actions from RVSD.
2. vsd.debuglog: Detailed information of actions from RVSD.

css

css is a comprehensive directory containing important files related to the switch. Table 2-1 provides a list of files in /var/adm/SPlogs/css directory and their location.

Table 2-1 Files included in css directory

File	Description	Location
cable_miswire	Information about cable miswire from nodes.	Primary node
cssadm.debug	Detailed information about the Switch admin daemon. It is in chronological order. Take a look at the time near the error.	CWS
cssadm.stderr	Standard error output from the switch admin daemon.	CWS
cssadm.stdout	Standard output for switch admin daemon.	CWS
daemon.log	Standard output for SP Switch2 daemon.	Nodes
daemon.stderr	Standard error output for fault service daemon (Worm) for the SP Switch.	Nodes
daemon.stdout	Standard output for fault service daemon (Worm) for the SP Switch.	Nodes
dist_topology.log	Error messages when the primary node distributes the topology file across all nodes.	Primary node
dtbx_failed.trace	Trace file about diagnostic for the SP Switch adapter.	Node
dtbx.trace	Information about diagnostic for the SP Switch adapter.	Node
dtbxworm.stderr	Standard error output for messages from adapter diagnostics.	Nodes
Eclock.log	Information from all **Eclock** commands.	CWS
Ecommands.log	Information from all Ecommands issued.	CWS
Emonitor.log	Information from all **Emonitor** commands.	CWS
Emonitor.Estart.log	Information from **Estart** command issued by the Emonitor daemon.	CWS, nodes
Eunpart.file	Information from an **Eunpartition** operation.	Primary node
fs_daemon_print.file	Trace file of fault service daemon messages.	Nodes

File	Description	Location
flt	All information from a Switch fault. This file is very useful in problem determination. Always look at the time of the error to try to understand what happened.	Nodes
logevnt.out	stdout and stderr of the Event Management Resource Monitor and Methods.	CWS
msdg.log	SP Switch and SP Switch2 advanced diagnostics Message Daemon log.	CWS
out.top	Information about the current topology file.	Primary node
rc.switch.log	Information about the initialization of the SP Switch support code. This is another good file to start looking at.	Nodes
router.log	Log from switch router generation.	Nodes
router_failed.log	Error information when a problem is detected from a router generator.	Nodes
spd.trace	SP Switch advanced diagnostics tests and architecture components log.	CWS, nodes
spd_gui.log	SP Switch advanced diagnostic GUI log.	CWS

summlog	Summary of entries in AIX error log for all nodes. Entries in the log have the following fields, which are separated by blanks:
	`Timestamp: MMDDhhmmYYYY`
	`Node: Reliable hostname`
	`Snap: If Y there is snap capture else N`
	`Partition name`
	`Index: AIX error log sequence number`
	`Label: AIX error log label.`
		CWS

summlog.out	stdout and stderr for the CSS logging daemon's Event Management client.	CWS
topology.data	Status of the current network topology.	Primary node
worm.trace	Worm trace file from switch initialization.	Primary node

css0 and css1

There are two subdirectories in /var/adm/SPlogs called *css0* and *css1*, one for each switch adapter. These directories are available only in PSSP 3.2 or later release. These directories have two more subdirectories *p0* and *p1*, for each port of the switch adapter. This structure is located on the nodes and all information is for the SP Switch2 only. Table 2-2, list the files within this structure. The level column can assume two values:

1. Adapter: The file is located in /var/adm/SPlogs/cssX, where X is the adapter ID (0 or 1).
2. Port: The file is located in /var/adm/SPlogs/cssX/pY, where X is the adapter ID (0 or 1), and Y is the port ID (0 or 1).

Table 2-2 New structure of SP Switch2 logs

File	Description	Level
cadd_dump.out	Output of the **`cadd_dump`** command. This file is a dump from the adapter device driver message buffer.	Adapter
ifcl_dump.out	Output of the **`ifcl_dump`** command. This file is a dump from the adapter device driver IP message buffer.	Adapter
col_dump.out	Output of the **`col_dump`** command. This file is a dump from the microcode trace buffer.	Adapter

odm.out	Contains the following output for each adapter:
	`/bin/odmget -q "attribute=adapter_status"PdAt`
	`/bin/odmget -q "attribute=adapter_status"CuAt`
		Adapter

spdata.out	Output of the following commands:
	`/usr/lpp/ssp/css/splstdata -f -G`
	`/usr/lpp/ssp/css/splstdata -s -G`
	`/usr/lpp/ssp/css/splstdata -n -G`
	`/usr/lpp/ssp/css/splstdata -b -G`
		Port

netstat.out	Output of the following commands:
	`netstat -I css0`
	`netstat -m`
		Adapter

scan_out.log scan_save.log	The TBIC, which is a chip on the SP Switch2 adapter, is scanned into this file and saved as part of the snapshot by the css.snap script. The scan_save.log is for the previous TBIC scan.	Adapter

File	Description	Level
DeviceDB.dump	This file contains the latest dump of the device data base from the fault service daemon.	Port
adapter.log	This file contains the tracing of the fault service daemon adapter events.	Port
flt	This file is used to log hardware error conditions on the switch, recovery actions taken by the fault service daemon, and other general operations that alter the switch configuration.	Port
fs_daemon_print.file	This file contains the tracing of the fault service daemon port events.	Port
out.top	This file is now located on each node. It contains the actual topology for the network. Whenever a change in the topology occurs, the file will be updated in all nodes.	Port
topology.data	This file is located only in the primary node. It contains error messages that occur during the distribution of the topology file to the other nodes. The topology file distribution is done via the SP Switch2.	Port
cable_miswire	This file is located only in the primary node. It reports any miswire detected by the fault service daemon during SP Switch2 initialization (running the Worm subsystem).	Port
colad.trace	The file contains trace information from css0 adapter diagnostics.	Adapter
spd.trace	This file contains tracing of advanced switch diagnostics.	Port

filec

The /var/adm/SPlogs/filec, located on the nodes, contains information about the **supper** command. Basically, you see two files in the following format:

1. sup<date>.<time>: The output of the **supper** command.
2. sup<date>.<time>r: The actions performed by the **supper** command.

get_keyfiles

The /var/adm/SPlogs/get_keyfiles directory, located on the nodes, contains the file get_keyfiles.log which has the output of the last **get_keyfiles** command issued.

jm

The /var/adm/SPlogs/jm directory contains information regarding the resource manager. The file jmd_out contains the messages of the Resource Manager, and the file jm_err contains the errors from the Resource Manager.

kerberos

The /var/adm/SPlogs/kerberos file contains information about Kerberos Version 4 authentication. Table 2-3 list the files in this directory.

Table 2-3 Kerberos log files

File	Description	Location
admin_server.syslog	This file contains information about the authentication database administration daemon.	Primary authentication server
kerberos.log	This is the log of the primary authentication server.	Primary authentication server
kerberos.slave_log	This is the log of the secondary authentication server.	Secondary authentication server
kpropd.log	This file contains the log of the authentication database propagation daemon.	Secondary authentication server

kfserver

The kfserver subdirectory contains information about srvtab files. In kfserver.log, you can find messages generated from the transfer of the srvtab files to the nodes. The regserver.log file has the messages generated by the registration process for the kfserver program.

pman

The pman subdirectory contains messages generated by the problem management daemon. On the nodes, the file is called pmand.log; whereas, on CWS it is referred to as pmand.<partition_name>.log.

sdr

Information about the sdr daemon is kept in the sdr subdirectory. On CWS, there are two files:

1. SDR_config.log; contains configuration messages from SDR.
2. sdrdlog.<partition>.<pid>; contains error messages from the sdr daemon.

On the nodes there is only the SDR_config.log file.

spacs

The login control messages, on the nodes, are stored in the var/adm/SPlogs/spacs/spacs.log file.

spmgr

The /var/adm/SPlogs/spmgr directory stores information about extension nodes, such as the 9077 SP Switch Router. The spmgrd.log is maintained by the spmgrd daemon.

spmon

The spmon directory is located on the CWS. The Table 2-4 shows the spmon directory structure.

Table 2-4 spmon logs

File	Description
hmlogfile.<julian-date>	hardmon initialization and error messages. When the hardmon starts to respawn, this is a good log to look at.
nc/nc.<frame>.<node>	Messages from the **nodecond** command. We recommend you run tail -f in this file during the install/migration, diagnostic, and maintenance processes.
nfd/nfd.<frame>.log.<julian-date>	Netfinity daemon information.
s70d/s70d.<frame>.log.<julian-date>	Messages from the hardware monitor s70d daemon.
splogd/splogd.<pid>	Contains the PID of the splogd daemon.
splogd.state_changes.<timestamp>	Changes reported by the splogd daemon.
ucode/ucode_log.<frame.node>	Microcodes download messages when using the **smit supervisor** command.
spmon_ctest.log	Output of the **spmon_ctest** command.
spmon_itest.log	Output of the **spmon_itest** command.

st

The information and error messages from the Job Switch Resource Table and Services is in the /var/adm/SPlogs/st/st_log file on the nodes.

sysctl

Information about the Sysctl server is kept in /var/adm/SPlogs/sysctl/sysctld.log file both on nodes and the CWS.

sysman

The sysman subdirectory is useful during installation, migration and customization of nodes. Table 2-5 lists its contents.

Table 2-5 sysman subdirectory log files

File	Description	Location
mirror.out	Contains AIX error messages when mirroring a volume group the SP volume group commands.	MACN (CWS), nodes
<node>.config.log.<pid>	Contains messages from the **`pssp_script`** command. This command runs during the installation/migration and customization of the nodes. When you have a node that was set to customize and does not return automatically to disk, check this file to see what happened.	Nodes
<node>.configb.log.<pid>	The **`pssp_script`** commands calls the **`psspfb_script`**, which uses this file as output. It is also very useful for installation/migration and customization problems.	Nodes
<node>.console.log	All the console messages for the node.	MACN (CWS), nodes
spfbcheck.log	Output of the **`spfbcheck`** command that runs after installation/migration or when the node is set to customize.	Nodes
SYSMAN_test.log	The /var/adm/SPlogs/SYSMAN_test.log file contains information and error messages from the **`SYSMAN_test`** command.	MACN (CWS), nodes
unmirror.out	Contains AIX error messages when unmirrorring a volume group the SP volume group commands.	MACN (CWS), nodes

CSS_test.log

The /var/adm/SPlogs/CSS_test.log file contains the output of the **CSS_test** command. It is useful for debugging purposes. This file resides only on the CWS.

SDR_test.log

The SDR_test.log file contains the output of the **SDR_test** command on both nodes and the CWS. This file is located in the /var/adm/SPlogs directory.

SPdaemon.log

There is general information about system daemons and hardware errors in the /var/adm/SPlogs/SPdaemon.log file. This file resides both on nodes and the CWS.

2.1.9 RSCT log files

The information from the RSCT daemons is stored in the /var/ha directory. The following are the important log files:

hags

The var/ha/log/hags* files contain information collected from the hags daemon. The files have following two formats:

1. hags.<partition>_<node>_<incarnation>.<partition> on the CWS
2. hags_<node>_<incarnation>.<partition> on the nodes.

To find out the active hags file, execute **ls -ltr hags*** command and check the last updated file as shown in Example 2-6.

Example 2-6 Determining the active hags log file

```
# cd /var/ha/log
/var/ha/log # ls -ltr hags*
-rw-r--r--   1 root     system    675838 Nov 02 11:14 hags.sp3en0_0_14.sp3en0
-rwxr-xr-x   1 root     system      5065 Nov 02 11:15
hagsglsm.default.sp3en0.0_14
-rwxr-xr-x   1 root     system      8716 Nov 02 11:15 hags.default.sp3en0.0_15
-rw-r--r--   1 root     system    467013 Feb 15 10:04
hagsglsm.sp3en0_0_14.sp3en0.bak
-rw-r--r--   1 root     system    635601 Feb 26 10:39
hags.sp3en0_0_15.sp3en0.bak
-rw-r--r--   1 root     system    149613 Feb 26 14:49
hagsglsm.sp3en0_0_14.sp3en0
-rw-r--r--   1 root     system    164547 Feb 26 14:50 hags.sp3en0_0_15.sp3en0
-rwxr-xr-x   1 root     system      6659 Feb 26 14:50
hagsglsm.default.sp3en0.0_15
-rwxr-xr-x   1 root     system      7876 Feb 26 14:50 hags.default.sp3en0.0_16
```

```
-rw-r--r--   1 root     system      1966 Feb 26 17:11
hagsglsm.sp3en0_0_15.sp3en0
-rw-r--r--   1 root     system    315278 Feb 26 17:11 hags.sp3en0_0_16.sp3en0
-rwxr-xr-x   1 root     system      3552 Feb 26 18:34
hagsglsm.default.sp3en0.0_16
-rwxr-xr-x   1 root     system      7237 Feb 26 18:34 hags.default.sp3en0.0_17
-rw-r--r--   1 root     system    557502 Feb 28 11:04 hags.sp3en0_0_17.sp3en0
-rw-r--r--   1 root     system     23505 Feb 28 11:30
hagsglsm.sp3en0_0_16.sp3en0

/var/ha/log # date
Wed Feb 28 11:38:24 EST 2001
```

Based on the **`date`** command, we see that hags.sp3en0_0_17.sp3en0 is the active log file.

hagsglsm

The hagsglsm daemon stores its trace and log information in the /var/ha/log/hagsglsm* files. The format of the files is as follows:

1. hagsglsm.<partition>_<node>_<incarnation>.<partition> on the CWS.
2. hagsglsm_<node>_<incarnation>.<partition> on the nodes.

To determine the active log, follow the procedure in Example 2-6 on page 40, and look for the latest hagsglsm log file.

haem

The activity log for the *haem* daemon is in the /var/ha/log/em.default.<partition> file. This file reside on nodes and on CWS.

hats

The hats daemon stores its information in the /var/ha/log/hats.<dd>.<hhmmss>.<partition-name> file, where dd specifies the day the daemon gets started and hhmmss represents the start time in hours, minutes, and seconds. Use the procedure described in Example 2-6 on page 40 to determine the active log file.

Note: Remember to change hags to hats.

The topology information from the hats startup is collected in the /var/ha/log/hats.<partition-name> file.

haem

Messages from the hardmon resource monitor are stored in the /var/ha/run/haem.<hostname>/IBM.PSSP.hmrmd/IBM.PSSP.hmrmd_log.<julian-date> file.

2.1.10 Miscellaneous log files

The following are two important log files that do not reside on the locations mentioned in the previous section:

- When the file_collection is set to *true* in the environment, the command **filec_config** is called during **install_cw** and during the installation of the nodes. The logs for this command is maintained in the /var/sysman/logs/* directory.
- The **setup_server** command creates the file /tmp/spot.out.<PID> (when the SPOT is being created). The file /tmp/spot.update.out.<PID> contains information about the update of the SPOT. Both files are useful in determining problems that occur during **setup_server** procedure.

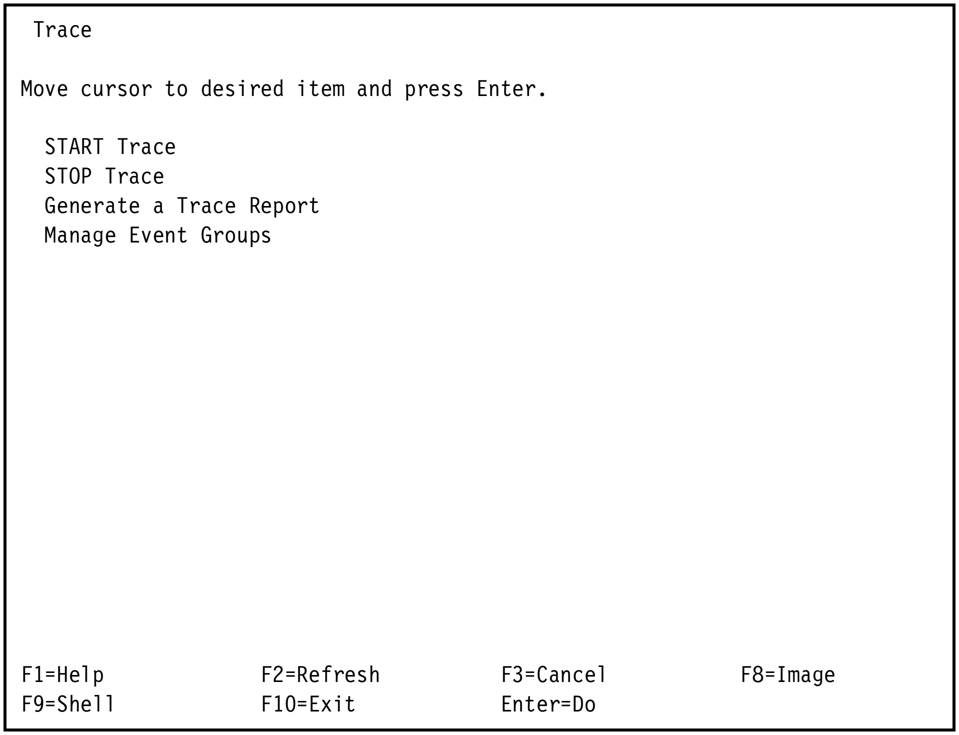
2.1.11 The trace facility

Tracing is a cumbersome process and it requires commitment and dedication to understand the trace report. Although the base system (bos.rte) includes minimal services for trace, you still need to install bos.sysmgt.trace, an optional installable component, to activate the trace daemon and generate trace reports.

Tracing works in a two-step mode: You turn on trace on selected subsystems and/or calls, and then you analyze the trace file through report tools.

The events that can be included or excluded from the tracing facility are listed in the /usr/include/sys/trchkid.h header file. They are called hooks and subhooks. With these hooks, you can specify to the tracing facility the specific event you want to trace. For example, you could generate a trace for all CREAT calls, which include file creation.

The **smit trace** command provides a user friendly interface to the trace facility. The trace menu provides the option Manage Event Group, which lets you manipulate the event listed in /usr/include/sys/trchkid.h. Figure 2-6 on page 43 shows the SMIT menu for trace.



```
 Trace

Move cursor to desired item and press Enter.

  START Trace
  STOP Trace
  Generate a Trace Report
  Manage Event Groups









F1=Help             F2=Refresh          F3=Cancel           F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 2-6 Smit trace screen

To learn more about tracing, refer to Chapter 11, “Trace Facility” in *AIX V4.3 Problem Solving Guide and Reference*, SC23-4123.

2.1.12 The system dump facility

AIX generates a system dump when a severe error occurs. A system dump can also be initiated by users with root authority. It creates a snapshot of your system’s memory contents.

A system dump can help in determining what took the machine out of order. A good system dump in the right hands can point to the faulty component.

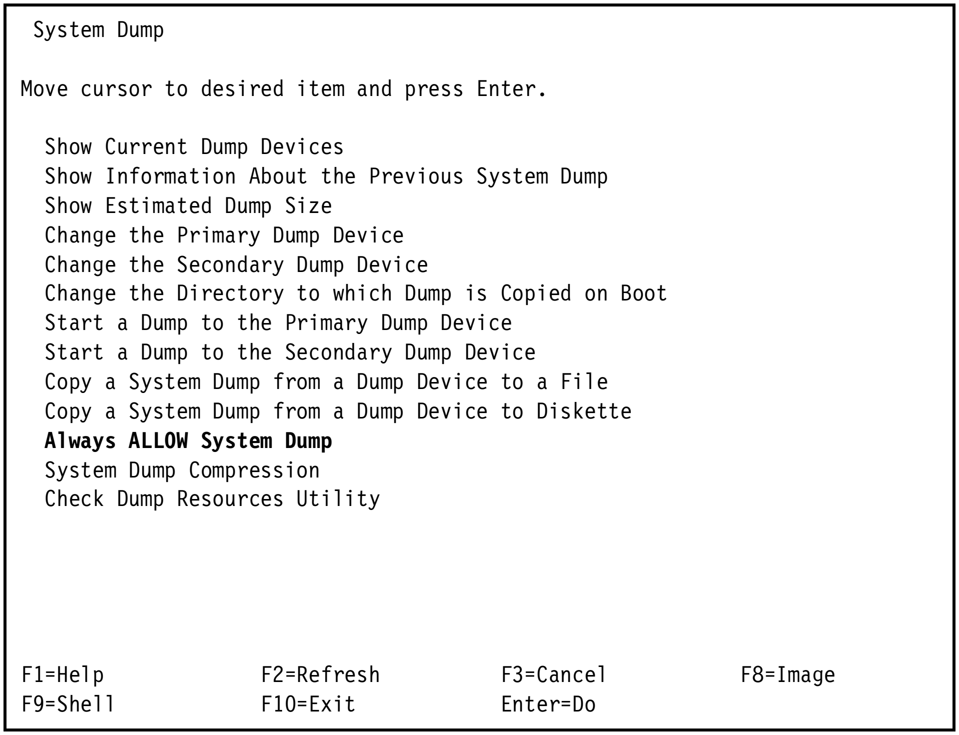
A system dump is a copy of selected areas of the kernel. These areas contain information about the processes and routines running at the moment of the crash. However, while it is easier for the operating system to keep this information in memory-address format, the output is not so friendly to humans. Therefore, for a good system dump analysis, you need the table of symbols. It is a good practice to send the copy of /unix, along with the system dump, to IBM centers.

There are four possible ways to initiate a system dump:

1. If the software service aid package is installed

   If you have *bos.sysmgt.serv_aid* installed on your system, you can use the command line or SMIT to initiate a system dump.

   You can initiate a system dump with the **`sysdumpstart`** command. Issue the **`sysdumpdev -l`** command to verify the current dump device. The **`sysdumpdev`** can also be used to changed the dump device. Alternatively, use the **`smit dump`** fastpath to get to the dump screen. Figure 2-7 shows the system dump options available through SMIT.



```
 System Dump

Move cursor to desired item and press Enter.

  Show Current Dump Devices
  Show Information About the Previous System Dump
  Show Estimated Dump Size
  Change the Primary Dump Device
  Change the Secondary Dump Device
  Change the Directory to which Dump is Copied on Boot
  Start a Dump to the Primary Dump Device
  Start a Dump to the Secondary Dump Device
  Copy a System Dump from a Dump Device to a File
  Copy a System Dump from a Dump Device to Diskette
  Always ALLOW System Dump
  System Dump Compression
  Check Dump Resources Utility





F1=Help             F2=Refresh          F3=Cancel           F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 2-7 SMIT screen for system dump options

2. If the software service aid package is NOT installed

   If you do not have bos.sysmgt.serv_aid installed, you can use one of the following methods to initiate a system dump:

3. Using the reset button

   Start a system dump with the reset button by following the directions in the list below. This procedure works for all system configurations and will work in circumstances where other methods for starting a dump will not.

  - Turn your machine’s mode to service position or set always allow system dump to true. This can be done by selecting the always allow system dump option on the SMIT dump menu as shown in Figure 2-7 on page 44. By default, this option is set to false.
  - Press the reset button.

   This will write the dump information to the primary dump device.

4. Using special key sequences

   You may start a system dump with special key sequences by doing the following:

   - Turn your machine’s mode switch to the service position, or set always system dump to true. This can be done by selecting the always allow system dump option on the SMIT dump menu as shown in Figure 2-7 on page 44. By default this option is set to false.
   - Press the CTRL+ALT+1 key sequence to write dump information to primary dump device. Use CTRL+ALT+2 key sequence to write the dump information to the secondary dump device.

> **Important:** You can start a system dump by this method *only* on the native keyboard.

For more information on AIX system dumps, refer to Chapter 12, “System Dump Facility,” in *AIX V4.3 Problem Solving Guide and Reference*, SC23-4123.

2.1.13 The alog command

The **alog** command provides another important tool to reference the logs specified in the alog configuration database. It also works with log files that are specified on the command line. The **alog** command reads the standard input, writes to standard output, and copies the output into a fixed-size file.

By default, the following logs are available:

- boot
- bosinst
- nim
- console
- dumpsymp

The bootlog, for example, helps you view the messages appear during the boot process. The **alog** is a circular log and when this file is full (you can always change the size of the log), new entries are written over the oldest existing entries. Depending on the size of the file, you can log multiple boot records.

The logs are kept in /var/adm/ras directory. Use the following command to view bootlog.

```
alog -o -f bosinstlog
```

Alternatively, you can use the **smit alog** fastpath command to manipulate the alog options.

A new log type sample could be added to the alog configuration database by creating the alog.add file in ODM database. To learn more about alog; refer to *The AIX Commands Reference, Volume 1*, SBOF-1877.

2.1.14 Hardware diagnostic – the diag command

Not all problems are software related. If you are facing a problem, it could well be caused by some fault in the hardware; a disk may have dirty blocks and causing read/write errors, a tape drive is failing intermittently during the backup, a network adapter might have failed completely and caused problems in communication with other systems, etc.

The **diag** command provides a variety of hardware diagnostics on your system. It is the starting point to run a wide choice of tasks and service aids. Most the tasks and services aids available through this command are platform specific. Example 2-7 provides a complete list of tasks and services available through the **diag** command.

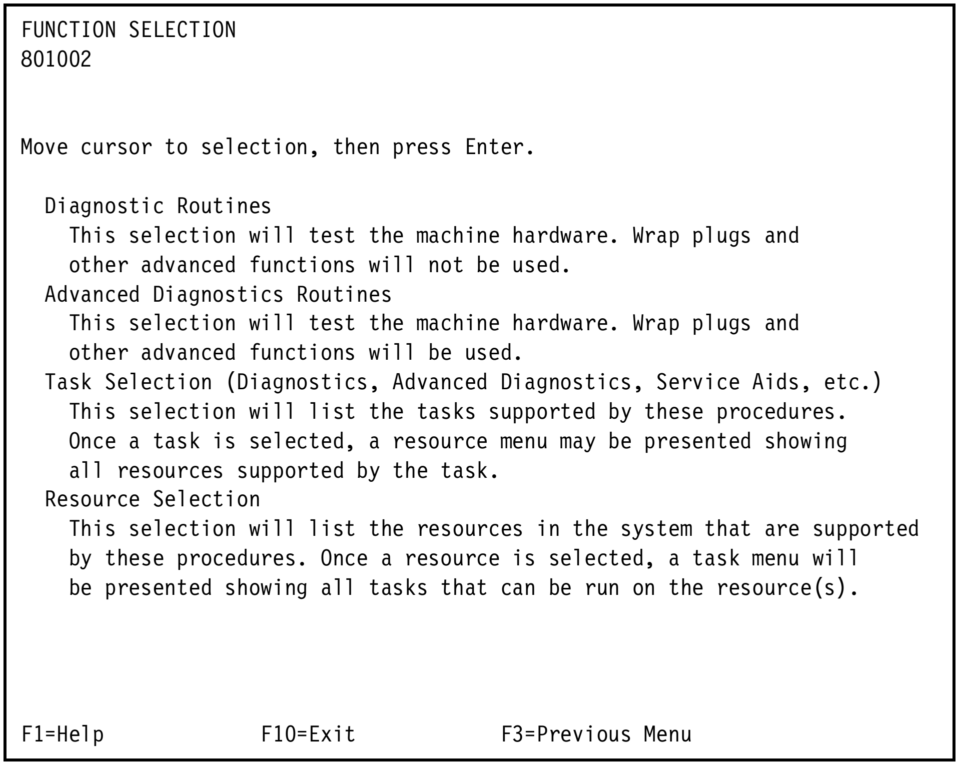
Example 2-7 Tasks and service aids available through diag command

```
Run Diagnostics
     Display or Change Diagnostic Run Time Options
     Display Service Hints
     Display Previous Diagnostic Results
     Display Hardware Error Report
     Display Software Product Data
     Display Configuration and Resource List
     Display Hardware Vital Product Data
     Display Resource Attributes
     Change Hardware Vital Product Data
     Format Media
     Certify Media
     Display Test Patterns
     Local Area Network Analyzer
     Add Resource to Resource List
```

```
Delete Resource from Resource List
SCSI Bus Analyzer
Download Microcode
Display or Change Bootlist
Periodic Diagnostics
Backup and Restore Media
Disk Maintenance
Configure Dials and LPFkeys
Add or Delete Drawer Config
Create Customized Configuration Diskette
Update Disk Based Diagnostics
Configure ISA Adapter
AIX Shell Prompt (Online Service Mode only)
Display or Change Multiprocessor Configuration
     Enable and disable individual processors
Display or change BUMP Configuration
     Update the flash EPROM with a new binary image
     Display or change diagnostic modes
     Display or change remote phone numbers and modem configurations
Display or Change Electronic Mode Switch
Process Supplemental Media (Standalone Mode only)
Generic Microcode Download
Run Error Log Analysis
Service Aids for Use with Ethernet
Update System Flash (RSPC)
Configure Ring Indicate Power-On (RSPC)
Configure Service Processor (RSPC)
Save or Restore Service Processor Configuration (RSPC)
Display Machine Check Error Log (RSPC)
7135 RAIDiant Array Service Aids
SCSI Device Identification and Removal
SCSD Tape Drive Service Aid
Escon Bit Error Rate Service Aid
SSA Service Aid
PCI RAID Physical Disk Identify
Configure Ring Indicate Power On Policy (CHRP)
Configure Surveillance Policy (CHRP)
Configure Reboot Policy (CHRP)
Configure Remote Maintenance Policy (CHRP)
Save or Restore Hardware Management Policies (CHRP)
Display Firmware Device Node Information (CHRP)
Spare Sector Availability
7318 Serial Communication Network Server
Update System or Service Processor Flash (CHRP)
Display System Environmental Sensors (CHRP)
Display Checkstop Analysis Results
Analyze Adapter Internal Log
Flash SK-NET FDDI Firmware
```

```
    Display Microcode Level
```

The **diag** command can be invoked from the command line or through the fastpath **smit diag** screen. Figure 2-8 shows the initial menu, which appears when you execute the **diag** command. Further selections can be made through this screen.



```
FUNCTION SELECTION
801002


Move cursor to selection, then press Enter.

  Diagnostic Routines
    This selection will test the machine hardware. Wrap plugs and
    other advanced functions will not be used.
  Advanced Diagnostics Routines
    This selection will test the machine hardware. Wrap plugs and
    other advanced functions will be used.
  Task Selection (Diagnostics, Advanced Diagnostics, Service Aids, etc.)
    This selection will list the tasks supported by these procedures.
    Once a task is selected, a resource menu may be presented showing
    all resources supported by the task.
  Resource Selection
    This selection will list the resources in the system that are supported
    by these procedures. Once a resource is selected, a task menu will
    be presented showing all tasks that can be run on the resource(s).




F1=Help             F10=Exit            F3=Previous Menu
```

Figure 2-8 diag command initial selection screen

2.1.15 Service Director

Service Director is a separately installable program product and is offered as a part of the IBM warranty or IBM maintenance service package for no additional charge. Due to entitlement checking at the IBM server, machines not on IBM warranty or under service agreement can take benefit from this service.

Service Director is a widely used product, but is now functionally replaced by a Java based electronic service agent. However, Service Director is still supported and available from IBM. We encourage you move to the electronic service agent.

Note: Service Director and Service Agent require remote dialing and support infrastructure which is not implemented in all the countries. Check with your local IBM representative for the availability of this service in your country.

Functions provided by Service Director

The Service Director application can automatically report hardware related problems to IBM through a modem on a local server. It can also automatically perform problem analysis on those issues before calling for the service. This level of Service Director supports all classic RS/6000, pSeries and SP systems. Classic RS/6000 refers to machines that have concurrent diagnostics installed. Some of the PCI-based PowerPC machines had diagnostics on CD-ROM and were not concurrent.

Service Director aids system errors to be dynamically monitored and analyzed; no customer intervention is required. Service Director event log can further facilitate error analysis for some errors once the service representative is onsite. Customers' hardware error logs can now be reduced, as errors are being maintained within the Service Director event.

The Service Director performs following functions:

- Automatic problem analysis
- Manual creation of problem reports
- Automatic customer notification
- Automatic problem reporting
  - Service calls placed to IBM without operator intervention
  - Vital Product Data (VPD) reported to IBM
- Common focal point for service interface
- Problem-definable threshold levels for call placement
- Reduced hardware error logs
- High-availability cluster multiprocessing (HACMP) support for full fallback; includes high-availability
- High availability cluster workstation (HACWS) for SP
- Simple and complex environment support with minimum analog lines

Service Director components

Service Director contains three major components:

Product Support Application (PSA)

The PSA determines appropriate error disposition, then captures and passes information required to resolve any problem identified on a particular product or option to the analysis engine.

Analysis Routine

The analysis routine within Service Director schedules the execution of the PSAs. They can be configured to run constantly or on a specific time schedule. When the analysis routine runs, it monitors errors or events identified by a PSA. errors or events are logged. Depending on customer configured options, the analysis routine can automatically notify, for example, customer's system administrator, and also automatically transmits the hardware errors and associated problem information, to IBM support center for remote analysis and action. If necessary, an IBM service representative is sent to the customer site with the parts needed to correct the problem reported.

Display Routine

The display function is the user's interface to Service Director for RS/6000. It provides a structured view of problem management information,.

How to get the Service Director code

Service Director code can be obtain by the following ways:

- SD is delivered with any new RS/6000 SP machine.
- Your IBM service sales representative can get it for you.
- SD may be obtained from:

  `ftp://ftp.software.ibm.com/aix/servdir_client_code/`

2.1.16 Electronic Service Agent

The electronic Service Agent (SA) is no-charge software that resides on your system to monitor events and transmit event data to the IBM service center. The customers within a warranty period or with a valid service agreement can take advantage of this tool.

Service Agent also helps you prepare for imminent errors, such as an increasing number of bad sectors on the hard disk. This information assists in shortening the service turnaround time. In some cases, IBM may notify a customer of an impending error and provide a resolution prior to an outage.

Information collected through Service Agent will be made available to IBM service support representatives when they are helping to answer questions or diagnosing problems.

Note: Service Director or Service Agent requires remote dialing and support infrastructure which is not implemented in all countries. Check with you IBM representative for the availability of this service.

Electronic service agent – components

The following are three major components of ESA infrastructure:

1. IBM Service Agent Server (SAS)

   The Service Agent Server (SAS) is located at the IBM service center. It is the machine to which your computer(s) sends information that is stored, analyzed, and acted upon by IBM.

2. Gateway machine

   The gateway machine is a customer system where Service Agent is installed using the svcagent.installp fileset. The gateway is located at a customer site and provides a central database and processes for communication with the IBM service center, via a modem.

3. Forwarder

   It is a system interface defined to gateway a database as a client or interface that forwards requests to or from other client machines. Typically, you can have several forwarders and one gateway.

Electronic service agent – functions

The following basic functions are available through the electronic service agent:

- Automatic problem analysis.
- Problem-definable threshold levels for error reporting.
- Automatic problem reporting; service calls placed to IBM without intervention.
- Automatic customer notification.
- Commonly viewed hardware errors; you can view hardware event logs for any monitored machine on the network from any Service Agent host user interface.
- High-availability cluster multiprocessing (HACMP) support for full fallback; includes high-availability cluster workstation (HACWS) for SP.
- Network environment support with minimum telephone lines for modems.

Electronic service agent – working principle

The Service Agent code is installed on the host by using the Service Agent user interface. Once installed, they are registered with the IBM Service Agent Server (SAS).

During the registration process, an electronic key is created that becomes part of your resident Service Agent program. This key is used each time Service Agent places a call for service. The IBM Service Agent Server checks the current customer service status from the IBM entitlement database; if this reveals that you are not on a warranty or maintenance agreement, then the service call is refused and posted back via e-mail notification.

Service Agent is not designed to arbitrarily pick up just any general information without it having been programed to do so. There is some data that Service Agent does bring to IBM to help with problem resolution. In some cases, this information may very likely be used by IBM for other purposes. This information consists of the problem or error information itself and Vital Product Data (VPD) or Inventory data.

If concerned about sensitivity of data, you can review the actual data that is being sent to IBM by using the Service Agent User Interface and take actions and prevent data transmission to IBM service center. Today the only data, besides error information, being sent to IBM is Vital Product Data (VPD), which is generated by either the **`lscfg`** command or the new **`invscout`** program. You can run one of these commands on a system and determine if this information is of a sensitive nature or not.

2.1.17 Electronic service agent prerequisites

The following describes the list of prerequisites for electronic service agent:

- AIX Version 4.1 or above with concurrent diagnostic installed.
- PSSP 1.2 or above.
- Java 1.1.6 to 1.3 on all monitored machines. Java for AIX 4.3.3 is shipped and installed with base AIX. You can obtain Java from IBM service representatives for releases below 4.3.3.
- A serial port for modem connectivity is required on gateway servers. A TTY (0 - 16) device must be available and configured on the gateway.
- An asynchronous modem with a minimum communications speed of 9600 baud and error correction (in the United States) is required. The modem is required in order to call IBM Service Agent Server (SAS). For security reasons, only outbound calls are required by Service Agent, so the auto answer capability of the modem should be disabled. IBM ships the following

modem types for use with Service Agent on some products: 7852 Model 400, 7857-017 or 7858-336.

- On your gateway server, ensure that the bos.net.ppp fileset is installed and configured to support the Point-to-Point Protocol (PPP). This is only required if a modem is to be used for error reporting to IBM.

Electronic service agent in SP/Cluster environment

Electronic service agent in SP or cluster environment differ from basic RS/6000 and pSeries service agent in the following two ways:

1. In an SP or in SP-attached server environment, the CWS (or SA Gateway) must be set to Machine Type (M/T) 9076.
   - The nodes are added using the Add SP Nodes function
2. In the CES environment, the servers are added using the Add machines function.
   - In the model type enter C80. This information is used to create a proper RETAIN record at IBM. With the C80 information the RETAIN call is forwarded to the correct queue monitored by CES skilled people.

> **Note:** For the performance reason, sometime it is not recommended to install the SA Gateway on CWS.

How to obtain electronic service agent code

The following URL could be used to download the ESA code:

```
ftp://ftp.software.ibm.com/aix/service_agent_code
```

2.1.18 Inventory scout (invscout)

The inventory scout is a tool that contains microcode discovery service and, in SP/CES environment, Vital Product Data (VPD) capture service. These two services are made possible through a new AIX command **invscout**, and a new daemon, invscoutd.

Functions provided by invscout command

The following are the basic functions performed by the **invscout** command:

1. Scouts the user's system for microcode levels and compares it with an IBM database of the latest levels. When used with web-based microcode discovery service, an HTML report is generated that includes links to the latest level of microcode for the system, should it find the currently installed level is at a lower level.

In case of SP and CES systems, customers will not be given a link to the latest microcode. Instead, the user will be directed to Contact CE to obtain the new level of microcode.

In the HTML report for RS/6000 systems, a link will be provided to the README of the latest level of microcode.

2. Gathers Vital Product Data (VPD) from the user's machine and, when used with web-based VPD capture service, uploads it to the IBM MRPD database at the plant. The VPD will be useful in determining the correct components to ship when a Miscellaneous Equipment Specification (MES) upgrade is ordered. At this time, only SP VPD data will be stored in the MRPD database. All other RS/6000 systems are targeted to work with MRPD by the end of 2001.

   Inventory scout runs on AIX V4.1.5 or higher and can be invoked by Java applets or be run from the command line. The last level of inventory scout is 1.2.0.1.

How to obtain the invscout code

The **`invscout`** and invscoutd are included in the 10/00 AIX 4.3.3 update and AIX 5.0 releases. You do not need to be at one of these levels to use this command/daemon and services. These can also be obtained from the following internet service links:

Microcode Discovery Service

`http://techsupport.services.ibm.com/rs6k/mds.html`

VPD Capture Service

`http://techsupport.services.ibm.com/rs6k/vcs.html`

2.1.19 Problem management subsystem

The problem management subsystem (pman) provides an infrastructure for recognizing and acting on problem events in your SP system. This infrastructure is based on an event management application that provides configurable access to event management client and resource monitor function without the necessity of writing programs to use the event management APIs. The problem management subsystem is available on PSSP 2.2 or later version.

The pman subsystem consists of four major components:

1. pmand
2. pmanrmd
3. pmandef
4. sp_configd

In the following sections we will cover pman functions in detail and provide an example to explain how pman works.

2.1.20 pmand

The pmand daemon is a client of event management; it can be configured to register for event management events and perform actions when those events do occur. Event management provides access to events throughout an SP system partition; therefore, pmand can monitor and react to events on the node on which it is running, as well as on all other nodes in the system partition and CWS.

When you install your system, a pmand daemon is automatically configured on each node in a system partition. Additionally, there is a pmand daemon running on the control workstation for each system partition in the SP system. When running on a node, the pmand daemon:

- Monitors events occurring on the node on which the daemon is running.
- Monitors events on all other nodes in the system partition.
- Monitors events not associated with a node, such as frame events, as supplied by event management.

There are no restrictions on what a pmand daemon can monitor:

- Any number of pmand daemons can monitor and act on a single event.
- A single pmand daemon can monitor any number of events locally or remotely.
- A single pmand daemon can monitor the same event multiple times. All actions associated with all event registrations are taken by the daemon when the event occurs.

Each pmand daemon has access to events on the node on which it is running, as well as to the other nodes in the system partition. The access to the events is provided by event management, which can, due to its distributed nature, monitor resource variables throughout the system partition and generate events based on the values of these resource variables. Figure 2-9 on page 56 provides an example of how event management communicates with local and other remote pmand daemons.

When an event, which any of the pmand daemons has subscribed to, occurs (whether that event is local or remote), all the pmand daemons registered for it will perform the actions they are configured for (if any). Because pmand is a daemon, its subscriptions to events are persistent. This means the daemon continues to subscribe to events even after the process or user who created the subscription has gone away. A system administrator, for example, can set up automated operations with unattended monitoring and recovery actions.

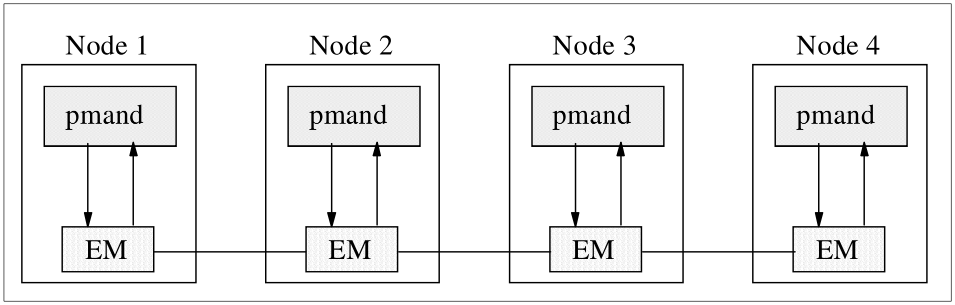


Figure 2-9 Event management communicating with pmand in a four-node system

Controlling pmand

The pmand daemon is under the System Resource Controller (SRC) and can be controlled by the following commands:

- To start pmand on a node:

  ```
  startsrc -s pman
  ```

- To start pmand on a CWS:

  ```
  startsrc -s pman.system_partition_name
  ```

- To stop pmand on a node.

  ```
  stopsrc -s pman
  ```

- To stop pmand running on a control workstation.

  ```
  stopsrc -s pman.system_partition_name
  ```

- To refresh pmand

  ```
  refresh -s pman
  ```

The **refresh** command causes pmand to update its internal configuration from the SDR and start pairing actions with events as specified by the **pmandef** command.

If a pmand refresh occurs, all currently monitored events get unregistered and the configuration information is reread from SDR. The SDR contains persistent information, so that a refresh results only in configuration changes that have been put into the SDR. If you have not deleted or modified a configuration record for a particular event, refreshing the daemon results in reregistering for the same event. For example:

- To refresh pmand on the control workstation:

  ```
  refresh -s pman.system_partition_name
  ```

- To receive status on pmand daemon running on a node:

  ```
  lssrc -ls pman
  ```

- To check the status of pmand running on control workstation:

  ```
  lssrc -ls pman.system_partition_name
  ```

These commands provide the following status information:

- When pmand was started.
- When pmand was last refreshed.
- Whether tracing (debug mode) is on or off. When debug mode is on, all SRC requests and all events are logged to the /var/adm/SPlogs/pman directory.
- Events for which registrations are as yet unacknowledged.
- Events for which actions are currently being taken.
- Events currently ready to be acted on by this daemon.

2.1.21 pmanrmd

The pmanrmd daemon is a resource monitor daemon that provides resource variables to event management services. When you install SP, a pmanrmd daemon is automatically configured on each node in a system partition. Additionally, there is a pmanrmd daemon running on the control workstation for each system partition in the SP system.

The problem management subsystem provides 16 resource variables; IBM.PSSP.pm.User_state1 through IBM.PSSP.pm.User_state16. These are predefined resource variables that have been set aside for system administrators to create their own resource monitors.

A resource monitor that you create through problem management is a command that gets executed repeatedly by the pmanrmd daemon at a specific interval. The standard output from the command is supplied to the event management subsystem as the value for the resource variable. You can then use the pmandef command to subscribe to events for that resource variable.

The resource variable name, resource monitor command, sampling interval, and list of nodes for which the resource monitor is defined, are stored in the SDR. The **pmanrmdloadSDR** command is used to store those definitions in SDR.

You define your resource monitor to the pmanrmd daemon by doing the following:

1. Make a copy of the /spdata/sys1/pman/pmanrmd.conf sample configuration file. Example 2-8 lists the content of system provided sample file.

Example 2-8 /spdata/sys1/pman/pmanrmd.conf sample configuration file

```
* IBM_PROLOG_BEGIN_TAG
* This is an automatically generated prolog.
*
* Licensed Materials - Property of IBM
*
* (C) COPYRIGHT International Business Machines Corp. 1996,2000
* All Rights Reserved
*
* US Government Users Restricted Rights - Use, duplication or
* disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*
* IBM_PROLOG_END_TAG

* "@(#)70   1.8   src/ssp/pman/pmanrmd.conf, probmgmt, ssp_rlyn, rlynt500
10/19/
99 23:02:28"

*  Problem Management resource monitor configuration file.
*
* This will run a user script resource monitor every 1 minutes and
* place the stdout of the monitor into event management as a resource
*
* This will run a user script resource monitor every 1 minutes and
* place the stdout of the monitor into event management as a resource
* variable when it runs. It will run on the CWS only.
*
*TargetType=NODE_LIST
*Target=CWS
*Rvar=IBM.PSSP.pm.User_state1
*SampInt=60
*Command=/u/joe/checker
*
* This will provide a resource monitor that will run every 10 minutes
* and will provide the most recently changed file in the /etc directory.
* It runs on all the nodes that belong to the SERVERS node group.
*
*TargetType=NODE_GROUP
*Target=SERVERS
```

```
*Rvar=IBM.PSSP.pm.User_state2
*SampInt=600
*Command="/bin/ls -lt /etc | /bin/head -2 | /bin/grep -v total"
```

2. Edit your copy of the configuration file. Provide the following:
   - The name of the resource variable (for example, IBM.PSSP.pm.User_state1)
   - The resource monitor command
   - A sampling interval (in seconds)
   - The nodes number on which to run the resource monitor command

> **Attention:** When typing commands in the pmanrmd.conf file, be aware that the command string of the pmrmCommand line will get enclosed in single-quotes and the Korn Shell rules apply to it. Keep in mind that a single quote cannot occur within single quotes. If a command contains single quotes, each must be replaced by the four characters '\'' (single quote, backslash, single quote, single quote). For example: Command="mount | awk '\''{print $3}'\'' | tail -l"
>
> The marks between awk and tail are single quotes, the others are double quotes.

3. Loading the configuration information into the SDR by using the **pmanrmdloadSDR** command.
4. Stopping and restarting the pmanrmd daemon on the nodes that are affected by this change:

   On the control workstation:

   ```
   stopsrc -s pmanrm.syspar_name
   startsrc -s pmanrm.syspar_name
   ```

   where syspar_name is the name of system partition.

   On the node:

   ```
   stopsrc -s pmanrm
   startsrc -s pmanrm
   ```

For further details on Problem Management resource monitors, refer to chapter 27 “Using the Problem Management Subsystem” in *Parallel System Support Programs for AIX, Administration Guide*, SA22-7348.

2.1.22 pmandef – creating problem management subscriptions

The **pmandef** command provides a mechanism for creating problem management subscriptions for pmand daemon to event management services. The **pmandef** command provides:

- Registration for event management events (full file systems, disk drive failures, etc.).
- Actions to take when management events are triggered. For example:
  - Running a command
  - Issuing an SNMP trap
  - Write to the AIX error log and the BSD syslog facilities

The **pmandef** command also provides the following functions:

- Activating a problem management subscription
- Deactivating a problem management subscription
- Querying a problem management subscription
- Removing a problem management subscription

Users interface with PMAN through the **pmandef** command, which is restricted to authorized users only. A user needs to have a Kerberos principal, and this principal needs to be listed in the /etc/sysctl.pman.acl file. Example 2-9 provides a listing of this file.

To learn more about the **pmandef** command, refer to *PSSP: Command and Technical Reference, Volume 1*, SA22-7351.

Example 2-9 Sample /etc/sysctl.pman.acl file

```
#acl#

# These are the kerberos principals for the users that can configure
# Problem Management on this node.  They must be of the form as indicated
# in the commented out records below.  The pound sign (#) is the comment
# character, and the underscore (_) is part of the "_PRINCIPAL" keyword,
# so do not delete the underscore.

_PRINCIPAL root.admin@MSC.ITSO.IBM.COM
#_PRINCIPAL root.admin@PPD.POK.IBM.COM
#_PRINCIPAL joeuser@PPD.POK.IBM.COM
```

In this case, the Kerberos principal, root.admin@ MSC.ITSO.IBM.COM realm is authorized to use the PMAN subsystem on this node. Make sure you have authorization on every other node where you want to use this facility.

Important: Make sure you stop and start the Sysctl SRC subsystem every time you modify this file.

For more information on **pmandef** command, refer to *Parallel System Support Program for AIX: Command and Technical Reference, Volume 1*, SA22-7351.

2.1.23 Using pmandef command – an example

You can use **pmandef** to specify a command to run when a specified event or re-arm event occurs. In Example 2-10, **pmandef**, running on node 5, causes the command **echo program has stopped >/tmp/myevent.out** to execute on node 5 whenever the number of processes named **mycmd**, owned by user bob, on node 12 becomes 0 (the event). When this number increases back to 1 (the re-arm event), the command **echo program has restarted >/tmp/myrearm.out** runs on node 5.

Example 2-10 pmandef -– output received on the node issuing the command

```
[root@sp4n05]:/> pmandef -s Program_Monitor \
> -e 'IBM.PSSP.Prog.pcount:NodeNum=12;ProgName=mycmd;UserName=bob:X@0==0'\
> -r





-c "echo program has stopped >/tmp/myevent.out" \
> -C "echo program has restarted >/tmp/myrearm.out"
```

You can specify to run commands on a node other than the one from which the **pmandef** was issued. Example 2-11 on page 62 causes the command to run on nodes 1, 2, 3 and 7, whenever bob's program dies or gets restarted on any of nodes 1, 2, 3, 4, 5 or 13. If bob's program dies on node 4, then the command **/usr/local/bin/start_recovery** runs on nodes 1, 2, 3 and 7. Any number of commands can run simultaneously

Example 2-11 Output received on nodes other than the node issuing the command

```
[root@sp4n05]:/> pmandef -s example \
-e 'IBM.PSSP.Prog.pcount:NodeNum=1-5,13;ProgName=mycmd;UserName=bob:X@0==0'\
-r "X@0>0" -c /usr/local/bin/start_recovery \
-C /usr/local/bin/stop_recovery -n 1-3,7
```

You can specify a timeout, in seconds, for each command. The minimum timeout that can be specified is 10 seconds. If the command has not exited before the specified timeout, the command gets killed.

The environment variables used by the pmandef command

The problem management subsystem makes all of the contents of an event management notification available in the command's environment when the command is run. Example 2-12 on page 62 provides a list of the variables obtained from the event management notification environment.

Example 2-12 pmandef command environmental variable

```
PMAN_HANDLE
   The name that identifies this subscription to the Problem Management
   subsystem. This name was given as the argument of the -s flag to pmandef.
PMAN_PRINCIPAL
   The name of the Kerberos V4 principal that owns this subscription, if one
   exists.
PMAN_DCEPRIN
   The name of the DCE principal that owns this subscription, if one exists.
PMAN_RVNAME
   The Event Management resource variable.
PMAN_IVECTOR
   The Event Management resource identifier.
PMAN_PRED
   Either the Event Management expression or rearm expression, depending on
   whether this is an event or rearm event.
PMAN_TIME
   The time that the event was reported to the Problem Management subsystem.
PMAN_LOCATION
   The node number of the node on which the event was generated, usually (but
   not always) the node on which the event occurred.

PMAN_RVTYPE
   One of long, float or sbs, depending on whether the type of the resource
   variable value is a long integer, a floating point value or a Structured
   Byte String. Note: If the PMAN_RVTYPE is either long or float, then the
   resource variable value is stored in PMAN_RVVALUE, and PMAN_RVVALUE is to
   be interpreted as type PMAN_RVTYPE.
```

```
If PMAN_RVTYPE is sbs, then the resource variable value is composed of one
or more structure elements. There is no PMAN_RVVALUE environment variable.
Instead there is a separate environment variable for each element, and the
PMAN_RVCOUNT environment variable defines the number of elements. For
example, if there are 3 structure elements within the Structured Byte
String, the PMAN_RVCOUNT will be 3, and there will be 3 separate
environment variables for the 3 structure elements: PMAN_RVFIELD0,
PMAN_RVFIELD1 and PMAN_RVFIELD2. Each of these 3 environment variables
contains a name=value pair, where name is the structure element name, and
value is the structure element value.
```

In Example 2-13 on page 64, the **pmandef** command requests the /usr/local/bin/recovery_cmd to run on node 12 when the number of processes named mycmd, owned by root, on nodes 9, 10, or 11 becomes zero. If mycmd program terminates on node 10, the command /usr/local/bin/recovery_cmd runs on node 12, and the following environment variable values are included in its environment:

PMAN_HANDLE (Program_Monitor)

PMAN_PRINCIPAL (root.admin@MSC.ITSO.IBM.COM)

PMAN_DCEPRIN (/.../test_dcecell/cell_admin)

PMAN_RVNAME (IBM.PSSP.Prog.pcount)

PMAN_IVECTOR (ProgName=mycmd;UserName=bob;NodeNum=10)

PMAN_PRED (X@0==0)

PMAN_TIME (Thu Aug 22 00:42:08 1996)

PMAN_LOCATION (10)

PMAN_RVTYPE (sbs)

PMAN_RVCOUNT (3)

PMAN_RVFIELD0 (CurPIDCount=0)

PMAN_RVFIELD1 (PrevPIDCount=1)

PMAN_RVFIELD2 (CurPIDList=)

This information could be used by any command. Two utilities that report this information are provided as part of the Problem Management subsystem:

- **notify_event**
- **log_event**

These commands are provided to get you started. You may want to write more sophisticated commands.

Example 2-13 pmandef - environment variables example

```
pmandef -s example \
-e 'IBM.PSSP.Prog.pcount:NodeNum=9-11;ProgName=mycmd;UserName=root:X@0==0' \
-c "/usr/local/bin/recovery_cmd" -n 12
```

Obtaining problem management access

The **pmandef** command is built upon the Sysctl facility, which uses the SP security services to provide authorized users, which may include both root and non-root users with the ability to create, modify, and delete problem management subscriptions.

How a user is authorized to access problem management depends on which SP trusted services authentication methods have been enabled.

When Kerberos V4 or compatibility is the only authentication method enabled, access to problem management is protected by the /etc/sysctl.pman.acl file by adding their Kerberos V4 principals to the /etc/sysctl.pman.acl file.

When no authentication method is enabled, access to problem management is protected by the /etc/sysctl.pman.acl text file. You can choose to authorize all or none un-authenticated users to access the problem management subsystem. You cannot authorize individual unauthenticated users to access the PMAN.

For further details refer to:

– Chapter 27, “Problem Management Access,” in *Parallel System Support Programs for AIX, Administration Guide*, SA22-7348.

– Chapter 23, “Controlling Remote Execution by Using sysctl” in *Parallel System Support Programs for AIX, Administration Guide*, SA22-7348*.*

Authorizing event response actions

After the pmand daemon receives notification that an event has occurred, and before it performs the action for that event, the **pmand** daemon checks to see whether the subscription owner is authorized to perform the requested action on the node where it is running. If the requested action is the execution of a command, the subscription owner must have AIX remote commands access to the node as the target user. The target user is, by default, the same user who has issued the **pmandef -s** command to create the subscription. A different user can be specified to the **pmandef** command by using the -U flag.

The underlying principal is that the pmand daemon will execute a command in response to an event only if the subscription owner has the ability to execute the same command by other means. If the user can log in to the node as the target user and execute the command directly from the command line, or at least run the command as the target user by invoking the **rsh** command from a remote node, then no extra privileges can be gained from using problem management. The only thing the end user gains is the automation of responses to events within the SP system.

How event response authorization takes place is dependant on the type of the subscription's ownership and the AIX remote command authentication methods enabled by the system administrator.

The steps that pmand uses to determine whether the subscription owner is authorized to run a command as the requested target user on the local node are listed below in the order they are checked.

- If the subscription contains a DCE principal, and if Kerberos V5 is enabled as an AIX remote command authentication method, then the subscription owner can be authorized if the DCE principal's underlying Kerberos V5 principal has been listed in the target user's $HOME/.k5login file.
- If the subscription contains a Kerberos V4 principal, and if Kerberos V4 is enabled as an AIX remote command authentication method, then the subscription owner can be authorized if the Kerberos V4 principal has been listed in the target user's $HOME/.klogin file.
- If no SP trusted services authentication methods are enabled, if the subscription contains both source AIX user name (the user who issued the **pmandef -s** command to create the subscription) and source host name (the node where the **pmandef -s** command was issued), and if the standard UNIX is enabled as an AIX remote command authentication method, then the subscription owner can be authorized if the source AIX user name and source host name combination have been listed in the target user's $HOME/.rhosts file. If this step also fails, the subscription owner is not authorized, so the event response is not executed.

When the action to be performed is an entry in the AIX error log and BSD syslog or the generation of an SNMP trap, all of the preceding rules apply, except the target user is always root, regardless of which target user is contained in the subscription. This restricts these actions to system administrators.

Authorization checking for AIX error log and BSD syslog actions and SNMP trap actions are done separately from authorization checking for command execution actions. Therefore, if a subscription requests that a command executes, for instance, as *joeuser* and an SNMP trap gets generated in response to an event, the command authorization check will use joeuser as the target user, while the SNMP trap authorization check will use root as the target user.

For further details read the sections:

- Chapter 27, “Problem Management Access,” in *Parallel System Support Programs for AIX, Administration Guide*, SA22-7348.
- Chapter 23, “Controlling Remote Execution by Using sysctl” in *Parallel System Support Programs for AIX, Administration Guide*, SA22-7348*.*

Querying the problem management subscription information

Use the **pmanquery** command to query the SDR for a description of a problem management subscription. The **pmanquery** command provides detail of the subscription information in raw format, which can then be used by other applications. The following example queries all subscriptions:

```
[root@sp4en0]:/> pmanquery -n all -k all -p all -U all -H all
```

For more information on **pmanquery**, refer to *Parallel System Support Programs for AIX, Command and Technical Reference, Volume 1*, SA22-7351.

Monitoring default events

The problem management subsystem provides a set of default events to be monitored in the /usr/lpp/ssp/install/bin/pmandefaults script. This script contains a series of **pmandef** commands that request an event notification to be mailed to root on control workstation when the specified event occurs. These are events that would interest many system administrators. Events are defined for all nodes in the current system partition and all events are monitored from the control workstation. The list of events includes:

- The /var file system is more than 95 percent full.
- The /tmp file system is more than 90 percent full.
- An error log record of type PERM has been written to the AIX Error Log.
- The inetd daemon has terminated.
- The sdrd daemon has terminated (control workstation only).
- The sysctld daemon has terminated.
- The hrd daemon has terminated (control workstation only).
- The fsd daemon has terminated (nodes only).

The /usr/lpp/ssp/install/bin/pmandefaults script is a suggested starting point for configuring the problem management subsystem on your SP system. You can choose to run the script as is, or you can make your own copy of the script and modify it to suit your needs, or you can choose not to run the script at all.

Logging an event using the pmandef command

You can specify that pmand write event notification information, along with some optional specified text, to the AIX error log and BSD syslog facilities. In Example 2-14 whenever the file system associated with the mylv logical volume and the myvg volume group on node 11 becomes more than 95% full, the text “file system is almost full” gets written to the AIX error log and BSD syslog facilities on node 11 (via the -h local option).

Example 2-14   pmandef - logging an event with user supplied text

```
pmandef -s Filesystem_Monitor \
-e 'IBM.PSSP.aixos.FS.%totused:NodeNum=11;VG=myvg;mylv:X>95' \
-l "filesystem is almost full" -h local
```

2.1.24 sp_configd (SNMP proxy agent)

With this daemon, PMAN can send simple network management protocol (SNMP) traps to SNMP managers to report predefined conditions. The ability to issue an SNMP trap in response to an event allows you to report problem events occurring in your SP system to a network manager, such as Tivoli Netview, existing on a remote node (a network manager application is not supplied with the SP).

The problem management subsystem provides an SP SNMP proxy agent, sp_configd, (sometimes referred to as subagent daemon) that runs on the control workstation and every SP processor node. The SP proxy agent provides the following functions:

- A Management Information Base (MIB)
- The SNMP GET (Requests the SNMP agent to retrieve the value of the specified variable and return it to the manager) and GET NEXT commands that allow data in the MIB to be accessed by a network manager application
- Creation and transmittal of SNMP traps to a network manager application when any of the following events occur on the node on which the SP proxy agent is running:
  - A cold start trap is issued when the agent is activated
  - An enterprise-specific trap is issued when an entry with an "alert=true" attribute is written in the AIX error log

   – An enterprise-specific trap is issued when user-specified event is detected within event management services

Example of an SNMP trap issued for an event management event

Below is an example on how an SNMP trap could be collected and send to the SNMP manager, such as Tivoli Netview.

1. Use the **pmandef** command to subscribe to an event management event and specify that an SNMP trap be issued for that event.

In Example 2-15, whenever the filesystem associated with the mylv logical volume and the myvg volume group on node 10 gets more than 95% full, an SNMP trap will be generated on the control workstation.

Example 2-15 Generating SNMP trap

```
pmandef -s Filesystem_Monitor /
-e'IBM.PSSP.aixos.FS%totused:NodeNum=10;VG=myvg;LV=mylv:X>95' /
-t 1234 -n 0
```

2. The pmand subscribes to the event as specified in the pmandef command. The event must occur on the local node on which pmand is running.
3. The pmand writes the contents of the event manager subsystem-supplied event response and the user-specified event configuration information into a FIFO file
4. The sp_configd reads the data and creates an SNMP trap from it
5. The sp_configd sends the trap to the SNMP managers specified in the /etc/snmpd.conf file on the node.
6. You specify a particular trap ID in the configuration information for an event manager event.

2.1.25 Monitoring an error condition using PMAN – test case

Now you know that the PMAN subsystem provides 16 resource variables for user-defined events. In this section, we use one of these variables to monitor a specific condition where PSSP does not provide a resource variable.

In this example, you will setup PMAN to get a notification on the console screen each time there is an authentication failure for remote execution. The remote shell daemon (rshd) logs these errors to the /var/adm/SPlogs/SPdaemon.log. A monitor could be created to search for the specific error created in this log.

First, identify the error that gets logged into this file every time somebody tries to execute a remote shell command without the proper credentials. Destroy the Kerberos ticket using **kdestroy** and execute, for example, the **rsh** command. Observe the error log for error entry, as below:

```
Feb 27 14:30:16 sp3n01 rshd[17144]: Failed krb5_compat_recvauth
Feb 27 14:30:16 sp3n01 rshd[17144]: Authentication failed from
sp3en0.msc.itso.ibm.com: A connection is ended by software.
```

From this, we see that `Authentication failed` seems to be a nice string to look for. The idea here is to notify the operator (console) that there was a failed attempt to access this machine through the remote shell daemon.

However, there is a small problem to solve. If we are going to check this log file every few minutes, how do we know whether the log entry is new or was already reported? Fortunately, the way user-defined resource variables work is based on strings. The standard output of the script you associate with a user-defined resource variable is stored as the value of that variable. This means that if we print out the last `Authentication failed` entry every time, the variable value will change only when there is a new entry in the log file.

Let’s create the definition for a user-defined variable. To do this, PMAN needs a configuration file that has to be loaded to the SDR by using the **pmanrmdloadSDR** command.

PSSP provides a template for this configuration file. It is located in the /spdata/sys1/pman directory on the CWS. Let’s make a copy of this file and edit it:

```
TargetType=NODE_RANGE
Target=0-5
Rvar=IBM.PSSP.pm.User_state1
SampInt=60
Command=/usr/local/bin/Guard.pl
```

In this file, you can define all 16 user-defined variables (there must be one stanza per variable). In this case, we have defined the IBM.PSSP.pm.User_state1 resource variable. The resource monitor (pmanrmd) will update this variable every 60 seconds as specified in the sample interval (SampInt). The value of the variable will correspond to the standard output of the /usr/local/bin/Guard.pl script. Example 2-16 describes this script.

Example 2-16 /usr/local/bin/Guard.pl script

```
#!/usr/lpp/ssp/perl5/bin/perl

my $logfile="/var/adm/SPlogs/SPdaemon.log";
my $lastentry;
```

```
open (LOG,"cat $logfile|") ||
        die "Ops! Can't open $logfile: $!\n";

while (<LOG>) {
   if(/Authentication failed/) {
        $lastentry = $_;
        }
   }

print "$lastentry";
```

The script prints out the `Authentication failed` entry from the log file. If there is no new entry, the old value is the same as the new value, so we have to create a monitor that gets notified every time the value of this variable changes. Let's take a look at the monitor's definition:

```
[sp5en0:/]# /usr/lpp/ssp/bin/pmandef -s authfailed \
-e 'IBM.PSSP.pm.User_state1:NodeNum=0-5:X@0!=X@P0' \
-c "/usr/local/bin/SaySomething.pl" \
-n 0
```

This command defines a monitor, through PMAN, for the IBM.PSSP.pm.User_state1 resource variable. The expression X@0!=X@P0 means that if the previous value (X@P0) is different from the current value (X@0), then the variable has changed. This special syntax is due to the fact that these user-defined variables are structured byte strings (SBSs), so to access the value of this variable you have to index this structure. However, these variables have only one field, so only index 0 is valid.

You can get a complete definition of this resource variable (and others) by executing the following command:

```
[sp5en0:/]/usr/sbin/rsct/bin/haemqvar "" IBM.PSSP.pm.User_state1 "*"|more
```

This gives you a good explanation along with examples of how to use it.

Now that we have subscribed our monitor, let's see what the /usr/local/bin/SaySomething.pl listed in Example 2-17, does:

Example 2-17 /usr/local/bin/SaySomething.pl

```
#!/usr/lpp/ssp/perl5/bin/perl

$cwsdisplay = "sp4en0:0";
$term="/usr/dt/bin/aixterm";
$cmd = "/usr/local/bin/SayItLoud.pl";
$title = qq/\"Warning on node $ENV{'PMAN_LOCATION'}\"/;
```

```
$msg = $ENV{'PMAN_RVFIELD0'};
$bg = "red";
$fg = "white";
$geo = "60x5+200+100";

$execute = qq/$term -display $cwsdisplay -T $title -geometry $geo -bg $bg -fg
$fg -e $cmd $msg/;

system($execute);
```

This script opens a warning window with a red background notifying the operator (on node 0, the CWS) about the intruder.

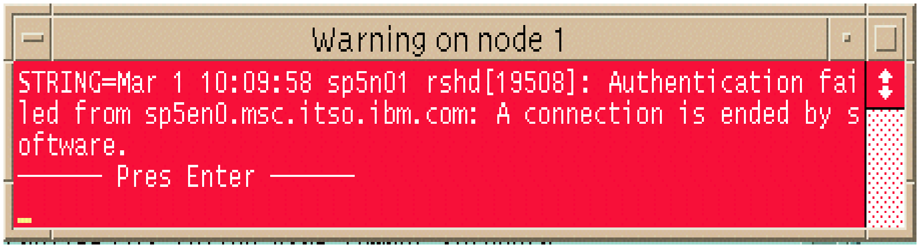
The script /usr/local/bin/SayItLoud.pl displays the error log entry (the resource variable value) inside the warning window. Let's take a look at this script:

```
#!/usr/lpp/ssp/perl5/bin/perl

print "@ARGV\n";
print "------ Press Enter ------\n";
<STDIN>
```

Now that the monitor is active, let's try to access one of the nodes. We destroy our credentials (**kdestroy** command) and then try to execute a command on one of the nodes:

```
[sp5en0:/]# kdestroy
[sp5en0:/]# dsh -w sp5n01 date
sp5n01: spk4rsh: 0041-003 No tickets file found. You need to run "k4init".
sp5n01: rshd: 0826-813 Permission is denied.
dsh:  5025-509 sp5n01 rsh had exit code 1
```

After a few seconds (a minute at most), we receive the warning window, shown in the warning message at the CWS illustrated in Figure 2-10.



```
STRING=Mar 1 10:09:58 sp5n01 rshd[19508]: Authentication fai
led from sp5en0.msc.itso.ibm.com: A connection is ended by s
oftware.
------ Pres Enter ------
```

Figure 2-10   User-defined resource variables – Warning window example

The example shown here is very simple. It is not intended to be complete, but instead, to illustrate the use of these user-defined resource variables.

2.1.26 SP perspectives

The SP perspectives provide access to a set of applications, each with a graphical user interface (GUI). The perspectives applications enable you to perform monitoring and system management tasks for your SP system by directly manipulating icons that represent system objects.

The perspectives launch pad is started using the `perspectives` command, which resides in the /usr/lpp/ssp/bin directory. Other perspectives applications, such as the event perspective or hardware perspective, can be started from the launch pad. If you are launching perspectives remotely, be sure that the DISPLAY environment variable is set to the machine that you want to display the SP Perspective. Also, be sure that you are permitted to display to that machine by running the `xhost` command.

Following two perspectives tools are useful for monitoring the system status and detecting problem situations:

- Event perspective
- Hardware perspective

Each perspectives application provides its own unique capabilities. For the purposes of problem monitoring and determination, we recommend that the SP event perspective be used to monitor conditions of interest for the SP system. When SP event perspective indicates that a hardware failure condition exists, the SP hardware perspective should be used to examine the current status of the system hardware and obtain more detailed information about the hardware problem.

2.1.27 Event perspective

Event perspective provides a graphical interface to event management and the problem management subsystems. Through this interface, you can create monitors for triggering events based on defined conditions and generate actions by using the problem management subsystem when any of these events are triggered.

Using the SP event perspective, you can create event definitions that tell the system to let you know automatically when changes that are important to you have occurred in your SP system; when a node goes down or becomes unreachable, when system is close to running out of paging space, when there's something wrong with the switch – these are the kinds of things you want to know about, and these are the kinds of things that you can have the system automatically tell you about when you use the SP event perspective.

When you create an event definition, you specify a condition that defines the state of a system resource and then you identify the specific resources that you want the system to monitor. You can also specify that you want the system to respond to the event by displaying a notification on your screen and/or take an action to in response to that event. Finally, you activate the event definition by registering it.

To effectively use this perspective, you must understand the following terminology:

Condition

Conditions are the circumstances within the system that are of interest to the system administrator. Conditions can be created, viewed, and modified through the conditions pane in the SP Event Perspective. To specify a condition, the system administrator must provide the necessary components to form the condition, including an event expression and, optionally, a re-arm expression.

A set of predefined event definitions can be loaded to let you quickly and easily set up monitoring for events that are commonly of interest. Some of these predefined events are; the status of nodes and frames, the usage of paging space and file systems, etc. You may use the predefined event definitions and conditions as is, or use them as templates for creating new event definitions and conditions. You can also create a new condition from a large list of resource variables included with PSSP.

List of predefined event definitions

The following is a list of predefined event definitions:

- errLog nodePowerLED
- fileSystems
- frameControllerNotResponding
- framePowerOff
- hostResponds
- keyNotNormal
- LCDhasMessage

- nodeEnvProblem
- nodeNotReachable
- nodePowerDown
- nodeSerialLinkOpen
- pageSpaceLow
- sdrDown
- switchNotReachable
- switchPowerLED
- switchResponds
- tmpFull
- varFull

Re-arm expressions

The re-arm expression indicates when the SP Event Perspective should consider the event to have stopped. For example, a file system is considered almost full when space utilization reaches 90% of the capacity. The system administrator may want to consider the condition to exist until the space utilization drops to 87%. The event expression would then be set to 90% and the re-arm expression to 87%. As with the event expression, the system administrator can indicate an action to take when the re-arm expression occurs, such as deactivating reserve resources that had been activated when the event occurred.

Event expression

A relational expression that specifies the circumstances under which an event is generated.

Event definition

An association made by the system administrator between a condition and a response to the presence of that condition.

Registration

The activation of an event definition. By registering an event definition, the system administrator instructs the perspectives to begin monitoring for the condition and to take the associated action if the condition should occur.

Once the user registers the event definition, the action will be run whenever the event or re-arm expression occurs. This is independent of whether the event perspective is active at the time of event or re-arm expression occurs.

Event

A change in the state of a system resource. For the purposes of this discussion, an event is more narrowly defined as the presence of the condition within the system.

2.1.28 Using the event perspective – a case study

The following case study provides processes to analyze the problem and procedures to correct it.

Case:

You are required to monitor free space of /tmp file system on all the nodes. A notification is required when file system utilization reaches over 90%; the condition will remain true until file system utilization goes below 70%, at which point file system utilization is considered to be normal and the condition will be reset. However, at any point in time, if the utilization again crosses the 90% mark, a new notification is required.

Solution

Lets begin by defining the condition that could be monitored by the event manager. You cannot create a monitor before defining the condition. Follow the steps.

1. Defining the condition

Since we need to monitor the file system utilization, let's first identify the resource variable representing the file system's full condition. If one is not already available, you will be required to create your own. Luckily, PSSP provides several commonly used pre-defined variables, so lets check them out first.

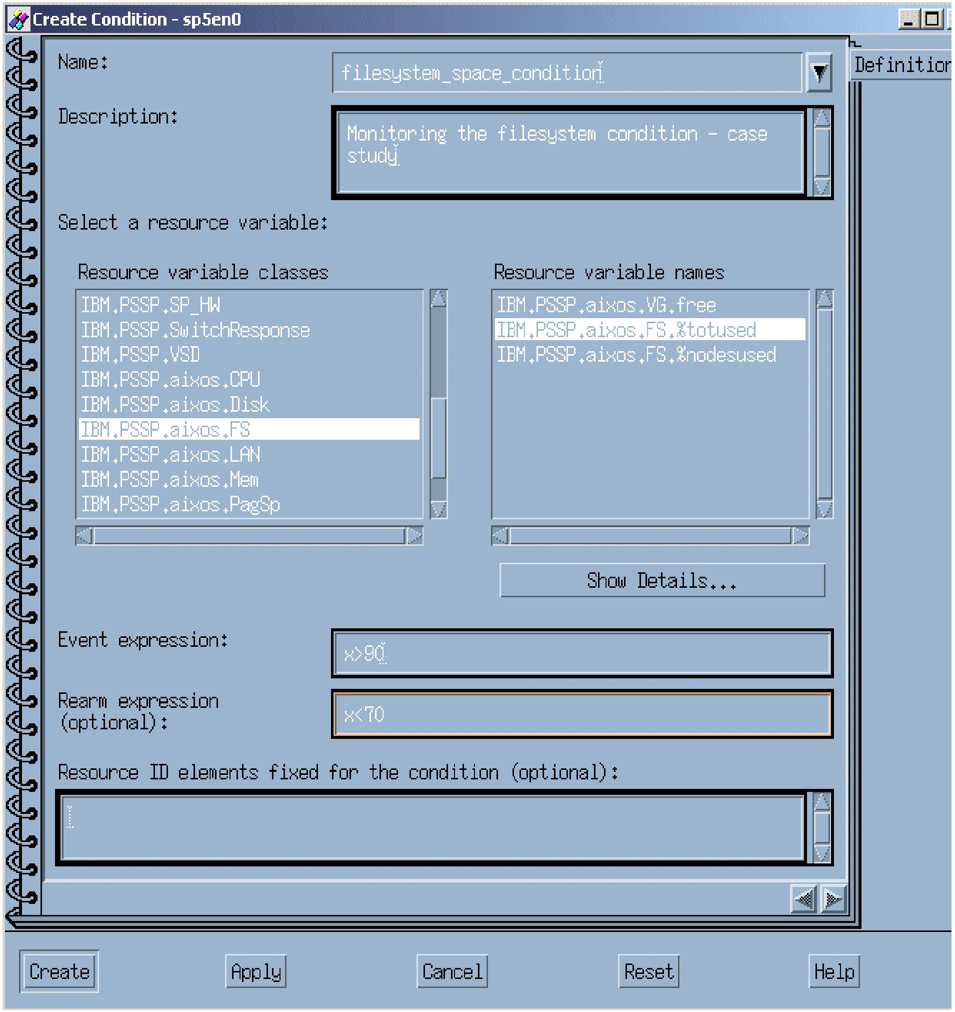
You can either use the **haemqvar** command to find out about those pre-defined variables, or use the perspectives panels (select show details in creating a condition pane). The variable format is IBM.PSSP.Membership.Node.State.To list all variables related to a file system (IBM.PSSP.aixos.FS class), run the following command:

```
[sp3en0:/]# haemqvar -d IBM.PSSP.aixos.FS "" "*"
```

The above command, among others, will list IBM.PSSP.aixos.FS.%totused variable with a description of Used space in percent. Further information can be obtained by running the **haemqvar** command on this variable, as follows.

```
sp3en0:/]# haemqvar  "IBM.PSSP.aixos.FS" IBM.PSSP.aixos.FS.%totused "*"
```

Now we know that IBM.PSSP.aixos.FS.%totused is the variable we want. Run the perspectives command and make condition pane active (by clicking once in the pane). Now choose **Actions -> Create** to get to create condition pane as shown in Figure 2-11.


Create Condition - sp5en0

Name: filesystem_space_condition

Description: Monitoring the filesystem condition - case study

Select a resource variable:

Resource variable classes

IBM.PSSP.SP_HW
IBM.PSSP.SwitchResponse
IBM.PSSP.VSD
IBM.PSSP.aixos.CPU
IBM.PSSP.aixos.Disk
IBM.PSSP.aixos.FS
IBM.PSSP.aixos.LAN
IBM.PSSP.aixos.Mem
IBM.PSSP.aixos.PagSp

Resource variable names

IBM.PSSP.aixos.VG.free
IBM.PSSP.aixos.FS.%totused
IBM.PSSP.aixos.FS.%nodesused

Event expression: x>90

Rearm expression (optional): x<70

Resource ID elements fixed for the condition (optional):

Figure 2-11 Defining a condition pane

The following data has been entered in this pane:

Name

Use any arbitrary (but meaningful) string to name your condition. filesystem_space_condition.

Description

This Provides a description to tell more about the condition. It serves a good documentation purpose in the long run. Monitoring the file system full condition casestudy is the description used in this example.

Select a resource variable – resource variable classes

This lists all the pre-defined resource variable provided with PSSP. Select *IBM.aixos.FS* resource variable for this example.

Select a resource variable – resource variable names

Within a resource class, select the resource variable of your choice. Choose IBM.PSSP.aixos.FS.%totused in this example.

Show details

If chosen, this will provide more details about IBM.PSSP.aixos.FS.%totused resource variable. The information here is equivalent to the **`haemqvar "IBM.PSSP.aixos.FS" IBM.PSSP.aixos.FS.%totused "*"`** output. Since you already have executed this command and are aware of the resource variable, you do not need this option in this example.

Event expression

Set event expression to >90.

Re-arm expression

Set re-arm expression to <70.

You will now be able to see your defined condition in the condition pane, among several others conditions.

2. Define an event definition

Now that the condition has been defined, you need to create a monitor for this condition. Make event definition as your active pane and select **Actions -> Create** to display pane as shown in Figure 2-12 on page 79.

Provide the data to the relevant fields as follows:

Event definition name

Provide any arbitrary name. Use filesystem_space_monitor for this example.

Condition - name

From the scroll bar, select your previously defined condition, filesystem_space_monitor.

Resource ID elements and their values

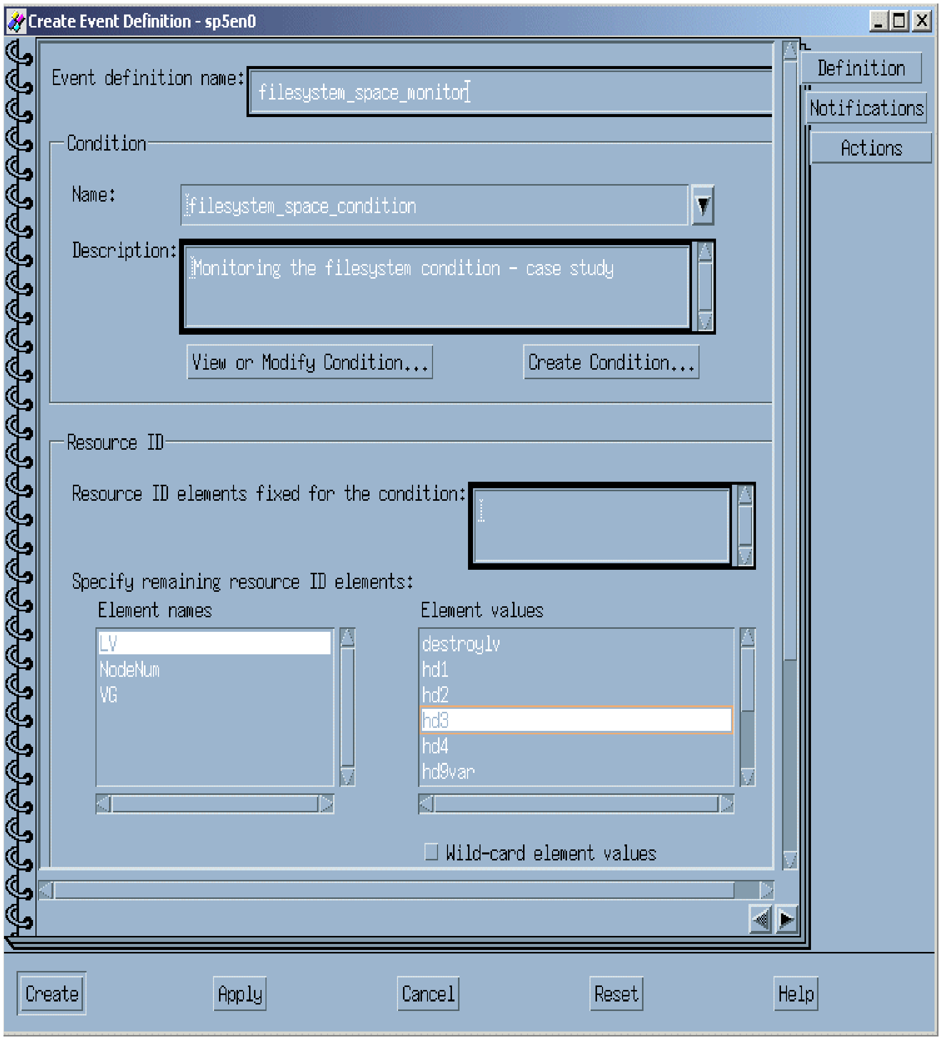
The following list describes the resource ID elements and their values:

- LV
  - Element LV can take possible values of all the logical volumes known to the system. Since we need to monitor /tmp our choice is hd3.

> **Tip:** Use the **`df`** command to find out relationship between the file system and the logical volume.

- NodeNum
  - Determine the nodes where the conditions will be tested. Scroll bar menu will list all the node number for the selection. Click on the wild-card element values to select all the nodes.
- VG
  - This element take values of all VGs defined to the system. Select rootvg for this example.

.



Create Event Definition - sp5en0

Event definition name: filesystem_space_monitor

Condition

Name: filesystem_space_condition

Description: Monitoring the filesystem condition - case study

Resource ID

Resource ID elements fixed for the condition:

Specify remaining resource ID elements:

Element names	Element values
LV	destroylv
NodeNum	hd1
VG	hd2
	hd3
	hd4
	hd9var

□ Wild-card element values

Figure 2-12 Create Event Definition - Pane

Once all the information has been filled out, you can create the monitor by clicking the Create button. For this example, we have not selected any special action, so only a window notification will be displayed when the condition evaluates true. However, you may add actions to your monitor by selecting the Actions tab on the right-hand side of the monitors pane.

The new monitor has been defined and is displayed in the events pane.

3. Testing the defined event

To test the new monitor, we fill up /tmp in one of the nodes by using the **dd** command as follows:

```
dd if=/dev/zero of=/tmp/file
```

Once the file system is over 90% full, we will receive a notification at the Control Workstation's display.

The re-arm events and notifications work in a similar way. To trigger the re-arm event (X < 70), we remove the file from the /tmp file system. The re-arm event notification and the notification windows are very similar to the ones we just saw.

For a detailed discussion of event perspective and monitors, refer to *SP Perspectives: A New View of Your SP System*, SG24-5180.

2.1.29 Hardware perspective

The RS/6000 SP hardware Perspective provides a view of your entire RS/6000 SP system at one place. You can modify your view of system elements (by adding panes) and perform actions on the system via the graphical toolbar and window menus.

The hardware perspective tells you about the current status of various types of SP hardware objects. To launch, click on the hardware perspective icon from the main perspective pane. Figure 2-13 on page 81 provides a view of hardware perspectives that has switch and frame panes added to the default view.

Unlike the SP event perspective, the SP hardware perspective does not permit the user to associate an action with the presence of a condition. Users that wish to automate a response to a specific system condition should use the SP event perspective. Also, the SP hardware perspective only monitors conditions while it is active. If the perspective is shut down, any monitoring of a hardware status is also shut down.

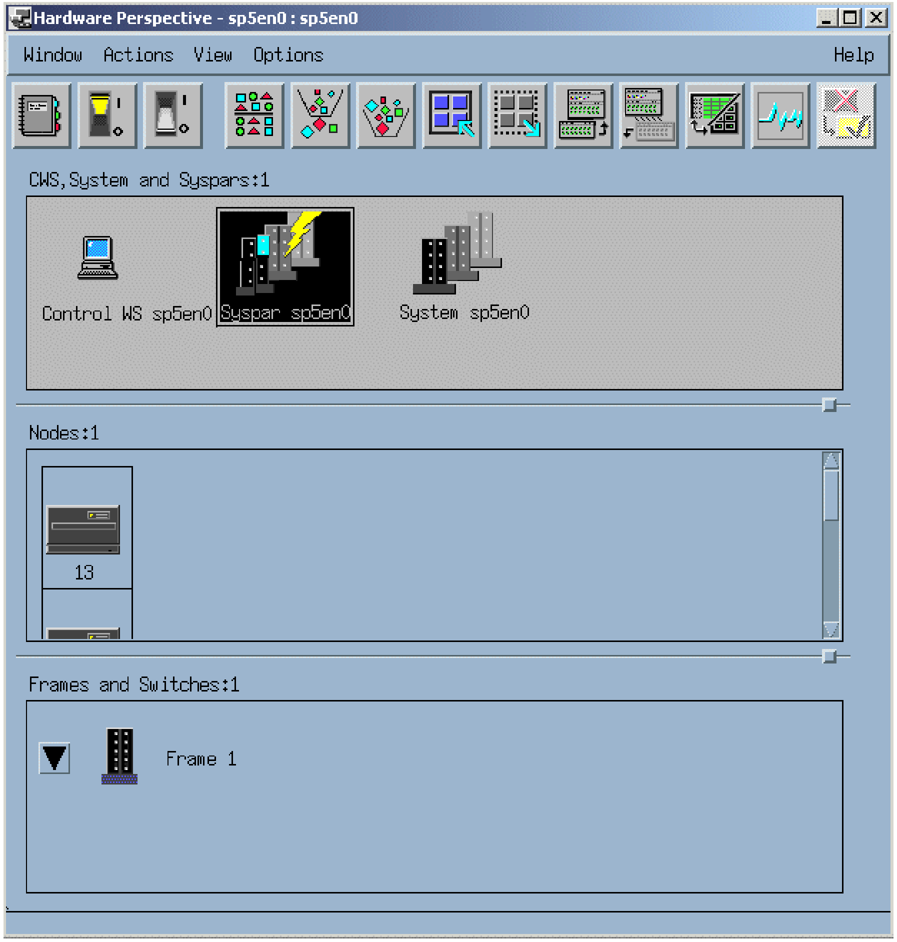


Figure 2-13 Hardware perspective pane

Examining the hardware objects

To examine the current status of a hardware device, select the hardware device by single-clicking on the device's icon in the particular pane.

Open the device's notebook by single clicking the notebook icon (leftmost icon) in the toolbar at the top of the perspective window. This displays a new window (notebook) that contains the device's current status, settings, and monitored conditions. This is useful for examining a node's LED values, its responsiveness to the network and the switch, its network configuration, and other information.

For further assistance in using the notebook to view hardware status, consult the perspective's online help. To access this help, click on the Help button from the SP hardware perspective display and select the Tasks. Assistance in viewing hardware status is available in the viewing hardware attributes topic.

Opening a notebook for multiple entities (such as a series of nodes) can be time-consuming. The SP hardware perspective offers an alternative method for displaying this information in table form. To switch from the icon to table view in a pane, first select your hardware object (by clicking once on the icon) than click on the icon on the right of the toolbar, which shows a table and an icon.

Hardware objects managed by hardware perspective

The following objects can be managed by the SP hardware perspective.

- System partition, nodes and CWS pane
  - From this pane, actions can be taken and conditions can be monitored for CWS, nodes and system partitions.
  - The CWS, system and partition pane enables you to do aggregate monitoring on a system-wide or partition-wide basis.
  - You can also use this pane to set the currently active system partition in which you wish to work.
  - This pane enables you to monitor the control workstation and view any of its attributes.
- Node pane
  - The *nodes pane* displays all nodes configured in the current system or system partition.
  - Nodes are represented according to their type (thin, wide, or high) which reflect the number of frame slots that they occupy.
  - SP-attached server is not physically contained in an SP frame, but functions as a logical node. It is displayed in the nodes pane outside a frame.
- Frame and switches pane
  - The frames and switches pane allows you to control frame and switch hardware and do aggregate monitoring on a frame basis.
  - Frame and switch attributes can be viewed via the notebook function.
  - From this pane, actions can be taken and conditions can be monitored.
- Node group
  - The Node Groups pane enables you to view and manage a collection of nodes at one time.

  - You can create node groups according to many attributes, including:
    - The type of work performed on the nodes
    - Departmental use of those nodes
    - Node types
  - From this pane, actions can be taken and conditions can be monitored for node groups.
- Netfinity nodes
  - The Netfinity nodes pane displays all netfinity nodes configured in the current system or system partition.

Hardware perspective - monitor

From the perspective main pane you can select `Hardware: Monitor for three important conditions`. This is in essence a customized view of three selected hardware objects and conditions (hostresponds, switchrespnds and nodespower), also available through the hardware perspective.

Figure 2-14 on page 84 provides an example of the hardware monitor. You can instantly observe the problems on node 1 and node 13.

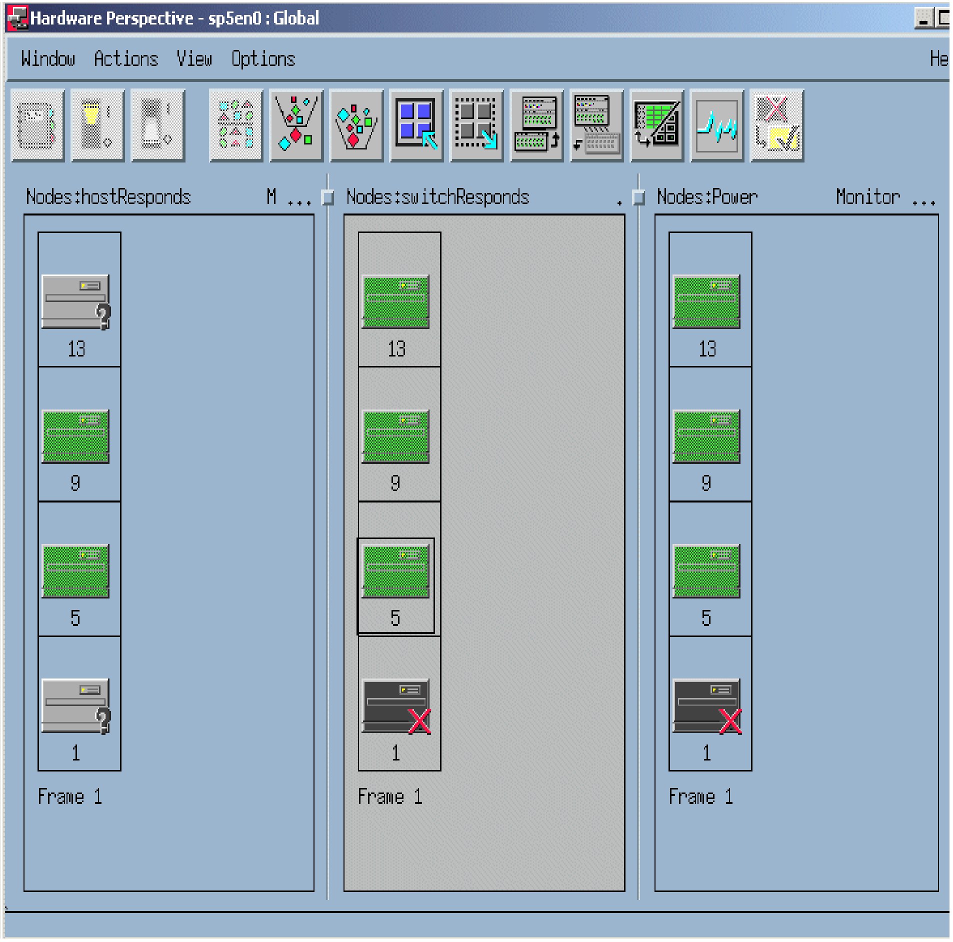


Figure 2-14 Hardware perspective – monitor for three conditions

2.1.30 Command line tools

PSSP provides command-oriented tools for system administration in addition to graphical tools for system administration and monitoring. These tools require no special workstation capability or high-speed connection, making them usable by almost any terminal type in any mode of access. Use these tools when examining system status through a modem connection or through a node's S1 serial port. The tools discussed in this section are documented in greater detail in *Parallel System Support Programs for AIX: Command and Technical Reference,* SA22-7351; *Parallel System Support Programs for AIX: Administration Guide,* SA22-7348; *AIX Version 4 Commands Reference, SBOF-1877*. Although these tools do not possess the same ease-of-use characteristics as their perspectives based counterparts, they do provide the same basic functions.

Several commands are useful for monitoring the system status and detecting problem situations:

- **spmon**
- **hmmon**
- **df**
- **dsh**
- **lsps**
- **lssrc**

spmon command

The **spmon** command requires user to have specific authorizations. To learn how a user can acquire these authorizations, see Chapter 20, "Using the SP System Monitor" in *Parallel System Support Programs for AIX: Administration Guide, SA22-7348.*

The **spmon** command permits the user to control and monitor SP hardware resources through a command-line interface without requiring a graphics-capable terminal or high-speed connection. The **spmon** command does not provide the capability to examine software status (such as paging space, file system space, or software subsystem activity). The **spmon** command provides a predefined system query to check the most basic problem conditions within the SP system.

To perform a basic diagnostic check on the entire SP system, issue the **/usr/lpp/ssp/bin/spmon -G -d | more** command from the control workstation: Example 2-18 provides the output of this command.

Example 2-18 spmon output

```
1.  Checking server process
    Process 6204 has accumulated 360 minutes and 24 seconds.
    Check successful

2.  Opening connection to server
    Connection opened
    Check successful

3.  Querying frame(s)
    1 frame
    Check successful
```

```
4.  Checking frames

      Controller Slot 17 Switch Switch   Power supplies
Frame Responds   Switch  Power  Clocking A     B     C     D
----- ---------- ------- ------ -------- ----- ----- ----- -----
   1      yes      yes     on      N_A     on    on    on   N/A

5.  Checking nodes
----------------------------------- Frame 1 ----------------------------------
                      Host     Switch   Key     Env   Front Panel      LCD/LED
Slot Node Type  Power Responds Responds Switch  Error LCD/LED          Flashes
---- ---- ----- ----- -------- -------- ------- ----- ---------------- -------
  1     1  high   on     yes      no      N/A     no  LCDs are blank      no
  5     5  high   on     yes      yes     N/A     no  LCDs are blank      no
```

Note that these tests are numbered. This makes it easy to detect if a test was omitted. The results of this command indicate potential problems if any of the following conditions exist:

- The command does not run.
- The command does not perform all five verification checks.
- The fourth test indicates that the frame's controller is not responding, the switch power is not on, or any of the power supplies are listed as off.
- The fifth test indicates any abnormal conditions: a node's power is off, the host responds does not read yes, an environment failure is indicated, or the LCD or LED of the node is not blank (but not flashing).
- The fifth test indicates that the node's LCDs or LEDs are flashing. This indicates that a system dump was attempted.
- The fifth test indicates that the node is not responding to the switch device.

hmmon command

The **hmmon** command provides hardware monitoring functions similar to the **spmon** command. In addition, the **hmmon** command provides detailed hardware information about frames and switches. The **hmmon** command provides the capability to monitor frame and switch status as well as node status. This command is intended to be a general-purpose SP hardware monitor. Although it has access to more SP information than the **spmon** command, it does not have access to some of the node-specific information that the **spmon** command does. The **hmmon** command does not provide a predefined system query, which is available through the **spmon** command.

dsh command

The **dsh** (distributed shell) command permits the user to issue a command locally, for selected remote nodes and view the results on the local node. Using **dsh**, you can issue the commands listed previously on any SP node from a single location. This removes the need to login to each node individually. Like the **spmon** command, users must have specific authorization to use the **dsh** command.

df command

The **df** command is an AIX command that examines the current status of file systems, such as current file system size and current available space within these file systems. While this command is designed to examine the AIX system on which it is issued, it can be invoked remotely with the **dsh** and **rsh** commands to acquire this information for all nodes on SP. Three file systems are of particular importance for all SP nodes:

- /spdata

  This directory contains configuration information for PSSP software and also contains copies of information from the SDR. By default, this directory resides in the / (root) file system. Insufficient space in this file system can result in failures in PSSP software, especially those dependent on the SDR for proper operation. As a rule of thumb, you should ensure that this file system has at least 5% of its capacity available at any time. One highly recommended method for avoiding space problems for the /spdata directory is to create a separate file system for this directory.

- /var

  This file system contains AIX system logs, such as error logs and user access logs. It also contains logs maintained by PSSP software for serviceability purposes. Some of these logs are never cleared except by explicit system administrator actions. If left unattended, they can grow to consume all available space. As a rule of thumb, you should ensure that 10 MB of space is available within this file system at all times. If the file system reaches this threshold, consider either extending the file system's capacity with the **chfs** command, or examine the file system to determine where the space is being consumed and remove unwanted files.

- /tmp

  This file system is used by various user level applications, software products, and PSSP programs for temporary storage. Some legacy PSSP applications use this file system to store trace logs used for serviceability purposes. Some applications may inadvertently leave temporary files in the /tmp file system, or these applications may terminate before removing these files. Insufficient space in /tmp can cause PSSP software to fail. As a rule of thumb, you should ensure that at least 8 MB of space is available in this file system at any time; 8 MB is the amount of space the **snap** command will require if the system has to produce a dump to be sent to the IBM Support Center.

lsps command

The **lsps** command provides an instant assessment of the currently available paging space for an AIX system. As with the **df** command, the **lsps** command provides information for the AIX system on which it runs. Using the **lsps** command with the **dsh** or **rsh** command, you can obtain the assessment for all nodes in your SP.

lssrc command

The **lssrc** command provides information for software services currently installed on an AIX system. Using **lssrc**, you can determine if a software service is active or inactive. Use this command in cases where a software service does not appear to be responding to requests for service on a specific node. To check software service status on multiple nodes, use this command through the **dsh** or **rsh** commands.

2.1.31 Useful sites

Table 2-6 lists sites that can prove useful in problem diagnosis and resolution process.

Table 2-6 Useful sites

IBM Redbooks `http://www.redbooks.ibm.com/`
AIX manuals `http://www.rs6000.ibm.com/resource/aix_resource/Pubs/index.html`
PSSP manuals `http://www.rs6000.ibm.com/resource/aix_resource/sp_books/index.html`
APARs and PTFs `http://techsupport.services.ibm.com/rs6000/fixes`
Forums `http://techsupport.services.ibm.com/rs6000/forums`
General AIX technical documents `http://techsupport.services.ibm.com/rs6000/techKnow`
Main entry for IBM support page `http://techsupport.services.ibm.com/eserver/support`

3

Installation

In this chapter, we step through the common problems that are encountered during the universal cluster installation process. Each section within this chapter references a step within the installation process. In addition to this, each section contains problems and solutions that could be encountered during installation.

3.1 Overview of the installation process

To successfully set up and configure a clustered system, we suggest that the *Installation and Migration Guide*, GA22-7347-02 be followed. The installation guide includes procedures for command line, SMIT menus and perspectives. In this chapter, we have focused on the command line methodology. Problem determination of a script is much easier to debug than a simple icon on a perspective screen.

3.1.1 Setup of the CWS

Use the following steps to configure the CWS:

1. Update root’s $PATH to include the SP-specific executable.
2. Ensure that the bos.net and perfagent.tools are installed on the system.
3. Connect the frames to the CWS.
4. Configure the RS-232 tty ports.
5. Configure and tune the network adapters on the CWS.
6. Ensure the System Resource Controller (SRC) is active.
7. Increase the maximum number of runable processes per user to 256.
8. Tune network options.
9. Create additional file systems for /spdata.
10. Create additional file systems for /tftpboot.
11. Create the directory structures needed under /spdata.
12. Copy the AIX LPP images to their lppsource directory.
13. Copy the PSSP installation images to the pssplpp directory structure.
14. Copy a basic mksysb images to the images subdirectory.
15. Install PSSP on the CWS.
16. Enter site environment variables.
17. Set Authentication Methods for AIX remote commands on the CWS.
18. Initialize Kerberos.
19. Set the authentication method for SP trusted services on the CWS.
20. Obtain authentication credentials.
21. Run install_cw.

3.1.2 Enter Information into the SDR

The following steps describe the process to enter information into the SDR:

1. Enter Frame Information and Reinitialize the SDR.
2. Update the state of the supervisor microcode.
3. Enter the required node information.
4. Acquire the hardware ethernet addresses.
5. Configure additional adapters for nodes.
6. Configure initial host names for the nodes.
7. Setup security capabilities required on nodes.
8. Enable authentication methods for AIX remote commands.
9. Enable authenticating methods for SP trusted services.
10. Start system partition-sensitive subsystems.
11. Enter volume group information for nodes into the SDR.
12. Change the default network tunable values.
13. Modify script.cust and tuning.cust files.
14. Select a switch topology file.
15. Set the clock source for all switches.
16. Verify the switch primary and primary backup nodes.
17. Configure control workstation as a Boot Install Server (setup_server).

3.1.3 Power on and install nodes

To install the nodes, follow the next steps:

1. Network boot nodes.
2. Verify node installation.
3. Run verification test on all nodes.
4. Start the optional switch.
5. Tune network adapters on the nodes.
6. Run post installation procedures on the nodes.

3.2 Prepare the control workstation

This section describes the problems you might encounter during the installation of the control workstation. Improper configuration of the control workstation can cause problems that might not be apparent until after the system is in production. There is no substitute for proper planning prior to the installation process. However, even the best plan can run into a fair share of unforeseen problems along the way. This section will outline some of the potential problems and or pitfalls that might occur while preparing the control workstation.

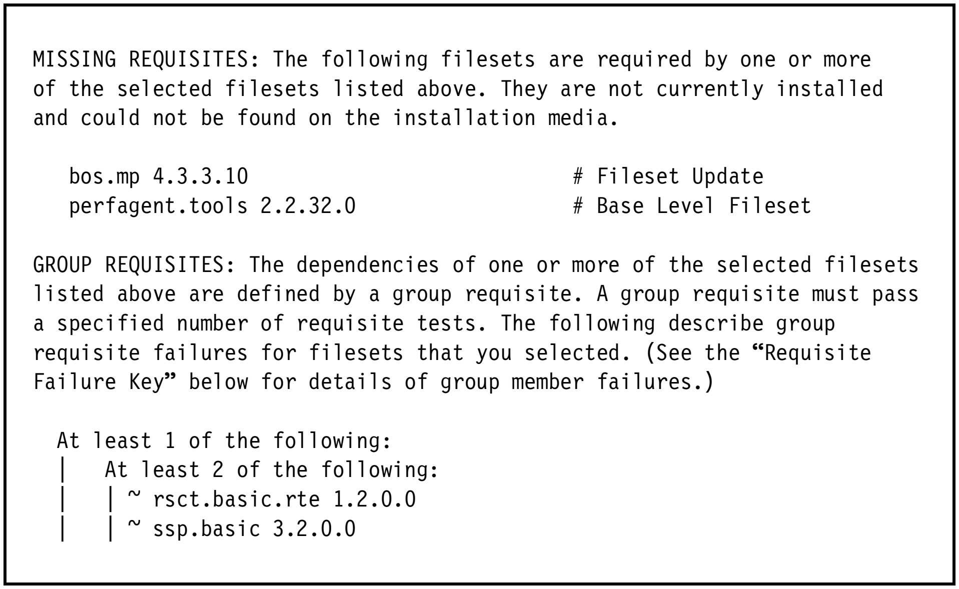
3.2.1 Failures when installing PSSP filesets on the CWS

Prior to installing the PSSP code, there are two requisites that are often overlooked. This causes the missing requisite failures, as shown in Figure 3-1.

The fileset bos.mp 4.3.3.10 can be downloaded from:

```
http://techsupport.services.ibm.com/rs6k/fixdb.html
```

The fileset for perfagent.tools 2.2.32.0 is located on the AIX 4.3.3 media.



```
MISSING REQUISITES: The following filesets are required by one or more
of the selected filesets listed above. They are not currently installed
and could not be found on the installation media.

   bos.mp 4.3.3.10                          # Fileset Update
   perfagent.tools 2.2.32.0                 # Base Level Fileset

GROUP REQUISITES: The dependencies of one or more of the selected filesets
listed above are defined by a group requisite. A group requisite must pass
a specified number of requisite tests. The following describe group
requisite failures for filesets that you selected. (See the “Requisite
Failure Key” below for details of group member failures.)

  At least 1 of the following:
  |   At least 2 of the following:
  |   | ~ rsct.basic.rte 1.2.0.0
  |   | ~ ssp.basic 3.2.0.0
```

Figure 3-1 Failures on the CWS

3.2.2 Configure the RS-232 ports

Each frame within the universal cluster will require a RS-232 port on the control workstation. This port is used for the console (s1term) communications link to the nodes within each frame. In addition to this port, a second port is needed for nodes that are not housed within an SP frame (SP-attached servers). The second port takes over the function of the frame supervisor card.

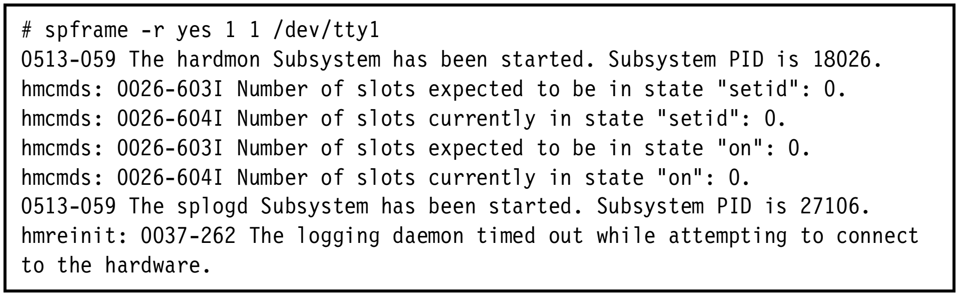
Frame connected to wrong tty port

Each frame has a RS-232 serial connection to the serial port on the CWS, RAN or eight-port break-out-box on the control workstation. It is important to document which port is connected to each frame. We recommend connecting the ports in frame number order to avoid confusion.

Some SP-attached servers (S70, S7A, S80 and S85) utilize a SAMI port as well as the serial port for connectivity to the control workstation. The connection for the SAMI should be the lower of the two tty addresses. It is necessary to document this information as it is needed in later installation steps.

The spframe command times out

If the **spframe** command does not find the frame on the tty devices specified, the command will time out, as shown in Figure 3-2.



```
# spframe -r yes 1 1 /dev/tty1
0513-059 The hardmon Subsystem has been started. Subsystem PID is 18026.
hmcmds: 0026-603I Number of slots expected to be in state "setid": 0.
hmcmds: 0026-604I Number of slots currently in state "setid": 0.
hmcmds: 0026-603I Number of slots expected to be in state "on": 0.
hmcmds: 0026-604I Number of slots currently in state "on": 0.
0513-059 The splogd Subsystem has been started. Subsystem PID is 27106.
hmreinit: 0037-262 The logging daemon timed out while attempting to connect
to the hardware.
```

Figure 3-2 spframe time out

A quick look at the following errpt output shows a few possible reasons for this timeout to occur.

Attention: The timeout here does not occur within the spframe script, but in hmreinit/splogd. If the same problem exists, you won't see a timeout from the command **spframe -r no 1 1 /dev/tty1**, but you will see it when you run **hmreinit**.

```
---------------------------------------------------------------------------
LABEL:          SPMON_EMSG100_ER
IDENTIFIER:     4CEF5A08

Date/Time:       Sat Jul 14 21:48:17
Sequence Number: 53
Machine Id:      000354794C00
Node Id:         sp5en0
Class:           H
Type:            PERM
Resource Name:   sphwlog
Resource Class:  NONE
Resource Type:   NONE
Location:        NONE

Description
LINK ERROR

Probable Causes
CABLE
SYSTEM MICROCODE

Failure Causes
CABLE LOOSE OR DEFECTIVE
SYSTEM MICROCODE

        Recommended Actions
        CHECK CABLE AND ITS CONNECTIONS
        CHECK FOR CORRECT MICROCODE FIX

Detail Data
DETECTING MODULE
LPP=PSSP,Fn=splogd.c,SID=1.16.1.33,L#=1363,
DIAGNOSTIC EXPLANATION
0026-100 Failure; Frame 1:0; controllerResponds; Frame supervisor does not
respond.
---------------------------------------------------------------------------
```

Figure 3-3 spframe frame supervisor does not respond errpt output

The errpt output shown in Figure 3-3 gives two possible explanations for the timeout:

```
Cable and its connections
Microcode level
```

In addition to the two recommended actions listed in the errpt output, be sure to check that:

- The correct serial port is defined to the tty devices specified.
- The correct tty port has been selected.
- The frame supervisor card is functioning correctly.

For an SP-attached server that requires the SAMI interface, the **spframe** command is a bit more complex, as the additional tty port needs to be defined. The tty port that is defined as the SAMI should be the lower of the two tty devices connected to the attached server as shown in Figure 3-4.

The switch node number will also need to be entered into the **spframe** command. This may be any valid switch node number that is not in use by another node. Consider a system with two frames and two switches with 30 thin nodes as shown in Figure 3-5. In this example, there are two switch nodes available (sn30 and sn31). In this particular setup we picked switch node 31 to assign to the S-80 leaving switch node30 to be used as a thin, wide or an SP-attached node.

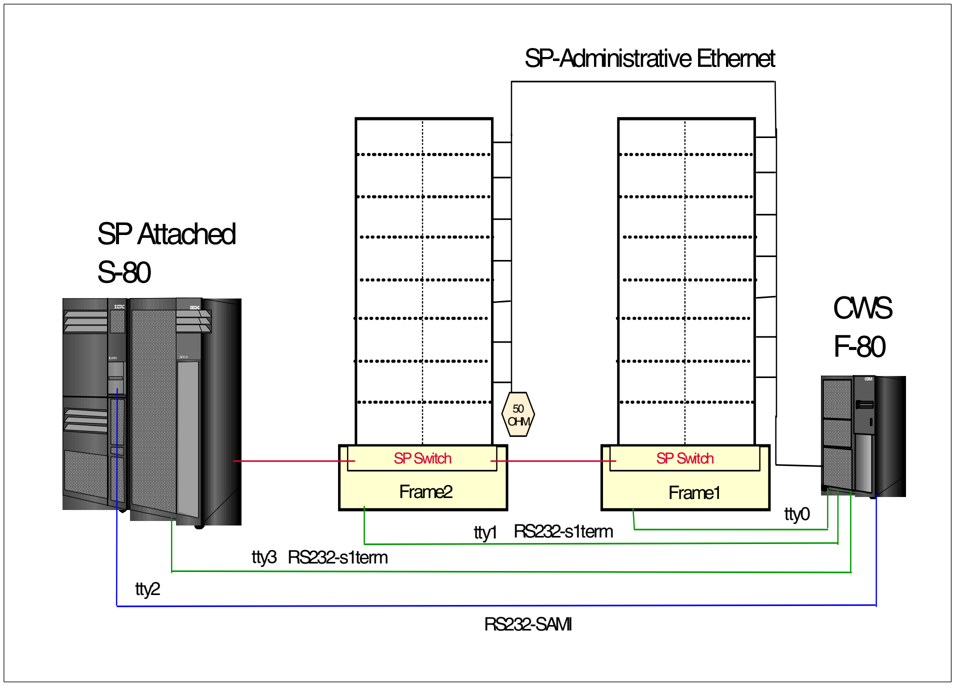


Figure 3-4 tty attachments

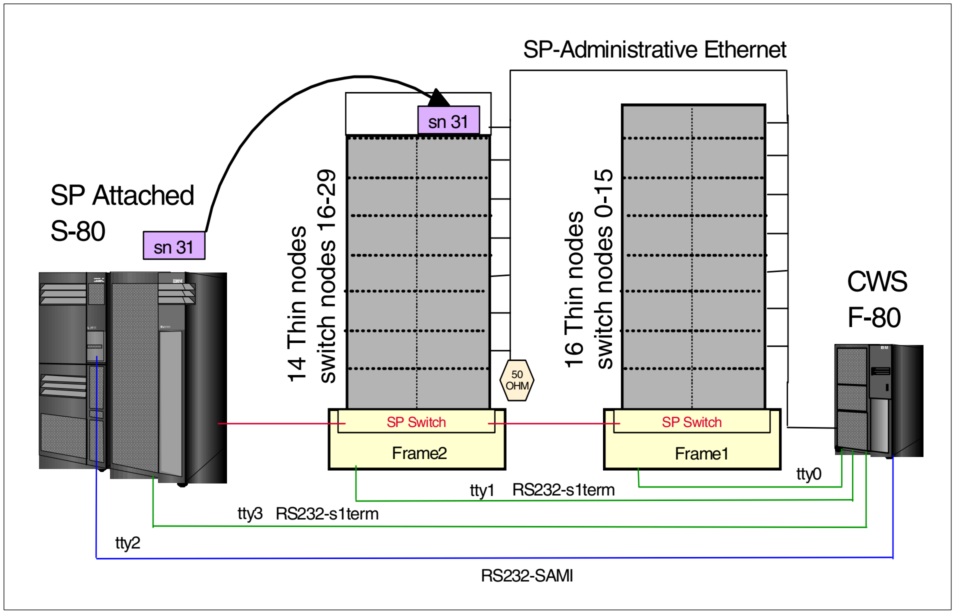


Figure 3-5 Switch node number assignment

In a switchless SP system, SP-attached nodes must also be given a switch node number. We recommend that when adding an SP-attached node into a switchless system, place a hypothetical switch into each frame to determine which switch node number would be available, if such a switch or group of switches exist. This method will yield a valid switch node number that could also be used if the system is converted to a switched system in the future.

Incorrect baud rate for tty

The baud rate for all rs232 connections should be set to 9600. The hardmon daemon will change this to 19200 for the serial terminal connections. However, hardmon does not change this value to 19200 for the SAMI interface. By allowing the hardmon daemon to make this change, rather than manually changing the rate of the device, the SAMI connections will remain at 9600. This is important, as the SAMI interface is not functional at 19200 baud.

3.2.3 Creation of optional spdatavg volume group

Although this is not a necessary step, it may be useful for larger universal clusters to create an additional volume group for the /spdata directory structures. Within this volume group, the following file systems can be utilized:

- /spdata

- /spdata/sys1/install/images
- /tftpboot

3.2.4 setup_authent failure

After the PSSP installation, the first test to ensure that the installation was successful is **setup_authent**. This command will conduct the first **SDRChangeAttrValues** command on the system as shown in Figure 3-6.

```
# setup_authent
/usr/lpp/ssp/bin/SDRChangeAttrValues: 0025-004 Item specified for query,
insertion or deletion was not found.#
```

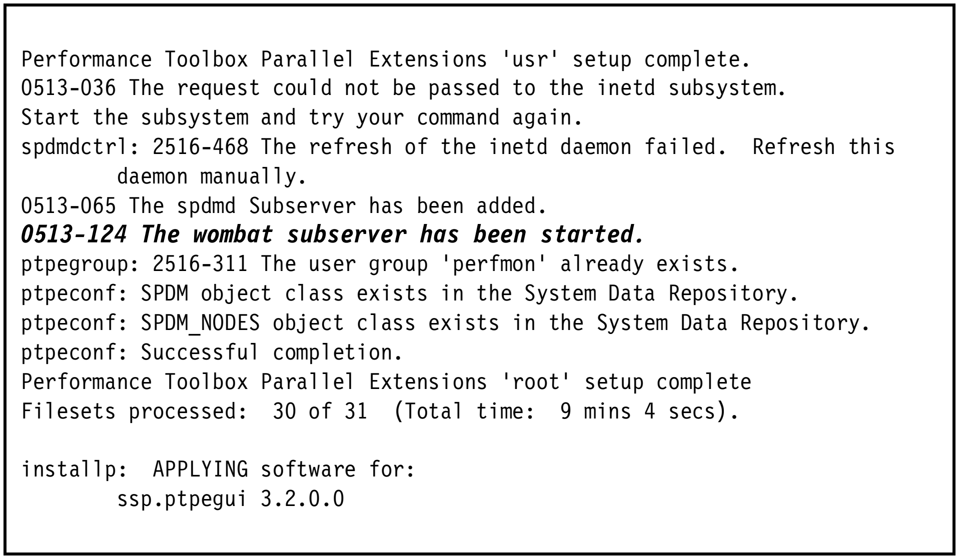
Figure 3-6 setup_authent fails due to SDR failure

If a problem is encountered at this stage, check the output of the **lslpp -l ssp*** command as in Figure 3-7.

```
ssp.authent              3.2.0.0  COMMITTED  SP Authentication Server
 ssp.basic                3.2.0.0  COMMITTED  SP System Support Package
 ssp.cediag               3.2.0.0  COMMITTED  SP CE Diagnostics
 ssp.clients              3.2.0.0  COMMITTED  SP Authenticated Client
 ssp.css                  3.2.0.0  COMMITTED  SP Communication Subsystem
 ssp.docs                 3.2.0.0  COMMITTED  SP man pages and PDF files and
 ssp.gui                  3.2.0.0  COMMITTED  SP System Monitor Graphical
 ssp.ha_topsvcs.compat    3.2.0.0  COMMITTED  Compatability for ssp.ha and
                                              ssp.topsvcs clients
 ssp.jm                   3.2.0.0  COMMITTED  SP Job Manager Package
 ssp.perlpkg              3.2.0.0  COMMITTED  SP PERL Distribution Package
 ssp.pman                 3.2.0.0  COMMITTED  SP Problem Management
 ssp.ptpegui              3.2.0.0  COMMITTED  SP Performance Monitor
 ssp.public               3.2.0.0  COMMITTED  Public Code Compressed
 ssp.spmgr                3.2.0.0  COMMITTED  SP Extension Node SNMP Manager
 ssp.st                   3.2.0.0  COMMITTED  Job Switch Resource Table
 ssp.sysctl               3.2.0.0  COMMITTED  SP Sysctl Package
 ssp.sysman               3.2.0.0  COMMITTED  Optional System Management
 ssp.tecad                3.2.0.0  COMMITTED  SP HA TEC Event Adapter
 ssp.tguides              3.2.0.0  COMMITTED  SP TaskGuides
 ssp.top                  3.2.0.0  COMMITTED  SP Communication Subsystem
 ssp.top.gui              3.2.0.0  COMMITTED  SP System Partitioning Aid
 ssp.ucode                3.2.0.0  COMMITTED  SP Supervisor Microcode
 ssp.vsdgui               3.2.0.0  COMMITTED  VSD Graphical User Interface
```

Figure 3-7 Check PSSP filesets on the CWS.

The output of the **lslpp** command shows that the fileset for the PSSP software were installed. When installing the PSSP software as in any installation, make sure to keep a log file of the output from the installation commands or the smit.log file. Investigation of the installation log shows a very interesting situation. Note that in Figure 3-8, there is a new subserver that has appeared, called wombat. For some unknown reason the system has started a subserver that is not a part of the PSSP code.



```
Performance Toolbox Parallel Extensions 'usr' setup complete.
0513-036 The request could not be passed to the inetd subsystem.
Start the subsystem and try your command again.
spdmdctrl: 2516-468 The refresh of the inetd daemon failed.  Refresh this
        daemon manually.
0513-065 The spdmd Subserver has been added.
0513-124 The wombat subserver has been started.
ptpegroup: 2516-311 The user group 'perfmon' already exists.
ptpeconf: SPDM object class exists in the System Data Repository.
ptpeconf: SPDM_NODES object class exists in the System Data Repository.
ptpeconf: Successful completion.
Performance Toolbox Parallel Extensions 'root' setup complete
Filesets processed:  30 of 31  (Total time:  9 mins 4 secs).

installp:  APPLYING software for:
        ssp.ptpegui 3.2.0.0
```

Figure 3-8 What is wombat?

> **Note:** As you were driving down the installation path, where did the wombat come from?

Upon additional investigation, it was determined that there is an entry for wombat in the /etc/services file as shown in Figure 3-9 on page 100.

```
isode-dua       17007/tcp
isode-dua       17007/udp
dtspc           6112/tcp
spseccfg                6681/tcp
sysctl          6680/tcp        sysctld
hardmon         8435/tcp
sdr             5712/tcp
sdrprot         1712/tcp
spmgrd-trap             162/udp
supfilesrv      8431/tcp
kfcli 32802/tcp
wombat          10000/udp
```

Figure 3-9 What is wombat doing in /etc/services?

This entry in the /etc/services file for wombat is due to an application that has port 10000 hard coded into the software. There is now a port conflict between the application's wombat entry and the HA services entry in the /etc/service file. To resolve port conflicts for the HA daemons, we recommend the following procedure:

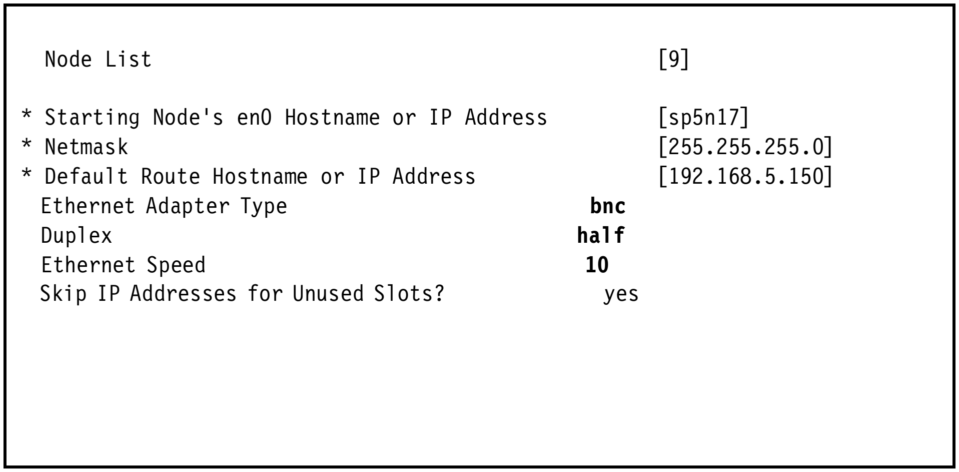
- Un-install the PSSP code
- Remove the entry in /etc/service for wombat
- Re-install the PSSP software

3.3 Entering Universal Cluster SDR Information

This section describes the steps to enter SDR information for the Universal cluster.

3.3.1 SP ethernet information

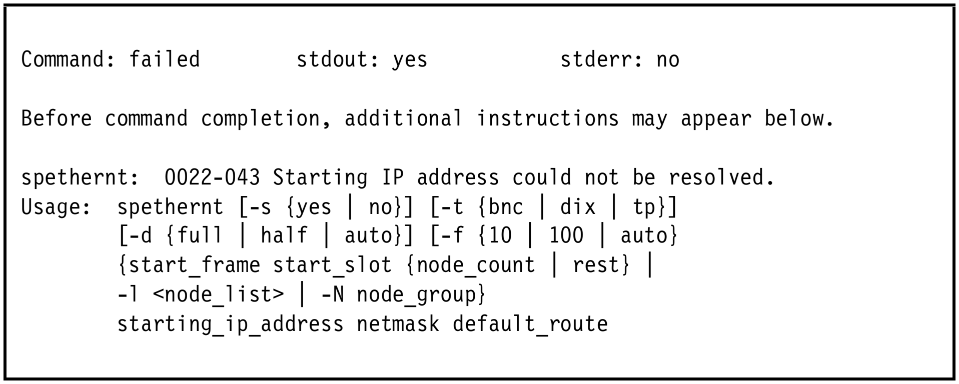
While entering the SP ethernet information, ensure that the entries for the netmask, duplex, ethernet speed and ethernet type are correct. In our system, we were using a coax BNC network and our settings are shown in Figure 3-10 on page 101.



```
  Node List                                         [9]

* Starting Node's en0 Hostname or IP Address        [sp5n17]
* Netmask                                           [255.255.255.0]
* Default Route Hostname or IP Address              [192.168.5.150]
  Ethernet Adapter Type                         bnc
  Duplex                                       half
  Ethernet Speed                               10
  Skip IP Addresses for Unused Slots?             yes
```

Figure 3-10 SP Ethernet information

Also keep in mind that the addresses must be resolvable. If the hostname is not resolvable, there will be an error, as shown in Figure 3-11. To correct this, verify the /etc/hosts file entries and ensure that the correct hostname is being used.



```
Command: failed       stdout: yes          stderr: no

Before command completion, additional instructions may appear below.

spethernt:  0022-043 Starting IP address could not be resolved.
Usage:  spethernt [-s {yes | no}] [-t {bnc | dix | tp}]
        [-d {full | half | auto}] [-f {10 | 100 | auto}
        {start_frame start_slot {node_count | rest} |
        -l <node_list> | -N node_group}
        starting_ip_address netmask default_route
```

Figure 3-11 SP Ethernet hostname not resolvable

3.3.2 Obtain the Ethernet hardware address

The following command will acquire the hardware address for node number 9:

```
      [root@sp5en0]: sphrdwrad -l 9
Acquiring hardware ethernet address for node 9 from /etc/bootptab.info
```

Using /etc/bootptab.info file

If the hardware address is already known for the adapters in the system, then the entries may be placed into the /etc/bootptab.info file. This file has the following characteristics, as shown in Figure 3-12.

```
# cat /etc/bootptab.info
1  02608CF57A7C
5  02608C2D4CA5
9  02608C2D63E2
13 02608CF518A4
```

Figure 3-12 /etc/bootptab.info file

/etc/bootptab.info file problems

If a bootptab.info file exist, then the script that obtains the hardware address of the node will not boot the node to acquire the hardware address. This can be helpful in saving time, as system reboots can be time consuming. However, if the bootptab.info file has become corrupted, or the entries in the file are incorrect, then the correct hardware address may never be acquired.

To remedy this problem, we suggest that the /etc/bootptab.info file be removed or the name changed. This will allow the node to be booted and the hardware address acquired. We suggest the following two steps:

1. Rename the file by issuing the command **`mv /etc/bootptab.info /etc/bootptab.info.bak`**.
2. Re-acquire the hardware address as shown in Figure 3-13.

```
   [root@sp5en0]:/etc> sphrdwrad 1 1 4
   Acquiring hardware Ethernet address for node 5
   Acquiring hardware Ethernet address for node 9
   Acquiring hardware Ethernet address for node 13
   Hardware ethernet address for node 5 is 02608C2D4CA5
   Hardware ethernet address for node 9 is 02608C2F9083
   Hardware ethernet address for node 13 is 02608CF518A4
```

Figure 3-13 Output from sphrdwrad

The following command will acquire the hardware address for node number 9:

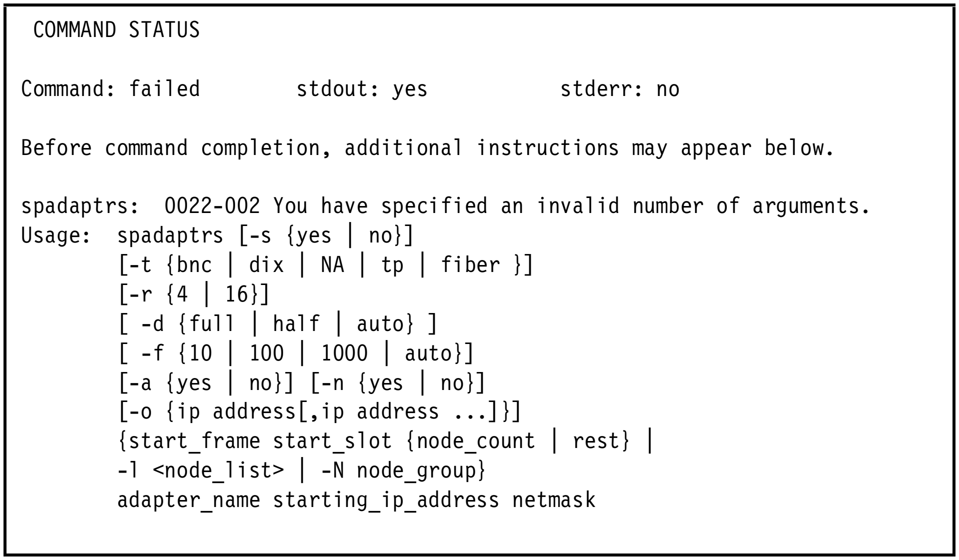
```
[root@sp5en0]:/spdata/sys1/install/images> sphrdwrad -l 9
Acquiring hardware ethernet address for node 9 from /etc/bootptab.info
```

Note: Acquiring the hardware address does not verify that the network connection between the CWS and the node(s) is viable. This command acquires the hardware address across the s1term connection.

3.3.3 Adding additional Network Adapters into the SDR

Enter all the network interfaces for each node into the SDR. If the system has HACMP configured on some nodes, then enter the adapter hostname information for the boot address and standby address. Allow HACMP to configure the service address when the cluster daemons have started.

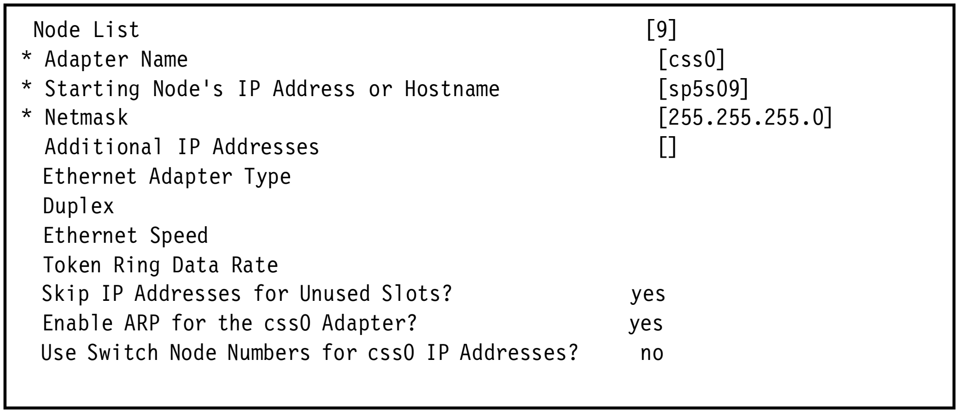
When entering information for the CSS0 adapter, do not enter a value for the Ethernet Adapter Type (Duplex or Ethernet Speed) as this will cause a failure as shown in Figure 3-14.



```
 COMMAND STATUS

Command: failed        stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

spadaptrs:  0022-002 You have specified an invalid number of arguments.
Usage:  spadaptrs [-s {yes | no}]
        [-t {bnc | dix | NA | tp | fiber }]
        [-r {4 | 16}]
        [ -d {full | half | auto} ]
        [ -f {10 | 100 | 1000 | auto}]
        [-a {yes | no}] [-n {yes | no}]
        [-o {ip address[,ip address ...]}]
        {start_frame start_slot {node_count | rest} |
        -l <node_list> | -N node_group}
        adapter_name starting_ip_address netmask
```

Figure 3-14 css0 configuration failed in SMIT

You will not be able to clear the entries for these field in SMIT. Therefore, exit pressing the F3 key, open the menu again, and re-try with the correct parameters, as shown in Figure 3-15 on page 104.



```
 Node List                                          [9]
* Adapter Name                                       [css0]
* Starting Node's IP Address or Hostname             [sp5s09]
* Netmask                                            [255.255.255.0]
  Additional IP Addresses                            []
  Ethernet Adapter Type
  Duplex
  Ethernet Speed
  Token Ring Data Rate
  Skip IP Addresses for Unused Slots?              yes
  Enable ARP for the css0 Adapter?                 yes
  Use Switch Node Numbers for css0 IP Addresses?     no
```

Figure 3-15 css0 configuration with node_data entry in SMIT

3.3.4 SP volume group information

The following section provides information on the SP volume group characteristics important during problem determination on an SP system.

Create volume group information

It is important to understand the directory structure in which the **spmkvgobj** and **spchvgobj** commands are referencing. There are expectations for files to exist in predetermined directory structures. If the files do not exist, the command will fail. For more detail information on the spdata directory structure, refer to Figure 3-16 on page 105.

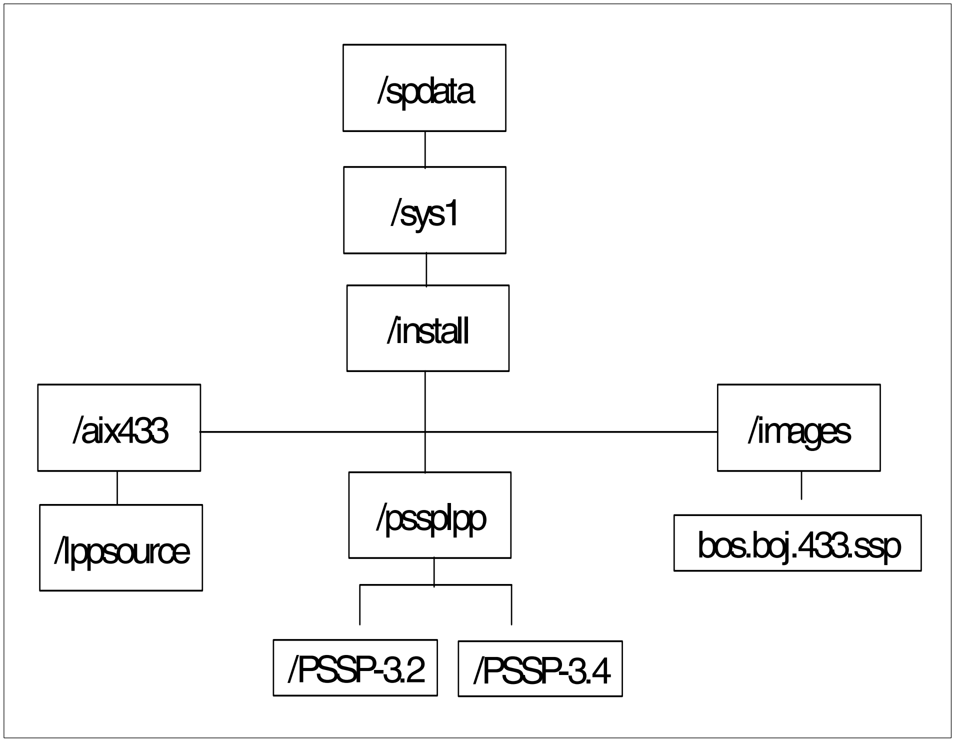


Figure 3-16 spdata directory structure

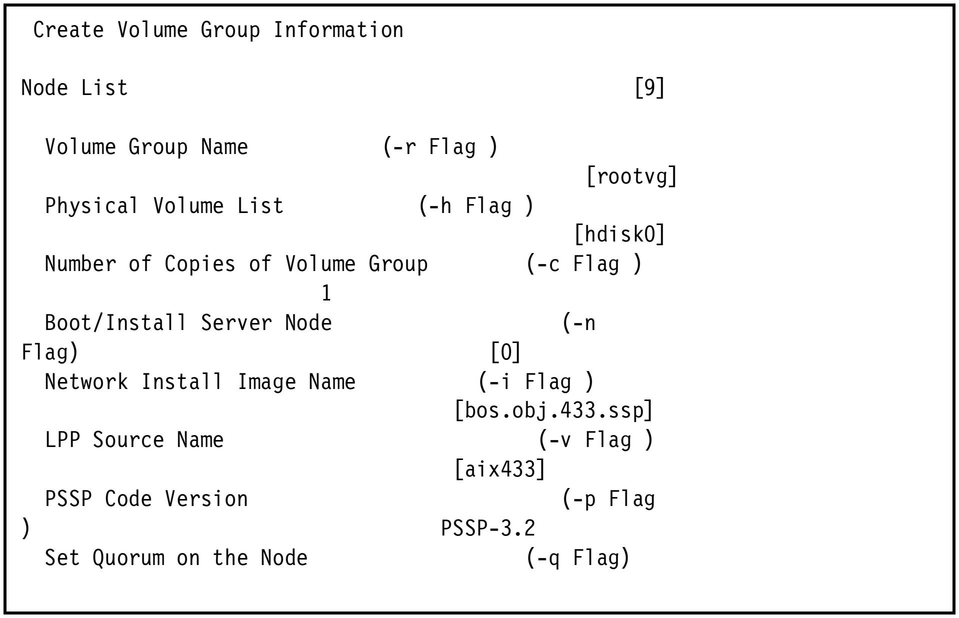
Flags in the spmkvgobj and spchvgobj commands

Table 3-1 displays the flags that are used with the **spmkvgobj** and **spchvgobj** commands. Each of these flags are represented in the smitty menu, as shown in Figure 3-17 on page 106.

Table 3-1 Flags used with the spmkvgobj and spchvgobj commands

Flag	Description
-r	Volume Group Name (For example, rootvg)
-h	Physical Volume List (For example, hdisk0)
-c	Number of Copies of Volume Group
-n	Boot Server
-i	Network Install Image Name; must be located in /spdata/sys1/install/images

Flag	Description
-v	LPP Source Name (For example, aix433); must be located in /spdata/sys1/install/<lppsourcename>; this should be a directory structure with a lppsource directory structure beneath it.
-p	PSSP Code Version (For example. PSSP-3.2) This is a pull down menu option in smitty The directory structure is /spdata/sys1/install/pssplpp



```
 Create Volume Group Information

Node List                                          [9]

  Volume Group Name           (-r Flag )
                                               [rootvg]
  Physical Volume List           (-h Flag )
                                              [hdisk0]
  Number of Copies of Volume Group        (-c Flag )
                         1
  Boot/Install Server Node                   (-n
Flag)                                  [0]
  Network Install Image Name          (-i Flag )
                                    [bos.obj.433.ssp]
  LPP Source Name                          (-v Flag )
                                    [aix433]
  PSSP Code Version                          (-p Flag
)                                  PSSP-3.2
  Set Quorum on the Node                  (-q Flag)
```

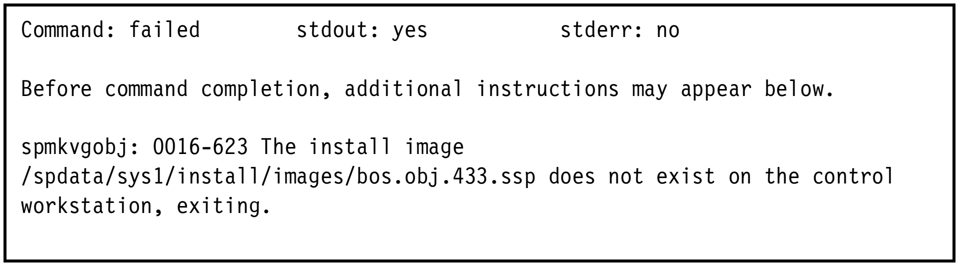
Figure 3-17 Create volume group information

Network install image name does not exist on the CWS

The SDR is expecting this image to be in the following directory:

```
/spdata/sys1/install/images
```

If the file is not there, then an error will occur as shown in Figure 3-18 on page 107.

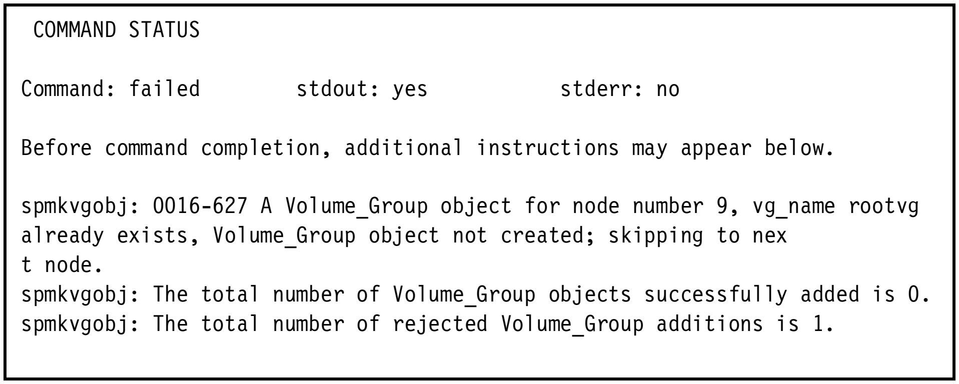


```
Command: failed        stdout: yes          stderr: no

Before command completion, additional instructions may appear below.

spmkvgobj: 0016-623 The install image
/spdata/sys1/install/images/bos.obj.433.ssp does not exist on the control
workstation, exiting.
```

Figure 3-18 spmkvgobj failed due to the install image not existing

A quick look at the /spdata/sys1/install/images directory shows that the name is not bos.obj.433.ssp, but instead is bos.obj.ssp.433.

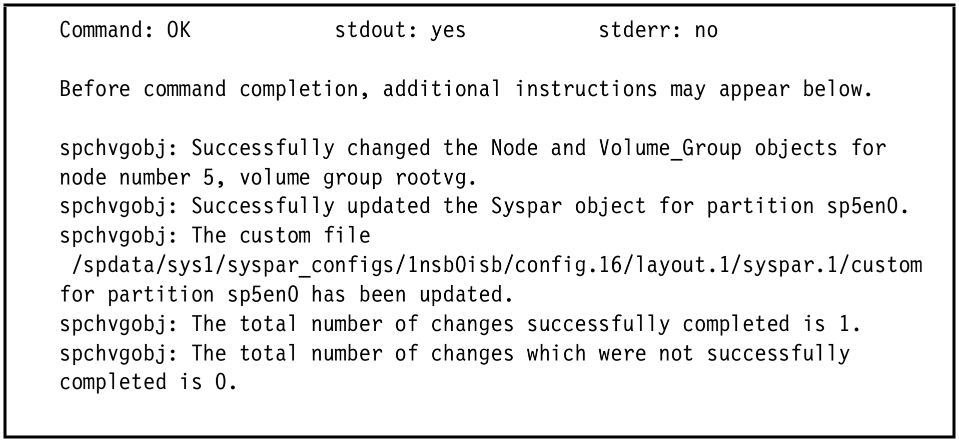
If there is already volume group information in the SDR for the node, then an error such as the one shown in Figure 3-19 will occur. This is simple to rectify with the use the **spchvgobj** command with the same options as used in the **spmkvgobj** command.



```
 COMMAND STATUS

Command: failed        stdout: yes          stderr: no

Before command completion, additional instructions may appear below.

spmkvgobj: 0016-627 A Volume_Group object for node number 9, vg_name rootvg
already exists, Volume_Group object not created; skipping to nex
t node.
spmkvgobj: The total number of Volume_Group objects successfully added is 0.
spmkvgobj: The total number of rejected Volume_Group additions is 1.
```

Figure 3-19 Failure due to the volume group already existing

The volume group can also be modified with smitty as shown in Figure 3-20.

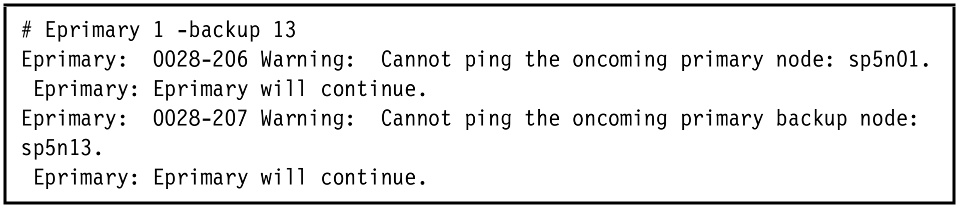


```
Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

spchvgobj: Successfully changed the Node and Volume_Group objects for
node number 5, volume group rootvg.
spchvgobj: Successfully updated the Syspar object for partition sp5en0.
spchvgobj: The custom file
 /spdata/sys1/syspar_configs/1nsb0isb/config.16/layout.1/syspar.1/custom
for partition sp5en0 has been updated.
spchvgobj: The total number of changes successfully completed is 1.
spchvgobj: The total number of changes which were not successfully
completed is 0.
```

Figure 3-20 update volume group information

3.3.5 Validate switch primary and primary backup nodes

Figure 3-21 on page 108 shows a failure occurred while validating both nodes. This is normal at this point of the installation as their are no nodes yet installed. If there are nodes installed (and they are functional), then see Chapter 6, “SP switches” on page 233 for switch problem determination procedures.



```
# Eprimary 1 -backup 13
Eprimary:  0028-206 Warning:  Cannot ping the oncoming primary node: sp5n01.
 Eprimary: Eprimary will continue.
Eprimary:  0028-207 Warning:  Cannot ping the oncoming primary backup node:
sp5n13.
 Eprimary: Eprimary will continue.
```

Figure 3-21 Eprimary failures

3.4 Setup_server

The SDR provides the information setup_server needs to configure NIM on the Boot Install Server (BIS). The setup_server script is divided into Perl wrappers each of which is an independent script. The use of wrappers assist in the problem determination of setup_server. The name of the wrappers are listed in Table 3-2 on page 110 by the order by which setup_server calls them.

Table 3-2 The wrappers of setup_server

services_config Location: /usr/lpp/ssp/install/bin	Checks the site environment information
setup_cws Location: /usr/lpp/ssp/bin	Checks to see if this is the CWS ensures krb files exist /etc/krb-srvtab /etc/krb.conf /etc/krb.realms Performs all kerberos updates on the CWS required for each node Creates and/or updates CWS files
delnimmast Location: /usr/lpp/ssp/bin	If the node parsed is a NIM master then unconfigure it.
delnimclient Location: /usr/lpp/ssp/bin	Undefines the client node on the master
mknimmast Location: /usr/lpp/ssp/bin	Creates the NIM masters
create_krb_files Location: /usr/lpp/ssp/bin	Creates the krb-srvtab file on CWS copies krb-srvtab to BIS (if necessary) created /spdata/sys1/k4srvtab
mknimint Location: /usr/lpp/ssp/bin	Craetes the NIM network objects.
mknimres Location: /usr/lpp/ssp/bin	Created the NIM resources
mknimclient Location: /usr/lpp/ssp/bin	Creates the NIM client definitions
mkconfig Location: /usr/lpp/ssp/bin	creates /tftpboot/<hostname>.config_info used for node customization
mkinstall Location: /usr/lpp/ssp/bin	creates /tftpboot/<hostname>.install_info used for installation and customization
export_clients Location: /usr/lpp/ssp/bin	exports the pssplpp filesystem to clients
allnimres Location: /usr/lpp/ssp/bin	Adds entries into /etc/bootptab if needed allocates NIM resources to clients

3.4.1 The flow of the setup_server command

The following flowcharts in Figure 3-22 and Figure 3-23 on page 112 describe the flow for the **setup_server** and **services_config** commands, respectively.

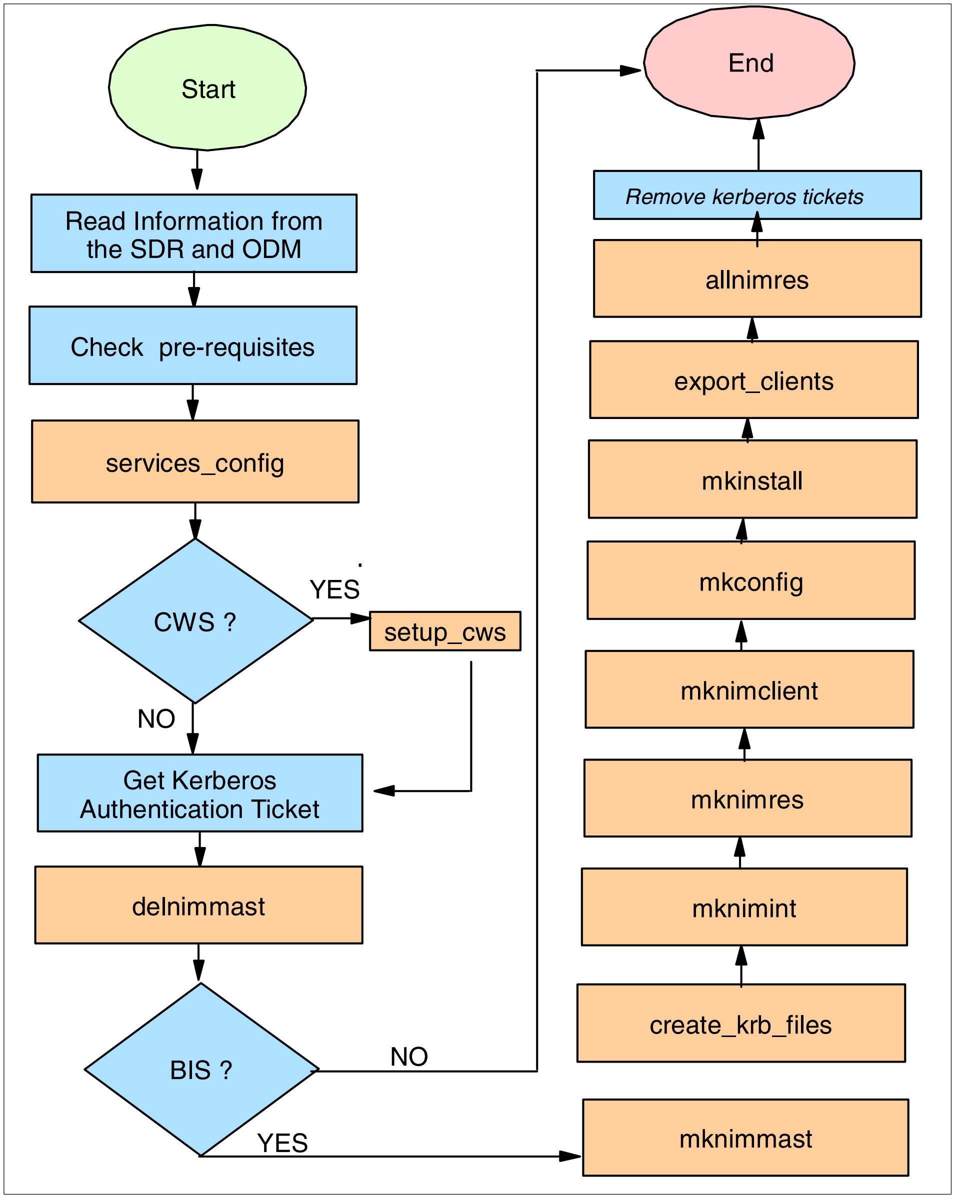


Figure 3-22 setup_server flowchart

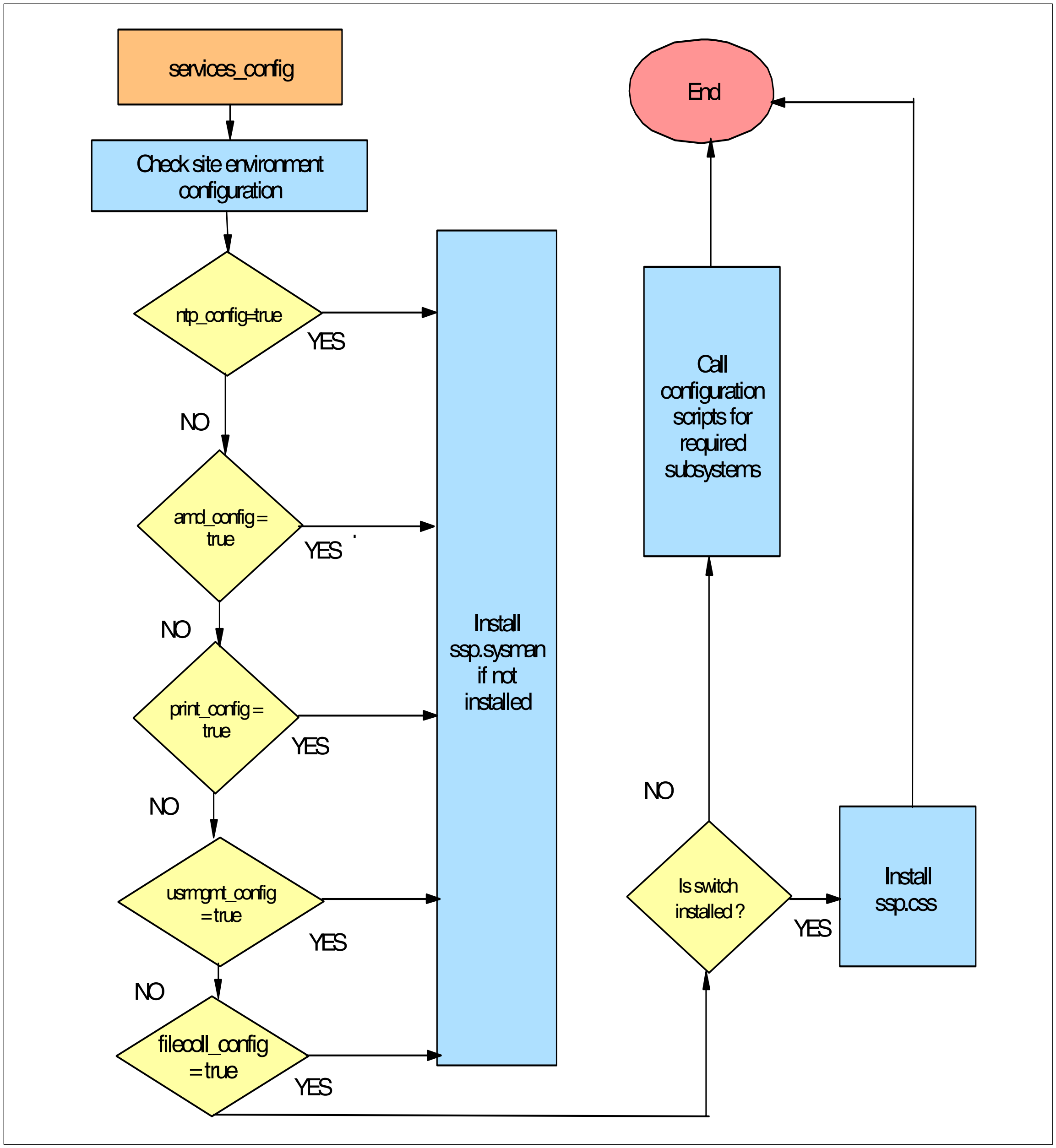


Figure 3-23 services_config flowchart

3.4.2 services_config

This script is invoked by /etc/rc.sp to set up designated services on the nodes, and by **setup_server** to set up services on the boot/install server.

It reads the SDR information stored in the SP object class and checks which services will be run on the node. Then, it calls the appropriate service configuration scripts. The possible services to invoke are the following:

- NTP
- Print management
- User management
- AMD
- File collection
- Accounting

Command Syntax:

```
services_config
This command has no input parameters.
```

The general flow and logic for services_config is shown in Figure 3-23 on page 112.

3.4.3 setup_CWS

The setup_CWS script reads the information from the SDR, checks prerequisites, and updates Kerberos files to reflect CWS and node network interface names. It then makes a setup of CWS-specific items.

Command Syntax:

```
setup_CWS [-h]
This command has no input parameters.
```

The general flow and logic for setup_CWS is shown in Figure 3-24 on page 114.

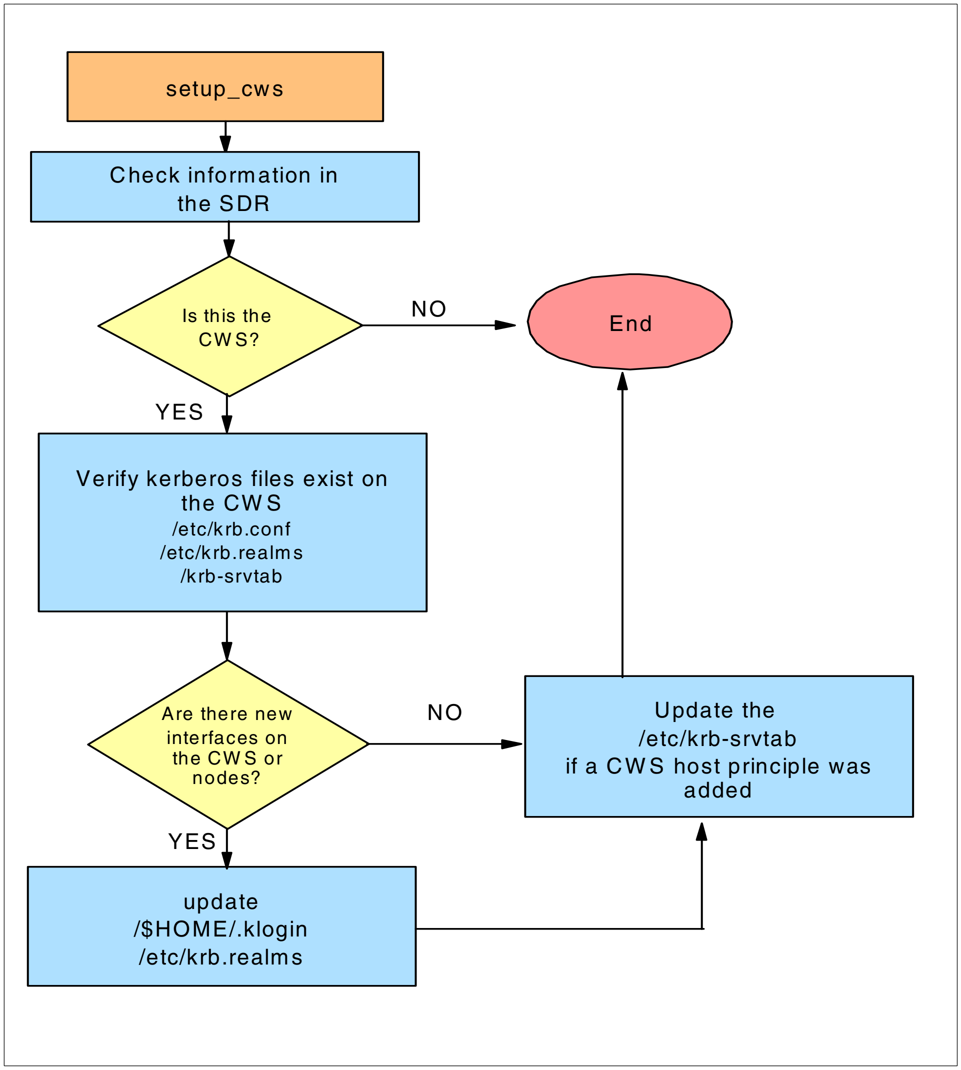

setup_cws
Check information in the SDR
Is this the CWS?
NO
End
YES
Verify kerberos files exist on the CWS
/etc/krb.conf
/etc/krb.realms
/krb-srvtab
Are there new interfaces on the CWS or nodes?
NO
Update the /etc/krb-srvtab if a CWS host principle was added
YES
update
/$HOME/.klogin
/etc/krb.realms

Figure 3-24 setup_cws flowchart

3.4.4 delnimmast

This wrapper deletes the NIM master definition, that is, nodes that were configured as NIM master are unconfigure, and the NIM filesets (master and SPOT filesets) are deleted from them.

Command syntax:

delnimmast -l <node_number_list>

where: <node_number_list> is the list of nodes to un-define as NIM master. Node number 0 refers to the CWS.

The general flow and logic for the **delnimmast** command is shown in Figure 3-25.

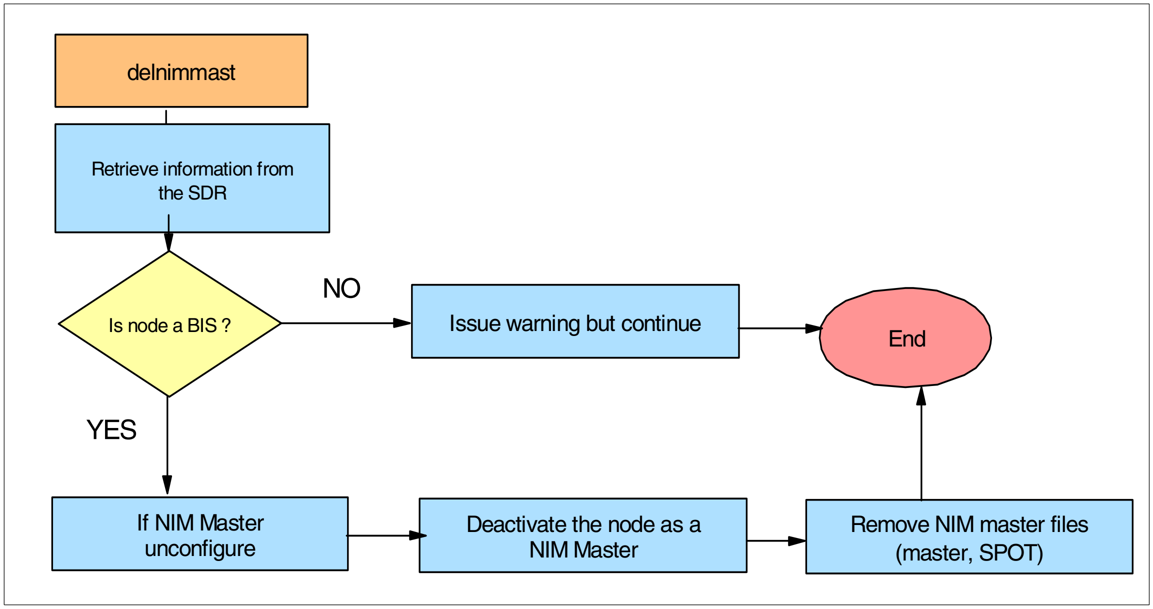

delnimmast
Retrieve information from the SDR
Is node a BIS ?
NO
Issue warning but continue
End
YES
If NIM Master unconfigure
Deactivate the node as a NIM Master
Remove NIM master files (master, SPOT)

Figure 3-25 delnimmast flow chart

3.4.5 delnimclient

The delnimclient script deletes the NIM client definitions from the NIM master. It searches for the nodes passed from the command line, resets these nodes, deallocates the NIM resources that were allocated to them, and then removes their client definitions on the NIM master.

Command syntax:

```
delnimclient -h | -l <node_number_list> | -s <server_node_list>
-h Display command syntax.
-l <node_number_list> - The list of node numbers to remove as NIM clients.
Node number 0 (the CWS) is not allowed.
-s <server_node_list - > The list of server (NIM master) nodes on which to
delete all clients that are no longer defined as boot/install clients in the
SDR.
```

The general flow and logic for the delnimclient script is shown in Figure 3-26 on page 116.

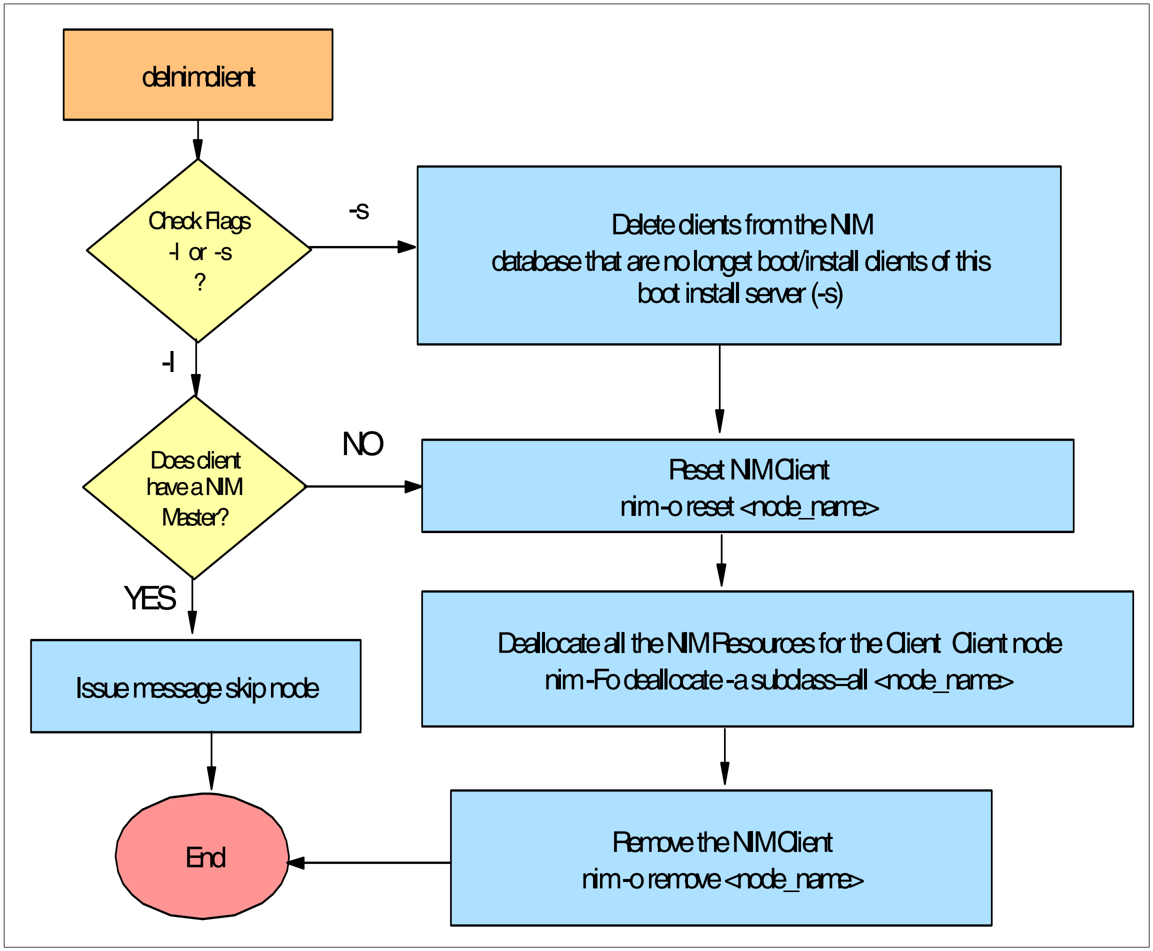

delnimclient
Check Flags
-l or -s
?
-s
Delete clients from the NIM database that are no longet boot/install clients of this boot install server (-s)
-l
Does client have a NIM Master?
NO
Reset NIM Client
nim -o reset <node_name>
YES
Issue message skip node
Deallocate all the NIM Resources for the Client Client node
nim -Fo deallocate -a subclass=all <node_name>
End
Remove the NIM Client
nim -o remove <node_name>

Figure 3-26 delnimclient flowchart

3.4.6 mknimmast

This wrapper creates a NIM master. To do this, the NIM master filesets (bos.sysmgt.nim.master and bos.sysmgt.nim.spot) must be installed. The node is configured as a NIM master by using the **nimconfig** command.

Command syntax:

```
mknimmast -h | -l <node_number_list>
-h Display command syntax.
```

```
-l <node_number_list> - List of node numbers to define as NIM masters. Node
number 0 indicates the CWS.
```

General flow and logic for mknimmast is shown in Figure 3-27.

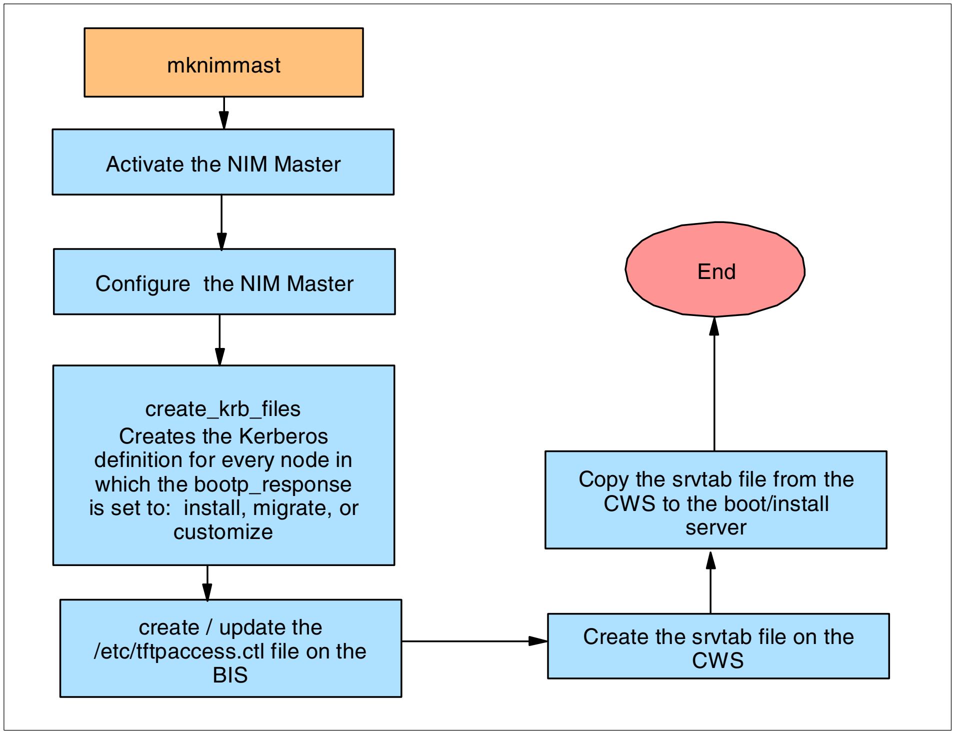

mknimmast
Activate the NIM Master
Configure the NIM Master
create_krb_files
Creates the Kerberos definition for every node in which the bootp_response is set to: install, migrate, or customize
create / update the /etc/tftpaccess.ctl file on the BIS
Create the srvtab file on the CWS
Copy the srvtab file from the CWS to the boot/install server
End

Figure 3-27 mknimmast flowchart

3.4.7 mknimint

This wrapper defines a new Ethernet network and interface objects on the NIM master (boot/install server). On the CWS, any network not previously defined are defined, and NIM interfaces are added. On a boot/install server that is not the CWS, all Ethernet network adapters and interfaces of the node are defined in the NIM master. Also, all CWS Ethernet and Token-Ring adapters and interfaces are defined. The CWS networks and interfaces are added to the boot/install server because there are resources, such as lppsource (which have the CWS as resource server), which the nodes have to access.

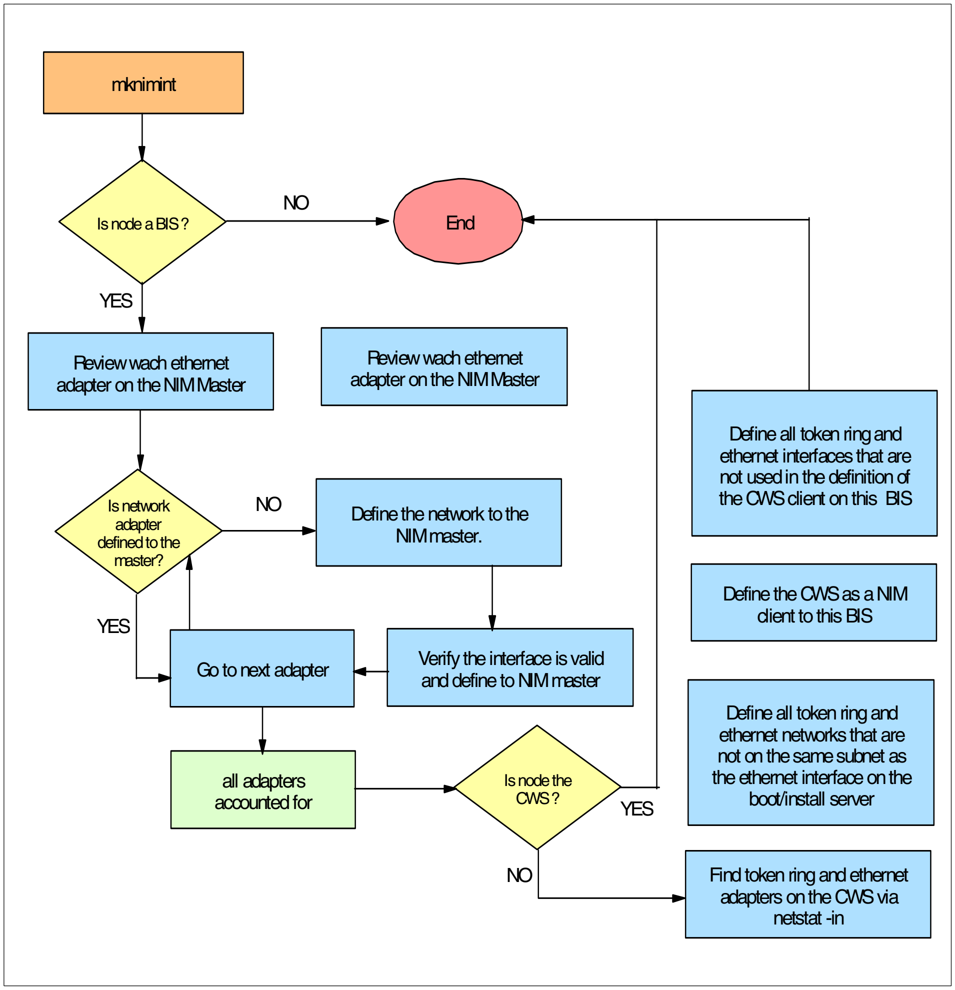
To serve a resource from the CWS to a client that is not on the same subnet as the CWS, routing is required. Routing is done by the mknimclient wrapper.

If mknimint is executed on a boot/install server that is not the CWS, it creates the CWS as a NIM client for the boot/install server, thus, making this server able to access the resources on the CWS.

Command syntax:

```
mknimint -h | -l <node_number_list>
-h Display command syntax
-l <node_number_list>List of node numbers to define as NIM masters.  Node 0 is
the CWS.
```

The general flow and logic for mknimint is shown in Figure 3-28.



mknimint
Is node a BIS ?
NO
End
YES
Review wach ethernet adapter on the NIM Master
Review wach ethernet adapter on the NIM Master
Define all token ring and ethernet interfaces that are not used in the definition of the CWS client on this BIS
Is network adapter defined to the master?
NO
Define the network to the NIM master.
Define the CWS as a NIM client to this BIS
YES
Go to next adapter
Verify the interface is valid and define to NIM master
Define all token ring and ethernet networks that are not on the same subnet as the ethernet interface on the boot/install server
all adapters accounted for
Is node the CWS ?
YES
NO
Find token ring and ethernet adapters on the CWS via netstat -in

Figure 3-28   mknimint flowchart

3.4.8 mknimres

This wrapper creates NIM resources that are needed for operations on the nodes. Depending on the value of the bootp_response field from SDR, the resources that are needed in the processes with the given bootp_response, and do not currently exist, will be created.

mknimres checks the Rstate attribute of resource objects, and if it is not ready for use, they will be removed (exception on boot resource).

Command syntax:

```
mknimres -h | -l <node_number_list>
-h Display command syntax.
-l <node_number_list> List of node numbers to be defined as NIM masters. Node
0 is the CWS.
```

The general flow and logic for mknimres is shown in Figure 3-29.

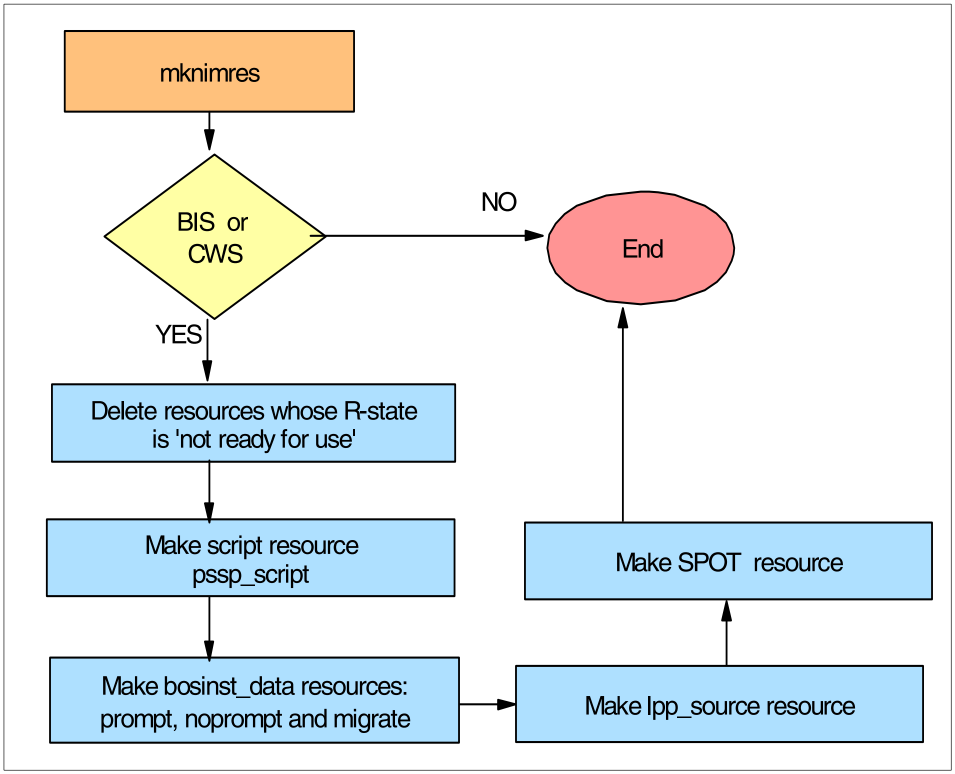

mknimres
BIS or CWS
NO
End
YES
Delete resources whose R-state is 'not ready for use'
Make script resource pssp_script
Make bosinst_data resources: prompt, noprompt and migrate
Make lpp_source resource
Make SPOT resource

Figure 3-29 mknimres flowchart

3.4.9 mknimclient

This wrapper creates NIM client definitions on the boot/install server. It searches for the processor type of a client (UP/MP) and the platform type (RS6K/chrp). The information about processor type and platform is used in the definition of the client, so that when creating the boot images for the nodes, it can construct the correct one. If the client node is not on the same subnet as the CWS, the mknimclient command builds the client routes to the CWS so that the node can use the lppsource.

Command syntax:

```
mknimclient -h | -l <node_number_list>
-h Display command syntax.
-l <node_number_list> List of node numbers to define as NIM clients. Node 0
(the CWS) is not allowed.
```

The general flow and logic for mknimclient is shown in Figure 3-30.

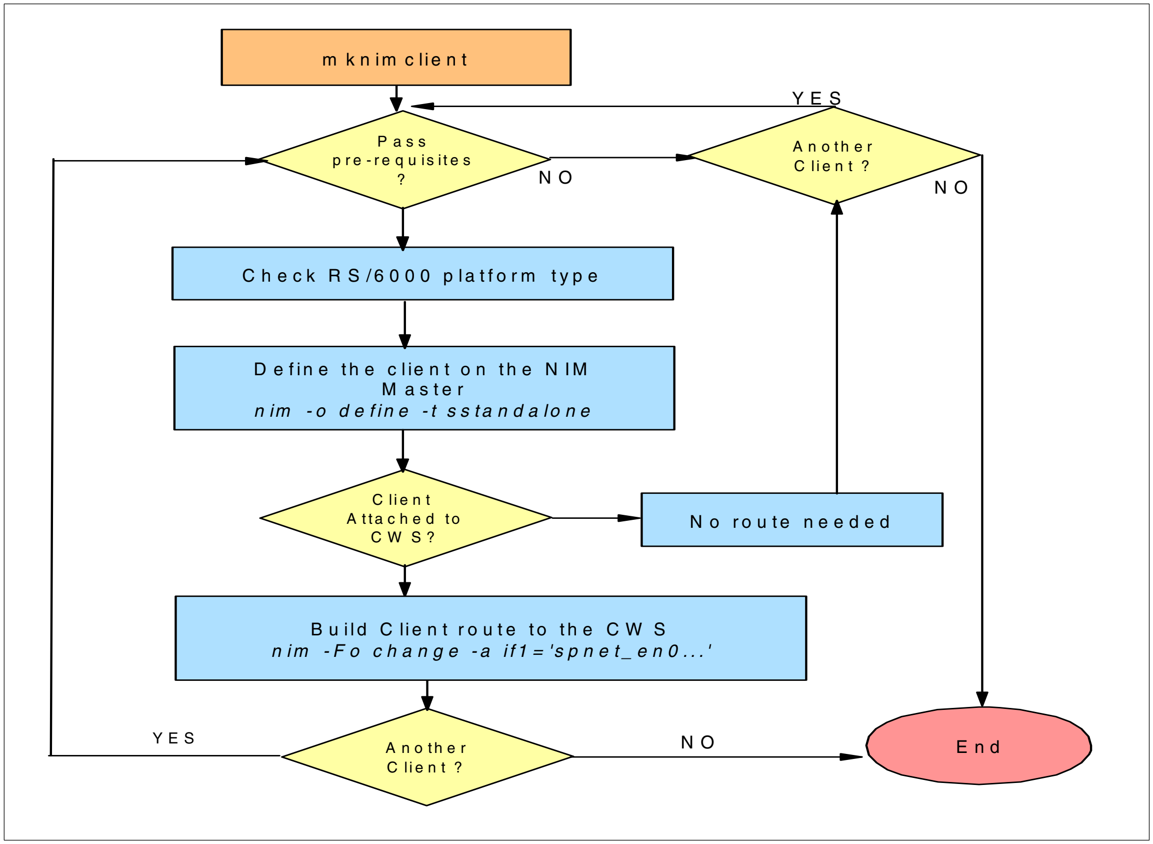

mknimclient
YES
Pass pre-requisites ?
NO
Another Client ?
NO
Check RS/6000 platform type
Define the client on the NIM Master
nim -o define -t sstandalone
Client Attached to CWS?
No route needed
Build Client route to the CWS
nim -Fo change -a if1='spnet_en0...'
YES
Another Client ?
NO
End

Figure 3-30 mknimclient flowchart

3.4.10 mkconfig

This wrapper creates the /tftpboot/<reliable_hostname>.config_info files for every node that has bootp_response not set to disk. These files are used when running pssp_script. The information is retrieved from the SDR for every node with bootp_response not set to disk.

This data is used in creating the config_info files. This mkconfig wrapper uses reliable hostnames for creation of the files. The general flow and logic for the mkconfig wrapper is shown in Figure 3-31.

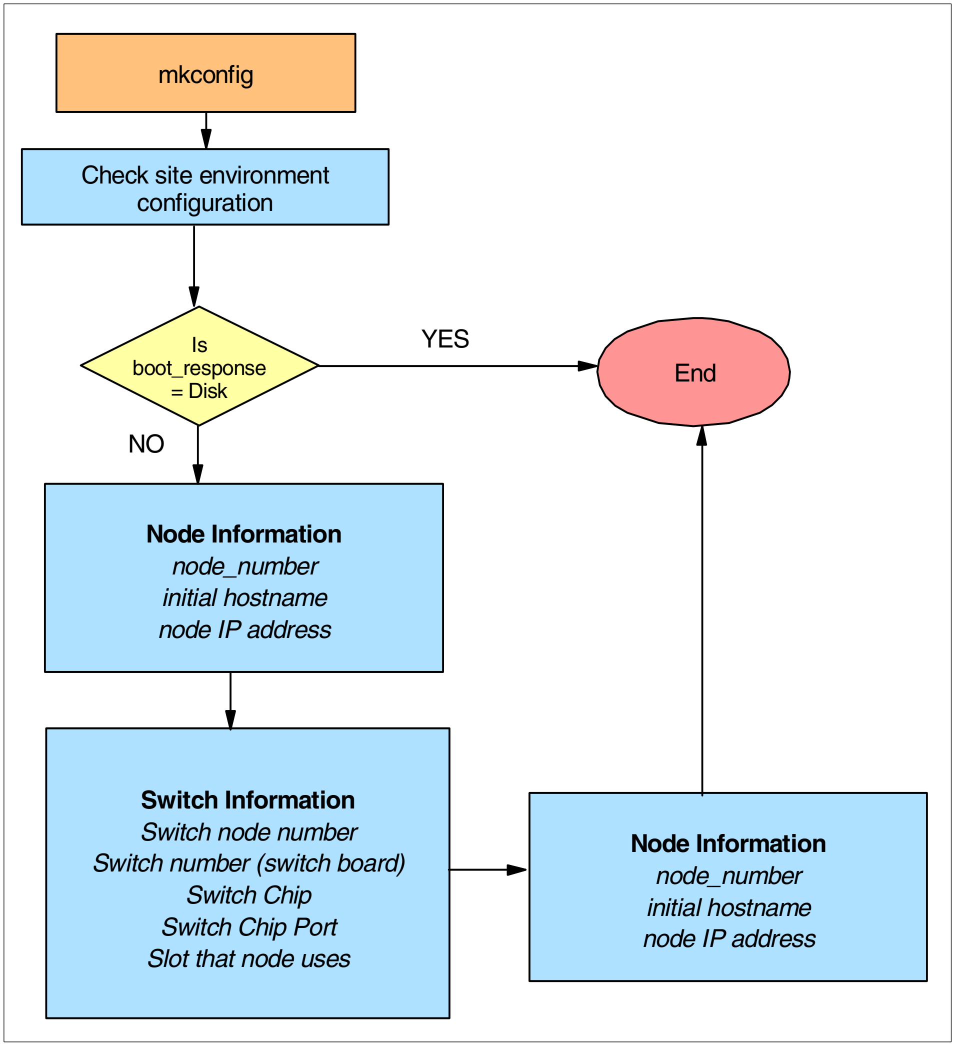


Figure 3-31 mkconfig flowchart

3.4.11 mkinstall

This wrapper creates the /tftpboot/<hostname>.install_info file for each node with bootp_response not set to disk. These files, like the config_info files, are used by pssp_script. The mkinstall wrapper retrieves site environment data from the SP SDR class. It gets the node information, and if the node bootp_response is not set to disk, the existing /tftpboot/<hostname>.install_info file is removed and a new install_info file is created for the node.

The general flow and logic for the mkinstall wrapper is shown in Figure 3-32.

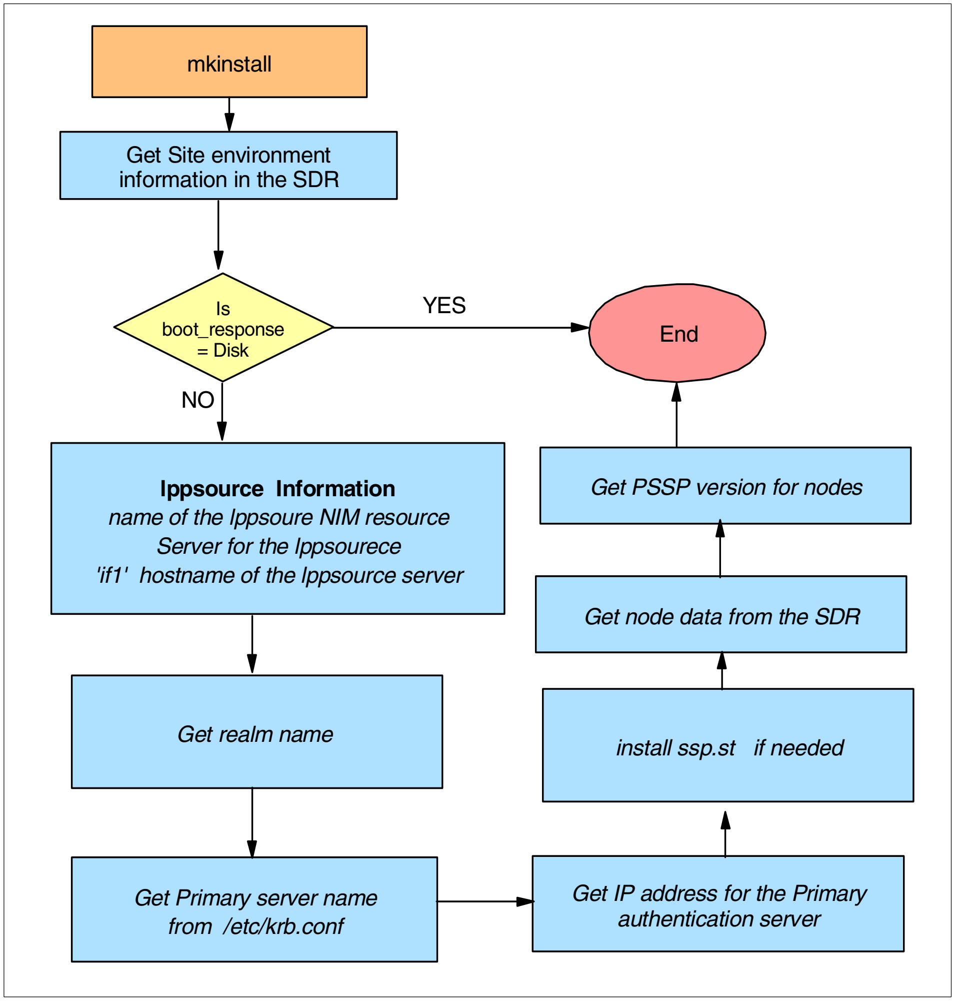

mkinstall
Get Site environment information in the SDR
Is boot_response = Disk
YES
End
NO
Ippsource Information
name of the Ippsoure NIM resource
Server for the Ippsourece
'if1' hostname of the Ippsource server
Get realm name
Get Primary server name from /etc/krb.conf
Get IP address for the Primary authentication server
install ssp.st if needed
Get node data from the SDR
Get PSSP version for nodes

Figure 3-32 mkinstall flowchart

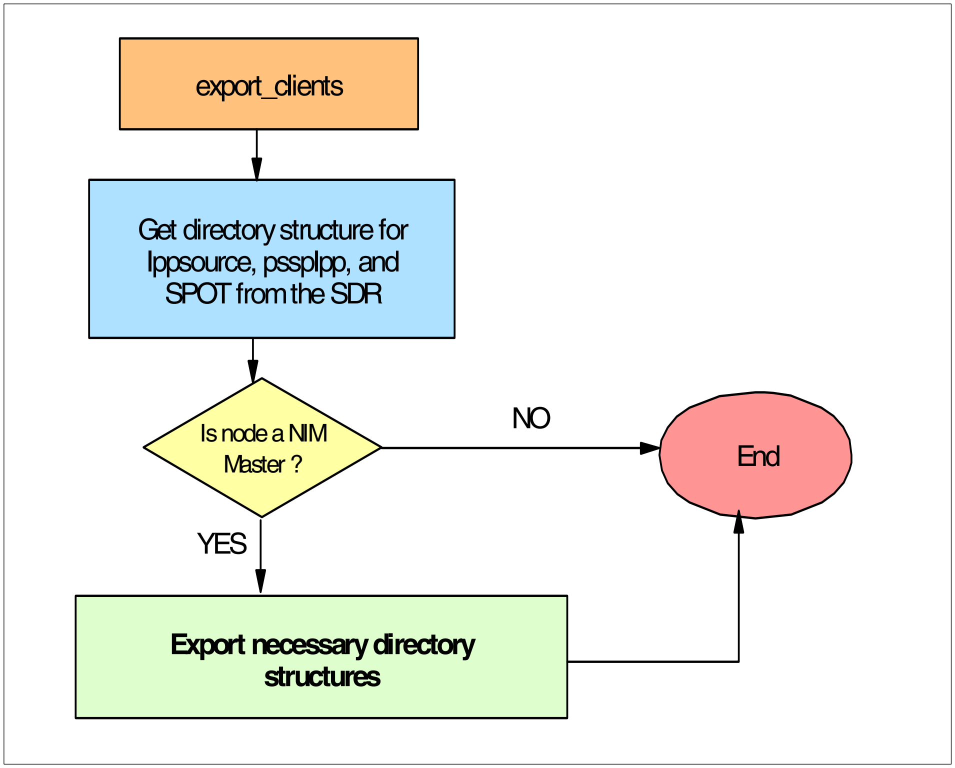
3.4.12 export_clients

This wrapper ensures that the required directories (pssplpp, images, lppsource, SPOT, and others) are exported from a NIM master to its clients. This command must be run on the NIM master.

Command syntax:

```
export_clients [-h]
This command has no input parameters.
```

The general flow and logic for export_clients is shown in Figure 3-33.



export_clients
Get directory structure for lppsource, pssplpp, and SPOT from the SDR
Is node a NIM Master ?
NO
End
YES
Export necessary directory structures

Figure 3-33 export_clients flowchart

3.4.13 allnimres

This wrapper allocates all necessary NIM resources to a client, depending on the client bootp_response in the SDR. This includes executing the **bos_inst** command for allocation of the boot and nimscript resources. When this command is done, nodes are ready for netboot/install, diagnostics, or maintenance. If a node bootp_response is set to “disk” or “customize”, then all NIM resources are deallocated from the node. The general flow and logic for the allnimres wrapper is shown in Figure 3-34 on page 125. The bootp_response values available in PSSP are:

- install
- customize
- disk
- maintenance
- diag
- migrate

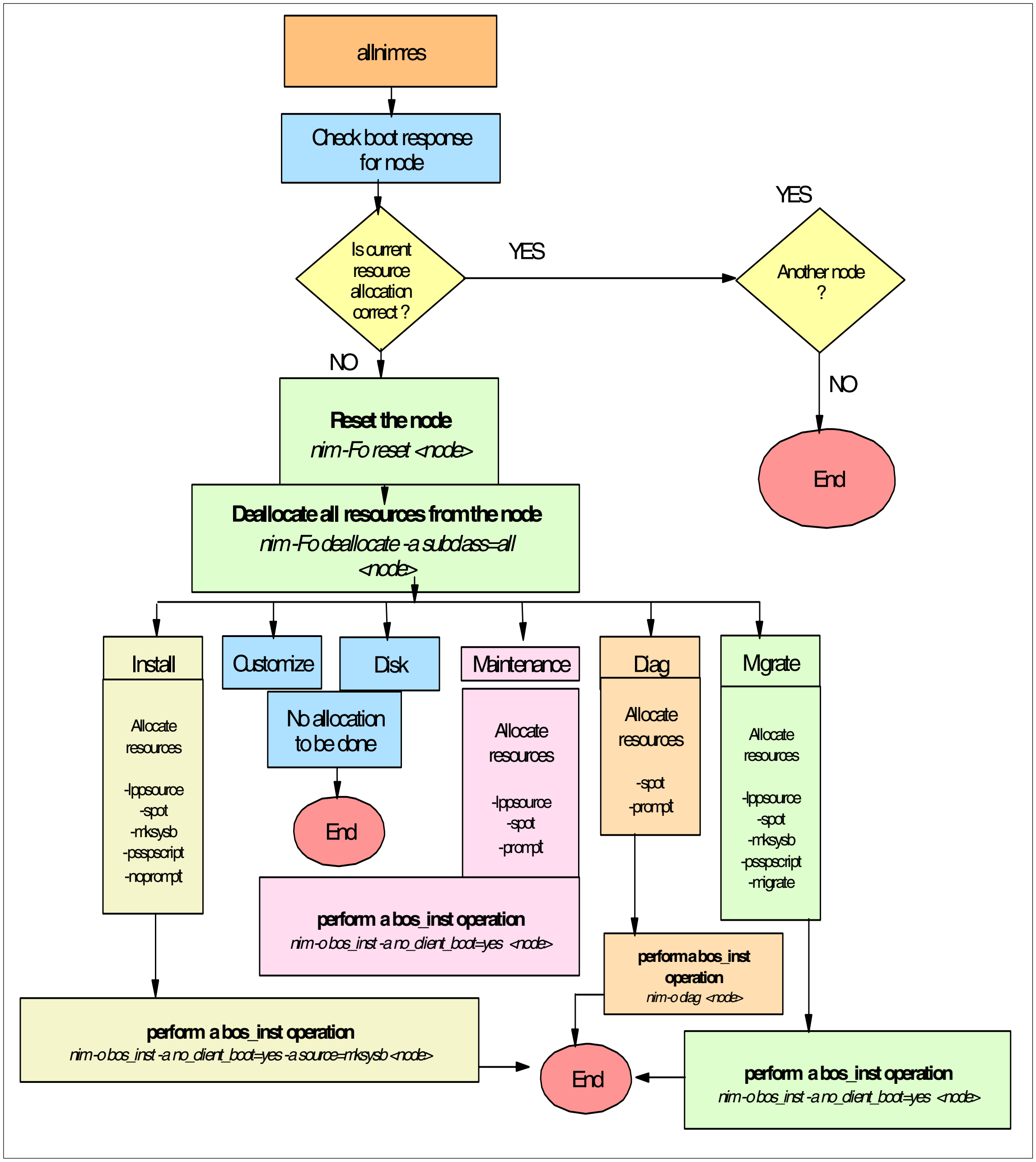

allnimres
Check boot response for node
Is current resource allocation correct ?
YES
NO
Another node ?
YES
NO
End
Reset the node
nim -Fo reset <node>
Deallocate all resources from the node
nim -Fo deallocate -a subclass=all <node>
Install
Allocate resources
-lppsource
-spot
-mksysb
-pspscript
-noprompt
Customize
Disk
No allocation to be done
End
Maintenance
Allocate resources
-lppsource
-spot
-prompt
Diag
Allocate resources
-spot
-prompt
Migrate
Allocate resources
-lppsource
-spot
-mksysb
-pspscript
-migrate
perform a bos_inst operation
nim -o bos_inst -a no_client_boot=yes <node>
perform a bos_inst operation
nim -o diag <node>
perform a bos_inst operation
nim -o bos_inst -a no_client_boot=yes -a source=mksysb <node>
End
perform a bos_inst operation
nim -o bos_inst -a no_client_boot=yes <node>

Figure 3-34 allnimres flowchart

3.4.14 unallnimres

This wrapper deallocates all NIM resources to a list of boot/install clients. The general flow and logic for unallnimres is shown in Figure 3-35.

Command syntax:

```
unallnimres -l <list_of_nodes>
```

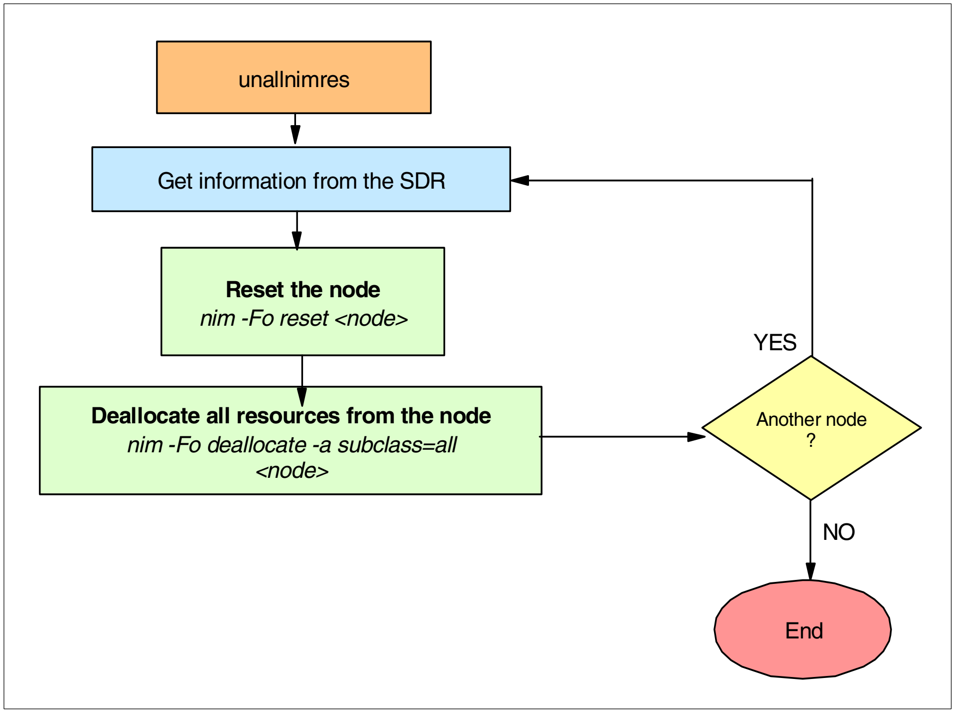

unallnimres
Get information from the SDR
Reset the node
nim -Fo reset <node>
Deallocate all resources from the node
nim -Fo deallocate -a subclass=all <node>
YES
Another node ?
NO
End

Figure 3-35 unallnimres flow chart

3.4.15 Site Environment errors

The following section describes possible site environment errors in your cluster.

Node data information invalid

Prior to running setup_server, all nodes which are physicaly installed into the system must have valid entries in the SDR.

Take, for example, a case in which the node information has been entered into the SDR for nodes 5, 9, and 13. However, in this specific example, the SDR information has not yet been entered for node1.

This configuration will lead to an immediate failure in setup_server shown in Example 3-1.

Example 3-1 Reliable hostname not configured in the SDR

```
# setup_server
setup_server: 0016-014 Problem found while querying SDR for reliable
hostnames.  SDR Return Code 2.
setup_server: Processing incomplete (rc= 2).
```

Check the reliable hostname in the SDR with the **splstdata** command as shown in Example 3-2.

Example 3-2 Check the reliable_hostname

```
# splstdata -n
                  List Node Configuration Information

node# frame# slot# slots initial_hostname  reliable_hostname
----- ------ ----- ----- ----------------- -----------------
1      1     1     4          ““““            ““““
5      1     5     4         sp5n05          sp5n05
9      1     9     4         sp5n09          sp5n09
```

Note that there are no entries for the initial_hostname or the reliable_hostname for node1. Correct this by entering the information in the SDR.

Another problem that is encountered when node data is not entered into the SDR is shown in Example 3-3.

Example 3-3 setup_server fails

```
mknimres: Copying /usr/lpp/ssp/install/config/bosinst_data_migrate.template to
/spdata/sys1/install/pssp/bosinst_data_migrate.
0042-001 nim: processing error encountered on "master":
   0042-001 m_mk_lpp_source: processing error encountered on "master":
   0042-154 c_stat: the file or directory

 "/spdata/sys1/install/default/lppsource" does not exist

mknimres: 0016-375 The creation of the lppsource resource named
lppsource_default
 had a problem with return code 1.
setup_server: 0016-279 Problem of internally called command:
/usr/lpp/ssp/bin/mknimres; rc= 2.
Tickets destroyed.
setup_server: Processing incomplete (rc= 2).
```

Once again there is incomplete information in the SDR for node 1. After the **spframe** command is run, the lppsource placed in the SDR is the default. However, in this installation, the lppsource directory is under a subdirectory in /spdata/sys1/install/aix433. To correct this situation, enter the appropriate lppsource information for node1 into the SDR and then re-run setup_server.

PSSP filesets missing

The mknimres script ensures that the pssp files exist in /spdata/sys1/install/pssplpp/PSSP-3.2 directory. See Example 3-4 for the output of the error that occurs in setup_server when these files do not exist.

Example 3-4 Sample error output from setup_server

```
mknimres: Successfully created the bosinst_data resource named prompt on
mknimres: Successfully created the lpp_source resource named mknimres:
Successfully created the mksysb resource named mksysb_1 from dir
 /spdata/sys1/install/images/bos.obj.ssp.433 on sp5en0.
mknimres: 0016-395 Could not get size of pssplpp PSSP-3.2
 files on control workstation, return code of 1, missing files are
/spdata/sys1/install/pssplpp/PSSP-3.2/pssp.installp.
setup_server: 0016-279 Problem of internally called command:
/usr/lpp/ssp/bin/mknimres; rc= 2.
Tickets destroyed.
setup_server: Processing incomplete (rc= 2).
```

When this error occurs, first check for the presence of this file:

```
   # cd /spdata/sys1/install/pssplpp/PSSP-3.2
```

Example 3-5 Checking for the missing files

```
   # ls
   .toc
   (3) rsct.clients.1.2.0.0.I
   (1) ssp.3.2.0.0.I
   ssp.vsdgui.3.2.0.0.I
   ptpe.3.2.0.0.I
   (4) rsct.core.1.2.0.0.I
   ssp.hacws.3.2.0.0.I
   vsd.3.2.0.0.I
   (2) rsct.basic.1.2.0.0.I
   spimg.3.2.0.0.I
   ssp.ptpegui.3.2.0.0.I
```

In Example 3-5, the output of the **ls** command shows that four files are missing from this directory structure. The four files have been numbered in Example 3-5 to illustrate our discussion. These files are:

1. pssp.installp

2. rsct.basic
3. rsct.clients
4. rsct.core

Attention: The four files are not quite missing. The files are there, but they first need to be renamed by following step 15.2 in Chapter 2 of the *Parallel System Support Programs for AIX Version 3 Release 2: Installation and Migration Guide*, GA22-7347.

Invalid hardware address for a node

Each node must have a valid hardware address in the SDR. setup_server does not check to ensure that the address is the actual address on the node but rather that the address is within valid parameters. Example 3-6 shows the output of setup_server on a system with such a configuration. To correct this problem, acquire the hardware address for the node in question and re-run setup_server (see step 35 in the *Installation and Migration Guide for PSSP 3.2*, GA22-7347).

Example 3-6 Invalid hardware address for a node

```
setup_server: Running services_config script to configure SSP services.This may
take a few minutes...
0513-044 The /usr/sbin/xntpd Subsystem was requested to stop.
rc.ntp: Starting NTP daemon (xntpd)
mknimres: Copying /usr/lpp/ssp/install/config/bosinst_data_prompt.template to
/spdata/sys1/install/pssp/bosinst_data_prompt.
mknimres: Copying /usr/lpp/ssp/install/config/bosinst_data_migrate.template to
/spdata/sys1/install/pssp/bosinst_data_migrate.
mknimres: Creating the SPOT resource spot_aix433.
mknimres: The SPOT resource named spot_aix433 was successfully created on
sp5en0.
mknimclient: 0016-416 The SDR contains an invalid hardware Ethernet address for
node 1 (sp5n01). Correct the value using the sphrdwrad comma
nd and try again.
```

3.5 Node installation

The following section describes boot install server (BIS) installation as well as problems encountered while installing nodes.

3.5.1 Optional boot install server installation

When creating a boot install server (BIS), it is important to remember that the BIS network is to be an independent network from the spnet_en0 network as shown in Example 3-36 on page 131. The BIS must also act as a router between these two networks.

The procedure to define a node as a BIS is accomplished by defining the node as the BIS for another node with the **spchvgobj** command. After changing the boot install server for the node, the boot response should be set to either install or customize. After this scenario is set up and setup_server is run on the CWS, the node will be defined as a boot install server.

This procedure will also require that setup_server be run on the node being defined as the BIS.

When problems occur during the installation over a BIS, verify the following has been completed:

- BIS is a TCP/IP Router.
- IP forwarding is set to on BIS nodes.
- Route has been established from the CWS to nodes that will be installed via the BIS.
- Make sure that rootvg has enough space for the necessary file systems setup_server will create on the BIS.

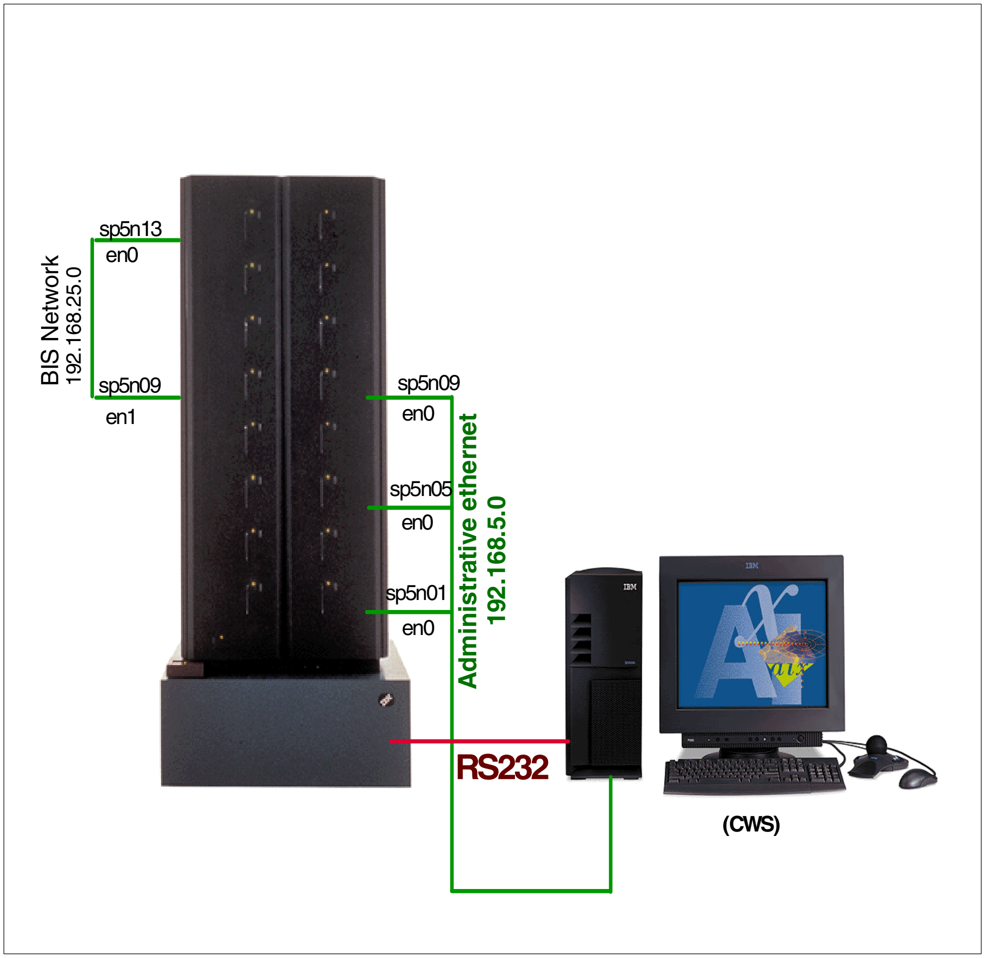


Figure 3-36 Boot install server

The trouble with tuples

When setup_server ran on the BIS, an error related to the incorrect number of tuples was encountered, as shown in Figure 3-37 on page 132.

```
sp5n09: cp: //.rhosts.nim: No such file or directory
sp5n09:
sp5n09: mknimint: 0016-201 There is an incorrect number of tuples in the IP
address.
sp5n09: mknimint: 0016-201 There is an incorrect number of tuples in the IP
address.
sp5n09: setup_server: 0016-279 Problem of internally called command:
/usr/lpp/ssp/bin/mknimint; rc= 2.
spbootins:  0022-038 There was a non-zero return code from issuing rsh
setup_server to node 9 .
sp5n05: 0513-044 The /usr/sbin/xntpd Subsystem was requested to stop.
sp5n05: rc.ntp: Starting NTP daemon (xntpd)
sp5n05: 0513-059 The xntpd Subsystem has been started. Subsystem PID is
10238.
```

Figure 3-37 The trouble with tuples

This error references the IP address and subnet of the ethernet and token ring adapters on the node that is becoming the BIS, as well as the CWS. To correct this problem, check:

1. IP addresses are valid for each interface on the BIS and CWS.
2. Subnets are valid on both the CWS and BIS.
3. lsattr -El (en0, en1, en1, tr0, etc...).
4. odmget -q name=en0 CuAt.
5. Remember to check for every adapter that shows up in netstat that is either an en(x) or a tr(x) distinction.

File systems place on the BIS Node

When the BIS node runs setup_server the wrapper mknimres creates three file systems that are placed in rootvg. Plan to have the necessary disk space available to rootvg to house the AIX lppsource, spot(s),and pssplpp files. If the space is not available, mknimres will display an error. This space must be in rootvg and not placed into a separate file system. Do not make these file systems on your own or mknimres will fail to create the necessary file systems due to the mount point already existing. Figure 3-38 on page 133 shows the logical volumes and file systems created on the BIS.

```
logical volumes mknimres creates on the non-cws boot install server (BIS)
/dev/install_images
/dev/install_pssplpp
/dev/spot_aix433

Filesystems mknimres create on the non-cws boot install server (BIS)
/spdata/sys1/install/images
/spdata/sys1/install/pssplpp
/spdata/sys1/install/aix433/spot
```

Figure 3-38 Logical volumes and filesystems created on the BIS

3.5.2 Network interface not supported when allocating SPOT

This error, as shown in Example 3-39, indicates that the spot has no information for the network adapter on the node. This is not always a true statement and can sometimes be rectified by resetting the SPOT.

```
export_clients: File systems exported to clients from server node 9.
0042-001 nim: processing error encountered on "master":
   0042-058 m_alloc_spot: unable to allocate "spot_aix433" to "sp5e1n13"
        because it does not support the network interface type
        of that client
```

Figure 3-39 Network interface not supported

Figure 3-40 conducts a forced reset of the SPOT with the spot_name being spot_aix433.

```
 # nim -Fo reset spot_aix433
```

Figure 3-40 Nim reset

If the force reset of the SPOT does not clear this error, then you will need to verify that the lpp for the adapter is actually in the SPOT lppsource. If it is not, then you can either add it to the lppsource directory and recreate the SPOT, or update the spot with the additional lpp files needed.

When setup_server runs on the new BIS, it creates NIM classes on the node for the new NIM master and its clients. Output of the **lsnim -l** shows that a new network interface, if2, is defined as shown in Figure 3-41. The reason for this is that the BIS needs to be able to route information from the CWS to the boot install server's client nodes.

```
# lsnim -l
master:
   class               = machines
   type                = master
   comments            = machine which controls the NIM environment
   platform            = rs6k
   netboot_kernel      = up
   if1                 = spnet_en0 sp5n09 02608C2D63E2
   if2                 = spnet_en1 sp5e1n09 02608c2d08d7 ent
   cable_type1         = bnc
   cable_type2         = bnc
   Cstate              = ready for a NIM operation
   prev_state          = ready for a NIM operation
   Mstate              = currently running
   serves              = boot
   serves              = nim_script
   master_port         = 1058
   registration_port   = 1059
   reserved            = yes
   max_nimesis_threads = 20
```

Figure 3-41 Output of lsnim -l

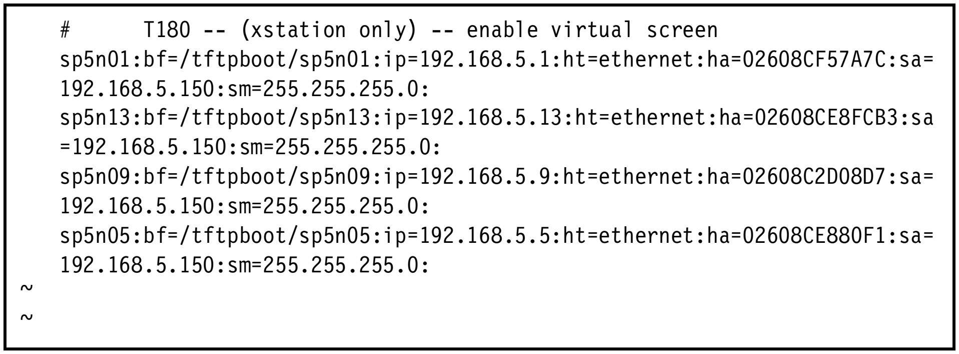
3.5.3 LED 231 - 239 in nodecond and ping test successful

The following section describes what LED 231 - 239 means when working on your cluster. Figure 3-42 shows BOOTP hangs while installing a node.

```
STARTING SYSTEM (BOOT)
Booting . . .  Please wait.
Ethernet:  Slot 0/3, BNC / modular Jacks
Hardware address .....................02608C2F9083
             Packets Sent        Packets Received

BOOTP             00006               00000
```

Figure 3-42 BOOTP hangs on node install

If the BOOTP hangs, check the /etc/bootptab file as shown in Figure 3-43.



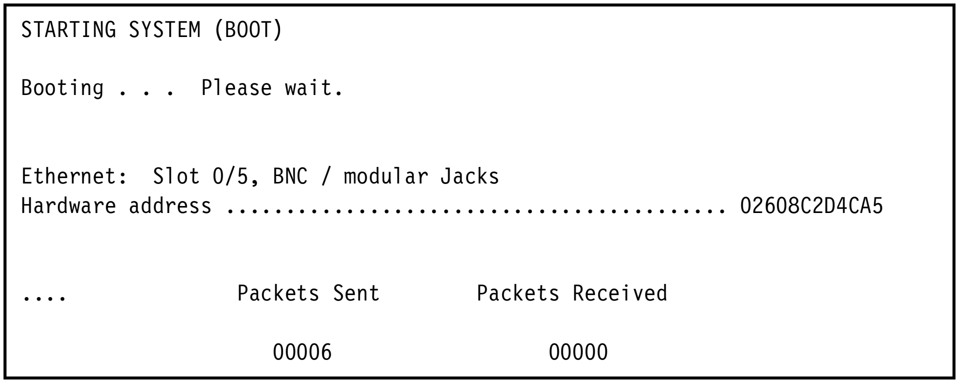
```
   #      T180 -- (xstation only) -- enable virtual screen
   sp5n01:bf=/tftpboot/sp5n01:ip=192.168.5.1:ht=ethernet:ha=02608CF57A7C:sa=
   192.168.5.150:sm=255.255.255.0:
   sp5n13:bf=/tftpboot/sp5n13:ip=192.168.5.13:ht=ethernet:ha=02608CE8FCB3:sa
   =192.168.5.150:sm=255.255.255.0:
   sp5n09:bf=/tftpboot/sp5n09:ip=192.168.5.9:ht=ethernet:ha=02608C2D08D7:sa=
   192.168.5.150:sm=255.255.255.0:
   sp5n05:bf=/tftpboot/sp5n05:ip=192.168.5.5:ht=ethernet:ha=02608CE880F1:sa=
   192.168.5.150:sm=255.255.255.0:
~
~
```

Figure 3-43 Checking /etc/bootptab file

Check the /etc/bootptab file for potential problems when node installation cycles at LED code 239 and 231. These LEDs usually indicate a network problem. However, if a new network card is being used for the en0 interface, then there is a chance that the bootptab file did not get updated with the new information. At the end of this file, each node is listed along with their boot parameters.

Under each node, there is a hardware address listed. Notice that, in Figure 3-43, the hardware address (ha) is 02608CE880F1.

In comparison, Figure 3-44 shows that the hardware address that is being used is 02608C2d4CA5.

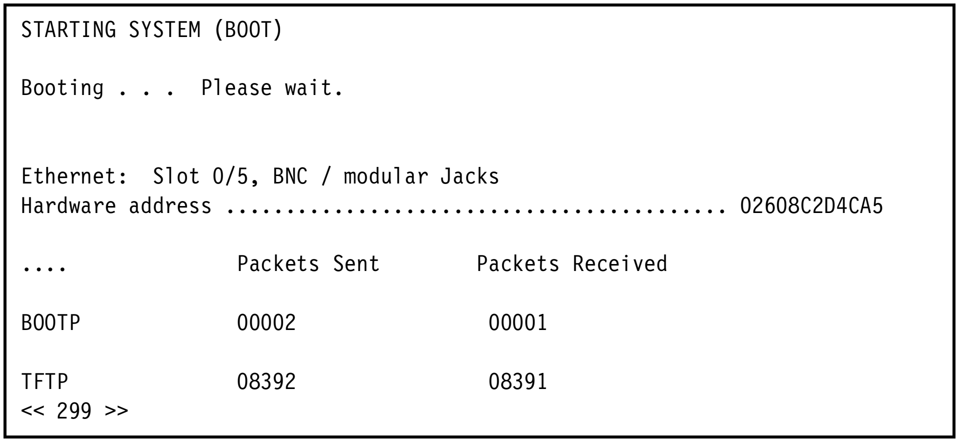


```
STARTING SYSTEM (BOOT)

Booting . . .  Please wait.


Ethernet:  Slot 0/5, BNC / modular Jacks
Hardware address ............................... 02608C2D4CA5


....             Packets Sent       Packets Received

                   00006                00000
```

Figure 3-44 BOOTP packets being sent

A quick check of the actual hardware adapter shows that slot 0/5 is the correct slot. Therefore, the /etc/bootptab file has the wrong information.

Manually update this file to reflect the correct hardware address and restart the node installation. The node should then be able to receive the BOOTP packets as shown in Figure 3-45.



```
STARTING SYSTEM (BOOT)

Booting . . .  Please wait.


Ethernet:  Slot 0/5, BNC / modular Jacks
Hardware address ......................................... 02608C2D4CA5

....              Packets Sent        Packets Received

BOOTP             00002                00001

TFTP              08392                08391
<< 299 >>
```

Figure 3-45 BOOTP success

Tip: Updating the /etc/bootptab file will not solve this problem permanently. setup_server will recreate this file the next time the node is to be installed by obtaining the incorrect MAC address from the SDR. A more permanent solution is to update the SDR with the correct address and also update the NIM definition.

3.5.4 Manual node conditioning

The following example is for node 9 in frame 1:

1. Power the node off by invoking:

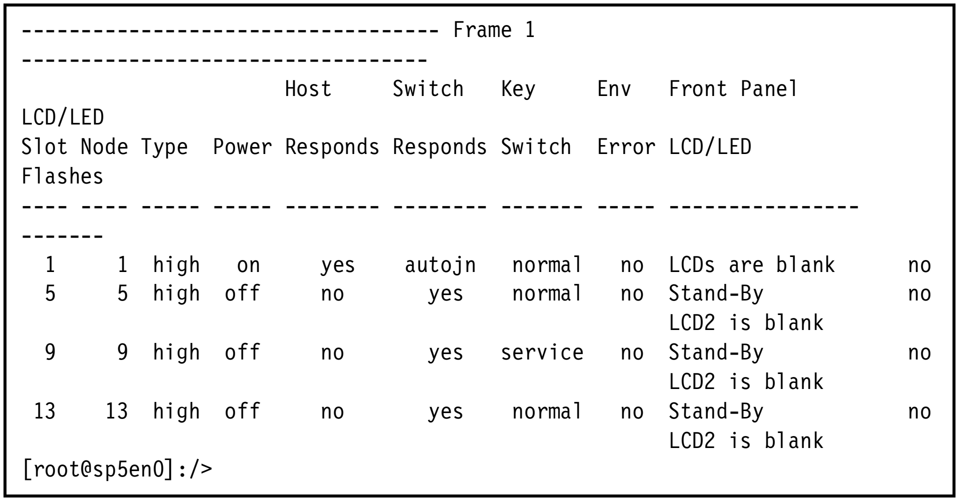
   ```
   spmon -p off node9
   ```

2. Set the key to service by invoking:

   ```
   spmon -k service node9
   ```

3. Check to ensure that the key is in service position by issuing (see Figure 3-46 on page 137):

   ```
   spmon -d
   ```



```
----------------------------------- Frame 1
----------------------------------
                      Host     Switch   Key     Env   Front Panel
LCD/LED
Slot Node Type  Power Responds Responds Switch  Error LCD/LED
Flashes
---- ---- ----- ----- -------- -------- ------- ----- ----------------
-------
  1     1  high   on     yes    autojn   normal   no  LCDs are blank      no
  5     5  high  off     no       yes    normal   no  Stand-By            no
                                                      LCD2 is blank
  9     9  high  off     no       yes   service   no  Stand-By            no
                                                      LCD2 is blank
 13    13  high  off     no       yes    normal   no  Stand-By            no
                                                      LCD2 is blank
[root@sp5en0]:/>
```

Figure 3-46 Nodes installed but no host responds

4. Power on the node

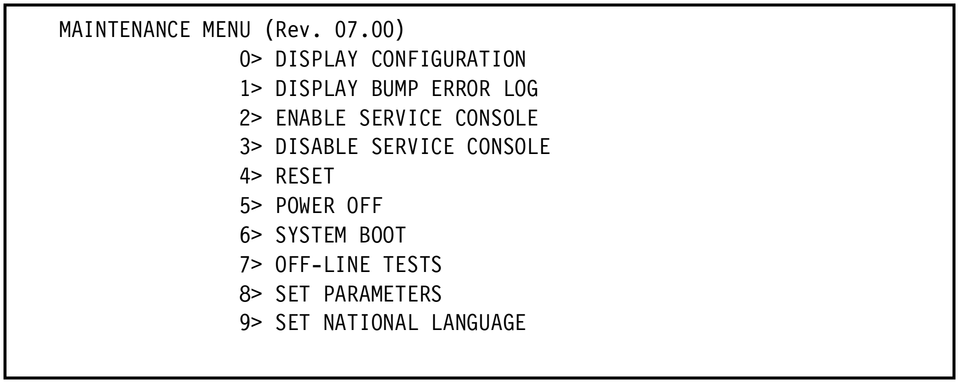
   `spmon -p on [targetnode]`

5. Open a writable tty connection to the node by executing:

   `s1term -w 1 9`

6. Wait for the MAINTENANCE MENU as shown in Figure 3-47.

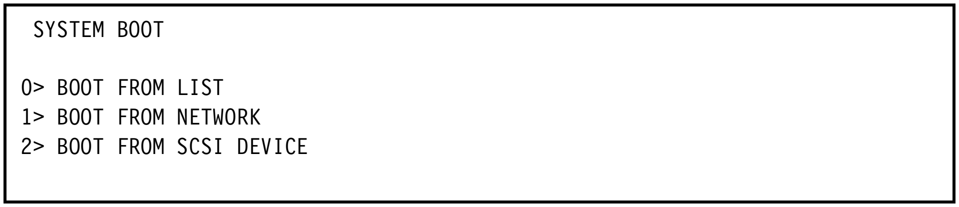
Note: The maintenance menu looks different for CHRP (Common Hardware Reference Platform) and RS/6000 MCA nodes.



```
   MAINTENANCE MENU (Rev. 07.00)
                  0> DISPLAY CONFIGURATION
                  1> DISPLAY BUMP ERROR LOG
                  2> ENABLE SERVICE CONSOLE
                  3> DISABLE SERVICE CONSOLE
                  4> RESET
                  5> POWER OFF
                  6> SYSTEM BOOT
                  7> OFF-LINE TESTS
                  8> SET PARAMETERS
                  9> SET NATIONAL LANGUAGE
```

Figure 3-47 Maintenance menu

7. Select option 6 SYSTEM BOOT.
8. Select option 1 from the SYSTEM BOOT menu as shown in Figure 3-48 on page 138.

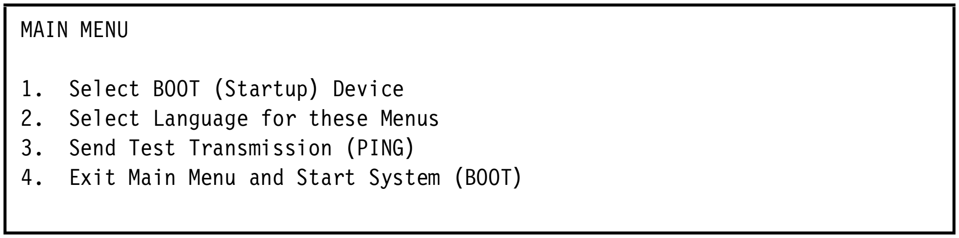


```
 SYSTEM BOOT

0> BOOT FROM LIST
1> BOOT FROM NETWORK
2> BOOT FROM SCSI DEVICE
```

Figure 3-48 Boot from the network

9. This will bring up the system boot menu as shown in Figure 3-49.

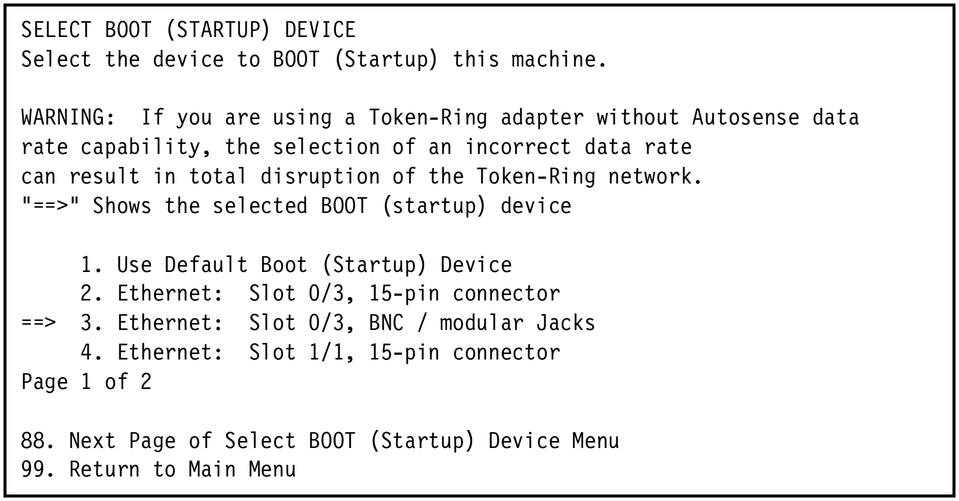
   Choose option 1 to `Select BOOT (Startup) Device`.



```
MAIN MENU

1.  Select BOOT (Startup) Device
2.  Select Language for these Menus
3.  Send Test Transmission (PING)
4.  Exit Main Menu and Start System (BOOT)
```

Figure 3-49 Conducting the ping test

10. Select the BOOT (Startup) Device as in Figure 3-50.

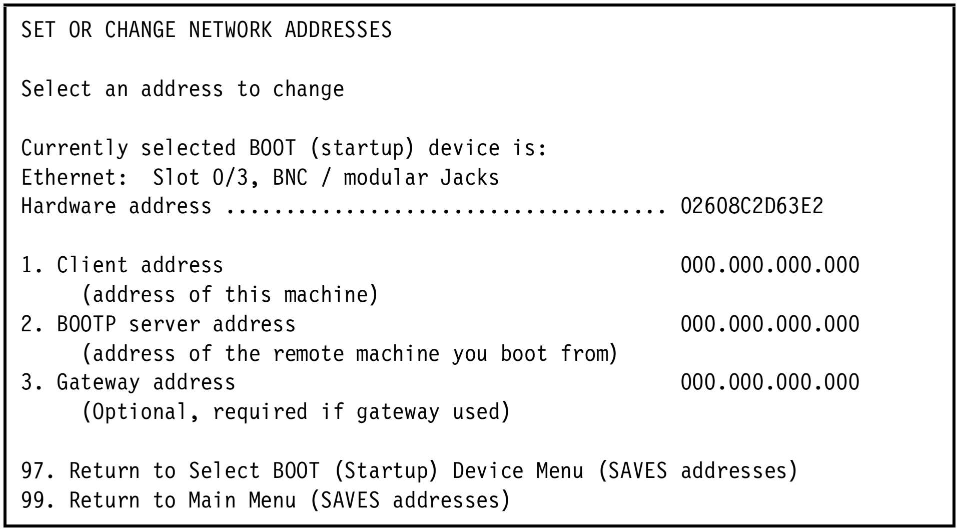


```
SELECT BOOT (STARTUP) DEVICE
Select the device to BOOT (Startup) this machine.

WARNING:  If you are using a Token-Ring adapter without Autosense data
rate capability, the selection of an incorrect data rate
can result in total disruption of the Token-Ring network.
"==>" Shows the selected BOOT (startup) device

     1. Use Default Boot (Startup) Device
     2. Ethernet:  Slot 0/3, 15-pin connector
==>  3. Ethernet:  Slot 0/3, BNC / modular Jacks
     4. Ethernet:  Slot 1/1, 15-pin connector
Page 1 of 2

88. Next Page of Select BOOT (Startup) Device Menu
99. Return to Main Menu
```

Figure 3-50   Select the boot device

11. In Figure 3-50 on page 139, we select option 3 to use the Ethernet adapter in Slot 0/3:

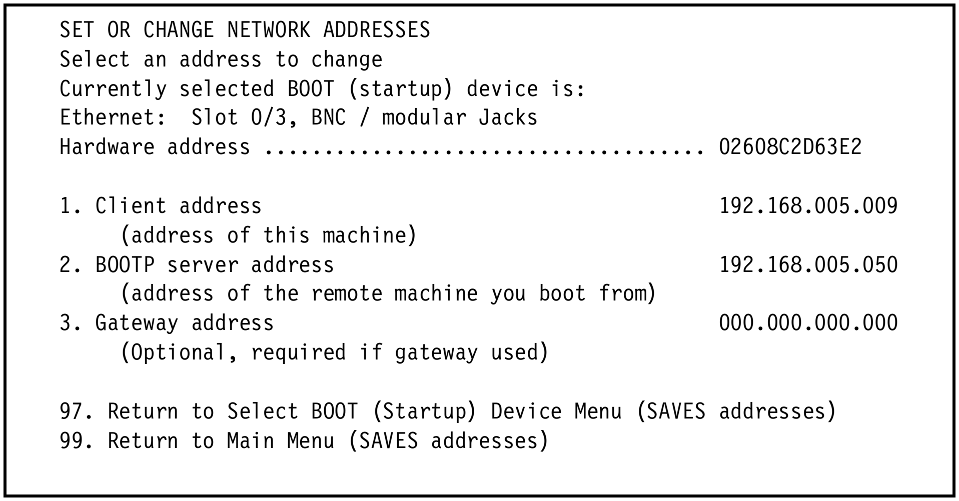
   - This will prompt you to set or change network addresses as shown in Figure 3-51.
   - Enter in the appropriate information for the client and the server's IP address. It should not be necessary to enter a gateway address.



```
SET OR CHANGE NETWORK ADDRESSES

Select an address to change

Currently selected BOOT (startup) device is:
Ethernet:  Slot 0/3, BNC / modular Jacks
Hardware address ..................................... 02608C2D63E2

1. Client address                                      000.000.000.000
     (address of this machine)
2. BOOTP server address                                000.000.000.000
     (address of the remote machine you boot from)
3. Gateway address                                     000.000.000.000
     (Optional, required if gateway used)

97. Return to Select BOOT (Startup) Device Menu (SAVES addresses)
99. Return to Main Menu (SAVES addresses)
```

Figure 3-51 Enter client and BOOTP server's IP

12. After setting the IP addresses, there should now be a entry for the client and BOOTP server similar to Figure 3-52 on page 141.

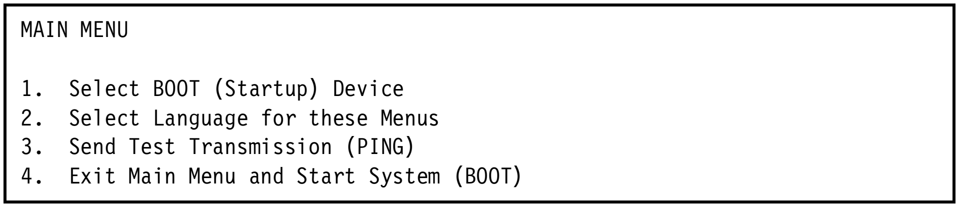


```
    SET OR CHANGE NETWORK ADDRESSES
    Select an address to change
    Currently selected BOOT (startup) device is:
    Ethernet:  Slot 0/3, BNC / modular Jacks
    Hardware address ..................................... 02608C2D63E2

    1. Client address                                      192.168.005.009
         (address of this machine)
    2. BOOTP server address                                192.168.005.050
         (address of the remote machine you boot from)
    3. Gateway address                                     000.000.000.000
         (Optional, required if gateway used)

    97. Return to Select BOOT (Startup) Device Menu (SAVES addresses)
    99. Return to Main Menu (SAVES addresses)
```

Figure 3-52 Verify setting for BOOTP server

13. Enter 99 to return to the Main Menu
14. To verify that the hardware is connected correctly conduct the ping test.

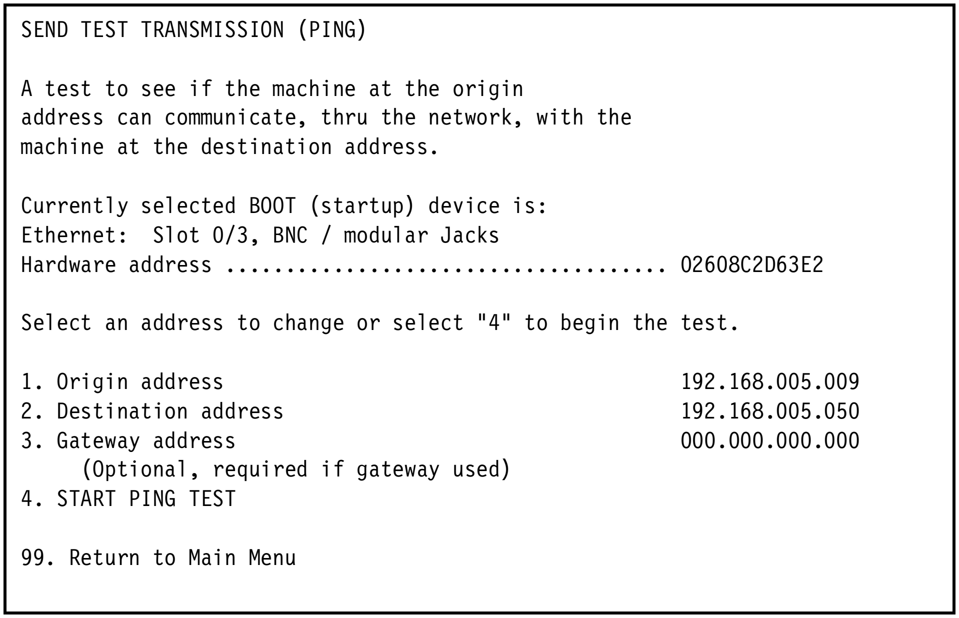
    Select option 3 as shown in Figure 3-53.



```
MAIN MENU

1.  Select BOOT (Startup) Device
2.  Select Language for these Menus
3.  Send Test Transmission (PING)
4.  Exit Main Menu and Start System (BOOT)
```

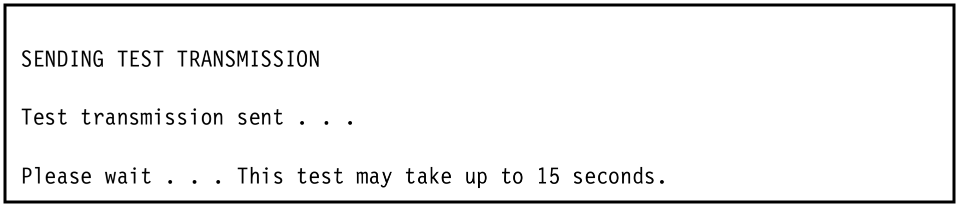
Figure 3-53 Send ping test

15. Verify that the IP address and correct if necessary. Then choose option 4 to start the ping test, as shown in Figure 3-54 on page 142.



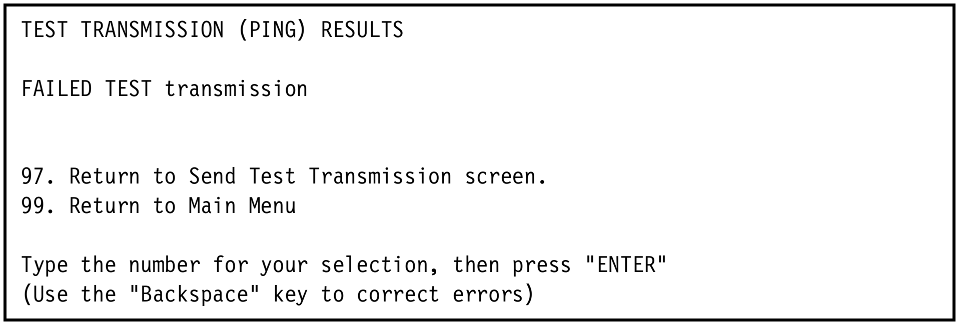
```
SEND TEST TRANSMISSION (PING)

A test to see if the machine at the origin
address can communicate, thru the network, with the
machine at the destination address.

Currently selected BOOT (startup) device is:
Ethernet:  Slot 0/3, BNC / modular Jacks
Hardware address .................................... 02608C2D63E2

Select an address to change or select "4" to begin the test.

1. Origin address                                     192.168.005.009
2. Destination address                                192.168.005.050
3. Gateway address                                    000.000.000.000
     (Optional, required if gateway used)
4. START PING TEST

99. Return to Main Menu
```

Figure 3-54 Verify all entries are correct



```
SENDING TEST TRANSMISSION

Test transmission sent . . .

Please wait . . . This test may take up to 15 seconds.
```

Figure 3-55 Ping test output



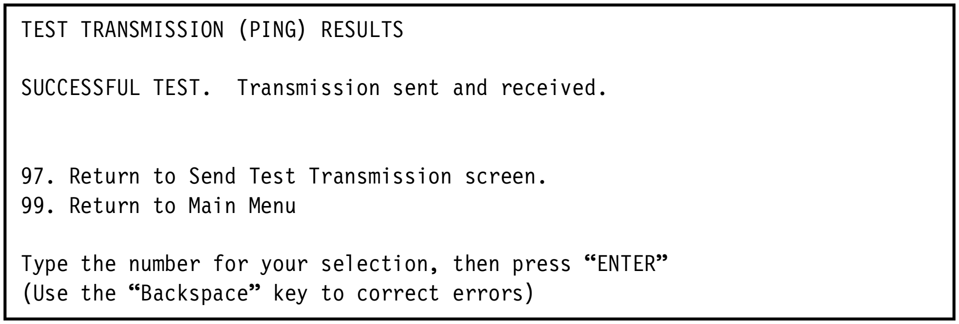
```
TEST TRANSMISSION (PING) RESULTS

FAILED TEST transmission


97. Return to Send Test Transmission screen.
99. Return to Main Menu

Type the number for your selection, then press "ENTER"
(Use the "Backspace" key to correct errors)
```

Figure 3-56 Ping test failed

16. If the ping test is unsuccessful, as shown in Figure 3-56 on page 142, then a network problem exist. Check the following:

- Are the network connections good?
- Does BNC network has the proper 50 Ohm termination?
- Is the IP address being used for the BOOTP server correct?
- Is there a functional route between the BOOTP server and the client node?
- Is the network adapter on the client node functioning?
- Is the network connected to the correct adapter on the BOOTP server and the client node? This is a likely problem if the node has multiple network adapters installed.

17. After the problem is resolved you should receive a successful ping test, as seen in Figure 3-57.

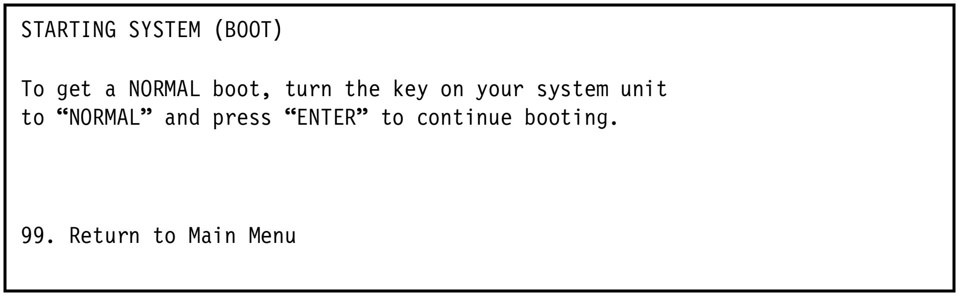


```
TEST TRANSMISSION (PING) RESULTS

SUCCESSFUL TEST.  Transmission sent and received.



97. Return to Send Test Transmission screen.
99. Return to Main Menu

Type the number for your selection, then press “ENTER”
(Use the “Backspace” key to correct errors)
```

Figure 3-57   Ping test successful

18. Exit out of the menus selecting 99 to return to the main menu.

19. Press 4 to continue booting and the following screen will appear.



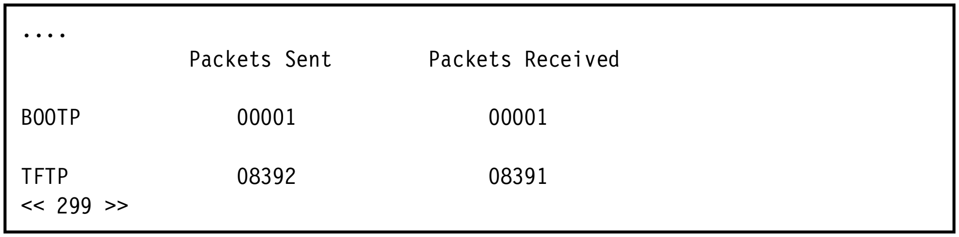
```
STARTING SYSTEM (BOOT)

To get a NORMAL boot, turn the key on your system unit
to “NORMAL” and press “ENTER” to continue booting.




99. Return to Main Menu
```

Figure 3-58   Press Enter to being system boot

20. It is important at this step to remember that a writable s1term is being utilized during manual node conditioning. Press the enter key to continue booting.

21. As soon as the booting process begins, make sure to close the s1term (ctrl-x) otherwise the BOOTP process will hang.

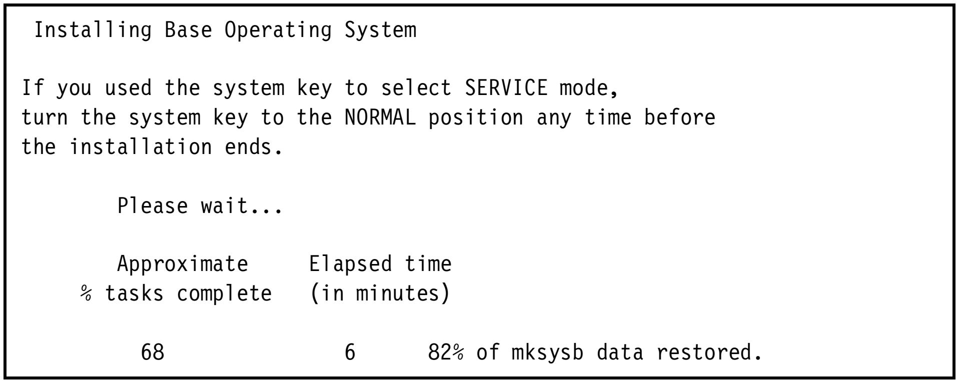
22. Open a non-writable console via the command **s1term 1 9** to make sure that the network installation is proceeding correctly (see Figure 3-59 for the TFTP progress).



```
....
              Packets Sent        Packets Received

BOOTP             00001                00001

TFTP              08392                08391
<< 299 >>
```

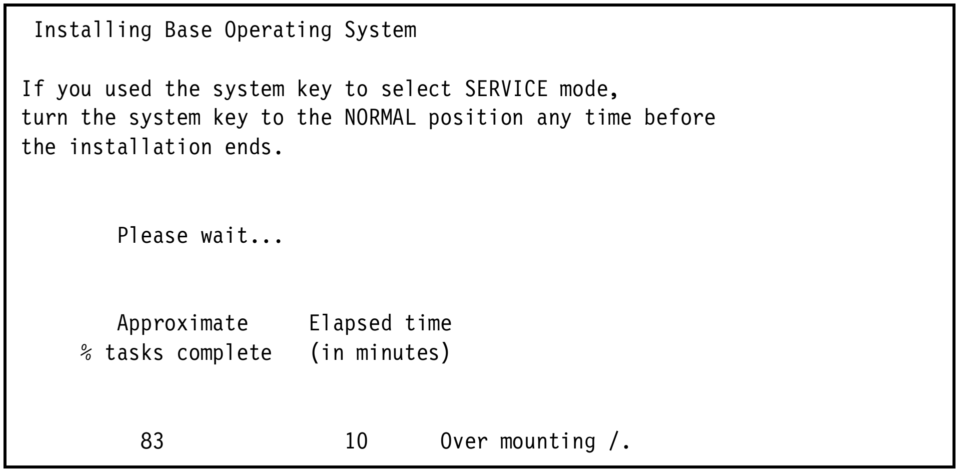
Figure 3-59 TFTP successful

23. Monitor the installation as shown in Figure 3-60 and Figure 3-61.



```
 Installing Base Operating System

If you used the system key to select SERVICE mode,
turn the system key to the NORMAL position any time before
the installation ends.

        Please wait...

        Approximate     Elapsed time
     % tasks complete   (in minutes)

          68              6      82% of mksysb data restored.
```

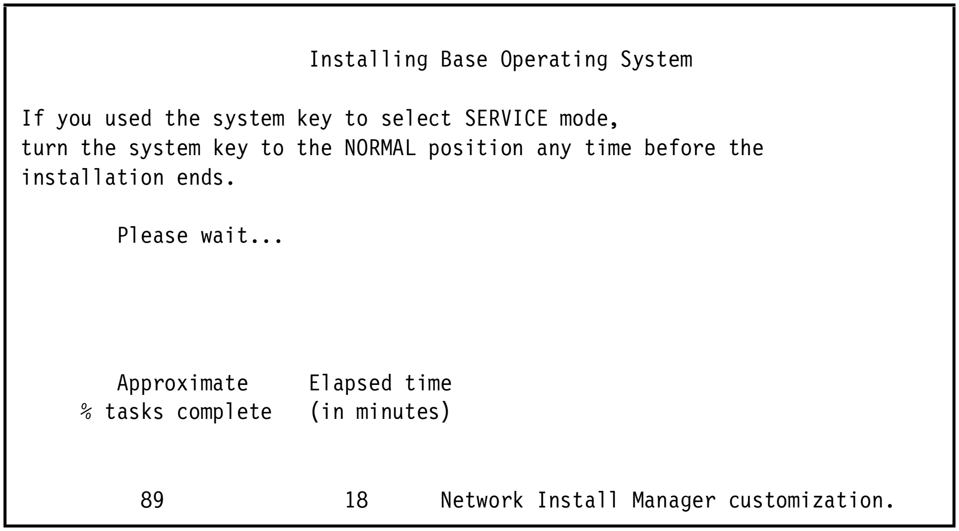
Figure 3-60 Installing BOS



```
 Installing Base Operating System

If you used the system key to select SERVICE mode,
turn the system key to the NORMAL position any time before
the installation ends.


        Please wait...


        Approximate     Elapsed time
     % tasks complete   (in minutes)


          83               10      Over mounting /.
```

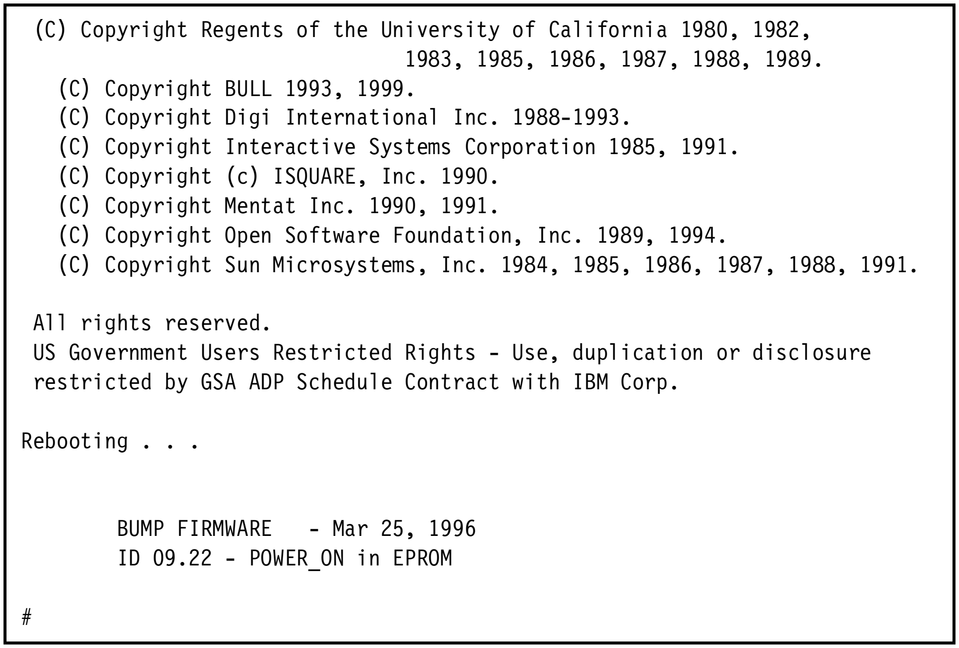
Figure 3-61 overmounting /

Monitor the installation as shown in Figure 3-62. The NIM customization shows LED U-84.



```
                        Installing Base Operating System

If you used the system key to select SERVICE mode,
turn the system key to the NORMAL position any time before the
installation ends.

        Please wait...




        Approximate     Elapsed time
     % tasks complete   (in minutes)


          89               18      Network Install Manager customization.
```

Figure 3-62 NIM customization



```
 (C) Copyright Regents of the University of California 1980, 1982,
                               1983, 1985, 1986, 1987, 1988, 1989.
   (C) Copyright BULL 1993, 1999.
   (C) Copyright Digi International Inc. 1988-1993.
   (C) Copyright Interactive Systems Corporation 1985, 1991.
   (C) Copyright (c) ISQUARE, Inc. 1990.
   (C) Copyright Mentat Inc. 1990, 1991.
   (C) Copyright Open Software Foundation, Inc. 1989, 1994.
   (C) Copyright Sun Microsystems, Inc. 1984, 1985, 1986, 1987, 1988, 1991.

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

Rebooting . . .



        BUMP FIRMWARE   - Mar 25, 1996
        ID 09.22 - POWER_ON in EPROM

#
```

Figure 3-63 System rebooting after installation

After the Network Install Manager customization has ended, the system will reboot as shown in Figure 3-63.

3.5.5 Network Install Manager (NIM)

The following section provides an overview of the Network Installation Manager (NIM).

Overview of NIM

Within the SP environment, the utilization of NIM is largely hidden in the setup_server. This isolates the configuration and customization on NIM resources to the information which is stored within the SDR. While conducting problem determination of a cluster installation, it is helpful to have a good understanding on how NIM interacts with the setup_server process.

NIM objects used in a SP environment ar the same as in a typical RS/6000 environment and as such utilize the same version of NIM. However, there are some constraints associated with the way the SP environment utilizes NIM.

The nodes within the cluster are defined as stand-alone machines as highlighted in the output of the **lsnim** command. These standalone machines must utilize the SP administrative LAN (spnet_en0) network for their installation. Individuals familiar with NIM on the classical RS/6000 platform might state that additional networks such as a public LAN or token-ring could be utilized as a NIM installation network, however, this is not supported. Additionally, the group resource is not used in the SP environment.

In the SP environment, NIM installs via a pull from the node in which the bootp response is requested from the node during a network boot (nodecond). The push method is not supported in the SP infrastructure.

```
# lsnim
master                machines        master
boot                  resources       boot
nim_script            resources       nim_script
spnet_en0             networks        ent
psspscript            resources       script
prompt                resources       bosinst_data
noprompt              resources       bosinst_data
migrate               resources       bosinst_data
lppsource_aix433      resources       lpp_source
mksysb_1              resources       mksysb
spot_aix433           resources       spot
sp5n05                machines        standalone
sp5n09                machines        standalone
sp5n13                machines        standalone
sp5n01                machines        standalone
```

Figure 3-64 Output of the lsnim command

In the following sections, the shared product object tree (SPOT) is mentioned as part of the common problems associated with NIM.

SPOT is a directory that contains AIX code and is equivalent in content to the /usr file system. The way in which setup_server creates the SPOT is via the lppsource that is defined as the NIM resource lpp_source.

Common problems associated with NIM

The following scenarios describe common problems associated with NIM.

HANG with LED 605

This LED code indicates that the node being installed is unable to identify the boot device.

At this step the node has received the boot images and booted into the BIOS mode. The problem has occurred in the configuration of the network object SP administrative LAN (spnet_en0) in NIM. To correct this problem, ensure that the device driver to support the network is in the SPOT resource. As the SP system is using only a limited number of adapters for the spnet_en0 network, this is an unlikely problem. Validate that the system is using a SP supported network adapter.

Hang with the LED 608

This LED code indicates that the NIM client was not able to get the <node_name>.install_info file from the SPOT server. The node will continue to retrieve this information.

This file is kept in the /tftpboot file system and needs to have a permission setting of 644 with the owner being root:system. If the file does not exist, then make sure the node is set to install and rerun setup_server to regenerate these files.

Hang with the LED 611

This LED indicates that the node was unable to mount the resources necessary via NFS from the CWS or BIS. Validate the following:

- Check that NFS is running on the CWS/BIS with the `lssrc -s nfsd` command.
- Verify that the resources necessary have been exported correctly.
- If you have included the latest fixes in the resource lpp_source, make sure that you:
  - Create a new lpp_source without the additional fixes.
  - Allocate the lpp_source to the node in question.
  - Create a new SPOT for the lpp_source.
  - Reinstall the node.
- Incorrect routing can cause this hang as well. Verify that the default route can reach the CWS.
- Run NIM in debug mode; see, “Enable NIM debug mode” on page 151.

Hang with LED 613

Route is incorrectly defined for the SP administrative LAN (spnet_en0) network in the NIM database.

To verify the route that NIM should be using check the SDR entries as shown inFigure 3-65. Not that the entry for node 9 has a different default route than the other nodes. Nodes 1, 5 and 13 install fine and have the spnet_en0 interface as their default route whereas node 9 has assigned an internal network interface as the default route. However, the network that supports this default route does not include the CWS. Therefore, the entry in the SDR will need to be modified to allow for a routable network to the CWS.

```
# SDRGetObjects Node node_number default_route
node_number  default_route
          5 192.168.5.150
          9 192.168.25.1
         13 192.168.5.150
          1 192.168.5.150
```

Figure 3-65   Check the default route in the SDR

Hang with LED u90

If node installation appears to hang during NIM customization on LED code u90 as in Figure 3-66, check the authentication methods by using the **lsauthent** command, as shown in Figure 3-67 on page 150.

```

5.  Checking nodes
----------------------------------- Frame 1
----------------------------------
                      Host     Switch   Key     Env   Front Panel
LCD/LED
Slot Node Type  Power Responds Responds Switch  Error LCD/LED
Flashes
---- ---- ----- ----- -------- -------- ------- ----- ----------------
-------
  1     1  high  off     no     notcfg  service   no  Stand-By          no
                                                      LCD2 is blank
  5     5  high   on     no     notcfg   normal   no   u90              no
                                                      LCD2 is blank
  9     9  high   on     no     notcfg   normal   no   u90              no
                                                      LCD2 is blank
 13    13  high   on     no     notcfg   normal   no   u90              no
                                                      LCD2 is blank
# shutdown -Fr
```

Figure 3-66   Node install hangs on u90

```
# lsauthent
Kerberos 4
```

Figure 3-67 Output of lsauthent command

Note that, in Figure 3-68, the output only shows Kerberos V4 as the authentication method. This is a problem for the installation of the nodes, as the nodes do not have Kerberos authentication credentials until after node customization. Therefore, the authentication will need to be changed to add standard AIX to the authentication method and the node re-installed.

```
chauthent -k4 -std

# lsauthent
Kerberos 4
Standard Aix
#
```

Figure 3-68 Change authentication to Kerberos V4 and standard AIX

Node installed AIX but not PSSP code and LED is blank

If a node installation has apparently completed but the PSSP code has not installed on the node and host response in “no”, check the /spdata/sys1/install/pssplpp/<PSSP_version> directory. Ensure that file permissions of 2755 with owner root:system

If the file permissions are not set correctly then the installation of the PSSP code will fail. This can be verified by NFS mounting this PSSP-3.2 directory on the node manually and attempting a installation over the NFS mount point. Output of such an installation is shown in Figure 3-69 on page 151.

Change the permission on the files and either re-install or customize the node in question.

```
installp:  APPLYING software for:
        rsct.basic.rte 1.2.0.0
        rsct.basic.sp 1.2.0.0
        rsct.basic.hacmp 1.2.0.0

restore: 0511-126 Cannot open /mnt/./rsct.basic: The file access permissions
do not allow the specified action.
Mount volume 1 on /mnt/./rsct.basic.
        Press the Enter key to continue.
0503-405 installp:  An error occurred while running the restore command.
        Use local problem reporting procedures.

installp:  CANCELLED software for:
        rsct.basic.rte 1.2.0.0
        rsct.basic.sp 1.2.0.0
        rsct.basic.hacmp 1.2.0.0
```

Figure 3-69 File access problems in PSSP install on nodes

If the permissions are correct, check that the instructions listed in Chapter 2, step 15.2 (see Figure 3-70) from *Update the Image Table of Contents in the Installation and Migration Guide*, GA22-7347 were followed correctly. The instructions state to:

```
cd /spdata/sys1/install/pssplpp/PSSP-3.2
mv ssp.3.2.0.0.I pssp.installp
mv rsct.basic.1.2.0.0.I rsct.basic
mv rsct.clients1.2.0.0.I rsct.clients
mv rsct.core.1.2.0.0.I rsct.core
```

Figure 3-70 Installation step 15.2 instructions

If you copy these file rather than re-naming them, setup_server will terminate with the rc=0 and does not check for this condition. Therefore, the effects are not seen until the very end of the installation procedure. This specific problem occurs when the LEDs are blank giving little indication as to what went wrong.

To correct this problem remove the unnecessary nodes and re-install them.

Enable NIM debug mode

To enable NIM debugging, perform the following steps on the boot/install server:

1. Confirm that the NIM resource object is not allocated.

```
#lsnim -a alloc_count spot_aix433
spot_aix433:
   alloc_count = 1
```

If the alloc_count in not 0, there are machine objects allocated to this resource object. To list what machine object is allocated to the SPOT, issue the command:

```
# lsnim -a spot
sp5n09:
   spot = spot_aix433
```

This output indicates that the machine object for node sp5n09 is allocated to the SPOT. Check the boot response for this node and, if it is not set to disk, then change the boot response to disk to unallocate the machine group from the SPOT resource.

```
# splstdata -b
                  List Node Boot/Install Information

node# hostname         hdw_enet_adr  srvr response     install_disk
----- ----------------- -------------- ---- ------------ --------------
1  sp5n01            02608CF57A7C     0 disk         hdisk0
5  sp5n05            02608C2D4CA5     0 disk         hdisk0
9  sp5n09            02608C2D63E2     0 install      hdisk0,hdisk1
13 sp5e1n13          02608CF518A4     9 install      hdisk0
#
# spbootins -r disk -l 9

# lsnim -a alloc_count spot_aix433
spot_aix433:
   alloc_count = 0
```

Figure 3-71 Perform these tasks prior to setting NIM to debug mode

Once the boot response has been set back to disk and setup_server has run successfully, the machine object class should be no longer allocated to the NIM SPOT resource. Note that, in Figure 3-71, node 13 also has a boot response of install. However, this node's boot install server is node9, which is not allocated to the same SPOT resource as node9. Therefore, node13's boot response does not need to be changed to install at this time. If the nodes boot response has been changed to disk, setup_server ran successfully and the machine object is still allocated to the SPOT, then run the following commands to force a reset of the machine object and then deallocate the machine from the SPOT as shown in Figure 3-72 on page 153.

```
nim -Fo reset <machine_object>
nim -o deallocate -a spot=<spot_object> <machine_object>
```

```
# lsnim -a alloc_count spot_aix433
spot_aix433:
   alloc_count = 1

# nim -Fo reset sp5n09

# nim -o deallocate -a spot=spot_aix433 sp5n09
# lsnim -a alloc_count spot_aix433
spot_aix433:
   alloc_count = 0
```

Figure 3-72 Deallocate resources from SPOT if necessary

After no NIM objects are allocated to the SPOT, NIM debug mode may be enabled.

```
nim -o check -a debug=yes <spot_object>
```

In our test example, we ran the command to place NIM into the debug mode and then verified that debug was set in NIM, as shown in Figure 3-73.

```
# nim -o check -a debug=yes spot_aix433

# lsnim -l spot_aix433
spot_aix433:
   class         = resources
   type          = spot
   enter_dbg     = "rs6k.mp.ent 0x0026a794"
   Rstate        = ready for use
   prev_state    = verification is being performed
   location      = /spdata/sys1/install/aix433/spot/spot_aix433/usr
   version       = 4
   release       = 3
   mod           = 3
   alloc_count   = 0
   server        = master
   if_supported  = rs6k.mp ent
   Rstate_result = success
   plat_defined  = chrp
   plat_defined  = rs6k
   plat_defined  = rspc
```

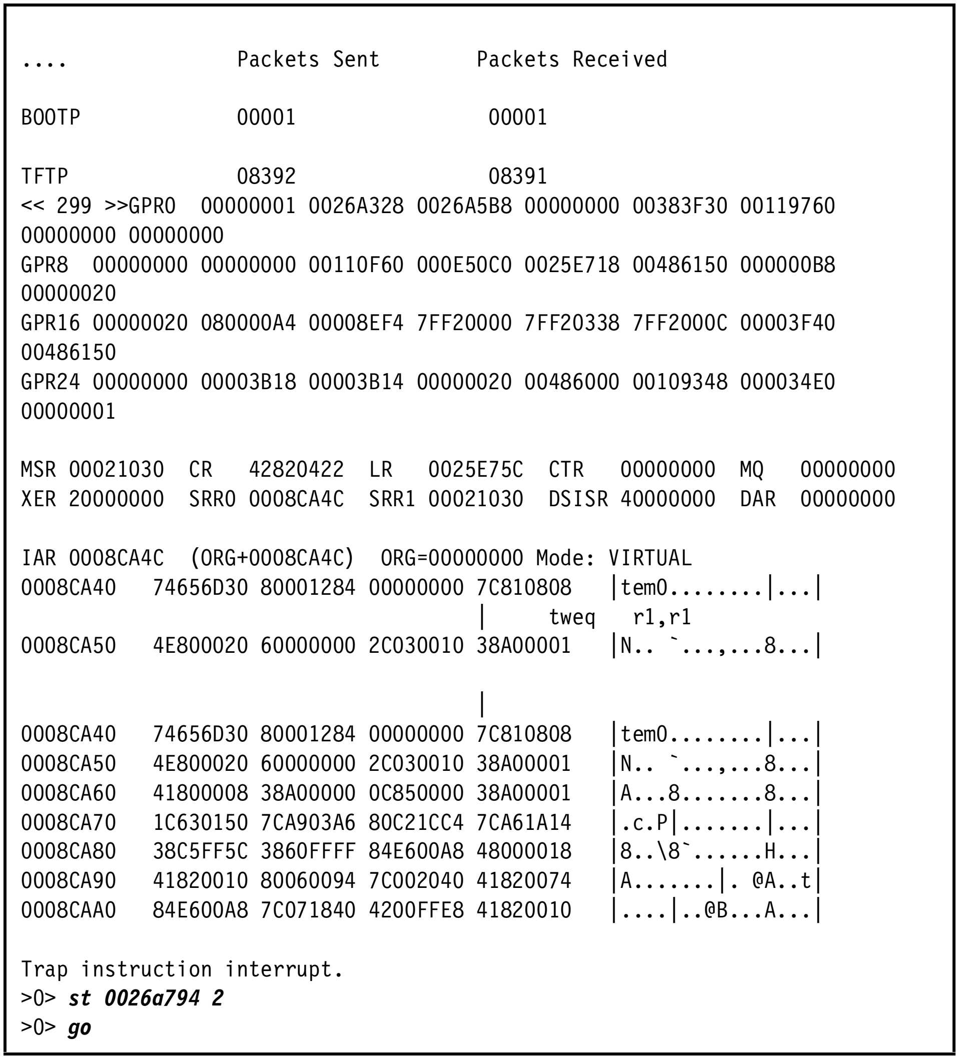
Figure 3-73 Enabling NIM debug mode

Now, you need to ensure that any nodes that need to have a boot response of install are set back to the install mode and then run setup_server to reallocate the machine objects to the NIM SPOT.

Make note of the output from the `lsnim -l` command and record the information from the enter_dbg attribute. The output from the enter_dbg attribute will be needed for the trap instruction interrupt of the manual conditioning procedure. Using Figure 3-73 as an example, you drop the 0x part proceeding the eight digit hex entry of 0026a794, which is the number that need to be recorded as part of the trap interrupt entry.

To follow the steps for manual node conditioning, see Section 3.5.4, “Manual node conditioning” on page 136.

Once the trap instruction interrupt is displayed on the s1term, enter the required information as shown in Figure 3-74 on page 155.



```
....              Packets Sent        Packets Received

BOOTP             00001                00001

TFTP              08392                08391
<< 299 >>GPR0  00000001 0026A328 0026A5B8 00000000 00383F30 00119760
00000000 00000000
GPR8  00000000 00000000 00110F60 000E50C0 0025E718 00486150 000000B8
00000020
GPR16 00000020 080000A4 00008EF4 7FF20000 7FF20338 7FF2000C 00003F40
00486150
GPR24 00000000 00003B18 00003B14 00000020 00486000 00109348 000034E0
00000001

MSR 00021030  CR   42820422  LR   0025E75C  CTR   00000000  MQ   00000000
XER 20000000  SRR0 0008CA4C  SRR1 00021030  DSISR 40000000  DAR  00000000

IAR 0008CA4C  (ORG+0008CA4C)  ORG=00000000 Mode: VIRTUAL
0008CA40   74656D30 80001284 00000000 7C810808   |tem0........|...|
                                      |     tweq   r1,r1
0008CA50   4E800020 60000000 2C030010 38A00001   |N.. `...,...8...|

                                      |
0008CA40   74656D30 80001284 00000000 7C810808   |tem0........|...|
0008CA50   4E800020 60000000 2C030010 38A00001   |N.. `...,...8...|
0008CA60   41800008 38A00000 0C850000 38A00001   |A...8.......8...|
0008CA70   1C630150 7CA903A6 80C21CC4 7CA61A14   |.c.P|.......|...|
0008CA80   38C5FF5C 3860FFFF 84E600A8 48000018   |8..\8`......H...|
0008CA90   41820010 80060094 7C002040 41820074   |A.......|. @A..t|
0008CAA0   84E600A8 7C071840 4200FFE8 41820010   |....|..@B...A...|

Trap instruction interrupt.
>0> st 0026a794 2
>0> go
```

Figure 3-74 NIM debug trap instructions

When the trap instruction interrupt prompt occurs, enter information in the following format:

```
st <Enter_dbg_Vaule> 2
```

The *st* is to store the information, <Enter_dbg_vaule> is the eight-digit hex number gathered from the **lsnim -l <spot>** command and the *2* indicates to print the output of the debug to the tty. After entering the string st 0026a794 2 and pressing Enter, refer to Figure 3-74 on page 155; you will see a second prompt with the word “go” after it. When this appears, press Enter again. The debug process will then begin.

Tips:

1. Once the debug process begins you can temporarily halt it with Ctrl-S and resume with Ctrl-Q.

2. If the process seems to hang on LED c58, press Ctrl-Q to continue.

After debugging has completed the boot image will need to be rebuilt in non-debug mode; use the following command to achieve this:

```
nim -Fo check SPOTname
```

From our example this would be:

```
nim -Fo check spot_aix433
```

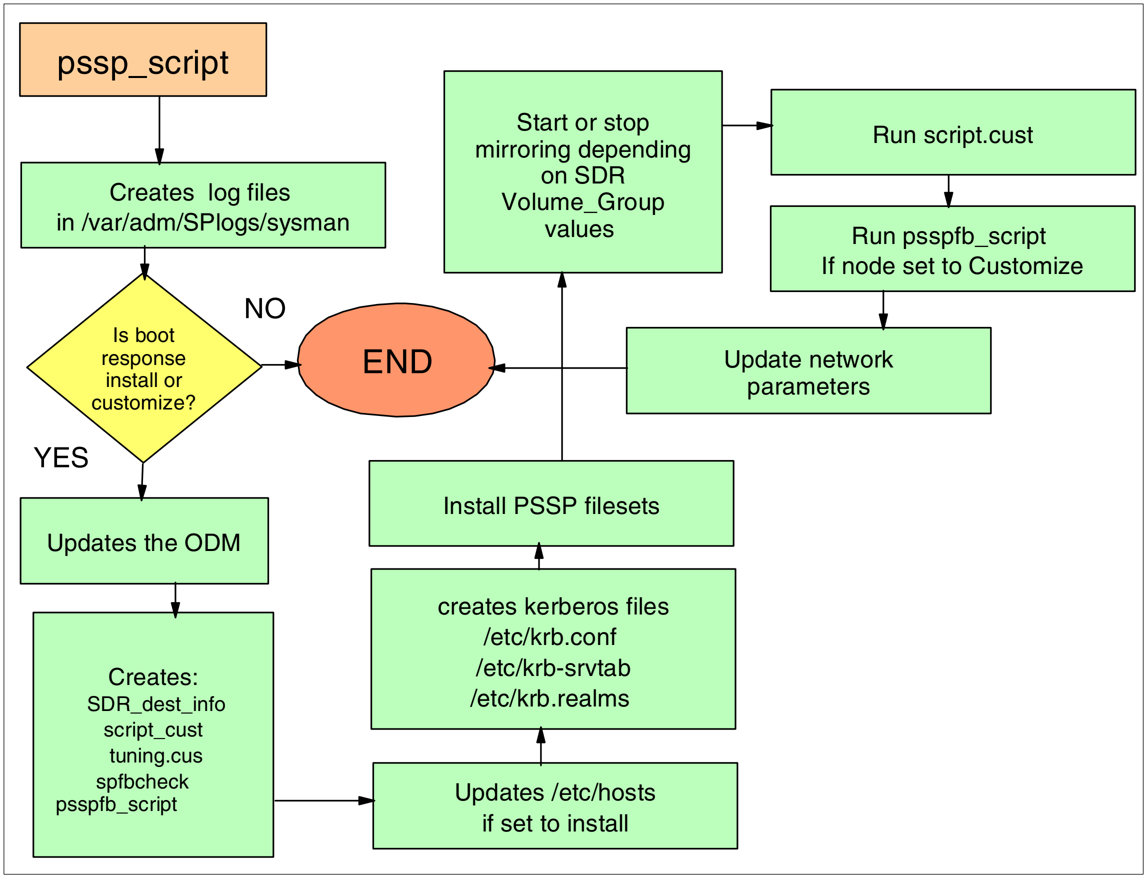
Note: NIM commands and their responses are recorded. Look for errors associated with command invocation, error associated with file set installation, and errors associated with NIM related activities. For more information on NIM debugging, refer to Chapter 7, “Diagnosing NIM problems” in *Parallel System Support Programs for AIX: Diagnosis Guide Version 3 Release 2*, GA22-7350.

3.6 pssp_script

After the installation of a node, /usr/lpp/ssp/install/pssp_script is run. This script is called by NIM and is run before the node reboots, at which time the RAM file system is in place. It installs required LPPs and does post-PSSP installation setup. Additional adapter configuration is performed after reboot by the psspfb_script.

The flow of the pssp_script is shown in Figure 3-75 on page 157.

If pssp_script fails or hangs, then check the log files in /var/adm/SPlogs/sysman. The pssp_script will create a series of files in this directory if the files created have the name NODE rather than the <node_name> as a prefix.



pssp_script
Creates log files in /var/adm/SPlogs/sysman
Is boot response install or customize?
NO
END
YES
Updates the ODM
Creates: SDR_dest_info script_cust tuning.cus spfbcheck psspfb_script
Updates /etc/hosts if set to install
creates kerberos files /etc/krb.conf /etc/krb-srvtab /etc/krb.realms
Install PSSP filesets
Start or stop mirroring depending on SDR Volume_Group values
Run script.cust
Run psspfb_script If node set to Customize
Update network parameters

Figure 3-75 setup_server flowchart

4

Security

This chapter discusses the security options available in the universal cluster and provides troubleshooting information for the cluster security services. It addresses the following topics:

- Available security options
- Changes in the installation process
- Debugging information
- Case studies

4.1 Available security options

Starting with PSSP 3.2, more flexibility has been added in installing and configuring the authentication services, aiming to better accommodate the security requirements needed in different environments. The security options can now be differentiated for remote commands and for trusted services. A feature has been added to restrict root remote command access within the SP and changes have been operated in the way the server keys are transferred from the CWS to nodes. Because of the new choices available, deciding for the initial or subsequent security options for a SP system requires a good planning and understanding of the relationship between the SP components, the local requirements and the selected security methods.

The security on the SP system is a two-step process. First, the system needs to *authenticate* the user or service, that is, to check its security credentials using specific mechanisms. In the second stage, assuming that the authentication has been successfully completed, the user or service is granted certain rights, based on an *authorization* mechanism.

The following four security attributes of the Syspar class in the SDR are available:

auth_install Set of authentication methods to be installed on nodes in the partition.

auth_root_rcmd Authorization methods used for root access to AIX remote commands.

ts_auth_methods Set of active SP trusted services authentication methods for the partition.

auth_methods Set of active AIX remote command authentication methods for the partition.

The previous attributes can be examined using the **SDRGetObjects Syspar** or **splstdata -p** commands, as shown in Example 4-1. The settings are valid for the selected partition (SP_NAME variable needs to be set, or the default partition is assumed).

Example 4-1 splstdata output

```
# splstdata -p | grep auth
auth_install    k4
auth_root_rcmd  k4
ts_auth_methods compat
auth_methods    k4:std
```

For authentication, PSSP allows the use of three mechanisms:

- Kerberos V5, provided by DCE.

- Kerberos V4
- Standard AIX

As of PSSP 3.2, authentication mechanisms on the SP can be differentiated in:

- Authentication used by *remote commands* (**rsh**, **rcp**, **rlogin**), plus the **ftp** and **telnet** for system management. It is specified by the auth_methods attribute.
- Authentication methods for SP trusted services (hardmon, sysctl, hats, hags, haem, pman, and so on). It is specified by the ts_auth_methods attribute of the Syspar class.

Authorization for the root user to use AIX remote commands is specified by the auth_root_rcmd attribute in the Syspar class and by the restrict_root_rcmd attribute in the newly introduced class, SP_Restricted. Its implementation depends on the type of authorization in use and is based on ACLs and special files.

4.1.1 Choosing the authentication option

Based on the security requirements in your environment, you have the choice of using one of the following mechanisms: DCE, Kerberos 5, Kerberos 4, and standard AIX. Here is how those mechanisms deal with authentication and authorization. Here is how these mechanisms deal with authentication and authorization:

Standard AIX The login authentication method is done through the DES-encrypted password stored in the /etc/security/passwd file and the UNIX login program. Remote command authentication is done by reading the IP address and user ID in the network request packet from the originating host. The authorization method is implemented through the normal UNIX base file permissions and ownership, through the AIX extended permissions, and, optionally, through mechanisms, such as .rhosts or hosts.equiv files.

Kerberos V4 The authentication method is implemented through the Kerberos daemon on the master security server (typically, the CWS), and the kinit program on the client machines, which include the nodes and the CWS. The authorization method is implemented through the kerberos daemon and mechanisms, such as the .klogin and /etc/krb-srvtab files.

DCE (Kerberos V5) This represents the authentication and authorization services provided by DCE or, more specifically, Kerberos V5. The authentication method is implemented by the **secd** daemon on the DCE Security server(s) and the **dced**

daemon on the DCE clients. The authorization method is implemented through DCE Access Control Lists (ACLs) and mechanisms, such as the .k5login file and DCE keytab files in the /spdata/sys1/keyfiles/ssp/<hostname>/ directory.

If your environment is isolated and runs scientific applications, using standard AIX security might be a good choice, but most commercial environments need a more sophisticated security mechanism, for example, the one based on Kerberos. DCE is a framework that offers more than security services and is more difficult to manage than Kerberos 4, which is delivered with PSSP. We recommend, unless you use more DCE facilities than security, using Kerberos 4 for authentication.

4.1.2 Using Restricted Root Access (RRA)

One of the advantages of the SP cluster versus standalone systems is the ability to control and manage all nodes from a central location, the CWS. In order to achieve that, several SP system components use remote commands (**rcp** and **rsh**) to run commands or copy files within the SP complex. Those remote commands run as root, which implies that gaining root access on any node in the SP will automatically grant root access on any other node, including the CWS. In certain server consolidation implementations, it is not desirable to have a single root user across the entire SP system. Beginning with PSSP 3.2, a feature called restricted root access (RRA) is available to limit the use of **rsh** and **rcp** within the SP. With RRA enabled, **rcp** and **rsh** is only allowed from the CWS to nodes. Nodes are no longer automatically authorized to **rcp** and **rsh** to other nodes or CWS. Instead, when the PSSP software on the nodes needs to execute **rcp** or **rsh** commands, the task is executed by **sysctl,** using the partition-wide selected authentication methods and its specific authorization mechanism.

When RRA is activated, remote commands authorizations are re-configured such that:

- When run as a root process on the control workstation, SP software can continue to issue **rsh** and **rcp** commands to any node within the SP system. Authorization file entries will be created by PSSP on the nodes for which a root process on the control workstation can obtain the necessary credentials to issue these commands.
- SP software run as a root process on a node no longer requires the capability to issue **rsh** and **rcp** commands to the control workstation or to any other node within the SP system. PSSP will no longer create authorization file entries on the control workstation or nodes that grant a root process on a node remote command access.

RRA can be activated with any of the three authentication mechanisms: DCE, Kerberos V4 and AIX.

4.1.3 Using a secure remote command process

In choosing a remote command process, consider that the PSSP code should be able to issue secure **rcp** and **rsh** without being prompted for a password or passphrase.

4.2 Security services implementation

The security settings discussed in the preceding paragraphs can be selected and implemented at the time of PSSP installation, or can be modified on an existing system.

The initial versions of PSSP were using their own “kerberized” commands of **rcp** and **rsh**. Starting with AIX V4.3.1, the **rcp** and **rsh** commands provided by the operating system are able to use Kerberos V4 or Kerberos V5 (if DCE is enabled) for authentication. The AIX commands are located in the /usr/bin directory. For compatibility reasons, the remote commands used in the versions of PSSP older than 3.1 are now symbolic links to the AIX commands:

- /usr/lpp/ssp/rcmd/bin/rcp -> /usr/bin/rcp
- /usr/lpp/ssp/rcmd/bin/remsh -> /usr/bin/rsh
- /usr/lpp/ssp/rcmd/bin/rsh -> /usr/bin/rsh

4.2.1 Security software installation

From the available authentication methods, only the DCE filesets can be optionally installed. Kerberos V4 is part of PSSP code, and Standard AIX security is part of AIX. If Kerberos V4 is selected, the Kerberos V4 configuration files will be installed.

To use DCE and/or Kerberos V5, you have to do the following before you begin to install and configure PSSP:

1. Obtain the server software and licenses.
2. Have AIX installed and operating on the control workstation.
3. Have the DCE core server software installed, configured, and operational.
4. Have the control workstation operational in the DCE cell as a client or a server.
5. Place the DCE client software in the lppsource file on the control workstation.

At installation time, you have the choice of selecting the authentication method(s) to be installed on the nodes. For more information refer to Chapter 2, Step 39 "Select Security Capabilities Required on Nodes" of the *Parallel System Support Programs for AIX: Installation and Migration Guide,* GA22-7347.

The auth_install attribute

At installation time, you have the choice of selecting the authentication methods to be installed on the nodes in the partition in the Chapter 2, Step 39 "Select Security Capabilities Required on Nodes" of the *Parallel System Support Programs for AIX: Installation and Migration Guide,* GA22-7347.

The command used is **spsetauth**. For example, to select Kerberos V4 to be configured in the sp3en0 (default) partition see Example 4-2.

Example 4-2 Setting the auth_install attribute

```
sp3en0:> spsetauth -p sp3en0 -i k4
sp3en0:> SDRGetObjects Syspar auth_install
auth_install
k4
```

The **spsetauth** command, when used with the **-i** flag, sets the auth_install attribute in the Syspar SDR class. Based on this attribute, the **spauthconfig** command installs and/or configure the nodes during the installation or customization process.

4.2.2 AIX local security settings for remote command authentication

When a remote command is called, it issues the get_auth_method*()* system call in the libauthm.a library, which returns the list of authentication methods enabled in AIX on the system. The following options are available and can be checked with the **lsauthent** command:

- Kerberos V5
- Kerberos V4
- Standard AIX

The authentication methods are tried in the order returned by the lsauthent command. If the first authentication method fails, the second authentication method will be used, and so on until all available authentication methods are exhausted. The methods are stored in the ODM on the local node.

4.2.3 PSSP partition level security settings for remote command authentication

For a standalone AIX system, you would set the order of precedence for authenticated commands with the command **chauthent**. However, PSSP maintains its own security settings in the SDR, at system partition level, in the auth_methods attribute of the Syspar class.

The auth_methods attribute

The available methods for remote commands, reflected in the auth_methods attribute of the Syspar class are:

k5 Specifies that the Kerberos Version 5 authentication method is to be made active for this partition. To activate k5, DCE needs to be installed and configured for the partition.

k4 Specifies that the Kerberos Version 4 authentication method is to be made active for this partition. To activate k4, Kerberos 4 needs to be installed and configured for the partition.

std Specifies that the Standard AIX authentication method is to be made active for this partition.

To query the partition-level enabled authentication methods use:

- **splstdata -p**
- **SDRGetObjects Syspar auth_methods**

To change the partition-level enabled authentication methods, use the **chauthpar** command.

Because the propagation of settings to running nodes is performed using the AIX **rsh** command to execute **chauthent**, the CWS and the nodes must have at least one common remote command authentication method active in order for propagation to succeed. When this in not the case, propagation can only be completed by the local root user running **chauthent** or **spauthconfig** on each node (or by a re-boot). The same applies to nodes that are not running, or are otherwise inaccessible, when this command is executed.

In order to be able to activate an authentication method, the software that enables that method needs first to be installed and configured. This is done with the **spsetauth -i** command. See Section 4.2.1, “Security software installation” on page 163 for more info.

Notes:

1. If the authentication methods enabled for use by SP Trusted Services (see “Authentication methods for SP trusted services” on page 168) includes DCE, the authentication methods enabled for use by the AIX remote commands must include Kerberos V5.
2. If the authentication methods enabled for use by SP Trusted Services includes compatibility, the authentication methods enabled for use by the AIX remote commands must include Kerberos V4.
3. You must activate the Kerberos Version 4 authentication method if any node in the partition is running a level of PSSP earlier than Version 3.2.

Attention: The chauthpar should be the sole tool to control the active authentication methods for remote commands on the SP.

Initial system installation setup

At installation time, the partition-level setting for remote command authentication is done in Chapter 2, Step 46, “Enable Authentication Methods for AIX Remote Commands” in the *PSSP: Installation and Migration Guide*, GA22-7347.

Example 4-3, shows how to enable Kerberos V4 and Standard AIX at the partition level.

Example 4-3 Enabling the authentication methods for remote commands

```
# chauthpar -c -p sp3en0 -v k4 std
The remote command authentication methods for this host are currently k4:std
The authentication methods by partition are currently

        sp3en0  k4
The partition to be modified is sp3en0
The auth_methods attribute of the partition has been set to k4:std
```

The **-c** flag specifies that the command is to operate only on the control workstation, changing settings in the System Data Repository (SDR) and in AIX as required, without attempting to make any changes on the nodes in the partition. This is what you want at installation time, when the nodes are not installed yet.

The **-p** flag allows you to specify the system partition. When not used, the SP_NAME variable will define the current partition. When SP_NAME is not set, the default partition is assumed.

The **-v** flag is useful for debugging purposes, as it returns verbose output from the command.

The shown command sets the SDR auth_methods attribute in the Syspar class. At installation time, the setting is propagated to the nodes by the **spauthconfig** command.

4.2.4 Authorization methods for remote root access

Once one or more authentication methods are configured, the authorization for the root user to issue remote commands is controlled on each individual host by the authorization files. The PSSP code maintains the authorization files at partition level, using the **updauthfiles** command. The authorization files are:

- .rhosts for standard AIX authentication
- .klogin for kerberos 4 authentication
- .k5login for DCE authentication

Selecting the desired settings is done at installation time in Chapter 2, Step 43, “Select Authentication Methods for AIX Remote Commands” in the *PSSP: Installation and Migration Guide*, GA22-7347.

For example, to select Kerberos V4 type authorization files for the sp3en0 partition and verify the result, use the command shown in Example 4-4.

Example 4-4 Selecting the authorization methods for remote root commands

```
sp3en0:> spsetauth -p sp3en0 -d k4
sp3en0:> SDRGetObjects Syspar auth_root_rcmd
auth_root_rcmd
k4
```

The auth_root_rcmd attribute

The **spsetauth** command, when used with the -d flag, updates or sets the auth_root_rcmd attribute in the SDR Syspar class. Based on this attribute, the **updauthfiles** command updates or creates (if necessary) the .k5login, .klogin, and .rhosts files on the control workstation and on all nodes in the system for which DCE , Kerberos 4, or Standard AIX are defined as Authentication Methods. The **updauthfiles** command runs at node boot time and is called by the /etc/rc.sp script.

The **updauthfiles** command can also be used to rebuild the authorization files on a running system.

Tip: If using Kerberos for authentication, do not select **std** as the last method for auth_root_rcmd. It creates the .rhosts files on the nodes, in which case the root user can issue remote commands even when not authenticated by Kerberos.

Manually editing the authorization files

Some applications, like HACMP, require the addition of customized entries in the authorization files. Because HACMP on SP is supported with enhanced (Kerberos V4) and with standard security, we will discuss the .rhosts and .klogin files.

- **.rhosts**: Do not place any entry between the following two lines:

```
# Updated by updauthfiles script
# End of generated entries by updauthfiles script
```

If you do, each time the **updauthfiles** command runs (for example at reboot), the added entries are overwritten.

- .klogin: The behavior is different based on the RRA setting. See also Section 4.2.7, “Implementing RRA” on page 172 for RRA setup information.
  - RRA inactive: Any manually added entry on nodes is overwritten by the **updauthfiles** command, which remote copies the authorization files from the CWS. The customized entries on the CWS are preserved. Because authorization files are identical across the SP, if you need any special entries, like for the godm principal in HACMP, you need to add them on the CWS.
  - RRA active: In this case, only the rcmd principal from the CWS is automatically added. However, for HACMP you might want to add more entries in the .klogin files. The **updauthfiles** command does not automatically remove the customized entries in the .klogin file when RRA is active.

Attention: An entry in .klogin without a corresponding entry in the Kerberos database does not provide any authorization, because the authentication process for the corresponding principal fails.

4.2.5 Authentication methods for SP trusted services

The SP System Monitor command-line interface, the SP Perspectives graphical user interfaces, the PSSP remote execution facilities **dsh** and **sysctl**, and other SP trusted services all use authentication services. The possible combinations of authentication methods to be enabled for mutual authentication by SP trusted services within each SP system partition are:

- DCE
- Compatibility (compat)
- DCE and compat
- None

The compatibility method lets the SP trusted services use whatever means of authentication they used before PSSP 3.2. Some used Kerberos V4 with access control lists, some had their own independent means, some simply required root access, and some did not require authentication.

The authentication methods for SP trusted services can be set at host level or at partition level. At node level, the **chauthts** command is used. As with the authentication methods for remote commands, the recommended command to be used if the one that acts at the partition level (**chauthpts**). The **chauthpts** command first checks whether the specified option is consistent with the other security settings at SP level, then uses the **rsh** command to execute the **chauthts** on nodes.

If the partition-level **chauthpts** command fails for some nodes when it is run from the CWS, then the **chauthts** command must be run locally on each failed node

The ts_auth_methods attribute

At installation time, the authentication method for the SP trusted services is selected in Chapter 2, Step 47 of the *PSSP: Installation and Migration Guide*, GA22-7347.

The command in use is **chauthpts**. The Example 4-5 shows the command output in verbose mode, for the sp3en0 (default) partition.

Example 4-5 Setting the authentication methods for SP Trusted Services on the CWS

```
sp3en0:> chauthpts -c -p sp3en0 -v compat
The trusted services authentication methods for this host are currently
The authentication methods by partition are currently

        sp3en0  compat
The partition to be modified is sp3en0
The trusted services authentication methods for this host are now set to compat
```

The **-c** flag operates the change on the CWS only and updates the ts_auth_methods attribute of the Syspar class in the SDR. This flag is used at system installation time.

The **-f** flag attempts to propagate the change to the nodes using **rsh**. This flag is used when changes need to be operated on a running system. The output in this case is shown in Example 4-6.

Example 4-6 setting the authentication methods for SP Trusted Services in the partition

```
sp3en0:> chauthpts -f -p sp3en0 -v compat
The trusted services authentication methods for this host are currently compat
The authentication methods by partition are currently

        sp3en0  compat
The partition to be modified is sp3en0
The chauths command was executed successfully on node sp3n05
The chauths command was executed successfully on node sp3n06
.................................................................
The chauths command was executed successfully on node sp3n12
The chauths command was executed successfully on node sp3n13
The chauths command was executed successfully on node sp3n14
The chauths command was executed successfully on node sp3n15
0513-095 The request for subsystem refresh was completed successfully.
```

If an un-installed method is selected, the command reports an error, as shown in the Example 4-7.

Example 4-7 Incorrect use of chauthpar command

```
sp3en0:> chauthpts -c -p sp3en0 -v dce
chauthpts: 0016-349 You cannot enable dce, because the partition has not been
configured for DCE use.
```

To query the authentication methods used by the SP Trusted Services, use one of the following commands as in Example 4-8.

Example 4-8 Quering the authentication methods for SP Trusted Services

```
sp3n06:> lsauths
Compatibility
sp3n06:> lsauthpts
Compatibility
sp3n06:> splstdata -p | grep auth
auth_install    k4
auth_root_rcmd  k4
ts_auth_methods compat
auth_methods    k4:std
sp3n06:> SDRGetObjects Syspar ts_auth_methods
ts_auth_methods
compat
```

Note: Querying the authentication methods commands.

- **lsauthpts**, at partition level
- **lsauthts**, at node level
- **splstdata -p**, at partition level
- **SDRGetObjects Syspar ts_auth_methods**

4.2.6 The kfserver and the server key files

In the PSSP versions earlier that 3.2, the server key files (/etc/krb-srvtab) files were transferred from the CWS to the nodes via the SP Ethernet. In order to make this transfer more secure, starting with PSSP 3.2, the serial line is used for that purpose. On the CWS, the server keys are still created by the **setup_server** command, but placed in a different directory, /spdata/sys1/k4srvtabs.

The **install_cw** command adds the following line to the /etc/inetd.conf file, so that *kfserver* runs as a sub-server of *inetd*:

```
kfcli stream tcp nowait root /usr/lpp/ssp/install/bin/kfserver kfserver
```

The *kfserver* script is run by inetd upon request from a node for its Kerberos V4 srvtab file. When the client connects to the server, kfserver queries the socket for the node's IP address requesting its srvtab file. kfserver locates the srvtab file and sends it over the s1term in write mode.

On the node side, the server-key file transfer is initiated by the get_keyfiles script.This program stops all getty processes and removes the "cons" entry from inittab so that it cannot restart until this command is completed. get_keyfiles sends a request to the control workstation and listens on /dev/tty0 for keyfiles. Keyfiles are sent in a uuencoded format over s1term by the control workstation. This program will uudecode the keyfile to its original format and place it in the /spdata/sys1/k4srvtabs directory of the node. Once the keyfile transfer is completed, the "cons" entry is added back to inittab, and inittab is refreshed.

The get_keyfiles script is invoked during node installation or customization, by the psspfb_script. The **get_keyfiles** command can also be used to transfer the server-key files from the CWS, in case they become corrupted on the nodes.

The log files are located in the /var/adm/SPlogs/kfserver directory on the CWS for the kfserver script, respectively /var/adm/SPlogs/get_keyfiles/get_keyfiles.log on the nodes, for the **get_keyfiles** command.

An example of how the logs look like during the transfer of the key files follows in Example 4-9 and Example 4-10.

Example 4-9 Log file from the get_keyfiles command on node

```
sp3n06:> cat get_keyfiles.log

************************************************
*   Beginning of logging for -- get_keyfiles
***  Thu Jul 12 14:33:22 EDT 2001
Parsing operands
Removing cons (getty) entry from inittab
Finding getty process
Getting my node number
Sending request for keyfile
Setting up the socket
Opening temporary keyfile for writing
get_keyfiles: 2545-113 KF_PORT was not set - trying port 32801
Connecting to port 32801
Opening /dev/tty0
Waiting for keyfile
Uudecoding the keyfile
Adding cons (getty) entry to inittab
Closing files and socket connection

************************************************
***  End of logging for -- get_keyfiles
***  Thu Jul 12 14:33:30 EDT 2001
```

Example 4-10 Kfserver log on CWS

```
sp3en0:> cat kfserver.log.20816

************************************************
*   Beginning of logging for -- kfserver
***  Thu Jul 12 14:33:27 EDT 2001

Received request from node 6
Uuencoding keyfile /spdata/sys1/k4srvtabs/sp3n06-new-srvtab
Sending keyfile to /dev/tty0 on sp3n06

************************************************
***  End of logging for -- kfserver
```

4.2.7 Implementing RRA

RRA can be turned on by changing the restrict_root_rcmd attribute to true in the SP_Restricted SDR class. This can be achieved by using the Site environment SMIT panel or by using the **spsitenv** command. This command performs the following:

- Checks prerequisites for RRA.

- Updates the restrict_root_rcmd attribute.
- Runs **setup_CWS**, which calls **updauthfiles** on the CWS.
- Acquires security credentials and uses **dsh** to run the **updauthfiles** command on all active nodes.

In RRA mode the remote commands principals used in scripts (rcmd.<interface>) are not automatically authorized to remotely access from nodes to CWS or from node to node. Remote commands, such as the rcmd principal, can only be executed from CWS to nodes. All the PSSP commands that use **rsh** or **rcp** check the restrict_root_rcmd attribute each time remote access is required. They will automatically use the sysctl method if RRA is enabled.

When RRA is enabled, the **updauthfiles** command removes all known SP-generated entries in the remote command authorization files and adds the following entries for each method specified by the auth_root_rcmd attribute:

- Standard AIX: /.rhosts is modified to contain only:
  - cwsname
  - Additional CWS interface names
- Kerberos V4: /.klogin is modified to contain only:
  - rcmd.cwsname@realm
- Kerberos V5: /.k5login is modified to contain only:
  - ssp/cwsname/spbgroot@ realm
  - hosts/cwsname/self@ realm

RRA limitations

RRA has the following limitations:

- All nodes and CWS need at least PSSP 3.2 in order to activate RRA.
- No VSD/GPFS node are allowed. During configuration and runtime, VSD and GPFS rely on existing combinations of **rsh** and **sysctl** commands and nested **sysctl** calls to access information on the control workstation as well as other nodes (node-to-node). The existing architecture relies on the existence of a common PSSP root identity that can be authorized successfully under **rsh** and **sysctl** ACLs. When restricted root access is enabled, the common PSSP root access required by VSD and GPFS is disabled.
- Boot/install servers are not automatically supported by PSSP when RRA is enabled. If multiple boot install servers are required, manual authorizations and configuration changes must be made by the customer.

- HACWS: Some manual operations may be required to maintain consistent authorization and configuration files between the CWS and backup CWS when using HACWS and RRA is activated.
- Certain Ecommands and system management commands only run from the CWS when RRA is enabled See *IBM RS/6000 SP: Planning, Volume 2, Control Workstation and Software Environment*, GA22-7281 for a complete list of such commands.
- HACMP: HACMP nodes make use of **rsh** and **rcp** to synchronize the HACMP ODM and to run c-spoc commands. In Enhanced Security mode, Kerberos 4 is used as authentication and authorization method. This mode requires modifications to the SP Kerberos 4 settings, which can be achieved by using the HACMP provided cl_setup_kerberos script. When RRA is enabled, the actions performed by cl_setup_kerberos script are no longer possible due to its rsh and rcp requirements from a node to the control workstation. Without using cl_setup_kerberos, the synchronization capability can be enabled by editing the /.klogin files on the HACMP nodes, explicitly allowing both systems to have root access to each other, and by manually adding the HACMP principals to the Kerberos V4 database.

sysctl

sysctl is an authenticated client-server application that runs commands with root privileges on nodes. It is implemented by the **sysctld** server daemon running with root privileges on all nodes and CWS. When a **sysctl** client connects to a **sysctld** server, two processes need to successfully complete before executing the request: authentication and authorization.

Sysctl authentication Sysctl servers come with a default svcconnect of AUTH, meaning the client must be authenticated by the server before allowing the client's command requests to be processed. This setting can be changed by editing the server's configuration file, /etc/sysctl.conf, and then restarting the server. However, that this is not a recommended way to run a Sysctl server because it takes away the first tier of Sysctl protection. The authentication is done using the methods specified in the ts_auth_methods attribute in the SDR Syspar class and should be a combination of Kerberos V5 and Kerberos V4, with Kerberos V4 as the last method in the list.

Sysctl authorization After a the server authenticates the client, it determines whether the client is authorized to execute the requested **sysctl** command. A **sysctl** command has one of four possible authorizations associated with it: NONE - any user can run the command; AUTH - any authenticated user can run the command; ACL - only principals that appear in the command's ACL can run the command; SYSTEM - only the server can run the command. Clients are not permitted to run the command. Sysctl ACLs come in two formats: Kerberos V4 ACLs, used in compat mode, and DCE ACL objects, used in dce mode. Note that a sysctl server can run under a combined dce:compat mode, case when the ACLs are checked in the same order. If the DCE ACL authorization succeeds, the Kerberos V4 ACL is no longer checked.

Sysctl use with no ts_auth_methods attribute selected

When no authentication method is set up, the trusted services use what is called the NONE/std method. This, in reality, is not an authentication method. It merely describes the fact that no authentication method has been set on the node for SP Trusted Services. When NONE/std method is used, sysctl behaves in a special manner. Although NONE/std implies that no authentication method is set on the node, sysctl authenticates and authorizes clients using their AIX identity.

Sysctl ACLs support two types of entries that implement the authorization of clients: _principal and _other_unauth. The _principal entry has an identity of form <user_name>@<host_name>, where <host_name> is the fully qualified host name of the node. The _other_unauth entry does not have any identity. The authorization policy can be described as follow: when the identity of the client can be matched with one _principal entry's identity, then access is permitted. If no match can be found, sysctl looks for the _other_unauth entry. If not found, the client is denied access. If found, the client is allowed permission. The _other_unauth entry is being used only for NONE/std. When an authentication method (for this discussion, compat) is set on the node and the client fails authentication, access is denied outright, regardless of the _other_unauth entry being present in the sysctl ACL file or not.

The following ACLs need to be modified in such an environment:

- On the CWS only:
  - /etc/sysctl.haem.acl
  - /etc/sysctl.install.acl
  - /etc/sysctl.rootcmds.acl
- On all the nodes:
  - /etc/logmgt.acl

For more information about Sysctl, see the **`sysctl`** and **`sysctld`** man pages, and the Sysctl online help.

Sysctl invocation in RRA mode

As an example of Sysctl use in RRA mode, we present the **spauthconfig** command. For the sake of simplicity, we deal with compat mode for the ts_auth_methods class. The **spauthconfig** script, among other things, remote copies the /etc/krb.conf and /etc/krb.realms from the CWS to nodes. But as the nodes are not allowed to remote copy files from CWS in RRA mode, sysctl is used instead. The sequence is as follows:

- Node executes **/bin/ksrvtgt root SPbgAdm** to get a Kerberos ticket as root.SPbgAdm principal.
- Node calls the client command to be executed by the **sysctld** server on the CWS:" **/usr/lpp/ssp/bin/sysctl -L -h $cw_name update_krb_files**" This is a sysctl procedure contained in the install.cmds module.
- The sysctld daemon on the CWS authenticates the root.SPbgAdm principal and then checks the ACL file /etc/sysctl.install.acl for the authorization to use the update_krb_files sysctl command.
- Because the ACL contains the _PRINCIPAL root.SPbgAdm line, the sysctld daemon on the CWS executes the request by getting a rcmd principal ticket-granting-ticket and pushing the files to the node.

Note: If you plan to use the firstboot.cust script in RRA mode, see the /usr/lpp/ssp/samples/firstboot.cust sample.

Troubleshooting security problems in RRA mode

For exemplification, consider the following scenario in a RRA environment:

Server keys are changed on a node, using the **ksrvutil change** command as in Example 4-11. We kill the **get_keyfiles** process, in order to allow the **psspfb_script** to continue.

Example 4-11 Changing the server keys

```
sp3n09 > klist -srvtab
Server key file:   /etc/krb-srvtab
Service         Instance        Realm      Key Version
-------------------------------------------------------
rcmd            sp3n09          SP3EN0         2
root            SPbgAdm         SP3EN0         3
sp3n09 > ksrvtgt root SPbgAdm
sp3n09 > klist
Ticket file:    /tmp/tkt10704
Principal:      root.SPbgAdm@SP3EN0

  Issued          Expires         Principal
Jul 24 09:46:38  Never           krbtgt.SP3EN0@SP3EN0
```

```
sp3n09 > kinit root.admin
Kerberos V4 Initialization for "root.admin"
Password:
sp3n09 > ksrvutil change

Principal: rcmd.sp3n09@SP3EN0; version 2
Changing to version 3.
Key changed.

Principal: root.SPbgAdm@SP3EN0; version 3
Changing to version 4.
Key changed.
Old key file in /etc/krb-srvtab.old.
```

Because the SPbgAdm principal is included in the server key on the node, its key version is changed on the node and in the Kerberos database. See Example 4-12. Notice also that the **ksrvtgt** command fails.

Example 4-12 Key versions on the CWS

```
sp3en0 > lskp root.SPbgAdm
root.SPbgAdm        tkt-life: Unlimited key-vers: 4  expires: 2037-12-31 23:59
sp3en0 > ksrvutil list
Version    Principal
   3     hardmon.sp3en0@SP3EN0
   3     rcmd.sp3en0@SP3EN0
   3     hardmon.sp3cws@SP3EN0
   3     rcmd.sp3cws@SP3EN0
   3     root.SPbgAdm@SP3EN0
sp3en0 > ksrvtgt root SPbgAdm
ksrvtgt: 2504-062 Incorrect Kerberos V4 password
```

A different node is set to customize and when pssp_script runs on it, it hangs while running the **get_keyfiles** command. This is happening because the **kfserver** on the CWS can no longer use the s1term, as it uses the **ksrvtgt root SPbgAdm** command to get a ticket-granting ticket, making use of the local /etc/krb-srvtab file to provide a Kerberos password. The root.SPbgAdm has now different key versions in the local /etc/krb-srvtab and in the Kerberos database, so the password provided by the node does not match the one stored in the database. See Example 4-13 on page 178 for the kfserver errors.

Example 4-13   Kfserver errors

```
sp3en0 > cat /var/adm/SPlogs/kfserver/kfserver.log.25342

************************************************
*   Beginning of logging for -- kfserver
***  Mon Jul 23 18:44:49 EDT 2001

Recieved request from node 12
Uuencoding keyfile /spdata/sys1/k4srvtabs/sp3n12-new-srvtab
ksrvtgt: 2504-062 Incorrect Kerberos V4 passwordSending keyfile to /dev/tty0 on
sp3n12
sp3en0 >
```

At some point, the psspfb_script attempts to get a ticket-granting ticket as root.SPbgAdm, in order to use the **sysctl complete_node** command. But as the key versions differ in the local /etc/srvtab and in the Kerberos database, the command fails. See the Example 4-14 for the psspb_script output.

Example 4-14   psspfb_script output

```
+ + /usr/lpp/ssp/bin/SDRGetObjects -Gx SP_Restricted restrict_root_rcmd
RestrictRSH=true
+ RestrictRSH=true
+ [[ true = true ]]
+ /bin/ksrvtgt root SPbgAdm
ksrvtgt: 2504-062 Incorrect Kerberos V4 password
+ rc=62
+ [[ 62 -ne 0 ]]
+ /usr/lpp/ssp/bin/spmsg_basic sminstall sminstall.cat emsg893 $pn: 0016-893
Failed to obtain rcmd ticket; rc=$rc. Continuing. psspfb_script 62
psspfb_script: 0016-893 Was not successful to obtain rcmd ticket; rc=62.
Continuing.+ print

+ /usr/lpp/ssp/bin/sysctl -L -h sp3en0 complete_node sp3n12 12 hdisk0
bos.obj.ssp.433
sp3en0: sysctl:  2501-122 complete_node: Insufficient Authorization.
```

To fix the problem, re-create the server keys on the CWS, for the CWS and for the node to be customized as in Example 4-15.

Example 4-15   Replacing the server keys on the CWS and node

```
sp3en0 > ksrvutil delete

Principal: hardmon.sp3en0@SP3EN0; version 3
Delete this key? (yes,no) [yes] no
Keeping this key.
```

```
Principal: rcmd.sp3en0@SP3EN0; version 3
Delete this key? (yes,no) [yes] no
Keeping this key.

Principal: hardmon.sp3cws@SP3EN0; version 3
Delete this key? (yes,no) [yes] no
Keeping this key.

Principal: rcmd.sp3cws@SP3EN0; version 3
Delete this key? (yes,no) [yes] no
Keeping this key.

Principal: root.SPbgAdm@SP3EN0; version 3
Delete this key? (yes,no) [yes] yes
Deleting this key.
Old key file in /etc/krb-srvtab.old.
sp3en0 > ext_srvtab -n SPbgAdm
Generating 'SPbgAdm-new-srvtab'....
sp3en0 > cat SPbgAdm-new-srvtab >>/etc/krb-srvtab
sp3en0 > rm SPbgAdm-new-srvtab
sp3en0 > ksrvutil list
Version    Principal
   3     hardmon.sp3en0@SP3EN0
   3     rcmd.sp3en0@SP3EN0
   3     hardmon.sp3cws@SP3EN0
   3     rcmd.sp3cws@SP3EN0
   4     root.SPbgAdm@SP3EN0
sp3en0 > export KRBTKFILE=/tmp/tkt$$
sp3en0 > ksrvtgt root SPbgAdm
sp3en0 > kdestroy
Tickets destroyed.
sp3en0 > ksrvutil -f ./sp3n12-new-srvtab list
Version    Principal
   1     rcmd.sp3sw12@SP3EN0
   1     godm.sp3n12@SP3EN0
   1     rcmd.sp3n12@SP3EN0
   1     root.SPbgAdm@SP3EN0
sp3en0 > lskp rcmd.sp3n12
rcmd.sp3n12          tkt-life: Unlimited key-vers: 1  expires: 2037-12-31 23:59
sp3en0 > ksrvutil -f ./sp3n12-new-srvtab delete

Principal: rcmd.sp3sw12@SP3EN0; version 1
Delete this key? (yes,no) [yes] no
Keeping this key.

Principal: godm.sp3n12@SP3EN0; version 1
Delete this key? (yes,no) [yes] no
Keeping this key.
```

```
Principal: rcmd.sp3n12@SP3EN0; version 1
Delete this key? (yes,no) [yes] no
Keeping this key.

Principal: root.SPbgAdm@SP3EN0; version 1
Delete this key? (yes,no) [yes] yes
Deleting this key.
Old key file in ./sp3n12-new-srvtab.old.
sp3en0 > ksrvutil -f ./sp3n12-new-srvtab add
Name: root
Instance: SPbgAdm
Realm: SP3EN0
Version number: 4
New principal: root.SPbgAdm@SP3EN0; version 4
Is this correct? (yes,no) [yes] yes
Password:
Verifying, please re-enter Password:
Key successfully added.
Would you like to add another key? (yes,no) [yes] no
Old key file in ./sp3n12-new-srvtab.old.
sp3en0 > ksrvutil -f ./sp3n12-new-srvtab list
Version    Principal
   1     rcmd.sp3sw12@SP3EN0
   1     godm.sp3n12@SP3EN0
   1     rcmd.sp3n12@SP3EN0
   4     root.SPbgAdm@SP3EN0
```

In conclusion, it is a good practice to verify if the key versions for the root.SPbgAdm principal are identical on the nodes and in the Kerberos database.

4.3 Troubleshooting

The following publications provide information about the SP Security Services:

- *IBM RS/6000 SP: Planning, Volume 2, Control Workstation and Software Environment*, GA22-7281
- *PSSP: Installation and Migration Guide*, GA22-7347
- *PSSP: Diagnosis Guide*, GA22-7350
- *PSSP: Messages Reference*, GA22-7352
- *PSSP: Administration Guide*, SA22-7348
- *PSSP: Command and Technical Reference*, SA22-7351

When troubleshooting security services, try to get a detailed description of the symptom, and if possible, reproduce the problem. Take a look at the logs and at the error messages and consult the Messages Reference. Find out about your security configuration. Determine the authentication and authorization methods in use. Keep track of the actions taken and be prepared to revert the changes made to the system.

4.3.1 Security-related log files

The SP security components use the following files:

- /var/adm/SPlogs/auth_install/log

  Contains the progress and completion status of the PSSP security configuration scripts.

- **/var/adm/SPlogs/get_keyfiles/get_keyfiles.log**

  On the nodes, it contains the output of the **`get_keyfiles`** command.

- **/var/adm/SPlogs/kfserver**

  Directory on the CWS containing the output of the **`kfserver`** command.

- /var/adm/SPlogs/kerberos/kerberos.log

  Kerberos log, when it runs as primary authentication server.

- /var/adm/SPlogs/kerberos/kerberos_slave.log

  Kerberos log, when running as a secondary authentication server.

- **/**var/adm/SPlogs/sysctl

  Directory containing the *syslogd* output.

For a list of DCE daemons logs, refer to *IBM DCE for AIX, Version 3.1: Administration Guide - Core Components,* LK3T-4401*.*

4.3.2 Debugging krshd

When debugging remote commands and Kerberos V4 is the authentication mechanism in use, the AIX syslog can be used to capture the krshd messages.

To use syslog, on the target system:

1. Create your log file using the **`touch`** command. The file must exist before syslog will write to it.
2. Edit the /etc/syslog.conf file and add the line:

***.debug file_name** Where file_name is your log file, with the full path name specified.

3. Refresh the syslog subsystem to start logging, by issuing these commands:

- `stopsrc -s syslogd`
- `startsrc -s syslogd`

On the source host, issue the command you are trying to debug. On the target host, check your log file for krshd or kerberos errors. Remember to un-configure the /etc/syslog.conf file when you are done and to refresh the syslogd daemon.

> **Tip:** To see all the error messages produces by remote commands, ensure that K5MUTE=0.

4.3.3 Case studies

The following case studies are only examples of how to approach an error in the SP security services. Fixing the problems in the presented case studies may also be achieved in different ways from the one shown in this chapter. The case studies presented do not cover all problems that could arise related to security. For a complete list of possible problems, see the *Parallel System Support Programs for AIX Diagnosis Guide*, GA22-7350-02 .

krshd: Kerberos authentication failed

Symptom: The remote commands from the CWS to one node fail with the error message in Example 4-16 on page 182.

Example 4-16 Kerberos authentication failure

```
sp3en0:> rsh sp3n06 date
krshd: Kerberos Authentication Failed.
spk4rsh: 0041-004 Kerberos V4 rcmd failed: rcmd protocol failure.
rshd: 0826-813 Permission is denied.
```

After configuring syslog on sp3n06 as in “Debugging krshd” on page 181, the messages from krshd are shown in Example 4-17.

Example 4-17 krshd output

```
sp3n06:> tail -f /var/adm/syslog.log
Jul 14 14:03:57 sp3n06 syslogd: restart
Jul 14 14:04:04 sp3n06 krshd[11950]: Failed krb5_compat_recvauth
Jul 14 14:04:04 sp3n06 krshd[11950]: Authentication failed from sp3en0: Unknown
code krb5 205
```

So far, the error cause remains undetermined, so we check the Kerberos tickets. We acquire new tickets to ensure we eliminate error causes. The actual commands used are shown in Example 4-18.

Example 4-18 Checking kerberos tickets

```
sp3en0:> kdestroy
Tickets destroyed.
sp3en0:> kinit root.admin
Kerberos V4 Initialization for "root.admin"
Password:
sp3en0:> klist
Ticket file:    /tmp/tkt29494
Principal:      root.admin@SP3EN0

  Issued          Expires         Principal
Jul 14 14:10:09  Aug 13 14:10:09  krbtgt.SP3EN0@SP3EN0
sp3en0:> rsh sp3n06 date
krshd: Kerberos Authentication Failed.
spk4rsh: 0041-004 Kerberos V4 rcmd failed: rcmd protocol failure.
rshd: 0826-813 Permission is denied.
sp3en0:> klist
Ticket file:    /tmp/tkt29494
Principal:      root.admin@SP3EN0

  Issued          Expires         Principal
Jul 14 14:10:09  Aug 13 14:10:09  krbtgt.SP3EN0@SP3EN0
Jul 14 14:10:16  Aug 13 14:10:16  rcmd.sp3n06@SP3EN0
sp3en0:>
```

We notice that after issuing the command, a service ticket for the node has been acquired, but the error persists. We investigate the following components, as shown in Example 4-19.

- /etc/krb.realms
- ./klogin
- /etc/krb.conf
- Authentication and authorization settings
- Server keys versions on node and in the Kerberos database

Example 4-19 Checking kerberos components

```
sp3n06:> grep -E "sp3en0|sp3n06" /etc/krb.realms
sp3en0 SP3EN0
sp3n06 SP3EN0
sp3n06:> grep sp3en0 .klogin
rcmd.sp3en0@SP3EN0
sp3n06:> cat /etc/krb.conf
SP3EN0
SP3EN0 sp3en0 admin server
sp3n06:> splstdata -p | grep auth
auth_install    k4
```

```
auth_root_rcmd  k4
ts_auth_methods compat
auth_methods    k4:std
sp3n06:> splstdata -e | grep restrict
restrict_root_rcmd       false
sp3n06:> lsauthent
Kerberos 4
Standard Aix
sp3n06:> klist -srvtab
Server key file:   /etc/krb-srvtab
Service         Instance        Realm      Key Version
------------------------------------------------------
rcmd            sp3sw06         SP3EN0         1
rcmd            sp3n06          SP3EN0         1
root            SPbgAdm         SP3EN0         1
sp3en0:> lskp rcmd.sp3n06
rcmd.sp3n06          tkt-life: Unlimited key-vers: 1  expires: 2037-12-31 23:59

sp3n06:> ksrvutil -k list
Version         Key           Principal
   1     01010101 01010101  rcmd.sp3sw06@SP3EN0
   1     01010101 01010101  rcmd.sp3n06@SP3EN0
   1     f72a1f70 fb794ffb  root.SPbgAdm@SP3EN0
```

The keys displayed on the node by the **ksrvutil -k list** command do not look normal, so we create new server keys for the node, using the following steps, as in Example 4-20 on page 184:

1. Delete the existing server key file on the CWS.
2. On the CWS, change node responds to customize.
3. Run setup_CWS to get the new server keys created in the /spdata/sys1/k4srvtabs directory.
4. Check the server keys.

Example 4-20 Building new server keys

```
sp3en0:> rm /spdata/sys1/k4srvtabs/sp3n06-new-srvtab
sp3en0:> spbootins -r customize -l 6 -s no
sp3en0:> setup_CWS
setup_CWS: 2545-105 kfcli service is already defined in /etc/inetd.conf
add_principal: 2502-037 rcmd.sp3n06 already exists in database.
add_principal: 2502-037 rcmd.sp3sw06 already exists in database.
setup_CWS: Control Workstation setup complete.
sp3en0:> ls -l /spdata/sys1/k4srvtabs/sp3n06-new-srvtab
-r--------   1 root     system       86 Jul 14 14:44
/spdata/sys1/k4srvtabs/sp3n06-new-srvtab
sp3en0:> klist -file /spdata/sys1/k4srvtabs/sp3n06-new-srvtab -srvtab
```

```
Server key file:   /spdata/sys1/k4srvtabs/sp3n06-new-srvtab
Service         Instance        Realm      Key Version
------------------------------------------------------
rcmd            sp3sw06         SP3EN0         1
rcmd            sp3n06          SP3EN0         1
root            SPbgAdm         SP3EN0         1
sp3en0:> ksrvutil -k -f /spdata/sys1/k4srvtabs/sp3n06-new-srvtab list
Version         Key           Principal
   1     01010101 01010101  rcmd.sp3sw06@SP3EN0
   1     01010101 01010101  rcmd.sp3n06@SP3EN0
   1     f72a1f70 fb794ffb  root.SPbgAdm@SP3EN0
```

At this point, we notice that the server key file created on the CWS has the same wrong keys. So we suspect a Kerberos database corruption, so we delete and re-add the rcmd principals for the node in cause, like shown in Example 4-21 on page 185.

5. Delete from the Kerberos database the rcmd principals for the node.
6. Delete the server key from the CWS
7. Repeat steps 3 and 4. **setup_CWS** re-adds the principals to Kerberos.
8. Change back the node response to disk.
9. On the CWS, get new Kerberos tickets.
10. On the node, run **get_keyfiles**.
11. Copy the server key to the /etc/krb-srvtab.

Example 4-21 Correcting the kerberos database

```
sp3en0:> rmkp rcmd.sp3n06
Confirm removal of principal rcmd.sp3n06? (yes or no): yes
sp3en0:> rmkp rcmd.sp3sw06
Confirm removal of principal rcmd.sp3sw06? (yes or no): yes
sp3en0:> rm /spdata/sys1/k4srvtabs/sp3n06-new-srvtab
sp3en0:> setup_CWS
setup_CWS: 2545-105 kfcli service is already defined in /etc/inetd.conf
setup_CWS: Control Workstation setup complete.
sp3en0:> lskp | grep 06
rcmd.sp3n06         tkt-life: Unlimited key-vers: 1  expires: 2037-12-31 23:59
rcmd.sp3sw06        tkt-life: Unlimited key-vers: 1  expires: 2037-12-31 23:59
sp3en0:>
sp3en0:> ksrvutil -k -f /spdata/sys1/k4srvtabs/sp3n06-new-srvtab list
Version         Key           Principal
   1     ba3eb045 1aae671f  rcmd.sp3sw06@SP3EN0
   1     10dad5bf eac7fd49  rcmd.sp3n06@SP3EN0
   1     f72a1f70 fb794ffb  root.SPbgAdm@SP3EN0
sp3en0:> spbootins -r disk -l 6 -s no
sp3en0:> kinit root.admin
```

```
Kerberos V4 Initialization for "root.admin"
Password:


sp3n06:> get_keyfiles sp3n06-new-srvtab sp3en0
Parsing operands
Removing cons (getty) entry from inittab
Finding getty process
Getting my node number
Sending request for keyfile
Uudecoding the keyfile
Adding cons (getty) entry to inittab
sp3n06:> cp /spdata/sys1/k4srvtabs/sp3n06-new-srvtab /etc/krb-srvtab
sp3n06:> ksrvutil -k list
Version         Key           Principal
   1     ba3eb045 1aae671f  rcmd.sp3sw06@SP3EN0
   1     10dad5bf eac7fd49  rcmd.sp3n06@SP3EN0
   1     f72a1f70 fb794ffb  root.SPbgAdm@SP3EN0
```

Now we can successfully issue a remote command from the CWS to the node.

telnetd: No authentication methods enabled

Consider the following example: Issuing telnet to a node returns the message in Example 4-22. The SDR settings are shown in Example 4-23.

Example 4-22 telnet error message

```
# tn sp3n06
Trying...
Connected to sp3n06.
Escape character is '^T'.
telnetd: No authentication methods enabled.
Connection closed.
```

Example 4-23 SDR settings

```
# SDRGetObjects Syspar auth_methods
auth_methods
k4
# lsauthent
Kerberos 4
Standard Aix
```

So, the partition has only k4 enabled, which is wrong because standard AIX should also be configured. We try to change it back, as shown in Example 4-24.

Example 4-24   Changing the auth_methods attribute

```
# chauthpar  -c -p sp3en0 -v k4 std
The remote command authentication methods for this host are currently k4:std
The authentication methods by partition are currently

        sp3en0  k4
The partition to be modified is sp3en0
The auth_methods attribute of the partition has been set to k4:std
# chauthpar  -f -p sp3en0 -v k4 std
The remote command authentication methods for this host are currently k4:std
The authentication methods by partition are currently

        sp3en0  k4:std
The partition to be modified is sp3en0
The chauthent command was executed successfully on node sp3n01
The chauthent command was executed successfully on node sp3n05
sp3n06: sp3n06: A remote host refused an attempted connect operation.
sp3n06: spk4rsh: 0041-011 Kerberos V4 rcmd failed:  :krshd: Kerberos
Authentication Failed.
sp3n06: .
dsh:  5025-509 sp3n06 rsh had exit code 4
chauthpar: 0016-226 A dsh problem has occurred on node sp3en0 (sp3n06). The
return code value is 1.
............................................................................
The chauthent command was executed successfully on node sp3n15

# SDRGetObjects Syspar auth_methods
auth_methods
k4:std
```

Now we notice that, on node sp3n06, the **rsh** command fails.

The only way of getting to the node is via the serial line. Use **s1term -w frame# slot#** to access the node and issue the commands shown in Example 4-25.

Example 4-25   Node accessed via serial line

```
sp3n06:> lsauthent
Kerberos 4
sp3n06:> klist -srvtab
Server key file:   /etc/krb-srvtab
Service         Instance        Realm      Key Version
------------------------------------------------------
root            SPbgAdm         SP3EN0         1

sp3n6:> cat .klogin
root.admin@SP3EN0
rcmd.sp3en0@SP3EN0
root.SPbgAdm@SP3EN0
```

```
rcmd.sp3n01@SP3EN0
rcmd.sp3n05@SP3EN0
rcmd.sp3n06@SP3EN0
rcmd.sp3n07@SP3EN0
rcmd.sp3n08@SP3EN0
rcmd.sp3n09@SP3EN0
rcmd.sp3n10@SP3EN0
rcmd.sp3n11@SP3EN0
rcmd.sp3n12@SP3EN0
rcmd.sp3n13@SP3EN0
rcmd.sp3n14@SP3EN0
rcmd.sp3n15@SP3EN0
sp3n06:> cat .rhosts
# Updated by updauthfiles script on Tue Mar 13 16:33:10 2001
sp3en0
sp3cws
sp3en0
sp3n15
sp3n14
sp3n13
sp3n12
sp3n11
sp3n10
sp3n09
sp3n08
sp3n07
sp3n06
sp3n05
sp3n01
# End of generated entries by updauthfiles script
```

We notice that the only authentication method in use is Kerberos 4. But as there is no service key for the rcmd.sp3n06 principal, the remote command from the CWS fails. We also notice that both the .rhosts and .klogin authorization files are in place.

Checking the auth_root_rcmd attribute in the SDR returns the output shown in Example 4-26.

Example 4-26 Node auth_root_rcmd attribute

```
sp3n06:> SDRGetObjects Syspar auth_root_rcmd
auth_root_rcmd
k4:std
```

Both authorization methods, k4 and std are configured, so that explains why both .klogin and .rhosts are present.

Next step, we take the following actions to solve the problem:

- Correct the settings on the CWS.
- Create the new server keys for the node and copy them to the node. The easiest way is by setting the node response to customize and then running the *pssp_script* on the node, or by rebooting the node.

The actual commands are presented in Example 4-27.

Example 4-27 Operations on the CWS

```
sp3en0:> splstdata -p
List System Partition Information

System Partitions:
------------------
sp3en0

Syspar: sp3en0
-------------------------------------------------------------------------------
syspar_name     sp3en0
ip_address      192.168.3.130
install_image   default
syspar_dir      ""
code_version    PSSP-3.2
haem_cdb_version 983230416,8126720,0
auth_install    k4:std
auth_root_rcmd  k4:std
ts_auth_methods compat
auth_methods    k4:std

sp3en0:> spsetauth -p sp3en0 -d k4
sp3en0:> splstdata -p
List System Partition Information

System Partitions:
------------------
sp3en0

Syspar: sp3en0
-------------------------------------------------------------------------------
syspar_name     sp3en0
ip_address      192.168.3.130
install_image   default
syspar_dir      ""
code_version    PSSP-3.2
haem_cdb_version 983230416,8126720,0
auth_install    k4:std
auth_root_rcmd  k4
```

```
ts_auth_methods compat
auth_methods    k4:std

sp3en0:> spbootins -r customize -l 6 -s no
sp3en0:> setup_server
.................................
setup_server: Processing complete (rc= 0).
```

Tip: Do not run pssp_script on the node from a serial line window because get_keyfiles, which transfers the server keys from the CWS, needs exclusive access to the serial line.

We have two choices to run the pssp_script without using the serial line for that:

1. Manually update the AIX authentication methods with the **chauthent** command, close the serial session, then **telnet** to the node from the CWS.
2. Reboot the node and close the serial connection.

The first option is used. After running pssp_script on the node, we check the server keys and the authorization files. The rcmd.sp3n06 principal has its own key in /etc/krb-srvtab and only the .klogin authorization file is present. See Example 4-28 for the actual commands.

Example 4-28 Node verification

```
sp3n06:> klist -srvtab
Server key file:   /etc/krb-srvtab
Service         Instance        Realm      Key Version
------------------------------------------------------
rcmd            sp3sw06         SP3EN0         1
rcmd            sp3n06          SP3EN0         1
root            SPbgAdm         SP3EN0         1
sp3n06:> lsauthent
Kerberos 4
Standard Aix
sp3n06:> ls -l .klogin .rhosts
ls: 0653-341 The file .rhosts does not exist.
-rw-r--r--   1 root     system      285 Jul 11 15:46 .klogin
```

5

RS/6000 Cluster Technology (RSCT)

This chapter applies to RSCT Version 1 Release 2 and provides a brief overview of the RSCT subsystems and problem determination techniques that assist in keeping the cluster stable.

In this chapter we discuss problem determination on the following:

- Topology Services (TS)
- Group Services (GS)
- Event Management (EM)

5.1 RS/6000 Cluster Technology (RSCT) overview

RSCT is a distributed group of subsystems, running across multiple nodes or machines, that communicate with each other through multiple networks to provide high availability, online monitoring and automatic recovery actions. This distributed group of subsystems, known as a *stack*, runs in a single partition on the RS/6000 SP. There may be more than one partition per SP but each RSCT stack is separate from the other.

The three principal components of the RSCT stack are:

1. Topology Services (TS)
2. Group Services (GS)
3. Event Management (EM)

This infrastructure is represented pictorially, as shown in Figure 5-1.

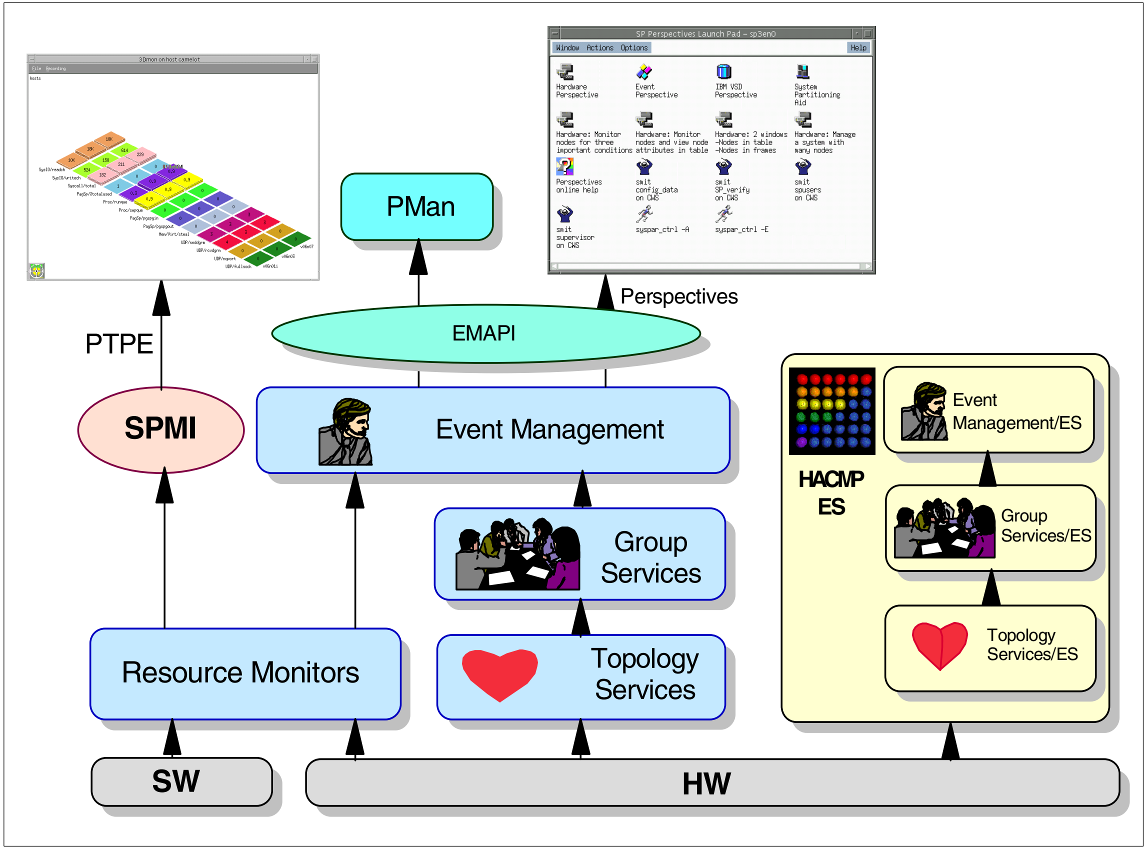


Figure 5-1 RSCT infrastructure

A HACMP/ES domain is also shown in this figure, which you can see contains another RSCT stack of subsystems. These stacks are independent of each other and although the problem determination of the RSCT stack in the HACMP environment is similar, it is not the intent of this book to detail the differences.

Refer to the following IBM Redbooks for more information about HACMP/ES:

- *HACMP Enhanced Scalability Handbook,* SG24-5328
- *HACMP/ES Customization Examples,* SG24-4498
- *HACMP Enhanced Scalability: User-Defined Events,* SG24-5327

5.2 Topology Services (TS)

Topology Services (TS) is the lowest level of the RSCT subsystems. It provides and maintains connectivity and availability information about the nodes and network adapters. Most problems are automatically recovered without intervention; however, often to understand or isolate a problem in a higher level of the RSCT system you need to examine the state of Topology Services.

5.2.1 TS overview

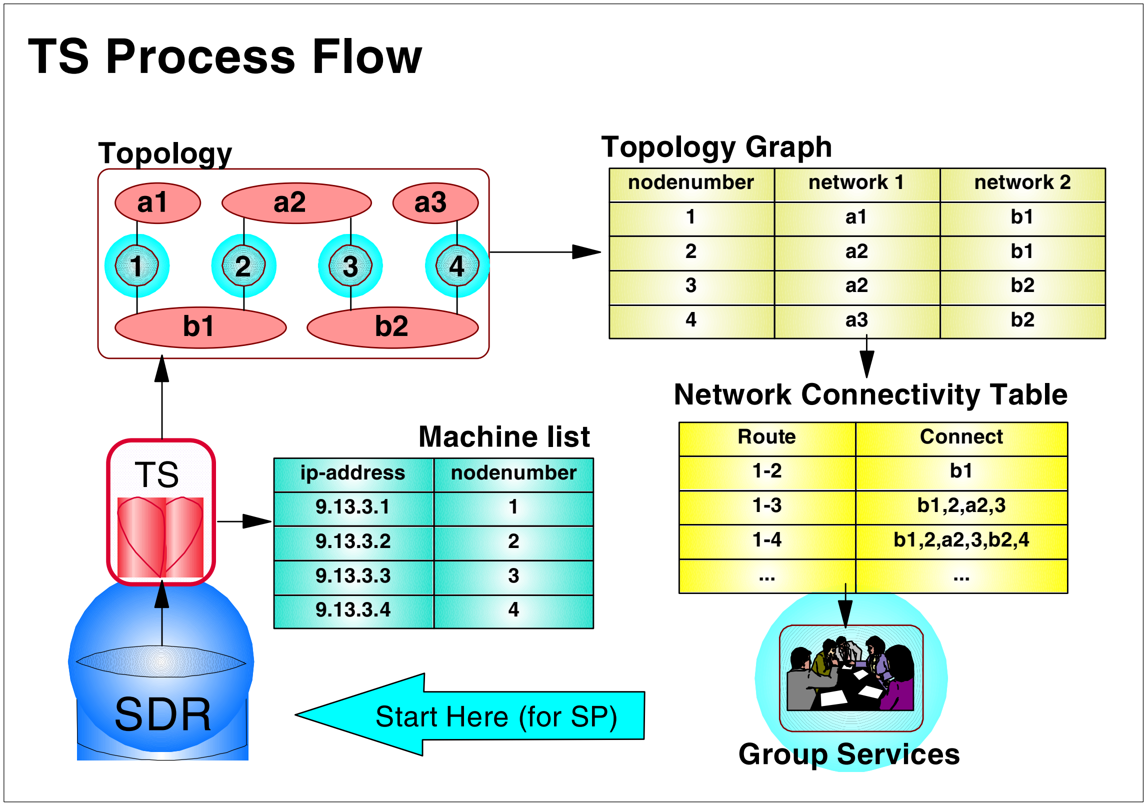
Topology Services is a distributed subsystem of the IBM RS/6000 Cluster Technology (RSCT) software on RS/6000 systems. The RSCT software provides a set of services that support high availability on your SP system. Other services in the RSCT software are the Event Management and Group Services distributed subsystems. These three distributed subsystems operate within a domain. A domain is a set of RS/6000 machines upon which the RSCT components execute and, exclusively of other machines, provide their services. On an SP system, a domain is a system partition. Note that a machine might be in more than one RSCT domain; the control workstation is a member of each system partition, and, therefore, a member of each RSCT domain. When a machine is a member of more than one domain, there is an executing copy of each RSCT component per domain.

Topology Services provides other high availability subsystems with network adapter status, node connectivity information, and a reliable messaging service. The adapter status and node connectivity information is provided to the Group Services subsystem upon request, Group Services then makes it available to its client subsystems. The Reliable Messaging Service, which takes advantage of node connectivity information to reliably deliver a message to a destination node, is available to the other high availability subsystems.

This adapter status and node connectivity information is discovered by an instance of the subsystem on one node, participating in concert with instances of the subsystem on other nodes, to form a ring of cooperating subsystem instances. This ring is known as a heartbeat ring, because each node sends a heartbeat message to one of its neighbors and expects to receive a heartbeat from its other neighbor. Actually each subsystem instance can form multiple rings, one for each network it is monitoring. Usually, each subsystem monitors two rings; the SP Ethernet and the SP switch. This system of heartbeat messages enables each member to monitor one of its neighbors and to report to the heartbeat ring leader, called the Group Leader, if it stops responding. The Group Leader, in turn, forms a new heartbeat ring based on such reports and requests for new adapters to join the membership. Every time a new group is formed, it lists which adapters are present and which adapters are absent, making up the adapter status notification that is sent to Group Services.

In addition to the heartbeat messages, connectivity messages are sent around all rings. Connectivity messages for each ring will forward its messages to other rings, so that all nodes can construct a connectivity graph. It is this graph that determines node connectivity and defines a route that Reliable Messaging would use to send a message between any pair of nodes that have connectivity.

Upon the startup of the Topology Services daemon, the initial configuration information is supplied from the SDR. This is used to build a Machine List file, and adapter groups are established and a topology table (connectivity and availability table) is built. This in turn is used to build the Network Connectivity Table (NCT) in shared memory, and this information is passed via the Reliable Messaging subsystem to Group Services (GS) as a client of Topology Services. This process flow is shown in Figure 5-2 on page 195.



TS Process Flow

Topology Graph

nodenumber	network 1	network 2
1	a1	b1
2	a2	b1
3	a2	b2
4	a3	b2

Network Connectivity Table

Route	Connect
1-2	b1
1-3	b1,2,a2,3
1-4	b1,2,a2,3,b2,4
...	...

Machine list

ip-address	nodenumber
9.13.3.1	1
9.13.3.2	2
9.13.3.3	3
9.13.3.4	4

Figure 5-2 Topology Services process flow

To manage the changes in heartbeat rings, the following roles are defined within Topology Services:

Group Leader (GL): The node whose adapter has the highest IP address; it *proclaims* the group and handles *join* requests and *death* notifications, coordinates with group members, and distributes connectivity information. The GL node will not necessarily be the same for the different heartbeat rings.

Crown Prince: The second highest IP address; detects the death of the Group Leader and takes over the GL role.

Mayor: Picked by the Group Leader to broadcast messages to the group members in a given subnet.

Generic: Any other member of the group, who monitors the heartbeat message from its neighbor and informs the Group Leader if there is a problem.

All of these roles are dynamic; they are continuously re-evaluated and reassigned.

5.2.2 What to use to diagnose Topology Services problems

The log files created by the TS daemon (hatsd) on the Group Leader, within each heartbeat ring contain the most relevant and main information in case of problems with TS.

> **Note:** With the introduction of First Failure Data Capture (FFDC) in PSSP 3.2, it is no longer expected that users do debugging using the HATS logs. The idea is to use lssrc and the error log entries. See Chapter 23, "Topology Services" in the *Parallel System Support Programs for AIX: Diagnosis Guide*, GA22-7350 for more information on debugging approach used.

TS service log glossary

To understand the TS logs you need to understand some of the more cryptic terms that are used throughout the logs.

PTC Prepare to commit. Message sent by GL to ring members.

Offset Network ring: [0] for SPether [1] for SPswitch. When the logs refer to adapter 0 it is referencing the SPether ring.

Tunstable Time-of-day when the instability timer was set. The main purpose is to ensure Group Services only receives a single "node" event. This is set when the initial group is formed and reset each time group membership changes.

TifUnstableSetNCT Related to previous timer, if this timer expires node membership is recomputed to form a valid NCT.

NCT Network connectivity table.

Refresh Quiesce Interval

Time period between a request for a refresh and actual changes to the hatsd data structures.

Refresh Propagation Interval

Time period for how often propagation messages will be sent.

Tunables Settings that apply to the hatsd. Some values come from the SDR class TS_Config others come from the daemon itself.

lssrc -ls

The TS subsystem is managed by the SRC, and since it does not use signals as its communication method we can get current status information by using the **lssrc -ls** command. We discuss this output in Example 5-1 on page 197.

> **Tip:** The output will be in English for all RSCT versions currently on the field.

*Example 5-1 **lssrc -ls** (on the CWS)*

```
          [3:root@sp3en0:]/home/root # lssrc -ls hats.sp3en0
          Subsystem         Group            PID     Status
Section 1  hats.sp3en0      hats             17440   active
          Network Name   Indx Defd Mbrs St Adapter ID      Group ID
Section 2 SPether        [ 0]   13   13  S 192.168.3.130   192.168.3.130
          SPether        [ 0] en0          0x43465891      0x43507660
Section 3 HB Interval = 1 secs. Sensitivity = 4 missed beats
            2 locally connected Clients with PIDs:
          haemd( 27040) hagsd( 32462)
Section 4  Configuration Instance = 983230443
            Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
            Control Workstation IP address = 192.168.3.130
            Daemon employs no security
            Data segment size: 6707 KB. Number of outstanding malloc: 346
            User time 953 sec. System time 1157 sec.
            Number of page faults: 40. Process swapped out 0 times.
            Number of nodes up: 13. Number of nodes down: 0.
```

Section 1 tells us the hatsd process id (PID), running on the CWS, and that it is active, or at least the socket connection is still open. The process could be hung.

> **Tip:** If you see any output at all from the **lssrc -ls** command, then the daemon cannot be hung, since all output from the **lssrc -ls** command is provided by the daemon itself. The “process could be hung” statement is valid for the **lssrc -s** command, but not for the **lssrc -ls** command.

Section 2 contains information about the heartbeat rings. This is the CWS so only the SPether network is present.

- Indx: Number used for the ring. [0] is for SPether [1] is for SPswitch.
- Defd: Number of nodes defined in this ring.
- Mbrs: Active members of this ring.
- St: State of the ring (S: stable, U: unstable, D: down).

- Adapter ID: The IP address of the adapter in this ring and a hexadecimal number used by TS to determine the instance number for the current node.
- Group ID: The IP address of the Group Leader of this ring and a hexadecimal number to determine the instance number of the group.

Section 3 shows the tunables for this specific ring. In this instance, they are set at the default.

Section 4, corresponds to the *InstanceNumber* of the machines.lst file and the contents of the machines.inst file.

To review these files you can run:

- # SDRRetrieveFile hats.machines.lst /tmp/machines.lst
- # SDRRetrieveFile hats.machines.inst /tmp/machines.inst
- Or look in the /var/ha/run/<syspar_name>/machines.lst file

The last portion of the output covers:

- The CWS IP address.
- The Security Setting of the hatsd (we will discuss this later in the chapter).
- The hatsd process CPU time in user and system space.
- The number of page faults and the number of times the process has been swapped out.
- The number of nodes either reachable (up) or not (down).

> **Attention:** The configuration instance changes every time the subsystem is refreshed.

Now we look at the output of **`lssrc -ls hats`** on a node, see Example 5-2.

Example 5-2 lssrc -ls (on a node)

```
[0:root@sp3n10:]/home/root # lssrc -ls hats
Subsystem         Group            PID     Status
 hats             hats             8544    active
Network Name   Indx Defd Mbrs St Adapter ID      Group ID
SPether        [ 0]   13   13  S 192.168.3.10    192.168.3.130
SPether        [ 0] en0          0x4350763f      0x43507660
HB Interval = 1 secs. Sensitivity = 4 missed beats
SPswitch       [ 1]   12   11  S 192.168.13.10   192.168.13.15
SPswitch       [ 1] css0         0x435076ed      0x4350770e
HB Interval = 1 secs. Sensitivity = 4 missed beats
  2 locally connected Clients with PIDs:
haemd(  7004) hagsd(  9988)
```

Section 1

```
  Configuration Instance = 983230443
  Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
  Control Workstation IP address = 192.168.3.130
  Daemon employs no security
  Data segment size: 6764 KB. Number of outstanding malloc: 383
  User time 40 sec. System time 52 sec.
  Number of page faults: 130. Process swapped out 0 times.
  Number of nodes up: 13. Number of nodes down: 0
```

Section 1 shows us the other ring being the [1] SPswtich network. Note that the **Defd** members show 12, as the CWS is not a member of this ring. Also the **Mbrs**, the active members of the ring only show as 11. One adapter in the SPswitch ring is not communicating. The Group Leader of the SPswitch ring is the highest IP address of the ring.

SDRGetObjects

`SDRGetObjects host_responds` is used for confirmation of the host_responds attribute in the *host_responds* class of the SDR. Shown in Example 5-3.

Example 5-3 SDRGetObjects host_responds

```
[root@sp4en0]:/> SDRGetObjects host_responds
node_number  host_responds
          1            1
          5            1
```

> **Note:** The information flow from Topology Services to the SDRGetObjects host_responds is as follows: Topology Services (ethernet adapter membership) to Group Services to Event Management to host responds (hr) daemon to finally the SDR host_responds object.

`SDRGetObjects TS_Config` shows the tunable parameters of the hats daemon.

Example 5-4 SDRGetObjects TS_Config

```
# SDRGetObjects TS_Config
Frequency    Sensitivity  Run_FixPri   FixPri_Value Log_Length   Pinning
          1            4            1           38         5000 ""
```

hatstune

This provides a easier interface to change the SDR attributes of the TS_Config class. A Usage screen shot is shown in Example 5-5 on page 200. The flags not directly related to the TS_Config tunable parameters are:

-d This resets all the tunable parameters to their default.

-v This displays the current values.

-r This *refreshes* hatsd after the tunable parameters are successfully set.

-h Help screen.

Note: The command **hatstune** was introduced in PSSP 3.2 as a replacement for making direct changes to TS_Config SDR class using the command **SDRChangeAttrValues**. HATS tunable parameters can be set by changing SDR attributes of certain SDR objects. hatstune performs consistency checking on on the tunable values.

- **hatstune** must be executed on the CWS (and not at the nodes) when used to change tunable values.

For more information on the **hatstune** command, refer to Parallel System Support Programs for AIX: Command and Technical Reference, Volume 1, SA22-7351.

Example 5-5 hatstune -h

```
# hatstune
Usage:
    hatstune [-f frequency] [-s sensitivity] [-p priority] [-l log_length]
        [-m pin_object] [-r]
    hatstune -d [-r]
    hatstune -v
    hatstune -h
```

A listing of the TS_Config class is shown in Example 5-4 on page 199. You could use the **SDRChangeAttrValues** instead of the **hatstune** command with the disadvantage that no consistency checking is done.

hatsctrl

The hatsctrl script is not usually executed from the command line. It is normally called by the **syspar_ctrl** command during installation of the system, and partitioning or repartitioning of the system.

The hatsctrl script provides a variety of controls for operating the Topology Services subsystem:

- Adding, starting, stopping, and deleting the subsystem.
- Cleaning or deleting the subsystem from all system partitions. This does not remove the SDR entries.

- Unconfiguring the subsystems from all system partitions. This removes the SDR entries.
- Turning tracing on and off.
- Refreshing the subsystem.

5.2.3 Topology Service logs and other useful information

Refer to the following logs and files for useful problem determination infrastructure:

- AIX Error Log: Entries for Topology Services are usually from the SRC master process and are related to starting or stopping the SRC subsystem. The tables in the *PSSP 3.2 Diagnosis Guide, Diagnosing Topology Services Problems*, GA22-7350 provide Error Labels and Error IDs, as well as an explanations and details.

> **Note:** The error log entries for HATS are always created by the HATS daemon itself, unless the daemon is killed or core dumps.

- /var/ha/log/hats.<syspar>: This is the log file for the hats script errors called by the SRC master. There will be multiple log files on the CWS if there are multiple system partitions.

> **Note:** The hats script controls the operations of the Topology Services subsystem, builds the machines.lst file, and then invokes the daemon.

- /var/ha/log/hats.<DOM>.<HHMMSS>.<syspar>: Where <DOM> is the day of the month that the daemon was started. This file is created by the hats daemon, if the hats script fails the reason, you would not see this file, and the reason should be in the hats.<syspar> file. The most relevant information will be on the node that is the Group Leader.
- /var/ha/run/hats.<syspar>: The Topology Services daemon current working directory. If the TS daemon abnormally terminates, the core file will be here, named core.<DOM>.<HHMMSS>.<syspar>.
- /var/ha/soc/hats/server_socket.<syspar>: This is the socket file used for intranode communication.
- /var/ha/run/hats.<syspar>/machine.lst: This contains all the IP addresses for all the supported networks. This is the information originally obtained from the SDR, verified and compared back against the SDR, if they are different this file will replace the one in the SDR.

5.3 Group Services (GS)

Group Services (GS) is a client of Topology Services, and is the next level in the RSCT structure. GS provides coordination and synchronization services to client subsystems, such as Event Management (EM) and Recoverable Virtual Shared Disk (RVSD).

Refer to the following IBM publications for more information about Group Services:

- *Parallel System Support Programs for AIX: Diagnosis Guide*, GA22-7350
- *Chapter 25, Group Services Subsystem in the Parallel System Support Programs for AIX: Administration Guide,* SA22-7348
- *RSCT Group Services: Programming Cluster Applications,* SG24-5523

5.3.1 GS overview

GS runs as a distributed daemon (hagsd) running on all nodes in a system partition, communication between the nodes is through the Reliable Messaging Library. On the CWS there will be one instance of the hagsd daemon running for each system partition. The GS structure is shown in Figure 5-3 on page 203.

.

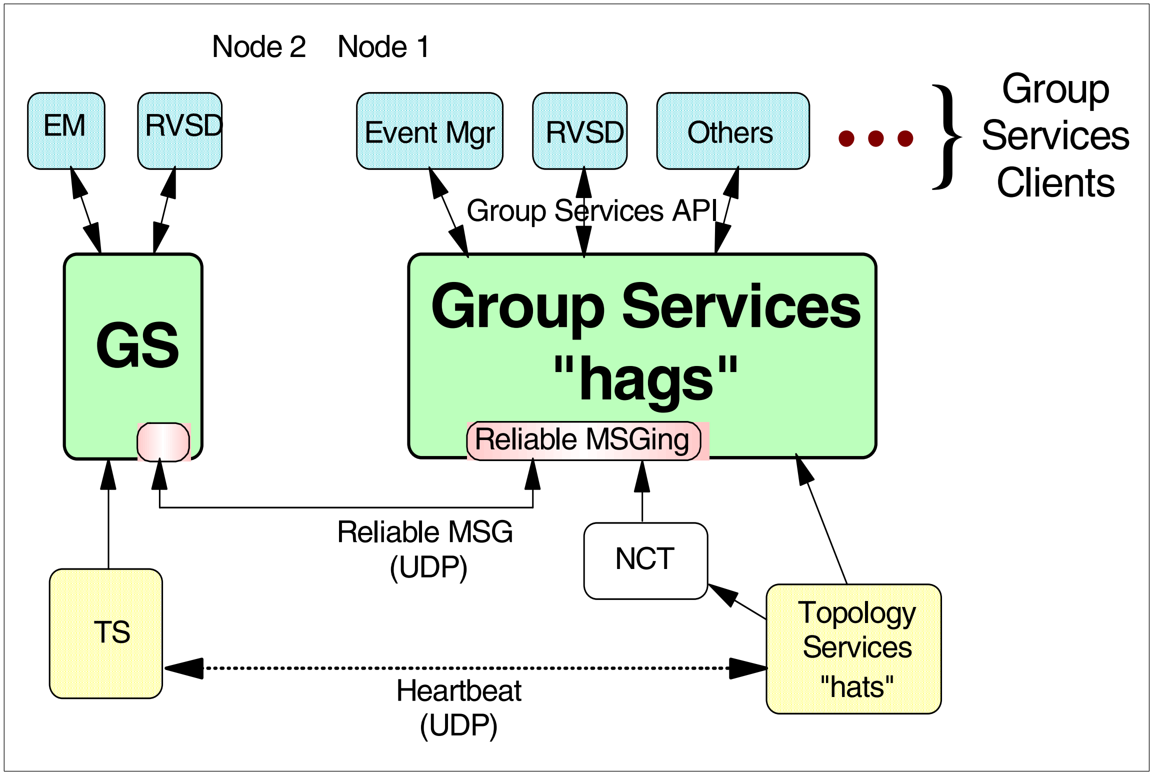


Figure 5-3 Group Services structure

GS clients are either providers, processes that join the group, or subscribers, that monitor the group. The group state, is maintained by the GS subsystem consisting of a group membership list, a list of the providers, and a group state value, this is controlled by the providers. Subscribers do not appear in the group membership lists, they are known to the GS subsystem but not by the providers. The client subsystems connect to GS and form groups by using the Group Services API (GSAPI).

Any provider in a group can initiate a change to the group state, by either joining or leaving the group. Changes to the group state are *serialized*, that is the change must complete before another change can start.

GS establishes a single group namespace across a SP partition. Each SP partition would be a separate namespace and can have more than one group within that namespace. To keep track of the changes to the client groups GS nominates a nameserver (NS), this is not the same as a DNS nameserver. The nameserver, if all nodes are booted at once, will be the node with the lowest IP address. If there is a GS daemon already running within a namespace, then it will be the NS and will remain so taking responsibility for tracking group state changes.

5.3.2 What to use to diagnose Group Services problems

The following commands can be utilized to diagnose GS problems:

lssrc -ls

The GS subsystem is also managed by the SRC, and since it does not use signals as its communication method we can get current status information by using the **lssrc -ls** command, we will discuss the output in Example 5-6.

Example 5-6 lssrc -ls hags.sp5en0

```
            # lssrc -ls hags.sp5en0
            Subsystem         Group            PID     Status
             hags.sp5en0      hags             14670   active
            2 locally-connected clients.  Their PIDs:
            16312(hagsglsmd) 25212(haemd)
            HA Group Services domain information:
Section 1   Domain established by node 5
            Number of groups known locally: 2
                               Number of   Number of local
            Group name         providers   providers/subscribers
Section 2   cssMembership            3           0          1
            ha_em_peers              4           1          0
```

Section 1 tells us the GS nameserver for this domain, node 5. Remember, although the GS nameserver would be the lowest IP address, if the nameserver is already established then other nodes join with the existing nameserver.

Section 2 shows the two groups in the domain. The number of providers are the nodes that can effect change to the group status. Since we are showing output on the CWS we see one local subscriber to the *cssMembership* group, the CWS does not have a switch adapter but wants to know the status of this group

Let us briefly describe the slightly different output show in Example 5-7.

Example 5-7 lssrc -ls hags.sp5en0 (problem)

```
# lssrc -ls hags.sp5en0
Subsystem         Group            PID     Status
 hags.sp5en0      hags             27662   active
2 locally-connected clients.  Their PIDs:
5832(haemd) 20078(hagsglsmd)
```

Section 1

```
HA Group Services domain information:
Domain not established.
Number of groups known locally: 0
```

Section 1 shows us that hagsd is active but the domain has not been established yet. Waiting a couple of minutes would normally allow the groups to be established but if there is a problem and the command times out check that hatsd is running.

hagsns

The **hagsns** command will also show information regarding the nameserver. Let's run **hagsns -s hags.sp5en0** on the CWS as in Example 5-8.

Example 5-8 hagsns -s hags.sp5en0

```
# hagsns -s hags.sp5en0
HA GS NameServer Status
NodeId=0.4, pid=24706, domainId=5.14, NS established, CodeLevel=PSSP-3.2(DRL=5)
NS state=kCertain, protocolInProgress=kNoProtocol,
outstandingBroadcast=kNoBcast
Process started on Jul 22 14:49:38, (0:12:26) ago. HB connection took (0:0:36).
Initial NS certainty on Jul 22 14:50:42, (0:11:21) ago, taking (0:0:27).
Our current epoch of certainty started on Jul 22 14:50:42, (0:11:21) ago.
Number of UP nodes: 4
List of UP nodes:  0 5 9 13
```

The domainId=5.14 shows us that node 5 is the nameserver and that the hagsd has been started 14 times.

More information is shown if run on the nameserver. Let's run the same command on the nameserver, node5, Example 5-9.

Example 5-9 hagsns -s hags (on GL nameserver)

```
# /usr/sbin/rsct/bin/hagsns -s hags
HA GS NameServer Status
NodeId=5.14, pid=15108, domainId=5.14, NS established,
CodeLevel=PSSP-3.2(DRL=5)
NS state=kBecomeNS, protocolInProgress=kNoProtocol,
outstandingBroadcast=kNoBcast
Process started on Jul 22 13:31:48, (1:36:40) ago. HB connection took (0:0:0).
Initial NS certainty on Jul 22 13:32:34, (1:35:55) ago, taking (0:0:45).
Our current epoch of certainty started on Jul 22 13:37:53, (1:30:35) ago.
Number of UP nodes: 4
List of UP nodes:  0 5 9 13
List of known groups:
```

```
1.1 ha_em_peers: GL: 9 seqNum: 10 theIPS: 5 9 13 0 lookupQ:
2.1 cssMembership: GL: 13 seqNum: 9 theIPS: 5 9 13 0 lookupQ:
```

We get the extra information of known groups in this GS domain showing the node numbers, **theIPS**, that are interested in the group.

hagsctrl

This is the script that is called by *syspar_ctrl,* which would be the normal way of invoking this command. However, calling this script from the command line provides a variety of controls for separately operating the Group Services subsystems:

- Adding, starting, stopping, and deleting the subsystems.
- Deleting them from all system partitions.
- Unconfiguring the subsystems from all system partitions.
- Turning tracing on and off.

5.3.3 Group Services logs & other useful information

The following list comprises GS logs and other useful information for problem determination:

- AIX Error Log: Entries for Group Services are usually from the SRC master process and are related to starting or stopping the SRC subsystem. The tables in the *PSSP 3.2 Diagnosis Guide, Diagnosing Group Services Problems,* GA22-7350 provide Error Labels and Error IDs, an explanation and details of each.
- /var/ha/log/hags.<syspar>: Which contains the standard out and standard error as the hags script is called from the SRC. Once the script completes it renames the file /var/ha/log/hags.default.<syspar_nodenum_instnum>.
- /var/ha/log/hags.default.<syspar_nodenum_instnum>: This is the renamed /var/ha/log/hags.<syspar> file, it contains output from the initial start of the daemon.
- /var/ha/log/hags_nodenum_instnum.syspar : This file holds current information about all the GS daemon's activities.
- /var/ha/soc/hagsdsocket.<syspar_name>, The socket files are used to communicate between GS clients and the daemon.
- /var/ha/lck/hags.tid.<syspar>: The lock file directory is used to ensure a single running instance of the Group Services daemon, and to establish an instance number for each invocation of the daemon.

- /var/ha/run/hags.<syspar>: current working directories. If the Group Services daemon abnormally terminates, the core dump file is placed in this directory.
- *Group Services Programming and Reference Guide,* SA22-7355.
- *RSCT Group Services: Programming Cluster Applications,* SG24-5523.

5.4 Event Management (EM)

Event Management (EM) is the top level of the RSCT subsystems, it is a client of Group Services. It provides a monitoring service of client requested system resources, such as file systems, processes, CPUs, and notifies those clients when certain conditions are met. It runs as a daemon, haemd. The functional flow is shown in Figure 5-4.

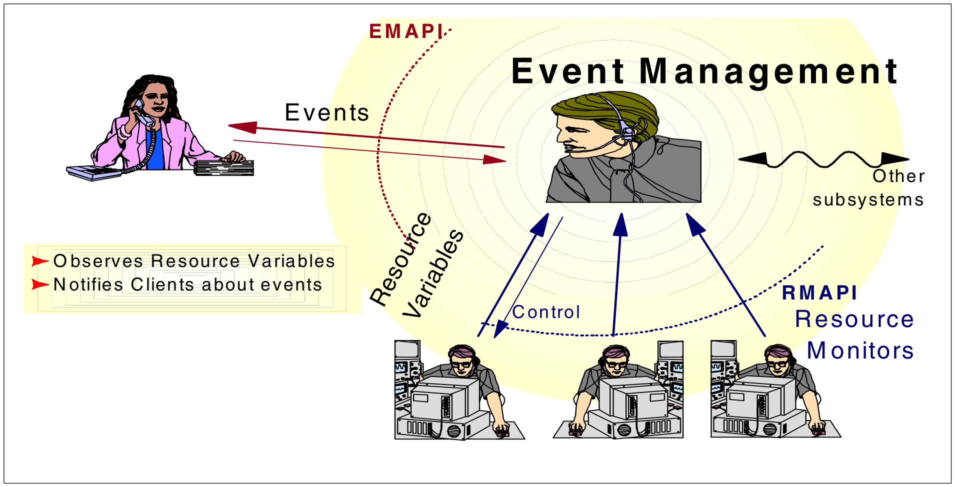


Figure 5-4 EM functional flow

5.4.1 EM overview

By monitoring the state of the resource conditions against the client system resources, the client is notified in advance of any event that can cause a possible system failure. Thus, using this information is useful in trying to recover from any events that can possibly cause system failures in advance of the problem. An example would be detecting a file system on a node starting to fill up, communicating this to the client and the client (such as *pmand*) then taking action to make available space in the monitored file system.

There are three components of EM:

1. Resource Monitors, which keep track of information related to system attributes, transform this information to resource variables and communicate them to the EM subsystem.
2. The EM subsystem communicates between the Resource Monitors and the EM clients. It receives and keeps track of information from the Resource Monitors, as well as tracking information for which the EM clients have expressed an interest in.
3. The EM client acts upon information regarding system resources. An EM client can be an application or a subsystem.

The Event Management Configuration Database (EMCDB) holds all the definitions of the resource monitors and the resource variables which are written to the SDR. It is a binary file that is created from the EM SDR classes.

5.4.2 What to use to diagnose Event Management problems

This section describes commands used to diagnose EM problems.

lssrc

There is a lot of information that is listed by the **lssrc -ls** command. We have left some parts out of the full listing. Let's look at the main sections of some typical output, shown in Example 5-10.

Example 5-10 lssrc -ls haem.sp5en0

```
# lssrc -ls haem.sp5en0
Subsystem         Group            PID     Status
 haem.sp5en0      haem             11984   active

No trace flags are set

Configuration Data Base version from SDR:
        995996645,800126650,0

Daemon started on Tuesday 07/24/01 at 13:44:28
Daemon has been running 0 days, 23 hours, 6 minutes and 22 seconds
Daemon connected to group services: Yes
Daemon has joined peer group:       Yes
Daemon communications enabled:      Yes
Daemon security:                    None
Peer count:                         2
Peer group state:
        995996645,800126650,0
        NOSEC
```

This small section of the **lssrc -ls** listing shows us quite a lot of information.

- That the haem daemon is alive and for how long it has been active.
- The instance value of the Event Management Configuration Database (EMCDB) from the SDR. This should be compared with the information shown in the Peer group state. If there is a mismatch all the EM daemons will have to be stopped, the group dissolved and reformed for a new EMCDB to be generated and used.
- Also shown are the connections between the GS daemon, the group, and communications to the EM clients are working.
- Whether security is in place for the EM daemons and how many peers are in the group.

This next portion of the **lssrc -ls** output, in Example 5-11, shows the file descriptors (FD) and the process ids (PIDS) of the local clients connected.

Example 5-11   lssrc -ls haem.sp5en0 (local client information)

```
Logical Connection Information for Local Clients
    LCID          FD           PID     Start Time
       0          11          14696    Tuesday 07/24/01 13:44:55
       1          12          20106    Tuesday 07/24/01 13:44:56
       2          15          30806    Tuesday 07/24/01 13:45:45
       3          16          30806    Tuesday 07/24/01 13:45:45
       5          22          30646    Wednesday 07/25/01 13:42:51
```

In our listing these local clients are: hrd (14696); pmand (20106); sp_configd (30806) and sphardware (30646).

This part, in Example 5-12, shows the resource monitors connected to EM.

Example 5-12   lssrc -ls haem.sp5en0 (resource monitor information)

```
Resource Monitor Information
         Name            Inst     Type      FD     SHMID     PID     Locked
IBM.PSSP.CSSLogMon         0        C        -1         -1      -2  00/00  No
IBM.PSSP.SDR               0        C        -1         -1      -2  00/00  No
IBM.PSSP.Switch            0        S        -1         -1      -1  00/00  No
IBM.PSSP.harmld            0        S        19    1572867   10812  01/01  No
IBM.PSSP.harmpd            0        S        18         -1    9428  01/01  No
IBM.PSSP.hmrmd             0        S        20         -1   13148  01/01  No
IBM.PSSP.pmanrmd           0        C        14         -1      -2  00/00  No
Membership                 0        I        -1         -1      -2  00/00  No
Response                   0        I        -1         -1      -2  00/00  No
aixos                      0        S        13    1572866      -2  00/02  No
```

This shows the resource monitor names and what type of connection (C: Client, S: Server, I: Internal). An internal EM file descriptor (FD), shared memory ID (SHMID), and whether the resource monitor is locked or not based on the counters (starts and successful connections) shown. A resource may become locked if a monitor cannot be started and remain running or there have been too many connections being lost.

The major portion, not shown, relates to internal EM counters, which may be used by IBM support.

haemctrl

The haemctrl script, called normally by the syspar_ctrl command, provides a variety of controls for operating the EM subsystem.

- Adds, starts, stops, and deletes the subsystem.
- Deletes the subsystem from all system partitions.
- Unconfigures the subsystem from all system partitions.
- Turns tracing on and off.
- Refreshes the subsystem.

5.4.3 Event Management Logs & other useful information

The following list comprises EM logs and other useful information for problem determination:

- *Event Management Programming Guide and Reference, SA22-7354.*
- /var/ha/log/em.default.<syspar_name>: This file contains any error messages that cannot be written to the error logging subsystem.
- /var/ha/log/em.trace.<syspar_name>: This file contains trace output from the EM daemon.
- /var/ha/log/em.msgtrace.<syspar_name>: This file contains message trace output from the EM daemon.
- /var/ha/log/em.loadcfg.<syspar_name>: A log of any errors that occurred while creating the EMCDB.
- /var/ha/log/em.mkgroup: A log of any errors making the haemrm group.
- /var/ha/log/em.mkdir: A log of any errors creating the /var/ha/lck/haem and /var/ha/soc/haem directories.
- /var/ha/run/haem.<syspar_name>: Runtime information and core files will be stored here.
- /var/ha/soc and /var/ha/soc/haem: Contain Unix domain sockets.

- *PSSP 3.2 Diagnosis Guide, Diagnosing Event Management Problems,* GA22-7350. Unlike TS and GS daemons there are only two error templates used. Other components, such as Resource Monitors, may log to the error subsystem as well. The important information is in the Detail Data Field of the error message.

5.5 Putting all the elements together

RSCT is a multi layered set of distributed subsystems operating across multiple nodes or SP attached systems in a clustered environment. Given the complex environments that can exist, RSCT is reasonably robust being able to correct a lot of problems and conditions without intervention, or where a serious problem exists, to work around the problem and keep operating in a reduced capacity.

Given this complexity we look to start problem determination, with the RSCT subsystems operating correctly, before we look to other conditions which may be causing problems. Start at the bottom layer, Topology Services (hats), confirming it is properly configured and working before looking at the group services (hags) subsystem and finally at the event management (haem) subsystem.

Problems in RSCT can be caused by many other factors besides the RSCT subsystems themselves, some of which we list here.

- Networking issues
  - Name resolution.
  - Intermediate nodes without ipforwarding = 1.
  - Communication adapters not working.
  - Wrongly configured communication adapters.
  - Netmasks incorrectly set.
  - Congestion, not allowing or delaying heartbeats.
  - Network outages, including router problems
  - Routing problems
  - ARP-related problems
  - Broadcast addresses incorrectly set
- Configuration problems
  - Node not defined in the SDR.
  - Duplicate or incorrect entries in the SDR.
  - Wrong IP address or MAC address in SDR.

  - SDR daemon not running.
  - Using the wrong instance number (adapter ID or group ID) in *hatsd*.
  - Missing entries in */etc/inetd.conf*.
- General problems
  - *hardmon* subsystem is not running.
  - Node powered off.
  - */var* filesystem is 100% full.
  - Time difference too great between the CWS and the nodes.
- Resource Starvation
  - The CPU is overloaded.
  - Not enough memory.

5.6 Case studies

Probably the most common problem with RSCT reported by customers is no host_responds. This can be caused by many different things since host_responds (hrd) depends on a multitude of components working properly together.

> **Attention:** The case study does not mention HAGS. It is assumed that HAGS is working properly. HAEM is a client for HAGS, so HAEM will not work if HAGS does not work correctly.

Chapter 22, “Topology Services” in the *Parallel System Support Programs for AIX: Diagnosis Guide*, GA22-7350, provides valuable debugging information. Topology Services debugging using the log files is very tricky, and requires much more knowledge of the protocols. We recommend trying to debug problems as much as possible using the AIX error log, and the Topology Services user log.

5.6.1 No host_responds (single node)

Description: Customer calls saying they have lost host_responds on node5. We start checking the system for host_responds as shown in Example 5-13 on page 213. Notice that some output lines from Example 5-13 have been omitted.

Example 5-13   Checking for host_responds

```
[root]:sp5en0 > spmon -d

5.  Checking nodes
----------------------------------- Frame 1 ----------------------------------
                      Host     Switch   Key     Env   Front Panel      LCD/LED
Slot Node Type  Power Responds Responds Switch  Error LCD/LED          Flashes
---- ---- ----- ----- -------- -------- ------- ----- ---------------- -------
  1     1  high  off     no     notcfg  service   no  Stand-By            no
                                                      LCD2 is blank
  5     5  high   on     no       yes    normal   no  LCDs are blank      no
  9     9  high   on     yes      yes    normal   no  LCDs are blank      no
 13    13  high   on     yes      yes    normal   no  LCDs are blank      no
```

Note: The customer has turned node1 off. So we will ignore it.

So we have switch_responds but not host_responds on node5.

Let's start looking by checking that the hatsd daemon is running on node5 as shown in Example 5-14.

Example 5-14   Checking for the hatsd daemon

```
            [root]:sp5en0 > dsh -w node5 lssrc -s hats
            node5: Subsystem         Group            PID     Status
            node5:  hats             hats             15772   active

            [root]:sp5en0 > dsh -w node5 lssrc -ls hats
            node5: Subsystem         Group            PID     Status
            node5:  hats             hats             15772   active
            node5: Network Name   Indx Defd Mbrs St Adapter ID      Group ID
Section 1   node5: SPether        [ 0]    5    1  U 192.168.5.5     192.168.5.5
            node5: SPether        [ 0] en0          0x436342b1      0x436342b1
            node5: HB Interval = 1 secs. Sensitivity = 4 missed beats
Section 2   node5: SPswitch       [ 1]    3    3  S 192.168.15.5    192.168.15.13
            node5: SPswitch       [ 1] css0         0x43632b76      0x43632b76
            node5: HB Interval = 1 secs. Sensitivity = 4 missed beats
            node5:   2 locally connected Clients with PIDs:
            node5: haemd(  7420) hagsd( 13058)
Section 3   node5:   Configuration Instance = 995987196
            node5:   Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
            node5:   Control Workstation IP address = 192.168.5.150
            node5:   Daemon employs no security
            node5:   Data segment size: 6933 KB. Number of outstanding malloc: 265
            node5:   User time 33 sec. System time 26 sec.
            node5:   Number of page faults: 4. Process swapped out 0 times.
```

```
node5:   Number of nodes up: 3. Number of nodes down: 2.
node5:   Nodes down : 0 1
```

Section 1 shows how the problem is with the SPether heartbeat ring, node5 is in a singleton group.

Section 2 shows us that the SPswitch ring is operating normally. Remember node1 is turned off.

Section 3. Is the configuration instance correct? We'll check on the CWS as shown in Example 5-15.

Example 5-15 Checking the configuration instance

```
[root]:sp5en0 > lssrc -ls hats.sp5en0
Subsystem         Group            PID     Status
 hats.sp5en0      hats             32722   active
Network Name   Indx Defd Mbrs St Adapter ID      Group ID
```

Section 1

```
SPether        [ 0]    5    3  S 192.168.5.150   192.168.25.13
SPether        [ 0] en0          0x4361d7fb      0x43632bab
HB Interval = 1 secs. Sensitivity = 4 missed beats
  2 locally connected Clients with PIDs:
haemd( 34602) hagsd( 32030)
```

Section 2

```
 Configuration Instance = 995987196
  Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
  Control Workstation IP address = 192.168.5.150
  Daemon employs no security
  Data segment size: 8981 KB. Number of outstanding malloc: 196
  User time 125 sec. System time 152 sec.
  Number of page faults: 1089. Process swapped out 0 times.
  Number of nodes up: 4. Number of nodes down: 1.
  Nodes down : 1
```

Section 1 shows that the GL of the SPether ring is 192.168.25.13. We'll check the machines.lst as shown in Example 5-16 to confirm who this is and if this is the same file as on the other nodes. We will only look at the en0 section to save some space.

Section 2 tells us the same configuration instance is being used.

Example 5-16 Checking the machines.lst for en0

```
root]:sp5en0 > grep en0 /var/ha/run/hats.sp5en0/machines.lst
0 en0 192.168.5.150
    1 en0  192.168.5.1
    5 en0  192.168.5.5
    9 en0  192.168.5.9
   13 en0  192.168.25.13
```

So node13 is the GL for the SPether ring. Talking with the customer, they have node9 as a boot install server for node13 so en0 is on a different network to the other nodes en0 interfaces. The host_responds has been working previously without a problem. Let's look at the hats log on node5 as shown in Example 5-17 to see what it thinks is happening.

Example 5-17 Checking the hats log on node5

```
[root]:sp5n05:/var/ha/log > ls -lrt hats*
-rwxr-xr-x   1 root     system      1942 Jul 24 13:45 hats.sp5en0.3
-rw-rw-rw-   1 root     system    306374 Jul 27 05:58 hats.24.134553.sp5en0.bak
-rw-rw-rw-   1 root     system      6418 Jul 27 17:08
hats.24.134553.sp5en0.en_US
-rw-rw-rw-   1 root     system    287166 Jul 27 17:08 hats.24.134553.sp5en0
-rwxr-xr-x   1 root     system      1263 Jul 27 17:08 hats.sp5en0.2
-rw-rw-rw-   1 root     system      2857 Jul 28 17:14
hats.27.170854.sp5en0.en_US
-rw-rw-rw-   1 root     system     45380 Jul 28 17:14 hats.27.170854.sp5en0
```

Note: We only show some of the relevant hats log to save space.

```
[root]:sp5n05:/var/ha/log > view hats.27.170854.sp5en0
.
.
07/27 17:09:02:hatsd[0]: Received a PTC request from (192.168.25.13:0x4361d7cf)
in group (192.168.25.13:0x4361d86c).
07/27 17:09:02:hatsd[1]: Node Connectivity Message stopped on adapter offset 1.
07/27 17:09:02:hatsd[0]: Received a COMMIT BROADCAST request from
(192.168.25.13:0x4361d7cf) in group (192.168.25.13:0x4361d86c).
07/27 17:09:02:hatsd[0]: Other group:
        0 (192.168.5.5:0x4361d86b)      1 (192.168.5.9:0x4361a44b)
        2 (192.168.5.150:0x4361d7fb)    3 (192.168.25.13:0x4361d7cf)
07/27 17:09:02:hatsd[0]: Sending a COMMIT BROADCAST ACK response to
(192.168.25.13:0x4361d7cf).
07/27 17:09:02:hatsd[0]: Received a COMMIT request from
(192.168.25.13:0x4361d7cf) in group (192.168.25.13:0x4361d86c).
07/27 17:09:02:hatsd[0]: Stable Time     = (996268142.605674) seconds.
07/27 17:09:02:hatsd[0]: Joining Adapters = 3: 8044
07/27 17:09:02:hatsd[0]: Reachable nodes (1 hop)    : 0 5-13(4)
07/27 17:09:02: - (0) - send_node_connectivity()          CurAdap 0
07/27 17:09:02: - (1) - send_node_connectivity()          CurAdap 1
07/27 17:09:02:hatsd[0]: Sending a COMMIT ACK response to
(192.168.5.5:0x4361d86b).
07/27 17:09:02:hatsd[0]: Sending adapter [Hb_Join] notifications.
07/27 17:09:02:hatsd[0]: Sending adapter [Hb_New_Group] notifications.
```

```
07/27 17:09:02:hatsd[0]: Node Connectivity Message stopped on adapter offset 0.
07/27 17:09:02:hatsd[0]: Reachable nodes (1 hop)    : 0 5-13(4)
07/27 17:09:03:hatsd[0]: My New Group ID = (192.168.25.13:0x4361d86c) and is
Stable.
        My Leader is              (192.168.25.13:0x4361d7cf).
        My Crown Prince is        (192.168.5.150:0x4361d7fb).
        My upstream neighbor is   (192.168.5.9:0x4361a44b).
        My downstream neighbor is (192.168.25.13:0x4361d7cf).
        I am                      (192.168.5.5:0x4361d86b).
```

So we see that node5 was indeed a member of the SPether ring. As we continue checking further down in the log file (almost 24 hours later) as shown in Example 5-18.

Example 5-18 Checking the log file 24 hours later

```
.
07/28 17:04:22:hatsd[0]: Missed Heartbeats = 1.
07/28 17:04:24:hatsd[0]: Missed Heartbeats = 2.
07/28 17:04:25:hatsd[0]: GROUP_PROCLAIM message from address 192.168.25.13
(defined: 192.168.25.13) rejected: adapter still is in my group.
07/28 17:04:26:hatsd[0]: Missed Heartbeats = 3.
07/28 17:04:28:hatsd[0]: Missed Heartbeats = 4.
07/28 17:04:28:hatsd[0]: Adapter (192.168.5.9:0x4361a44b) is dead.
07/28 17:04:28:hatsd[0]: Notifying leader (192.168.25.13:0x4361d7cf) of death.
FFDC id [.8hjsyyQXmMv.OWR/3I.e.z...................].
07/28 17:04:30:hatsd[0]: GROUP_PROCLAIM message from address 192.168.25.13
(defined: 192.168.25.13) rejected: adapter still is in my group.
07/28 17:04:30:hatsd[0]: Missed Heartbeats = 5.
07/28 17:04:30:hatsd[0]: Notifying leader (192.168.25.13:0x4361d7cf) of death.
FFDC id [---].
07/28 17:04:32:hatsd[0]: Missed Heartbeats = 6.
07/28 17:04:32:hatsd[0]: Netmon program returned. Local adapter is UP
07/28 17:04:32:hatsd[0]: Notifying leader (192.168.25.13:0x4361d7cf) of death.
FFDC id [---].
07/28 17:04:34:hatsd[0]: Missed Heartbeats = 7.
07/28 17:04:34:hatsd[0]: Notifying leader (192.168.25.13:0x4361d7cf) of death.
FFDC id [---].
07/28 17:04:35:hatsd[0]: GROUP_PROCLAIM message from address 192.168.25.13
(defined: 192.168.25.13) rejected: adapter still is in my group.
07/28 17:04:36:hatsd[0]: Missed Heartbeats = 8.
07/28 17:04:36:hatsd[0]: Broadcasting DISSOLVE GROUP message to group.
07/28 17:04:36:hatsd[0]: Received a DISSOLVE_GROUP message from
(192.168.5.5:0x4361d86b) in group (192.168.25.13:0x4361d86c).
07/28 17:04:36:hatsd[0]: DISSOLVE_GROUP message from a group member,
Dissolving!!!
07/28 17:04:36:hatsd[0]: Received a BAD MESSAGE message from
(192.168.5.9:0x4361a44b) in group (192.168.25.13:0x436328d4).
```

```
07/28 17:04:36:hatsd[0]: Received a BAD MESSAGE message from
(192.168.5.150:0x4361d7fb) in group (192.168.25.13:0x436328d4).
07/28 17:04:36:hatsd[0]: Re-initializing..................
07/28 17:04:37: - (0) - init_group()                      node_idx = 0
07/28 17:04:37:hatsd[0]: Dead Adapters    = 3: 8044
.
.
07/28 17:04:38:hatsd[0]: My New Group ID = (192.168.5.5:0x436328e5) and is
Unstable.
        My Leader is              (192.168.5.5:0x4361d86b).
        My Crown Prince is        (192.168.5.5:0x4361d86b).
        My upstream neighbor is   (192.168.5.5:0x4361d86b).
        My downstream neighbor is (192.168.5.5:0x4361d86b).
        I am                      (192.168.5.5:0x4361d86b).
07/28 17:04:38:hatsd[0]: Netmon program returned. Local adapter is UP
07/28 17:04:40:hatsd[0]: Received a Group Proclaim from
(192.168.25.13:0x4361d7cf) in group (192.168.25.13:0x436328d4).
07/28 17:04:40:hatsd[0]: Sending JOIN request to
(192.168.25.13:0x4361d7cf)[0:0:0:0:0:0] in group (192.168.25.13:0x436328d4).
07/28 17:04:50: check_state_3p hatsd[0]: 2523-097 JOIN time has expired.
PROCLAIM message was sent by (192.168.25.13:0x4361d7cf).
07/28 17:04:50:hatsd[0]: Received a Group Proclaim from
(192.168.25.13:0x4361d7cf) in group (192.168.25.13:0x436328d4).
```

Node5 is having problems communicating with other members of the SPether group, although it is receiving messages, so it dissolved the group, formed a singleton group, and attempted to rejoin unsuccessfully. Let’s look at the error log on node5 as shown in Example 5-19.

Example 5-19 Checking the error log in node5

```
LABEL:          TS_NODEDOWN_EM
IDENTIFIER:     4D9226A5

Date/Time:       Sat Jul 28 17:05:19
Sequence Number: 180
Machine Id:      00010023A400
Node Id:         sp5n05
Class:           U
Type:            PEND
Resource Name:   hats.sp5en0
Resource Class:  NONE
Resource Type:   NONE
Location:        NONE
VPD:

Description
Remote nodes down
```

```
Probable Causes
Remote nodes powered off
Remote nodes crashed
Networking problems render remote nodes unreachable
Remote nodes removed from configuration after refresh
Topology Services daemon on remote nodes stopped

User Causes
User powered off remote nodes
Remote nodes removed from configuration after refresh

        Recommended Actions
        Confirm that this is desirable

Failure Causes
Remote nodes crashed
Lost connection to remote nodes due to network problems
Remote nodes hang

        Recommended Actions
        Get system dump from remote nodes. Re-boot remote nodes
        Clear networking problems
        Re-start Topology Services daemon on remote node
        Contact IBM Service if problem persists

Detail Data
DETECTING MODULE
rsct,connect.C,           1.56,1475
ERROR ID
.ZOWYB/DYmMv.9sf.3I.e.z...................
REFERENCE CODE

File containing down node numbers and associated REFERENCE CODE
/var/adm/ffdc/dumps/hats.15998.20010728.170519
```

Attention: The TS_NODEDOWN_EM entry is related to “node reachability” as opposed to “adapter membership group.” For example, if you have problems with the SP Ethernet but the SP switch is still working fine, then you will NOT get the TS_NODEDOWN_EM entry.

Looking at the Probable Causes section, we can eliminate the remote nodes being powered off or hung. Let’s check if we have a network problem. What network adapters do we have in node5? See Example 5-20.

Example 5-20 Checking for network adapters

```
[root]:sp5n05:/var/ha/log > ifconfig -a
en0:
flags=e080863<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GROUPRT,64BIT>
        inet 192.168.5.5 netmask 0xffffff00 broadcast 192.168.5.255
en1:
flags=e080863<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GROUPRT,64BIT>
        inet 192.168.25.5 netmask 0xffffff00 broadcast 192.168.25.255
css0: flags=800843<UP,BROADCAST,RUNNING,SIMPLEX>
        inet 192.168.15.5 netmask 0xffffff00 broadcast 192.168.15.255
lo0:
flags=e08084b<UP,BROADCAST,LOOPBACK,RUNNING,SIMPLEX,MULTICAST,GROUPRT,64BIT>
        inet 127.0.0.1 netmask 0xff000000 broadcast 127.255.255.255
        inet6 ::1/0
```

Let’s check the routes on node5.

```
[root]:sp5n05:/var/ha/log > netstat -rn
Routing tables
Destination      Gateway           Flags   Refs     Use  If   PMTU  Exp  Groups

Route Tree for Protocol Family 2 (Internet):
default          192.168.5.150     UGc       0        0  en0     -   -
127/8            127.0.0.1         U         6      466  lo0     -   -
192.168.5/24     192.168.5.5       U         6    76232  en0     -   -
192.168.15/24    192.168.15.5      U         2   619648 css0     -   -
192.168.25/24    192.168.25.5      U         0     3350  en1     -   -
```

So node5 should be trying to communicate to node13 (192.168.25.13) through ethernet adapter en1. Let’s confirm that with the command shown in Example 5-21.

Example 5-21 Checking node5 communication with node13

```
[root]:sp5n05:/var/ha/log > traceroute 192.168.25.13
trying to get source for 192.168.25.13
source should be 192.168.25.5
traceroute to 192.168.25.13 (192.168.25.13) from 192.168.25.5 (192.168.25.5),
30 hops max
outgoing MTU = 1500
```

This does not succeed but does confirm that en1 is the path to communicate with node13. So we have a problem with en1 on node5 and need to have this fully diagnosed. While we are waiting for the H/W to be checked we can still alleviate the host_responds problem. We know the default route from node5 is through the CWS and that is a part of the SPether ring. Let’s detach the en1 interface on node5.

```
[root]:sp5n05:/var/ha/log > ifconfig en1 detach
```

Now we refresh the hatsd daemon on node5.

Attention: The **`hatsctrl -r`** command should never be issued on a node. It needs to be issued from the CWS. Also, the **`hatsctrl -r`** command only needs to be invoked when there is a change in the overall topology, such as:

- Adding or removing a node.
- Adding or removing an adapter.
- Changing the address of an adapter.

Adapters going up or down do not warrant calling **`hatsctrl -r`**.

```
[root]:sp5en0:/var/ha/log > /usr/sbin/rsct/bin/hatsctrl -r
0513-095 The request for subsystem refresh was completed successfully.
```

And we check the spmon -d on the CWS as shown in Example 5-22.

Example 5-22 spmon -d on the CWS

```
[root]:sp5en0 > spmon -d
.
.
5.  Checking nodes
----------------------------------- Frame 1 -----------------------------------
                      Host     Switch   Key     Env   Front Panel      LCD/LED
Slot Node Type  Power Responds Responds Switch  Error LCD/LED          Flashes
---- ---- ----- ----- -------- -------- ------- ----- ---------------- -------
  1     1  high  off     no     notcfg  service    no  Stand-By            no
                                                      LCD2 is blank
  5     5  high   on     yes      yes    normal   no  LCDs are blank      no
  9     9  high   on     yes      yes    normal   no  LCDs are blank      no
 13    13  high   on     yes      yes    normal   no  LCDs are blank      no
```

Host_responds is back because we are routing the SPether heartbeat traffic through the 192.168.5.xxx network. To configure host_responds, issue the command netstat -rn, as shown in Example 5-23.

Example 5-23 Configuring the host_responds

```
[root]:sp5n05:/var/ha/log > netstat -rn
Routing tables
Destination      Gateway           Flags   Refs     Use  If   PMTU  Exp  Groups

Route Tree for Protocol Family 2 (Internet):
default          192.168.5.150     UGc       0        0  en0     -   -
```

Section 1

```
127/8           127.0.0.1         U         6      468  lo0     -   -
192.168.5/24    192.168.5.5       U         7    78062  en0     -   -
192.168.15/24   192.168.15.5      U         2   628094 css0     -   -
192.168.25.13   192.168.5.9       UGHMW     0      601  en0     -     2

Route Tree for Protocol Family 24 (Internet v6):
::1             ::1               UH        0        0  lo0 16896   -
```

Section 1 shows us network traffic for 192.168.25.13 will now use the gateway 192.168.5.9.

5.6.2 No host_responds (all nodes)

Action: Customer shuts down the SP cluster for software maintenance and rebooted.

Description: Customer calls saying they have lost host_responds on all nodes. See Example 5-24 to check for signs of host_responds in the SP cluster.

Example 5-24 Checking for host_responds on the SP cluster

```
[root]:sp5en0 > spmon -d
.
.
5.  Checking nodes
----------------------------------- Frame 1 ----------------------------------
                      Host     Switch   Key     Env   Front Panel      LCD/LED
Slot Node Type  Power Responds Responds Switch  Error LCD/LED          Flashes
---- ---- ----- ----- -------- -------- ------- ----- ---------------- -------
  1     1  high  off     no     notcfg  service   no  Stand-By           no
                                                      LCD2 is blank
  5     5  high   on     no       yes    normal   no  LCDs are blank     no
  9     9  high   on     no       yes    normal   no  LCDs are blank     no
 13    13  high   on     no       yes    normal   no  LCDs are blank     no
```

Note: The customer has turned node1 off. So we’ll ignore it.

Let’s look at the **lssrc -ls hats** from one of the nodes, as shown in Example 5-25.

Example 5-25 Checking for hats on one of the nodes

```
[root]:sp5n05:/ > lssrc -ls hats
Subsystem         Group            PID     Status
 hats             hats             11588   active
Network Name   Indx Defd Mbrs St Adapter ID      Group ID
```

```
SPether        [ 0]    5    3  S 192.168.5.5     192.168.25.13
SPether        [ 0] en0         0x43670834      0x4367084d
HB Interval = 1 secs. Sensitivity = 4 missed beats
SPswitch       [ 1]    3    3  S 192.168.15.5    192.168.15.13
SPswitch       [ 1] css0        0x43670835      0x4367083e
HB Interval = 1 secs. Sensitivity = 4 missed beats
  2 locally connected Clients with PIDs:
haemd( 13336) hagsd( 12886)
  Configuration Instance = 995987196
  Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
  Control Workstation IP address = 192.168.5.150
  Daemon employs no security
  Data segment size: 6677 KB. Number of outstanding malloc: 263
  User time 3 sec. System time 3 sec.
  Number of page faults: 0. Process swapped out 0 times.
  Number of nodes up: 3. Number of nodes down: 2.
  Nodes down : 0 1
```

We see two heartbeat rings have successfully been formed but node0 (CWS) is down. Let’s look at the **lssrc -ls hats.sp5en0** output on the CWS.

```
[root]:sp5en0 > lssrc -ls hats.sp5en0
0513-085 The hats.sp5en0 Subsystem is not on file.
```

Let’s check what RSCT subsystems are running on the CWS as shown in Example 5-26.

Example 5-26 Checking for other RSCT subsystems

```
[root]:sp5en0 > lssrc -a|grep ha
 hardmon                           28670   active
 hags.sp5en0      hags             23172   active
 hagsglsm.sp5en0  hags             35376   active
 haem.sp5en0      haem             27858   active
 haemaixos.sp5en0 haem             30446   active
```

So only hats.sp5en0 appears to not be running. We have the customer run **/usr/lpp/ssp/bin/syspar_ctrl -D** on the nodes and the CWS to stop and remove the RSCT subsystems from the SRC, then run the command shown in Example 5-27 on the CWS to add and start the subsystems. We only show the errors produced in Example 5-27.

Example 5-27 Starting the RSCT subsystems

```
[root]:sp5en0 > syspar_ctrl -A
.
.
```

```
hatsctrl: 2523-638 Cannot set port number into /etc/services
syspar_ctrl:  0022-233 SP_NAME=sp5en0  /usr/sbin/rsct/bin/hatsctrl -a
  returned with a unsuccessful return code, rc = 1.
.
.
0513-085 The hats.sp5en0 Subsystem is not on file.
syspar_ctrl:  0022-233 SP_NAME=sp5en0  /usr/sbin/rsct/bin/hatsctrl -s
  returned with a unsuccessful return code, rc = 1.
.
.
syspar_ctrl:  0022-234 Add failed, calls to underlying control scripts returned
with unsuccessful return codes.
```

So the problem appears to be adding the port number for the hatsd into /etc/services file. Let's check that there is a port assigned in the SDR as shown in Example 5-28.

Example 5-28 Checking for port assignment in the SDR

```
[root]:sp5en0 > SDRGetObjects Syspar_ports
subsystem    port
hats                10000
spdmd               10004
hags                10001
haem                10002
```

Example 5-28 looks good. Let's look at the /etc/services file. We only show the lines we are interested in. See Example 5-29.

Example 5-29 /etc/services file contents simplified

```
wombat-monitor  10000/udp
hags.sp5en0     10001/udp
haem.sp5en0     10002/udp
haemd           10003/tcp
spdmd           10004/tcp
```

The subsystem wombat-monitor is using the udp port 10000 that was assigned to the hats.sp5en0 subsystem. This was apparently added as a monitoring program by the network administration people with a hard coded udp port number in the application.

Let's stop and delete the RSCT subsystems again on the CWS (syspar_ctrl -D) and clean out the SDR port number for the hats.sp5en0 and reassign another port using the /usr/sbin/rsct/bin/hatsctrl script options. See Example 5-30.

Example 5-30 Working with the RSCT subsystems

```
[root]:sp5en0 > hatsctrl -u

[root]:sp5en0 > SDRGetObjects Syspar_ports
subsystem    port
spdmd               10004
hags                10001
haem                10002

[root]:sp5en0 > hatsctrl -a
0513-071 The hats.sp5en0 Subsystem has been added.

[root]:sp5en0 > SDRGetObjects Syspar_ports
subsystem    port
spdmd               10004
hags                10001
haem                10002
hats                10005
```

Now let's add and start the RSCT subsystems again, first on the CWS and then on the nodes, so the new SDR configuration will be picked up by all the nodes. Example 5-31 shows the SP cluster with host_responds. Node 1 has been off for this exercise.

Example 5-31 Checking for host_responds

```
[root]:sp5en0 > spmon -d
.
.
----------------------------------- Frame 1 ----------------------------------
                      Host     Switch   Key     Env   Front Panel      LCD/LED
Slot Node Type  Power Responds Responds Switch  Error LCD/LED          Flashes
---- ---- ----- ----- -------- -------- ------- ----- ---------------- -------
  1     1  high  off     no     notcfg  service   no  Stand-By            no
                                                      LCD2 is blank
  5     5  high   on     yes      yes    normal   no  LCDs are blank      no
  9     9  high   on     yes      yes    normal   no  LCDs are blank      no
 13    13  high   on     yes      yes    normal   no  LCDs are blank      no
```

5.6.3 Changing CWS hostname (spmon -d broken)

Action: A customer wants to change the CWS hostname, not the IP address.

Description: We followed the procedure found in Appendix F of the *Parallel System Support Programs for AIX: Administration Guide*, SA22-7348, but found that we could not connect to the frame. Refer to Example 5-32 for more details on the problem.

Example 5-32 hostname change problem

```
[root@cws5en0]:/usr/dt/bin> spmon -d
1.  Checking server process
    Process 23632 has accumulated 0 minutes and 5 seconds.
    Check successful

2.  Opening connection to server
    Connection opened
    Check successful

3.  Querying frame(s)
spmon: 0026-064 You do not have authorization to access the Hardware Monitor.
spmon: 0026-059 Could not query frames.
```

Note: Are the RSCT subsystems (hats, hags) running?

```
[root@cws5en0]:etc> lssrc -a|grep ha
 hardmon                           23632   active
```

Note: The RSCT subsystems are not running. Let’s add and start them up.

note
SP_NAME

```
[root@cws5en0]:/etc> syspar_ctrl -A
The required function fails because SDR is not running correctly.
syspar_ctrl:  0022-233 SP_NAME=cwsen0  /usr/sbin/rsct/bin/hatsctrl -a
  returned with a unsuccessful return code, rc = 1.
hagsctrl: 2520-203 Cannot determine syspar name.
```

Note: Let’s check the error log.

```
[root@cws5en0]:/usr/dt/bin> errpt -a

LABEL:          HR_MESSAGE_ER
IDENTIFIER:     DE79DE5B

Date/Time:       Fri Jul 13 15:02:46
Sequence Number: 7476
Machine Id:      000354794C00
Node Id:         sp5en0
Class:           S
```

```
Type:            PERM
Resource Name:   hr

Description
INTERNAL ERROR: ERROR STRING

Probable Causes
SOFTWARE PROGRAM

Failure Causes
SOFTWARE PROGRAM

        Recommended Actions
        NO FURTHER ACTION REQUIRED UNLESS PROBLEM PERSISTS
        RESTART SOFTWARE SUBSYSTEM
        CONTACT YOUR LOCAL IBM SERVICE REPRESENTATIVE
        REFER TO PRODUCT DOCUMENTATION FOR ADDITIONAL INFORMATION

Detail Data
DETECTING MODULE
LPP=PSSP,Fn=hr.c,SID=1.11.1.25,L#=1524,
ERROR DATA
hrd: 2505-292 (cwsen0) Could not obtain authorization methods for trusted
services. Do not use DCE.
2502-600 An error occurred getting a hostname.
```

Note: What is in the SDR?

Wrong syspar_name

```
[root@cws5en0]:/etc> SDRGetObjects Syspar syspar_name ip_address code_version
syspar_name  ip_address code_version
cwsen0       192.168.5.150 PSSP-3.1.1
```

Note: What is written in the Syspar class of the SDR?

SDR has right syspar_name

```
[root@cws5en0]:/spdata/sys1/sdr/partitions/192.168.5.150/classes> cat Syspar
1=cws5en0 2=192.168.5.150 3=default
4=/spdata/sys1/syspar_configs/1nsb0isb/config.16/layout.1/syspar.1 5=PSSP-3.1.1
6=995062303,518021059,0 7=k4 8=k4:std 9=k4:std 10=compat
```

Note: The syspar_name value in the SDR is correct but the syspar_name shown in the SDRGetObjects is incorrect. It would seem a typographical error was made but then corrected to the SDR class without refreshing the SDR daemon.

Solution: Reset the SDR.

```
[root@cws5en0]:/etc> sdr reset
0513-044 The sdr.cws5en0 Subsystem was requested to stop.
0513-059 The sdr.cws5en0 Subsystem has been started. Subsystem PID is 5704.
```

Note: Check host_responds.

```
[root@cws5en0]:/etc> spmon -d
1.  Checking server process
    Process 20648 has accumulated 0 minutes and 6 seconds.
    Check successful

2.  Opening connection to server
    Connection opened
    Check successful

3.  Querying frame(s)
    1 frame
    Check successful

4.  Checking frames

    This step was skipped because the -G flag was omitted.

5.  Checking nodes
----------------------------------- Frame 1 ----------------------------------
                      Host     Switch   Key     Env   Front Panel      LCD/LED
Slot Node Type  Power Responds Responds Switch  Error LCD/LED          Flashes
---- ---- ----- ----- -------- -------- ------- ----- ---------------- -------
  1     1  high   on     yes      yes    normal   no  LCDs are blank      no
  5     5  high   on     yes      yes    normal   no  LCDs are blank      no
  9     9  high   on     yes      yes    normal   no  LCDs are blank      no
 13    13  high   on     yes      yes    normal   no  LCDs are blank      no
```

Note: Confirm by SDRGetObjects. All looks okay now.

```
[root@cws5en0]:/etc> SDRGetObjects host_responds
```

```
node_number  host_responds
          1            1
          5            1
          9            1
         13            1
```

5.7 Security for RSCT

The RSCT components are a part of the SP Trusted Services and the authentication methods used depend on the attributes set in the Syspar class of the SDR (ts_auth_methods). RSCT is able to work with all of the security environments, DCE (Kerberos Version 5), Kerberos Version 4, and Standard AIX. Should you wish to run your system at the DCE (Kerberos Version 5) level, then all nodes within the partition need to be running PSSP 3.2. For a more detailed explanation of the security settings and their interdependencies, please refer to Chapter 4, “Security” on page 159.

Restricted root access (RRA) has some implications for EM and some of the clients that depend on the RSCT subsystems, these are mainly to do with **rcp** and **rsh**. As an example, the EMCDB is normally copied via **rcp** to the nodes, this must now be run as a **sysctl** command under the Kerberos principle SPbgAdm. For a fuller explanation please refer to Chapter 4, “Security” on page 159.

5.8 Information collection for support

Here is a list of tools utilized for information collection:

phoenix.snap

This is a service tool and not a PSSP command, therefore, there is no documentation in the PSSP Command Reference Guides. It is supplied as part of RSCT and is a information collection tool used to assist in the analysis of problems. It can be run on a node or on the CWS and must be run by the root user.

When run on a node it will only collect information about that node.

The output will be archived and compressed in the created directory, /tmp/phoenix.snapOut. Successful collection will show similar output to the following:

```
[0:root@sp3n14:]/home/root # /usr/sbin/rsct/bin/phoenix.snap
   phoenix.snap version 1.7, args:
```

```
I am node 14 hostname sp3n14 running PSSP-3.2 and AIX 4.3.3.0

Determining if there is enough space in /tmp/phoenix.snapOut
compressed file will be about 919821 bytes
phoenix.snap requires about 5518928 bytes
Think we have enough space.
######################################################################
Send file /tmp/phoenix.snapOut/phoenix.snap.node14.07151429.out.tar.Z to
the RS6000/SP service team
```

When run without flags on the CWS, phoenix.snap will collect information from the Topology Services Group Leader and the Group Services Nameserver, storing the information on the CWS in the /tmp/phoenix.snapOut directory as a archive file all.<timestamp>.tar containing the archived, compressed files from the different nodes.

Other flags that can be used are:

- The -d flag directs the phoenix.snap output to another specified directory.
- The -p flag is used to specify a partition name, the default is ALL. Valid only on the CWS.
- The -x 1 flag can be used to determine if there is enough space in the default directory.
- The -l hostlist | ALL flag can be used only on the CWS directing phoenix.snap to gather information from the listed nodes. The parameter -l ALL would gather information about all the nodes, this should not be used on large systems unless space is not a problem.

Manual collection of data

Sometimes collecting data from all nodes in a large SP clustered environment may not be practical. If it is not feasible, collect the following files or output:

- errpt -a > /tmp/error.log
- lslpp -L rsct.*
- /usr/lpp/ssp/bin/lsauthpts -c
- /usr/lpp/ssp/bin/splstdata -p
- /usr/lpp/ssp/bin/splstdata -n
- lssrc -a

Topology Services

Collect data from at least the following nodes:

- The group leader (GL) on the all network rings. This is the node with the highest IP address in the ring. If you can identify that is is only one ring that is experiencing problems, then collect data only from that GL.
- The downstream neighbor of the node that has the problem. This is the node with the next lower IP address. The node with the lowest IP address has the GL as its downstream neighbor.
- The CWS and the node that has the problem.

- **lssrc -ls** command. On the nodes run **lssrc -ls hats**; on the CWS run **lssrc -ls hats.<syspar_name>**.
- The last ten files from /var/ha/log that are shown from **ls -lrt hats***.
- `SDRGetObjects SP cw_ipaddrs`
- `SDRGetObjects TS_Config`
- `SDRGetObjects Adapter`
- `SDRRetrieveFile hats.machines.lst /tmp/machines.output`
- `SDRGetObjects host_responds`
- `netstat -r -n`
- `netstat -in`
- `netstat -m`
- `netstat -D`

Group Services

Collect data from at least the following nodes:

- Node or nodes that have the problem.
- GS nameserver.
- CWS.

- On a node collect the **/usr/sbin/rsct/bin/nlssrc -c -ls hags** output.
- On the CWS collect the **/usr/sbin/rsct/bin/nlssrc -c -ls hags.<syspar_name>** output.
- The last ten files from /var/ha/log that are shown from ls **-lrt hags***.
- SDRGetObjects GS_Config.

Event Management

Collect the following files or output from the failing nodes and the CWS:

- /etc/hosts
- /etc/resolv.conf

- /etc/services
- /etc/netsvc.conf
- **`SDRGetObjects Syspar_ports`**
- **`SDRGetObjects SP_ports`**
- **`df -k`**
- **`no -a`**

5.9 Useful references

The following manuals contain useful information for problem determination:

- *http://www.rs6000.ibm.com/support/sp*
- */usr/sbin/rsct/README/rsct.basic.README*
- *PSSP 3.2 Administration Guide,* SA22-7348
- *PSSP 3.2 Diagnosis Guide,* GA22-7350

There are several manuals available that are detailed references to RS/6000 Cluster Technology (RSCT), such as:

- *RSCT: Event Management Programming Guide and Reference,* SA22-7354
- *RSCT: Group Services Programming Guide and Reference,* SA22-7355
- *RSCT Group Services: Programming Cluster Applications,* SG24-5523
- *RS/6000 SP Cluster: The Path to Universal Clustering,* SG24-5374

6

SP switches

In this chapter, we provide a hardware and software overview and we try to guide you through possible switch problems. This chapter is not a complete description of how to install and work with the switch network, but at least we will show some ways of avoiding possible problems. We give you quick APAR overviews of common SP Switch/SP Switch2 related software problems. The following books are still the common entry points for fixing problems:

- *PSSP Diagnosis Guide V3.2*, GA22-7350
- *PSSP Messages Reference V3.2*, GA22-7352
- *Understanding and using the SP Switch*, SG24-5161
- *RS/6000 SP System Service Guide*, GA22-7442
- *RS/6000 SP; SP Switch Service Guide*, GA22-7443
- *RS/6000 SP, SP Switch2 Service Guide*, GA22-7444
- *IBM 9077 SP Switch Router: Get Connected to the SP Switch*, SG24-5157
- *PSSP Version 3 Survival Guide*, SG24-5344

In this section we tried to gather common problems to provide you with some direction when you have problems with your switch. See Table 6-1 for a quick start to solving switch problems.

Table 6-1 SP switches start table

What are you looking for?	Where to go
SP Switch and SP Switch2 information	Go to “SP switches overview” on page 235
SDR related switch problems	Go to “SDR related switch problems” on page 250
Diagnostic tools and routines	Go to “Diagnostic tools and routines” on page 244
Switch problems fixed by PSSP APARs	Go to “Worm daemon problems - fixed by APARs” on page 258
Node Problems on the switch	Go to “Node problems on the switch” on page 252
SP Switch problems	Go to “SP Switch problems” on page 262
Multiple switches problems	Go to “Multiple switches problems” on page 265
SP Switch2 problems	Go to “SP Switch2 problems” on page 266
9077 GRF problems	Go to “9077 GRF overview and problems” on page 267

Several of the following sections have tables that illustrate common SP Switch related problems that are fixed by PSSP APARs. You will find the description of the failure, temporary fix, if available, and the APAR number. The fixes apply to PSSP 3.1.1 and 3.2. Older PSSP levels are not covered, but using the latest PTFs for PSSP will minimize the possibility of those software related errors. See also Appendix C, “Software maintenance strategy” on page 285.

6.1 SP switches overview

There are now two kinds of switches available for the current SP systems and SP-attached servers. The SP Switch (SPS) provides connectivity to withdrawn Microchannel Nodes and current PCI Nodes. This also includes the SP-attached servers (7017 S7A, S80, S85 and p680). The SP Switch2 provides higher performance, but is only available for PCI High Nodes. With the SP Switch2, we have better diagnostic capabilities and improved system reliability, availability and serviceability (RAS).

Restriction: SP Switch2 is not compatible with the SP Switch or the older HiPS (High Performance Switch). They cannot coexist in the same SP System.

6.1.1 SP Switch and SP Switch2 boards

Regardless if the switch board is a SP Switch or SP Switch2, it has components like the switch chips, supervisor card, power supplies, and cooling fans. There are several other hardware components used to make the switch board work, but, from the outside, it looks just like a black box with a power switch on the front and 32 connectors on the rear.

Sixteen ports are used for switch to node connections and 16 ports are used for switch-to-switch connections. The SP Switch, see Figure 6-1 on page 236, has specially assigned switch-to-node ports and switch-to-switch ports. You have to put Node1 (N1) to port J7. There is also an eight-port switch available for connectivity of eight nodes maximum.

The SP Switch2 gives us the possibility to plug nodes into any available node port on the switch and switches to any available switch-to-switch port. This gives us one more advantage; we can avoid possible misplugging while connecting the nodes.

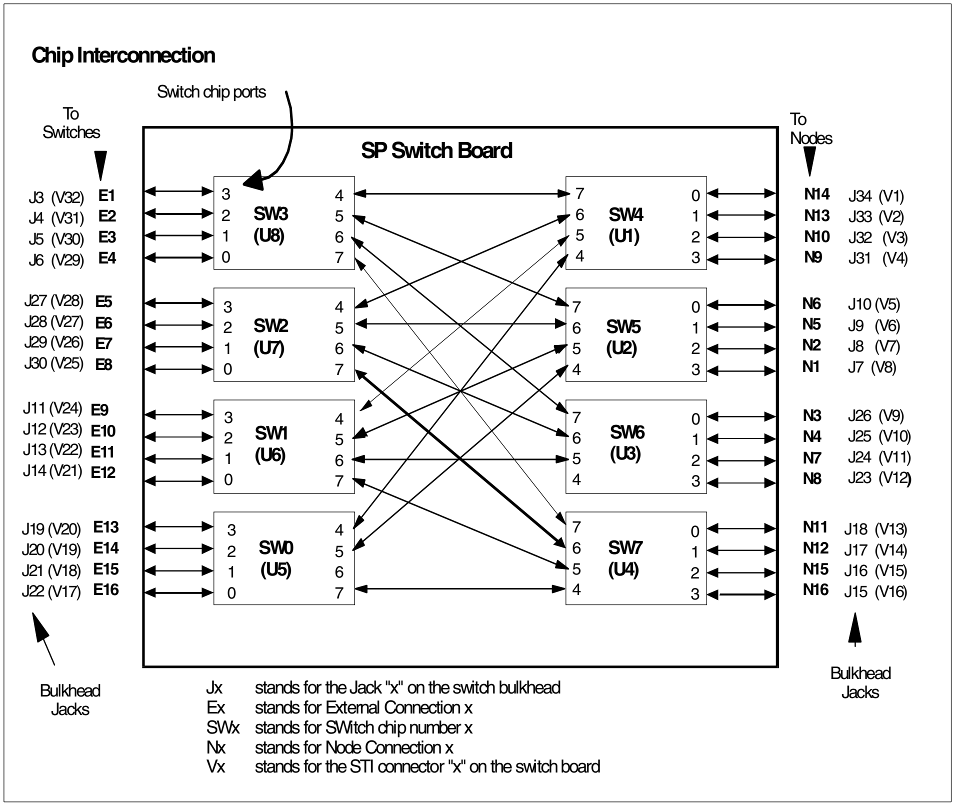


Figure 6-1 SP Switch layout

> **Important:** The SP Switch has a fixed Switch Node number to frame/slot combination. Also, the switch to switch connections have fixed ports. SP Switch2 does not have a fixed combination. You can connect any node to any switch port on an SP Switch2. Even if the node does not change its physical location, it can be connected to a different switch chip and port.

The heart of each switch board is the switch chip. There are eight switch chips on each board, and they have eight connectors each for sending and receiving data simultaneously. Each switch chip is interconnected with other chips through bi-directional redundant paths. Four chips are used for node communication and four chips are used for communication to switches in other frames. With the SP Switch2, we have no more master clocking, like we had on HiPS and the SP Switch. The SP Switch2 is designed to have multiple clock sources in the system. Instead of having a clock selection logic, the SP Switch2 has a separate oscillator for each switch chip.

See Example 6-1 for the **splstdata -s** output for a SP Switch2 system; it shows that there is no clock input used for the SP Switch2.

Example 6-1 SP Switch2 splstdata -s output (cutout)

```
...
switch# frame# slot# switch_partition# switch_type clock_input switch_name  swi
------- ------ ----- ----------------- ----------- ----------- ------------ ---
      1      1    17                 1         132           0 SP_Switch2 0
      2     10     2                 1         132           0 SP_Switch2 0
      3     10     4                 1         132           0 SP_Switch2 0
                                                 -> see here ^ all clock inputs
                                                    are 0
...
```

Note: On an SP Switch, you need a Switch configured as the master clock. That does not apply to an SP Switch2, where multiple clock sources are used. This could be a possible reason for error when using a SP Switch and the clock setting is not configured properly.

There is another enhancement on the SP Switch2. The JTAG interface is added to provide several supervisor function enhancements. The supervisor can use this interface now to write initialization data to the switch chips and read error information back from the switch chips.

Tip: To avoid problems with the switch board, ensure the following:

- If you have multiple SP Switches connected together check for a valid clock setting (not needed for SP Switch2).
- For SP Switches ensure that all cable connections are following the cabling rules. (see *SP Installation and relocation Guide*, GA22-7441).

6.1.2 SP Switch and SP Switch2 Adapters

For the current PSSP 3.2 Level we have SP Switch and SP Switch2 adapters available. There is no more HiPS support. Figure 6-2 on page 238 shows you the history and the future of Switch adapters.

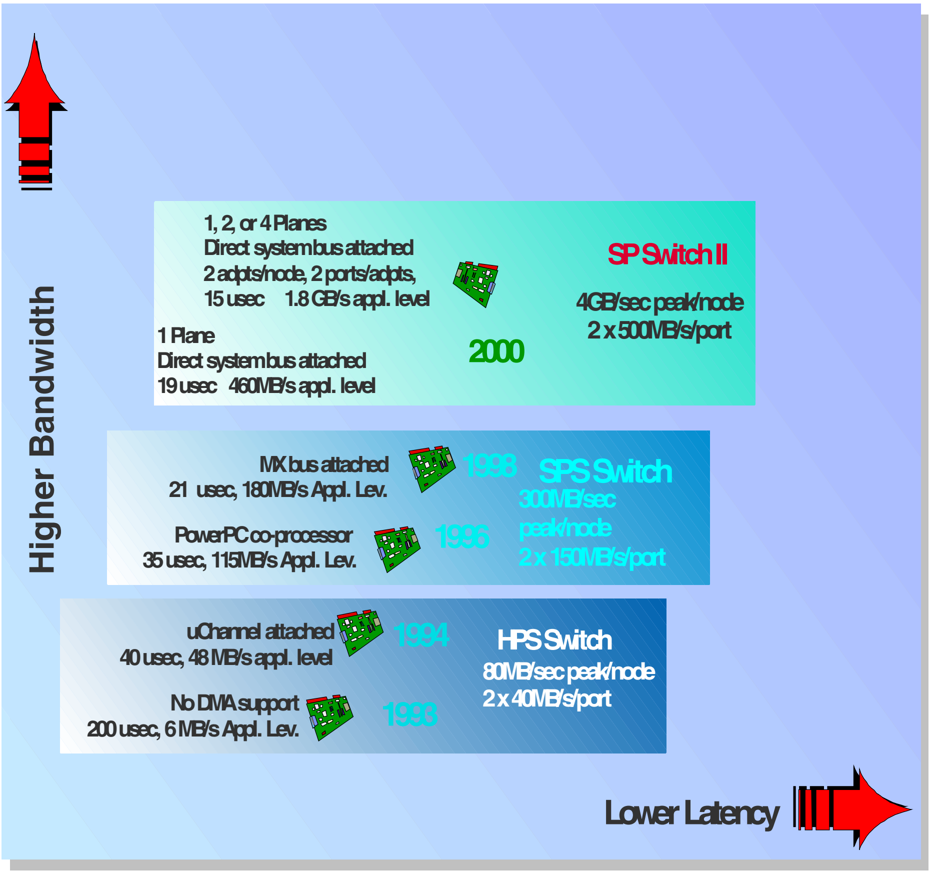


Figure 6-2 Switch history and future

Here is a overview of the current available adapters:

- SP Switch Adapter (F/C 4020)

  This adapter is used for the MCA nodes.

- SP Switch MX Adapter (F/C 4022)

  This adapter is used for 332 MHz PCI nodes.

- SP Switch MX2 Adapter (F/C 4023)

  This adapter is used for POWER3 SMP PCI Nodes and requires AIX V4.3.2 with APAR IX85409 or later with PSSP V3.1 (select F/C 9431).

- SP Switch Router Adapter (F/C 4021)

  This adapter is used for the 9077 SP Switch Router.

- IBM RS/6000 SP System Attachment Adapter (F/C 8396)

  This adapter is used to connect a 7017-S70, S7A, S80, S85. This adapter must be installed in slot #10 of the primary I/O drawer and slots #9 and #11 must remain empty. The adapter is also used for 7026 Models H80, M80, 6H0, 6H1.

> **Note:** SP Switch Adapter (FC 4020), SP Switch MX Adapter (FC 4022) and SP Switch MX2 Adapter can co-exist in an SP configuration. Applications will run faster with only SP Switch MX2 adapters in an SP system than with mixed SP Switch MX and MX2 Adapters.

The new SP Switch2 needs the following Adapter for the POWER3 High Node:

- SP Switch2 Adapter (F/C 4025)

  This feature needs an SP Switch2 Interposer (F/C 4032).

In every SP node there is a SP Switch or SP Switch2 Adapter. The data sent from another node connected to the switch board goes through the switch chips, through the cable to the switch adapter, where the node can use the data. Figure 6-3 illustrates an overview of the logical structure of a SP Switch adapter card.

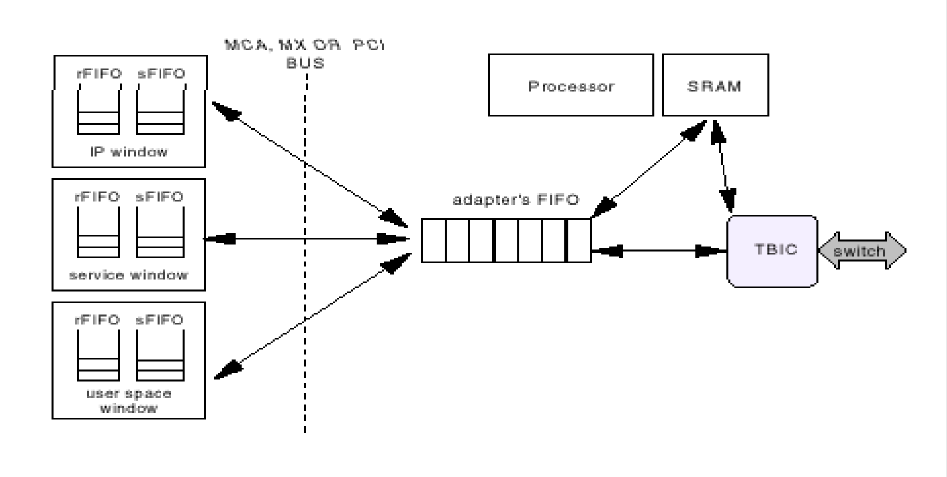


Figure 6-3 Overview of logical structure of an SP Switch adapter

On an SP Node, the data will be sent and received through memory areas called *windows*. When user space protocol is used for the data transfer, the application puts the data in windows. The SP Switch2 there also provides user space super packets. To achieve the full bandwidth of the SP Switch2, Super Packets are needed for sending large datagrams into the switch network. The data will be collected by the switch adapter microcode from the windows using the DMA engine of the adapter. After that another DMA engine on the adapter moves data to and from the switch chip.

The route tables are stored in the SRAM of the adapter. Responsible for the correct routing information for outgoing and incoming data packets is the microcode. This is the same for both SP Switch and SP Switch2. Based on the information provided by the primary node the fault service daemon creates the routing tables and updates it continuously for the adapter.

For more detailed information please refer to one or both of the following books: *Understanding and Using the SP Switch,* SG24-5161 and *RS/6000 SP Systems Handbook,* SG24-5596-01.

6.1.3 SP switches initialization

We are now talking about the switch initialization. Generally for both current Switch Types the initialization can be divided into three sections:

1. The switch adapter gets configured in the nodes.
2. The fault service daemon starts up on the nodes.
3. The **Estart** command is issued on the Control Workstation (CWS) or the switch admin daemon starts the switch automatically.

We want to give you a short overview of the things that are happening during initialization. During the boot phase, the cgfmgr is performing several actions. First of all, the css0 device is defined in the ODM CuDv class. The adapter's location itself is then verified and /dev/css0 is created in the /dev directory. When this is done, the device driver (cssdd3) is loaded, as well as the adapter dependent microcode. Assuming that everything is finer, the adapter device becomes the available state in the ODM CuDv class.

Finally, the adapter runs the Power on self test (POST) and, when everything is successful, the ODM CuAt class will be updated with the adapter status css_ready, as shown in Example 6-2 on page 241.

The fault service daemon, better known as the WORM, will then be started by the rc.switch script executed in the inittab file.

Note: Before you start the switch, the host responds must be yes. Check this with the **spmon -d** command.

Example 6-2 Adapter status

```
[root@sp6en0]:/> SDRGetObjects switch_responds
node_number  switch_responds autojoin     isolated     adapter_config_status sw
           1            1            1            0 css_ready  ""             "
           3            1            1            0 css_ready  ""             "
```

The fault service daemon takes care of several actions:

- Breadth first search: Explore the network, initialize it, find and report cable miswires.
- Generate routes and enable error reporting back to the Primary node.
- Send topology file only to nodes that need it (primary backup for example).
- Send database updates to all nodes.
- Send the KLOAD_ROUTE command to all nodes. The nodes will calculate routes and download the route tables to the adapters.

Since PSSP 3.1, there is a switch admin daemon called cssadm. This daemon monitors the nodes and switch adapter events and responds with an automatic **`Estart`**, if required.

6.1.4 How and when the switch initialization is invoked

The following scenarios provide information on when and how the switch initialization is invoked:

- Whenever an **`Estart`** command is issued.
- When **`Eunfence`** or **`Efence`** command fail with certain errors.
- When a switch scan fails.
- When the primary backup node takes over as primary node.

For more details about the switch and its initialization, please refer to one of the following publications: *PSSP 3 Survival Guide*, SG24-5344 or the *PSSP Diagnosis Guide*, GA22-7350.

6.2 SP switches diagnostics

We have two areas to diagnose for problems in the SP Switch and the SP Switch2: Software and hardware. Problems with the software and hardware prerequisites may manifest themselves as errors in the SP switches. You must consider both components as possible sources of errors. The following list describes the SP Switch and the SP Switch2 software and hardware prerequisites:

1. System Data Repository (SDR) component of PSSP, running on the control workstation.
2. Ethernet component of AIX: For SP Switch2 operation, there must be at least one connection between the control workstation and the primary node. SP Switch2 Time-Of-Day (TOD) recovery depends on an ethernet connection to all the nodes.
3. Group services hags and Topology services hats components of PSSP
4. SP system security services. Principal and group names for DCE entities use the default SP chosen names. These may not be the actual names on the system if you have overridden them using the spsec_overrides file.
5. SP Switch and SP Switch2 hardware monitor and control: hardmon component of PSSP.
6. Switch Fault Service: The Fault Service daemon (FSD) component of the SP Switch2 has to be operating on each node that is on the SP Switch2. When the daemon dies, the protocols (such as IP) are closed on this node. This is different than operation of the SP Switch. For the SP Switch, when the FSD dies, the protocols continue, only SP Switch recovery is not available.
7. Switch time of day (TOD): initiated and recovered by the emasterd daemon component of SP Switch2 software. This daemon runs on the control workstation and is responsible for selecting a suitable Master Switch Sequencer Node that maintain the switch TOD. When the daemon dies the switch TOD dies.

SP Switches themselves have internal subsystem components that are used for diagnostics. See Table 6-2 on page 242. For a detailed description, please refer to *PSSP Diagnosis Guide V3.2*, GA22-7350.

Table 6-2 Software subsystems differences between SP switches

Subsystem components	SP Switch	SP Switch2
worm	Yes	Yes
Ecommands	Yes	Yes
css	Yes	Yes

Subsystem components	**SP Switch**	**SP Switch2**
Pseudo device driver	Not available	Yes
Hardware Abstraction Layer (HAL)	Yes	Yes
Connectivity Matrix	Not available	Yes
CSS adapter device driver	Yes	Yes
CSS adapter microcode	Yes	Yes
CSS adapter diagnostics	Yes	Yes
MSS node recovery daemon: emasterd	Not available	Yes
Switch admin daemon: cssadm2	Not available	Yes

The SP Switch log and temporary files are located in the */var/adm/SPlogs/css* directory.The SP Switch2 log and temporary files are organized in a directory hierarchy. Next to each directory the specified file level is given.

See Table 6-3 for the file hierarchy.

Table 6-3 SP Switch2 log and temporary file hierarchy

Directory	**File level**
/var/adm/SPlogs/css	node
/var/adm/SPlogs/css0	**adapter**
/var/adm/SPlogs/css0/p0	port

Note:

- css - for global node level log files
- css0 - for adapter level log files (the 0 ins css0 is the adapter ID)
- css0/p0 - for port level log files (the 0 in p0 is the port number within the adapter)

First of all, view the AIX errorlog for isolating SP Switch/SP Switch2 or adapter errors. For switch related problems, login to the primary node. The **`Eprimary`** command lists the primary node by node number. In cases where the primary node failed, there will be no primary. In this case, login to the node listed under oncoming primary. In cases where the SP Switch/SP Switch2 continued working, it may have replaced the primary node several times. If you cannot locate the primary node using the **`Eprimary`** command, you can locate the primary node by looking inside the log files.

6.2.1 Diagnostic tools and routines

To diagnose problems on the switch or switch network we have some very useful tools available. Please refer to *PSSP: Command and Technical Reference*, SA22-7351 and *PSSP Diagnosis Guide V3.2,* GA22-7350 for more information.

Note: DO NOT run the commands/procedures on operational nodes

Adapter Error Log Analyzer (ELA)

When you suspect that the SP Switch or SP Switch2 adapter is not functioning properly, you can run the extended diagnostics. Adapter ELA is invoked on switch nodes to diagnose problems that occurred on the node. The command to run the adapter ELA directly is: **`diag -d css0 -A`** (Advanced diagnostics) or **`diag -d css0 -d`** (automatic POST tests).

CSS_test

This command verifies that the installation and configuration of the communications subsystem of the SP system completed successfully. It also makes an IP test for all nodes. A return code of 0 shows that the test completed without errors, 1 indicates a error occurred. In the /var/adm/SPlogs directory you will find the log file called CSS_log. See Example 6-3 for an example of this command.

Example 6-3 CSS_test output with some failing nodes

```
[0:root@sp3n14:]/home/root # CSS_test

CSS_test:  CSS Installation Verification Test started on
                   Sat Jul 21 15:50:42 EDT 2001.
-----------------------------------------------------------------------------
CSS_test:  Beginning Ethernet IP test of the nodes in partition sp3en0.
CSS_test:  Following failed Ethernet IP test - deleting from subsequent tests:
sp3n13
-----------------------------------------------------------------------------
CSS_test:  Show LPP ssp.basic installation levels:  (lslpp -Lq ssp.basic)

        -Node-          -LPP Name-  -Installation Level-
        sp3n01:         ssp.basic       3.2.0.7
        sp3n05:         ssp.basic       3.2.0.7
        sp3n06:         ssp.basic       3.2.0.11
        sp3n07:         ssp.basic       3.2.0.7
        sp3n08:         ssp.basic       3.2.0.7
        sp3n09:         ssp.basic       3.2.0.7
        sp3n10: rshd: 0826-826 The host name for your address is not known.
        sp3n10: spk4rsh: 0041-004 Kerberos V4 rcmd failed: rcmd protocol
failure.
```

```
        sp3n10: rshd: 0826-826 The host name for your address is not known.
        sp3n11:         ssp.basic       3.2.0.7
        sp3n12:         ssp.basic       3.2.0.11
       sp3n14:         ssp.basic       3.2.0.7
        sp3n15:         ssp.basic       3.2.0.7
        sp3n14:         ssp.basic       3.2.0.7

CSS_test:  Show LPP ssp.css installation levels:  (lslpp -Lq ssp.css)

        -Node-          -LPP Name-  -Installation Level-
        sp3n01:         ssp.css         3.2.0.6
        sp3n05:         ssp.css         3.2.0.6
        sp3n06:         ssp.css         3.2.0.11
        sp3n07:         ssp.css         3.2.0.6
        sp3n08:         ssp.css         3.2.0.6
        sp3n09:         ssp.css         3.2.0.6
        sp3n10: rshd: 0826-826 The host name for your address is not known.
        sp3n10: spk4rsh: 0041-004 Kerberos V4 rcmd failed: rcmd protocol
failure
.
        sp3n11:         ssp.css         3.2.0.6
        sp3n12:         ssp.css         3.2.0.11
        sp3n14:         ssp.css         3.2.0.6
        sp3n15:         ssp.css         3.2.0.6
        sp3n14:         ssp.css         3.2.0.6

CSS_test:  Show inconsistent ssp.css files for relevant nodes:  (lppchk)

 -Node-         -File Name-                     -Actual Size-   -Expected Size-

-----------------------------------------------------------------------------
CSS_test:  Beginning Switch IP test of the nodes in partition sp3en0.
CSS_test:  Following nodes failed Switch IP test:
sp3sw01 (192.168.13.1)
sp3sw06 (192.168.13.6)
sp3sw07 (192.168.13.7)
sp3sw10 (192.168.13.10)
sp3sw14 (192.168.13.14)      <-this node is fenced
-----------------------------------------------------------------------------
CSS_test:  CSS Installation Verification Test completed on
                   Sat Jul 21 15:51:26 EDT 2001.
```

mult_senders_test

This command stresses the communication over the switch in order to detect the node that is injecting damaged packets in the network. It is located in the /usr/lpp/ssp/bin/spd directory.

Attention: This command causes a very heavy load and possible performance degradation when used.

The **mult_senders_test** command is useful when diagnosing intermittent communication problems. You can specify the senders and receiver nodes, but neither the primary nor the secondary can be used as senders or receivers. You must have four or more nodes to run this test.

switch_stress

This command runs stress tests on a single switch chip. It determines if the chip fails under stress conditions. This command sends and receives data across the chip, but the primary and the secondary nodes do not participate in the test. You must have four or more nodes to run this test. Example 6-4 shows the **switch_stress** command. It is located in the /usr/lpp/ssp/bin/spd directory.

Attention: This command causes heavy load and possible performance degradation when used. First, decide which nodes will be used for the test. These nodes cannot run parallel applications during this test. Please refer to *PSSP: Command and Technical Reference*, SA22-7351 for more information.

Example 6-4 switch_stress command

```
sp3en0 > switch_stress -s 13
Test initialization might take several minutes. Please wait...
Switch Chip Stress Test is about to start. Please wait...


  Time      Node  Pers.    Messages
 ----------------------------------------
08:58:54      5    P    2548-421 Loading the scst_test library
08:58:54      5    P    2548-451 Starting SP Switch Diagnostic Framework.
08:58:54      5    P    2548-250 Port#0 of the switch chip is disabled; will
not be tested
08:58:54      5    P    2548-250 Port#1 of the switch chip is disabled; will
not be tested
08:58:54      5    P    2548-250 Port#2 of the switch chip is disabled; will
not be tested
08:58:54      5    P    2548-250 Port#3 of the switch chip is disabled; will
not be tested
08:58:54      5    P    SCST is going to test switch chip#13
08:58:54      5    P    2548-452 New model was initialized successfully.
modelID
 = 1.
08:58:54      5    P    Start iteration# 1 for ports 4, 5
sp3en0 > 08:58:54      5    P    Assistant node device ids are: 14, 6
```

```
08:58:55      6    S    2548-421 Loading the scst_test library
08:58:55     14    S    2548-421 Loading the scst_test library
08:58:55      6    S    2548-451 Starting SP Switch Diagnostic Framework.
08:58:55     14    S    2548-451 Starting SP Switch Diagnostic Framework.
08:59:45      5    P    2548-274 Iteration <4,5>: HAL packets: sent 1661410,
lost 9363. Switch chip failure count:    0
08:59:46      5    P    Start iteration# 2 for ports 4, 6
08:59:46      5    P    Assistant node device ids are: 14, 7
08:59:47      7    S    2548-421 Loading the scst_test library
08:59:47      7    S    2548-451 Starting SP Switch Diagnostic Framework.
09:00:46      5    P    2548-274 Iteration <4,6>: HAL packets: sent 1661410,
lost 3294. Switch chip failure count:    0
09:00:47      5    P    Start iteration# 3 for ports 4, 7
09:00:47      5    P    Assistant node device ids are: 14, 12
09:00:47     12    S    2548-421 Loading the scst_test library
09:00:47     12    S    2548-451 Starting SP Switch Diagnostic Framework.
09:01:37      5    P    2548-274 Iteration <4,7>: HAL packets: sent 1661410,
lost 1289. Switch chip failure count:    0
09:01:38      5    P    Start iteration# 4 for ports 5, 6
09:01:38      5    P    Assistant node device ids are: 6, 7
09:02:25      5    P    2548-274 Iteration <5,6>: HAL packets: sent 1661410,
lost 6268. Switch chip failure count:    0
09:02:26      5    P    Start iteration# 5 for ports 5, 7
09:02:26      5    P    Assistant node device ids are: 6, 12
09:03:13      5    P    2548-274 Iteration <5,7>: HAL packets: sent 1661410,
lost 5905. Switch chip failure count:    0
09:03:14      5    P    Start iteration# 6 for ports 6, 7
09:03:14      5    P    Assistant node device ids are: 7, 12
09:04:02      5    P    2548-274 Iteration <6,7>: HAL packets: sent 1661410,
lost 10024. Switch chip failure count:    0
09:04:03      5    P    The model has finished
09:04:03      5    P    2548-272 Model results summary: switch chip errors: 0
on the tested chip, 0 total
09:04:03      6    S    2548-455 Going to terminate SP Switch Diagnostic
Framework.
09:04:03     14    S    2548-455 Going to terminate SP Switch Diagnostic
Framework.
09:04:03      7    S    2548-455 Going to terminate SP Switch Diagnostic
Framework.
09:04:03     12    S    2548-455 Going to terminate SP Switch Diagnostic
Framework.
09:04:03      5    P    2548-273 Model results summary: HAL packets: sent
9968460, lost 36143
09:04:03      5    P    2548-274 Iteration <4,5>: HAL packets: sent 1661410,
lost 9363. Switch chip failure count:    0
09:04:03      5    P    2548-274 Iteration <4,6>: HAL packets: sent 1661410,
lost 3294. Switch chip failure count:    0
09:04:03      5    P    2548-274 Iteration <4,7>: HAL packets: sent 1661410,
lost 1289. Switch chip failure count:    0
```

```
09:04:03      5    P    2548-274 Iteration <5,6>: HAL packets: sent 1661410,
los09:04:03      5    P    2548-274 Iteration <5,7>: HAL packets: sent 1661410,
lost 5905. Switch chip failure count:    0
09:04:03      5    P    2548-274 Iteration <6,7>: HAL packets: sent 1661410,
lost 10024. Switch chip failure count:    0
09:04:03      5    P    2548-461 Model has ended.
09:04:03      5    P    2548-455 Going to terminate SP Switch Diagnostic
Framework.
09:04:03      5    P    2548-456 SP Switch Diagnostic Framework terminated.
09:04:03      5    P    2548-422 The scst_test library unloaded
09:04:03      5    P    2548-423 The SP Diagnostic is exiting
```

wrap_test

This command tests the physical link between the ports of the switch. It is located in the /usr/lpp/ssp/bin/spd directory. To run this command, the node must be fenced. It opens a Java console that guides you through the test. It is necessary to have two wrap plugs to test both the port and the cable. If you need the wrap plugs, please contact your IBM Hardware Support Center.

SP Switch External Clock Diagnostics

To verify if an external clock is operational on a node, we have a command called **read_tbic**. This command is located in the /usr/lpp/ssp/css/diags directory. Verify the operation of the external clock with the following steps:

1. Login to the suspected node as root.
2. Issue the following command as Example 6-5.

Example 6-5 External clock diagnostics

```
[0:root@sp3n14:]/home/root # /usr/lpp/ssp/css/diags/read_tbic -s
TBIC status register        : 12000000
```

3. Look at bits 3 and 4 (bits are numbered from left to right, starting with 0). Here the status register shows 12XXXXXX.

bits	012**34**567
status = 12	000**10**010

4. In this case, either bits 3 or 4 are OFF (equals zero). This means the external clock is not operational at this node (the node was fenced).
5. A running node will show a Trail Blazer Interface Chip (TBIC) status register like 78XXXXXX.

bits	012**34**567
status = 78	011**11**000

6. In this case, both bits 3 and 4 are ON (equal to one), the external clock is operational on the node.

css_cdn script - stopping the worm

This is a short and very useful script to kill the fault service daemon (FSD or WORM) without checking for the process ID. The script is very useful for terminating the worm daemon. This command is undocumented, but the mechanism works because the script looks for the process ID of the WORM and terminates this process. The script takes about eight seconds to complete, then the command prompt comes back. The script is located here:

```
/usr/lpp/ssp/css/css_cdn
```

You will get no message when the command is executed on the desired node. See Example 6-6 on page 249.

Example 6-6 The use of css_cdn on simply one node

```
0:root@sp3n09:]/usr/lpp/ssp/css # ps -ef |grep fault
   root  35094   4942   1 16:55:41  pts/0  0:00 grep fault
   root  36368      1   0 16:55:35      -  0:00
/usr/lpp/ssp/css/fault_service_Worm_RTG_SP -r 8 -b 1 -s 4 -p 3 -a TB3 -t 28

0:root@sp3n09:]/usr/lpp/ssp/css # /usr/lpp/ssp/css/css_cdn
```

After the command is issued no more WORM is running on the node.

```
0:root@sp3n09:]/usr/lpp/ssp/css # ps -ef |grep fault
   root  14876   4942   1 16:56:02  pts/0  0:00 grep fault
0:root@sp3n09:]/usr/lpp/ssp/css #
```

6.3 Switch problems – what can happen

Basically all problems occurring on an SP system with an SP Switch or SP Switch2 should be handled in the *SP Switch Service Guide*, GA22-7443, *SP Switch2 Service Guide*, GA22-7444, and the *Parallel System Support Program for AIX: Diagnosis Guide*, GA22-7350. Further problem determination may require you to contact your local IBM support team.

Attention: Use SDRArchive before performing ANY SDR changes do an SDRArchive.

There are several problems that look strange and are not easy to debug. Hardware related switch problems are caused by improper seated cables or even defective adapters. SP Switch board defects are rather rare. Many of the problems that are not so easy to fix are very often fixed by APARs or complete

PSSP PTF sets; see Appendix C, “Software maintenance strategy” on page 285 for more details. However, some of the software related problems occur pretty often, and we try to gather some of the solutions for easier and faster problem determination in this chapter.

6.4 SDR related switch problems

This part of the chapter will show some case studies and possible SDR related problems that can cause switch problems.

SDR modified due to wrong command usage

It is possible that a PSSP command was not used correctly or that even the SDR gets a problem without even using a specific modification to the system. Some of these problems can cause strange behaviors on an SP System with an SP Switch or SP Switch2. We try to gather some of these common problems to give you a fast way to solve the problem.

All node_number entries in switch_responds are the same

Let’s assume that a customer was trying to change the isolated flag in the switch_responds class from 1 to 0 for a particular node because there was already a SDR mismatch. The right command usage would be

```
SDRChangeAttrValues switch_responds node_number==1 isolated=1
```

However, a moment without full concentration can cause something like this:

```
SDRChangeAttrValues switch_responds node_number=1 isolated=1
```

Using the = instead of the appropriate == can cause severe problems. Using one equal sign will affect ALL nodes, not just the desired node_number 1. Example 6-7 shows an operational switch respond output before we entered the incorrect SDRChangeAttrValues command.

Example 6-7 SDRGetObjects switch_responds

```
node_number  switch_responds autojoin     isolated     adapter_config_status
1            0            1            1 css_ready    ""           ""
5            1            1            0 css_ready    ""           ""
6            1            1            0 css_ready    ""           ""
7            1            1            0 css_ready    ""           ""
8            1            1            0 css_ready    ""           ""
```

Now the wrong usage of the **`SDRChangeAttrValues`** command would cause the output shown in Figure 6-8 on page 251.

Example 6-8 All node_number entries are the same now

```
node_number  switch_responds autojoin     isolated     adapter_config_status
1            0            1            1 css_ready    ""           ""
1            1            1            0 css_ready    ""           ""
1            1            1            0 css_ready    ""           ""
1            1            1            0 css_ready    ""           ""
1            1            1            0 css_ready    ""           ""
```

The good thing is that the switch network itself is still working. In our example, aping will work successfully, but a **spmon -d** will show a positive switch_responds only for node1.

How to fix this SDR mismatch

Assuming that a recent SDRArchive was made of the SDR, it would be no big deal to make this problem disappear. From the archive you can extract just the class file switch_responds, as shown in Figure 6-9 on page 251.

> **Note:** You must know the partition's name.

Example 6-9 Extract the switch_responds class from archive

```
tar -xvf backup.01193.1453.SDR_config
/spdata/sys1/sdr/partition/192.168.3.130/switch_responds
and the file switch_responds will look good again:
# /spdata/sys1/sdr/partitions/192.168.3.130/classes # pg switch_responds
1=1 2=0 3=1 4=1 5=css_ready
1=5 2=1 3=1 4=0 5=css_ready
1=6 2=1 3=1 4=0 5=css_ready
1=7 2=1 3=1 4=0 5=css_ready
1=8 2=1 3=1 4=0 5=css_ready
```

Now that the file switch_responds is in the right place, we need to stop the sdr daemon now to make this change valid. Otherwise, an **Estart** would cause the same error to occur again. Use the following steps to start and stop the DRD daemon:

1. Stop the SDR daemon by issuing the **stopsrc -g sdr** command.
2. Verify that the switch_responds is working properly.
3. Start the SDR daemon by issuing the **startsrc -g sdr** command.

The **SDRGetObjects switch_responds** will look similar to Example 6-7 on page 250.

The following is a useful list, with the steps required to fix an SDR switch related problem:

- Do a SDRArchive (as often as possible).
- Replace the corrupted/defective switch_responds file from the SDRArchive.
- If no archive was done the file switch_responds can be modified by hand (it is fine on small systems but intensive on big systems).
- Stop the sdr subsystem with **stopsrc -g sdr**.
- Start the sdr subsystem with **startsrc -g sdr**.

6.5 Node problems on the switch

In some cases, only one node is affected with switch problems and some of the problems are not easy to find in the regular PSSP Diagnosis guide.

Single node not on the switch – WORM is dead

Tip: There are some reasons why the switch adapter may not be css_ready or the fault service daemon (WORM) wont start.

- /dev/css0 is corrupted
- /var file system is full
- Cable connection from the switch adapter to the switch board is bad
- Adapter POST has failed - see adapter status diag_fail

You noticed that a node is no longer communication over the switch network. So **spmon -d** was executed to verify the switch responds. See Example 6-10 on page 252.

Example 6-10 spmon shows switch responds autojoin

```
...
5.  Checking nodes
----------------------------------- Frame 1 ----------------------------------
                      Host     Switch   Key     Env   Front Panel      LCD/LED
Slot Node Type  Power Responds Responds Switch  Error LCD/LED          Flashes
---- ---- ----- ----- -------- -------- ------- ----- ---------------- -------
  1    1  high   on      yes     autojn   normal   no  LCDs are blank     no
  5    5  thin   on      yes      yes     normal   no  LEDs are blank     no
...
```

We found that node 1 has autojoin for its switch responds. We are now able to see also the changed status in the out.top file that is located on the primary node and the switch responds output provided from the **SDRGetObjects switch_responds** command. See Example 6-11, “out.top and SDRGetObjects output” on page 253.

Example 6-11 out.top and SDRGetObjects output

```
...
format 1
16 18
# Node connections in frame L01 to switch 1 in L01
s 15 3  tb3 0 0          E01-S17-BH-J7 to E01-N1   -4 R: device has been
remo)
s 15 2  tb3 1 0          E01-S17-BH-J8 to Exx-Nxx   -4 R: device has been
rem)
s 16 0  tb3 2 0          E01-S17-BH-J26 to Exx-Nxx   -4 R: device has been
re)
s 16 1  tb3 3 0          E01-S17-BH-J25 to Exx-Nxx   -4 R: device has been
re)
...

now look at SDRGetObjects

[2:root@sp3en0:]/home/root # SDRGetObjects switch_responds
node_number  switch_responds autojoin     isolated     adapter_config_status sw
          1            0            1            1 css_ready    ""           "
          5            1            1            0 css_ready    ""           "
          6            1            1            0 css_ready    ""           "
          7            1            1            0 css_ready    ""           "
          8            1            1            0 css_ready    ""           "
          9            1            1            0 css_ready    ""           "
         10            1            1            0 css_ready    ""           "
         11            1            1            0 css_ready    ""           "
         12            1            1            0 css_ready    ""           "
         13            1            1            0 css_ready    ""           "
         14            1            1            0 css_ready    ""           "
         15            1            1            0 css_ready    ""           "
```

We also find errorlog entries on the failing node. The errorlog shows a Switch Fault Service Daemon Terminated. See Example 6-12, “errorlog output” on page 253.

Example 6-12 errorlog output

```
[tty0:root@sp3n01:]/home/root # errpt |pg
IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
BEE2FB4A   0711133901 U U hats.sp3en0    Contact with a neighboring adapter
losr
E8817142   0711133901 P S Worm           Switch Fault Service Daemon Terminated
```

```
6E0AA114   0711133901 P S Worm           Switch daemon received SIGTERM
...

the detailed errorlog looks like:

[tty0:root@sp3n01:]/home/root # errpt -a |pg
LABEL:          SP_SW_FSD_TERM_ER
IDENTIFIER:     E8817142

Date/Time:       Wed Jul 11 13:39:50
Sequence Number: 545
Machine Id:      00091276A400
Node Id:         sp3n01
Class:           S
Type:            PERM
Resource Name:   Worm

Description
Switch Fault Service Daemon Terminated

Probable Causes
SYSTEM I/O BUS
Switch adapter failure
Switch clock signal missing

Failure Causes
SYSTEM I/O BUS
Switch cable faulty

        Recommended Actions
        Run adapter diagnostics

Detail Data
DETECTING MODULE
PSSP,fs_daemon_init.c,           1.46,1158
ERROR ID
.03LUczax6Hv.nsB//A.e.z...................
REFERENCE CODE
/SV/...ax6Hv.UgB//A.e.z...................
---------------------------------------------------------------------------
LABEL:          SP_SW_SIGTERM_ER
IDENTIFIER:     6E0AA114

Date/Time:       Wed Jul 11 13:39:50
Sequence Number: 544
Machine Id:      00091276A400
Node Id:         sp3n01
Class:           S
Type:            PERM
```

```
Resource Name:   Worm

Description
Switch daemon received SIGTERM

Probable Causes
Another process sent a SIGTERM

User Causes
Operator ran Eclock
Operator ran rc.switch on node and switch daemon was restarted
User program sent SIGTERM

        Recommended Actions
        Run rc.switch to restart switch daemon

Detail Data
DETECTING MODULE
PSSP,fs_daemon_init.c,           1.46, 980
ERROR ID
.I2e0i/ax6Hv.pw3//A.e.z...................
REFERENCE CODE
/SV/...ax6Hv.uf0//A.e.z...................
PID of process sending SIGTERM
       15742
Name of process sending SIGTERM
ksh
```

The easiest way to go is to restart the WORM daemon on the failing node. Right now there is no WORM daemon running. So we issue the **rc.switch** command to restart the daemon. See Example 6-13, “rc.switch” on page 255. If it still fails please REFER to the *PSSP Diagnosis Guide V3.2 GA22-7350*.

Example 6-13   rc.switch

```
[tty0:root@sp3n01:]/home/root # ps -ef |grep fault
    root 17756 15742   4 13:46:37      0  0:00 grep fault

[tty0:root@sp3n01:]/home/root # /usr/lpp/ssp/css/rc.switch
"adapter/mca/tb3"
/etc/inittab entry specified as once for the fault service daemon.

look now:
[tty0:root@sp3n01:]/home/root # ps -ef |grep fault
    root 17344     1   0 13:47:05      -  0:00 /usr/lpp/ssp/css/fault_service_W
    root 17794 15742   2 13:48:01      0  0:00 grep fault
```

After issuing the **rc.switch** command the WORM daemon started again and so the node was able to join the switch network again automatically (node was set to autojoin). In a case where the node is fenced you need to unfence the node either with the **Eunfence** command or through the smit menus.

SP Switch adapter has diag_fail

The SP Switch adapter usually has the status css_ready. You can check this status with the **SDRGetObjects switch_responds** command or look at the /var/adm/SPlogs/css/rc.switch.log file. See Example 6-14 on page 256.

Example 6-14 Adapter config status sample

```
[2:root@sp3en0:]/home/root # SDRGetObjects switch_responds
node_number  switch_responds autojoin     isolated     adapter_config_status sw
           1            0            1            1 css_ready    ""           "
           5            1            1            0 css_ready    ""           "
...
```

When a adapter is failing the diags it shows diag_fail. This status can be steady and the adapter can really be defective. If the node was powered on before the switch board was powered on and the switch adapter in the node has received no clock signal this can cause a diag_fail status. This can occur during a install. In this case the worm daemon will not start while rc.switch is running. The diag_fail status especially in a install phase is no defect of the adapter. To fix this problem we can do the following (ensure that switch board is powered on):

1. Reboot the node(s)
2. Reconfigure the adapter. This will require stopping any processes that could hold the css0 device, for example hats. The commands to be issued are:
   - **stopsrc -s hats**
   - **/usr/lpp/ssp/css/ucfgtb3 -l css0 -v**
   - **/usr/lpp/ssp/css/cfgtb3 -l css0 -v**
   - **startsrc -s hats**

Note: if there are other processes, including application processes, holding onto the css0 device, they must be stopped as well.

If the status is still diag_fail and there is no other solution possible you should contact your local IBM Support to analyze the problem and probably exchange the adapter hardware if needed.

Multiple nodes are not rejoining the switch

During a planned power off of several switch attached nodes are not able to rejoin the switch network via the **Estart** command. See Example 6-15.

Example 6-15 Estart fails for several nodes - flt output and out.top

```
flt output:
```

```
04/03/01 15:59:34 (i) : 2510-744 Estart initiated.
04/03/01 15:59:34 CSswitchInit: Switch network
Initialization Started!
04/03/01 15:59:35 (e) handlePh1SwSvcResponse: 2510-919 Bad
Device Signature detected. Device id = 100050.
04/03/01 15:59:35 (n) DisableChip: 2510-798 Disabling
Switch chip - device_id = 100050
04/03/01 15:59:36 (e) handlePh1SwSvcResponse: 2510-919 Bad
Device Signature detected. Device id = 100057.
04/03/01 15:59:36 (n) DisableChip: 2510-798 Disabling
Switch chip - device_id = 100057
04/03/01 15:59:55 CSswitchInit: Switch network
Initialization Ended!
04/03/01 16:00:02 (i) init_on_startup_msg: The Primary
backup is node testnode.itso.ibm.com
04/03/01 16:00:03 (i) init_on_startup_msg: Switch
initialization completed successfully!
```

out.top output:

```
,s 54 3  col 72 0          E17-S17-BH-J31 to E17-N9
,s 54 2  col 73 0          E17-S17-BH-J32 to E18-N9
,s 57 0  xxx xx x          E17-S17-BH-J18 to Exx-Nxx   -4
,L: device has been removed from network - faulty (link ha
,s been removed from network - not connected)
,s 57 1  xxx xx x          E17-S17-BH-J17 to Exx-Nxx   -4
,L: device has been removed from network - faulty (link ha
,s been removed from network - not connected)
,s 54 1  col 76 0          E17-S17-BH-J33 to E17-N13
,s 54 0  col 77 0          E17-S17-BH-J34 to E18-N13
,s 57 2  xxx xx x          E17-S17-BH-J16 to Exx-Nxx   -4
,L: device has been removed from network - faulty (link ha
,s been removed from network - not connected)
,s 57 3  xxx xx x          E17-S17-BH-J15 to Exx-Nxx   -4
,Ls been removed from network - not connected)
: device has been removed from network - faulty (link ha
```

Due to the fact that a power off was performed on these nodes it is unlikely that all adapters or switch ports are defective. First of all the cables should all be reseated and checked for proper connection. Then we do the following:

- **`Equiesce`** (to disable switch error recovery and primary node takeover).
- Either run **`dsh -a /usr/lpp/ssp/css/css_cdn`** from the CWS (will kill WORM on all nodes in that partition) or simply **`/usr/lpp/ssp/css/css_cdn`** on the node itself.
- Power cycle the switch board off/on. This is useful due to the fact that switch ports can be in undefined state and no soft reset can change the status.
- Either run **`dsh -a /usr/lpp/ssp/css/rc.switch`** on the CWS or **`/usr/lpp/ssp/css/rc.switch`** on each node itself.
- Run **`Estart`**.

After following these steps the nodes are back on the switch network and everything is running fine again.

Worm daemon problems - fixed by APARs

Some problems that you perhaps see on the SP Switch are probably already fixed by a PTF Set or just APAR. We wont list all available APARs available for PSSP in this section but we want to focus on some important software bugs that occur more often.You can search the APAR Database at: http://techsupport.services.ibm.com/rs6000/aix.CAPARdb. See Table 6-4.

Table 6-4 WORM related problems/fixed by APAR

Symptom	Description	APAR fix no.
Switch went to down	**ORIGINATING DETAILS:** flt file shows the errors: 2510-898 unable to access SDR to get the list of auto-join nodes rc= -1. 2510-195 The fault service daemon got a SIGTERM signal. **RESPONDER SUMMARY:** Closed the window so the child process will exit on SIGTERM without resetting the adapter. **RESPONDER CONCLUSION:** There is a small window where the primary forks a child process and an SDR test is run where a SIGTERM to the child will result in the child call the standard SIGTERM handler and reset the adapter.	IY17438

Symptom	Description	APAR fix no.
Fault Service Daemon (FSD) response time	**ORIGINATING DETAILS:** fsd response time **RESPONDER SUMMARY:** When a node is heavily loaded, the switch daemon can be delayed processing certain packets. In some cases, the node is dropped from the switch. **RESPONDER CONCLUSION:** The packet processing code in the fault service daemon has been changed to improve processing time.	IY17579
Primary daemon core dumps in CSRECOVERY	**RESPONDER SUMMARY:** This problem was caused by an array in switch recovery overflowing and wiping out pointers and other variables. The array now has one element assigned to each chip and node in the system, eliminating the overflow condition. **RESPONDER CONCLUSION:** Switch recovery was changed to have a fixed array containing error reset information. The previous array was coded to 100 entries. If more than 100 resets were pending then the array would overflow, and pointers to other structures would be wiped out. This would cause segmentation faults. The new arrays have one element for each switch chip/node in the system, so the overflow will not occur.	IY18011
Nodes drop off the switch with ucode: Link1_Bad_Svc_Packet	**ORIGINATING DETAILS:** Nodes drop of the switch, logging UCODE: Link1 Bad Svc Packet, 5:35>Port 0 CRC Err **RESPONDER SUMMARY:** Device driver was interrogating service buffer error packet incorrectly and reporting the error on the wrong link. The result is that the fault service daemon will ignore the packet instead of taking correct recovery action **RESPONDER CONCLUSION:** Device driver now reports correct link for a service buffer error to the fault service daemon.	IY15593

Symptom	Description	APAR fix no.
Node crash on detach of css0 interface	**ORIGINATING DETAILS:** A node crash can occur when detaching the css0 interface; e.g.: /usr/lpp/ssp/css/ifconfig css0 detach **LOCAL FIX AS REPORTED BY ORIGINATOR:** A delete of the interface may suffice instead of detaching the interface. **RESPONDER SUMMARY:** A node crash can occur when the css0 interface is detached as follows: "/usr/lpp/ssp/css/ifconfig css0 detach" **RESPONDER CONCLUSION:** The switch IP kernel extension, if_ls, has been changed to prevent a node crash when detaching the css0 interface.	IY16055
Switch responds green, IP stops working on a node after recovery	**ORIGINATING DETAILS:** A secondary nodes failed to respond to node initialization in time packets and were taken off the switch. However, the switch responds bit (switch_responds) was not set. The node's fault_service_daemon shows the error: Stuck TBIC interrupt - reinitializing adapter. **LOCAL FIX AS REPORTED BY ORIGINATOR:** **RESPONDER SUMMARY:** When a node is being isolated from the switch, transient errors can occur before the adapter is fully disabled. These can appear as hot interrupts and cause the adapter to be refreshed and not reset. This leaves IP in an inconsistent state. If the node is being isolated during initialization switch responds will not be updated. **RESPONDER CONCLUSION:** Transient errors that do not clear when a node is being isolated from the switch will be treated as link errors.	IY12824

Symptom	Description	APAR fix no.
Eunfence appears successful but has failed due to no primary backup nodes	**ORIGINATING DETAILS:** When the switch primary node tries to unfence a node, there must be an existing backup node for the fence to work properly, but the switch logs can indicate that such a fence is successful, and the SDR switch_responds Class can incorrectly show the node unfenced. The logs and SDR should show the fence failed. We should also log when the primary can't choose a backup node because there is no other node running the same code version as the primary. **RESPONDER SUMMARY:** Eunfence may appear successful but may actually fail if there is no operational primary backup node. In a system with dependent nodes, a backup candidate may be rejected if its PSSP level (as determined by the SDR code_version attribute in the Node class) is not the same as the primary node's code_version. **RESPONDER CONCLUSION:** The fault_service daemon will attempt to unfence the node even if there is no operational backup. In the pick_backup() function of the Worm, the backup candidate's eligibility test was changed: in a system with dependent nodes, the backup candidate must be at PSSP-2.3 or higher (as indicated by the SDR code_version attribute in the Node class).	IY12826

Symptom	Description	APAR fix no.
cfgtb3 should specify the -x option to suppress parent testing	**RESPONDER SUMMARY:** New with AIX 4.3.3, diagnostics are automatically invoked on parent devices whenever the diag command is run. This means that TB3, TB3MX and TB3PCI switch adapter configuration (which runs the diag command) will result in the invocation of sysplanar0 diagnostics. If a problem is detected on the sysplanar device, TBx switch adapter configuration will fail with ODM adapter_status set to diag_fail (refer to /var/adm/SPlogs/css/rc.switch.log). **RESPONDER CONCLUSION:** Support will be added to the TBx switch adapter configuration method for the new "-x" diag command flag to bypass diagnostics on the parent (sysplanar) device. "-x" will be specified only if AIX APAR IY04975 is installed, which provides diag command support for the new flag.	IY06775

6.6 SP Switch problems

Here we describe SP Switch related problems. The SP Switch has the possibility to recover automatically from node and switch clock problems. For this function there is the cssadm daemon available. The cssadm daemon is started in the /etc/inittab file on the control workstation (CWS). The SRC subsytem name for the daemon is swtadm. This SRC substyem is not running on a SP Switch2 system. Basically only the node switch recovery is enabled. You can modify the following file and add the line Switch 1 to /spdata/sys1/ha/css/cssadm.cfg. See Example 6-16.

Example 6-16 /spdata/sys1/ha/css/cssadm.cfg file on CWS

```
sp3en0 > pg cssadm.cfg
# IBM_PROLOG_BEGIN_TAG
# This is an automatically generated prolog.
#
#
#
# Licensed Materials - Property of IBM
#
# (C) COPYRIGHT International Business Machines Corp. 1998
# All Rights Reserved
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
# IBM_PROLOG_END_TAG
Node 1             <- This is the original entry
 Switch 1<- This is the added line
```

If you want to have no switch clock recovery, you have to explicitly change the value to Switch 0. After you modified the `/spdata/sys1/ha/css/cssadm.cfg` file you have to stop and restart the subsystem using the following commands:

- `stopsrc -s swtadm`
- `startsrc -s swtadm`

Now we can also enable the SP Switch power monitoring recovery. Therefore, you need to modify the /spdata/sys1/spmon/hmthresholds file. See Example 6-17 on page 263.

Example 6-17 /spdata/sys1/spmon/hmthresholds file

```
...
# 0x81 0x00 0xff .700 0xff .700 0xff 43.2 52.8 43.2 52.8 0x00 45.0 2.97 3.63
# Software thresholding disabled
# 0x81 0x00 0xff .700 0xff .700 0xff 0x00 0xff 0x00 0xff 0x00 0xff 0x00 0xff
#
# Software thresholding enabled to detect a switch master oscillator failure.
# PS1_POWERGOOD set to low threshold at .700
# PS2_POWERGOOD set to low threshold at .700
# In order to enable the thresholding to detect a switch master oscillator
# failure the following line should replace the default thresholds that
# follow the "DO NOT change these values......" line.  Please be sure you
# understand the purpose of these thresholds by reading the documentation
# concerning the switch admin daemon (cssadm).
# Set to 0x00 for defect 47883
 # 0x81 0x00 0xff .700 0xff .700 0xff 0x00 0xff 0x00 0xff 0x00 0xff 0x00 0xff
# +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
# DO NOT change these values without contacting IBM Support.
# +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
  0x81 0x00 0xff 0x00 0xff 0x00 0xff 0x00 0xff 0x00 0xff 0x00 0xff 0x00 0xff
...
```

Simply replace the last line shown with the line that is in grey. The line sets two values to.700 instead of 0x00. To enable the changes you need to restart the hardmon:

- `stopsrc -s hardmon`

- `startsrc -s hardmon`

> **Tip:** css.snap runs automatically. Delete the old *.css.snap.tar files from the /var/adm/SPlogs/css directory of the primary node. Otherwise the snap files get to big.

Table 6-5 shows the General APAR fixes for switch problems.

Table 6-5 General APAR fixes for switch problems

Symptom	Description	APAR fix no
Eclock -c on a 5nsb 0isb system creates a corrupted clock topology file	**ORIGINATING DETAILS:** Eclock -c on a 5nsb 0isb system creates a corrupted clock topology file. If this file is used with Eclock -f most likely no errors will result but when Estart is run: If the Eprimary is in the first frame, only the 1st frame will initialize. If the Eprimary is in any other frame, nothing will initialize. This causes the most serious problem when a customer runs: Eclock -c /etc/SP/Eclock.top.5nsb.0isb.0 Subsequent Eclock -f and Eclock -d will use this invalid file. **LOCAL FIX AS REPORTED BY ORIGINATOR:** If you suspect the default **/etc/SP/Eclock.top.5nsb.0isb.0** has been overwritten, you can verify it by looking in the file for the line: This Eclock topology file was generated by Eclock (-c option) If it has been create by Eclock you can copy a good topology from /usr/lpp/ssp/config. **RESPONDER SUMMARY:** Customers encountered problems running Eclock -a. In one case, Eclock -a was issued using a topology file created by the Eclock -c option, but Eclock -c does not generate alternate settings since they do not reside in the SDR. In another case, the specified alternate topology number did not exist in the topology file. In both cases, Eclock completed but used the wrong jack information. **RESPONDER CONCLUSION:** Changes were made to the Eclock command: - The Eclock -a, -f or -d commands will check to see if the topology file was created by Eclock and issue information message **0028-069**: File <xxxx> was created by the Eclock -c option". Eclock processing continues. - If the Eclock -a command is issued and the topology number specified cannot be found in the topology file, error message 0028-069 will be issued: "Invalid clock topology number <nn> specified." Eclock will exit with an error return code.	IY12801

Symptom	Description	APAR fix no
rc.switch fails with other than LANG=En_US	**ORIGINATING DETAILS:** In the Subject PMR Japan states that rc.switch fails when specify LANG=Ja_JP in /etc/environment. J. Pfau has recreated this problem on c676n05 with German language support. rc.switch is failing when executing /usr/sbin/lsdev and checking for adapter state of"css_ready". Since we are not En_US the css_ready state is not in english. **RESPONDER SUMMARY:** rc.switch fails with messages 0028-202 and **0028-192** if the locale is not set to English (C, en_US, or En_US). The messages in /var/adm/SPlogs/css/rc.switch.log are:0028-202 Device driver configuration failed for adapter css0 **0028-192** rc.switch failed **RESPONDER CONCLUSION:** The failing command is /usr/sbin/lsdev -C -l css0 | grep Available Before running the command, the environment variable **LC_ALL=C** is specified to force the reply in English.	IY14449

Multiple switches problems

When you have a system with multiple switch boards and this system is a SP Switch machine the clock settings need to be set. The most common problem seen on multiple switch systems is a wrong clock setting. This does not apply to the SP Switch2. SP Switch2 does not need to set a clock.

> **Note:** The **Eclock** command is disruptive for switch operations. **Eclock** will interrupt all activities on the switch and applications using the switch will fail. SP Switch2 does not need a clock setting.

The following is a checklist which you can use for solving SP Switch problems:

- Use **SDRGetObjects Switch** to check the clock setting. See Example 6-18.
- Use also **splstdata -s** to verify the clock settings.
- Check for switched off switch boards.
- Use **Eclock -d** and then **Estart** to re initialize clock/switch.
- Perform a Power off/on of the switch board when you encounter many uninitialized on Board connections in the out.top file.

Example 6-18 SDRGetObjects Switch

```
switch_number frame_number slot_number  switch_partition_number switch_type
clock_input  switch_level switch_name  clock_source clo
ck_change switch_plane switch_plane_seq
          1          1         17           1         129 0
1 SP_Switch               0 no           ""
           ""
          2          2         17           1         129 3
1 SP_Switch               1 no           ""
           ""
          3          3         17           1         129 3
1 SP_Switch               1 no           ""
           ""
          4          4         17           1         129 3
1 SP_Switch               1 no           ""
           ""
          5          5         17           1         129 4
1 SP_Switch               2 no           ""
```

6.7 SP Switch2 problems

The SP Switch2 has several improved self recovery mechanisms and diagnostic routines (emasterd, cssadm). But there are a few things that can happen. From a hardware point of view there is to mention the interposer card that is used for each cable connection to the switch board. This switch interposer card is hot swappable and has to be ordered for each new connection used on SP Switch2. This can be the first problem. Like cable misplugging failures in the past on all nodes also a not properly seated interposer card can cause problems.

Table 6-6 shows a sample APAR fix for an SP Switch2 problem.

Table 6-6 Sample APAR fix for an SP Switch2 problem

Symptom	Description	APAR fix no
emasterd takes a long time (up to 20 minutes) to establish the emaster on a large system	**RESPONDER SUMMARY:** On large SP Switch2 systems, the assignment of a new MSS node can take several minutes; this is too long a time for the system to be deprived of a synchronized switch clock. **RESPONDER CONCLUSION:** Changes were made to emastered to speed-up MSS failover on large systems.	IY18326

Symptom	Description	APAR fix no
LED 0765 hang at boot configuring SP Switch2 adapters	**ORIGINATING DETAILS:** Running AIX 4.3.3, recommended maintenance level 5 or later (bos.mp or bos.up 4.3.3.25 or higher), on Power3 SMP High Nodes (Nighthawk Nodes) attached to the SP-Switch2 may hang at LED 0765 during boot. **ADDITIONAL DATA:** LED765 765 hung ml5 ml6 IY12051 - AIX 4330-05 RECOMMENDED MAINTENANCE LEVEL **LOCAL FIX AS REPORTED BY ORIGINATOR:** If you encounter this problem you will need to boot the node in maintenance mode then: 1) mv /usr/lpp/ssp/css/cfgcol /usr/lpp/ssp/css/cfgcol.bak 2) boot the node from disk 3) After boot run: - /usr/lpp/ssp/css/cfgcol.bak -f -v -l css0 - /usr/lpp/ssp/css/rc.switch - Unfence the node Contact IBM Service for permanent circumvention. **RESPONDER SUMMARY:** Node boot hangs at LED 765. Further, the node is unresponsive to reset requests. **RESPONDER CONCLUSION:** We discovered that we cannot use the fp_open AIX service while the adapter is mapped using a BAT register. To solve the problem, we reorganized our code to avoid the above situation.	IY15381

6.8 9077 GRF overview and problems

The Lucent 9077 GRF SP Switch router is a router that can be attached to the SP Switch through a Switch adapter. The GRF is able to provide a very fast routing from different networks (ATM, FDDI, GIGABIT Ethernet and more) through the Switch adapter to the SP Switch. The dependent node (that is how the GRF is named on the SP) is only able to connect to the SP Switch and there is no SP Switch2 attachment possible. There are also some restrictions, differences between a standard SP node and a dependent node:

- The fault service daemon runs on all switch nodes in the RS/6000 SP, but not on the dependent node. Therefore, the dependent node does not have the full functionality of a normal RS/6000 SP Switch node.
- The dependent node requires the SP Switch’s primary node to compute its switch routes. Therefore, the primary node must have at

least PSSP 2.3 installed, otherwise the dependent node cannot work with the RS/6000 SP.

- In the RS/6000 SP, SP Switch nodes occasionally send service packets from one node to the next to keep track of status and links. Sometimes these packets are sent indirectly through another switch node. As the dependent node is not a standard RS/6000 SP Switch node, it cannot be used to forward service packets to other nodes.

Pease refer to the redbook *IBM 9077 SP Switch Router: Get Connected to the SP Switch*, SG24-5157, for more information.

CWS does not communicate with GRF through SNMP

In this case the cause of the problem was perhaps simple, but the result was not so pleasant. Please see Table 6-7 for the sample case study.

Table 6-7 GRF problem

Problem description
GRF configuration is correct,(de0 interface in GRF, and SDR in CWS), but snmp is not able to send the IOSTB3 configuration to the GRF. ► "/var/adm/SPlogs/spmgr/spmgr.log" file shows: EXCEPTIONS: Dependent node 08 managed by the SNMP agent host SPROUTER is not configured in the SDR ► "SDRGetObjects DependentNode" command gives: node_number switch_node_number switch_chip_port switch_chip 31 15 3 8 reliable_hostname management_agent_hostname extension_node_id sprouter.mop.ibm.com sprouter.mop.ibm.com 08 ► de0 interface was defined as follow in CWS in /etc/hosts file "192.168.253.200 **SPROUTER** SPROUTER.mop.ibm.com" **This was the root cause of the problem**
Solution
The right definition must be: "192.168.253.200 SPROUTER.mop.ibm.com **SPROUTER**" So the answer to the following command: "host 192.168.253.200" is "SPROUTER.mop.ibm.com is 192.168.253.200, Aliases: SPROUTER" then the primary name will fit the hostname defined in SDR.

A

SP Logs

The SP System uses a variety of logs. Some of them reside on the MACN (CWS), and some reside only on the SP nodes. Others reside on both.

Table A-1 on page 270 summarizes and show the location of the SP specific logs to use when diagnosing SP problems. The abbreviation CWS, stands for control workstation is used to represent both CWS and the management and control node (MACN).

Your IBM Support Center representative may ask you to provide information from these logs.

Table A-1 SP logs use when diagnosing problems

Type of Message	Log File Name	Location
Output of the phoenix.snap tool. See phoenix.snap Dump.	/tmp/phoenix.Snap/all.timestamp.tar.Z	CWS, nodes
Output of the **SDR_test** command when run without root authority	/tmp/SDR_test.log	CWS, nodes
Standard AIX error log entries including the SP Switch	/var/adm/ras/errlog	Nodes
Error messages and verbose messages from security programs	/var/adm/SPlogs/auth_install/log	CWS, nodes
Automounter messages	/var/adm/SPlogs/auto/auto.log	CWS, nodes
Messages from the **cstartup** command	/var/adm/SPlogs/cs/cstart.timestamp.pid	CWS
Messages from the **cshutdown** command	/var/adm/SPlogs/cs/cshut.timestamp.pid	CWS
Details of recovery actions for the IBM RVSD function	/var/adm/SPlogs/csd/vsd.debuglog	Nodes
Summary of recovery actions for the IBM RVSD function	/var/adm/SPlogs/csd/vsd.log	Nodes
Trace output of pssp_script, which performs the post install customization	/var/adm/SPlogs/css/$nim_client_shr.config.log	Nodes
Switch cable miswire information	/var/adm/SPlogs/css/cable_miswire	Primary node
SP Switch2 adapter diagnostics messages	/var/adm/SPlogs/css0/colad.trace (See SP Switch2 Log and Temporary File Hierarchy)	nodes
Messages from the css.snap script	/var/adm/SPlogs/css/css.snap.log	CWS, nodes
Switch admin daemon messages	/var/adm/SPlogs/css/cssadm.debug	CWS
Switch admin daemon messages (stdout)	/var/adm/SPlogs/css/cssadm.stdout	CWS
Switch admin daemon messages (stderr)	/var/adm/SPlogs/css/cssadm.stderr	CWS

Type of Message	Log File Name	Location
Fault service daemon messages for the SP Switch2	/var/adm/SPlogs/css/daemon.log	Nodes
Fault service daemon messages (stderr) for the SP Switch	/var/adm/SPlogs/css/daemon.stderr	Nodes
Fault service daemon messages (stdout) for the SP Switch	/var/adm/SPlogs/css/daemon.stdout	Nodes
System error messages that occurred while distributing the topology file to the nodes.	/var/adm/SPlogs/css/dist_topology.log	Primary node
Trace of SP Switch adapter diagnostics failures	/var/adm/SPlogs/css/dtbx_failed.trace	Nodes
SP Switch adapter diagnostics trace information	/var/adm/SPlogs/css/dtbx.trace	Nodes
Messages from adapter diagnostics (stderr)	/var/adm/SPlogs/css/dtbxworm.stderr	Nodes
Log from the **Eclock** command	/var/adm/SPlogs/css/Eclock.log	CWS
Log from all Ecommands issued	/var/adm/SPlogs/css/Ecommands.log	CWS
Log from the **Emonitor** command	/var/adm/SPlogs/css/Emonitor.log	CWS
Result of **Estart** commands issued by the Emonitor daemon	/var/adm/SPlogs/css/Emonitor.Estart.log	CWS
Trace of last Eunpartition operation	/var/adm/SPlogs/css/Eunpart.file	Primary node
Trace file of fault service daemon messages	/var/adm/SPlogs/css/fs_daemon_print.file	Nodes
Switch fault information	/var/adm/SPlogs/css/flt	Nodes
stdout and stderr of the Event Management Resource Monitor and Methods	/var/adm/SPlogs/css/logevnt.out	CWS
SP Switch and SP Switch2 advanced diagnostics Messages Daemon log	/var/adm/SPlogs/css/msdg.log	CWS

Type of Message	Log File Name	Location
Description of problems that arise while switch is initializing	/var/adm/SPlogs/css/out.top	Primary node
Initialization messages from the SP Switch support code	/var/adm/SPlogs/css/rc.switch.log	Nodes
Log from switch router generation	/var/adm/SPlogs/css/router.log	Nodes
Output log from switch router generation when it detects a failure	/var/adm/SPlogs/css/router_failed.log	Nodes
SP Switch advanced diagnostics tests and architecture components log	/var/adm/SPlogs/css/spd.trace	CWS, nodes
SP Switch2 advanced diagnostics tests and architecture components log	/var/adm/SPlogs/cssX/pY/spd.trace, where X and Y are variable. See SP Switch2 Log and Temporary File Hierarchy.	CWS, nodes
SP Switch advanced diagnostics GUI log	/var/adm/SPlogs/css/spd_gui.log	CWS
Summary records for events logged to AIX error log on nodes.	/var/adm/SPlogs/css/summlog	CWS
stdout and stderr for the CSS logging daemon's Event Management client	/var/adm/SPlogs/css/summlog.out	CWS
Current state of the switch network, details about attached nodes and the topology file	/var/adm/SPlogs/css/topology.data	Primary node
Worm trace file from switch initialization	/var/adm/SPlogs/css/worm.trace	Primary node
SP Switch2 log files. See SP Switch2 Log and Temporary File Hierarchy.	/var/adm/SPlogs/css0/*	Primary node
SP Switch2 log files. See SP Switch2 Log and Temporary File Hierarchy.	/var/adm/SPlogs/css1/*	Primary node
Results of the last CSS verification test	/var/adm/SPlogs/CSS_test.log	CWS

Type of Message	Log File Name	Location
Output of the **supper** command	/var/adm/SPlogs/filec/supdate.time	Nodes
Actions supper performs when updating file collections	/var/adm/SPlogs/filec/supdate.timer	Nodes
Messages generated by the last **get_keyfiles** command issued	/var/adm/SPlogs/get_keyfiles/get_keyfiles.log	Nodes
Kerberos V4 authentication database administration daemon	/var/adm/SPlogs/kerberos/admin_server.syslog	Primary authentication server
Kerberos V4 primary authentication server log	/var/adm/SPlogs/kerberos/kerberos.log	Primary authentication server
Kerberos V4 secondary authentication server log	/var/adm/SPlogs/kerberos/kerberos.slave_log	Secondary authentication server
Kerberos V4 authentication database propagation daemon	/var/adm/SPlogs/kerberos/kpropd.log	Secondary authentication server
Messages generated by transfer of srvtab files to nodes	/var/adm/SPlogs/kfserver/kfserver.log.PID	CWS
Messages generated by registration process for the kfserver program	/var/adm/SPlogs/kfserver/regserver.log	CWS
Messages generated by the Problem Management daemon	/var/adm/SPlogs/pman/pmand.log	Nodes
Messages generated by the Problem Management daemon	/var/adm/SPlogs/pman/pmand.partition name.log	CWS
System Data Repository configuration messages	/var/adm/SPlogs/sdr/SDR_config.log	CWS
Output of the **SDR_test** command when run with root authority	/var/adm/SPlogs/SDR_test.log	CWS, nodes
System Data Repository error messages	/var/adm/SPlogs/sdr/sdrdlog.partition.pid	CWS
Login control messages	/var/adm/SPlogs/spacs/spacs.log	Nodes

Type of Message	Log File Name	Location
SP configuration Vital Product Data directory	/var/adm/SPlogs/SPconfig/*	CWS, nodes
Vital Product Data output for the node	/var/adm/SPlogs/SPconfig/node number.umcl	CWS, nodes
`lscfg -v` command output	/var/adm/SPlogs/SPconfig/node number.lscfg	CWS, nodes
Messages generated by system daemons, including hardware errors	/var/adm/SPlogs/SPdaemon.log	CWS, nodes
SP extension node messages	/var/adm/SPlogs/spmgr/spmgrd.log	CWS
Hardware Monitor initialization and error messages	/var/adm/SPlogs/spmon/hmlogfile. julian_date	CWS
Node conditioning messages	/var/adm/SPlogs/spmon/nc/nc.frame.node	CWS
Netfinity daemon (nfd) messages	/var/adm/SPlogs/spmon/nfd/nfd.frame.log.julian-date	CWS
Hardware Monitor s70d daemon error messages	/var/adm/SPlogs/spmon/s70d/s70d.frame.log.julian_d ate	CWS
Activity of the logging daemon (splogd).	/var/adm/SPlogs/spmon/splogd.debug	CWS
Contains the PID of the logging daemon, (splogd).	/var/adm/SPlogs/spmon/splogd/splogd.pid	CWS
SP Logging Daemon state changes	/var/adm/SPlogs/spmon/splogd.state_changes. timestamp Note: this log is not shipped activated. To activate, see Configuration Test 1 - Check for State Logging.	CWS
Microcode download messages recorded when using smit (smitty supervisor command)	/var/adm/SPlogs/spmon/ucode/ucode_log.frame.node	CWS
Output of the **`spmon_ctest`** command	/var/adm/SPlogs/spmon_ctest.log	CWS
Output of the **`spmon_itest`** command	/var/adm/SPlogs/spmon_itest.log	CWS
Job Switch Resource Table Services information and error messages	/var/adm/SPlogs/st/st_log	Nodes
Sysctl server log messages	/var/adm/SPlogs/sysctl/sysctld.log	CWS, nodes

Type of Message	Log File Name	Location
AIX error messages from mirroring a root volume group using SP volume group commands	/var/adm/SPlogs/sysman/mirror.out	Nodes
System Management configuration messages	/var/adm/SPlogs/sysman/node.config.log.pid	Nodes
System Management first boot configuration messages	/var/adm/SPlogs/sysman/node.configfb.log.pid	Nodes
System Management console messages	/var/adm/SPlogs/sysman/node.console.log	CWS, nodes
System Management configuration messages	/var/adm/SPlogs/sysman/spfbcheck.log	Nodes
AIX error messages from unmirroring a root volume group using SP volume group commands	/var/adm/SPlogs/sysman/unmirror.out	Nodes
Informational and error messages from the SYSMAN_test command	/var/adm/SPlogs/SYSMAN_test.log	CWS, nodes
hags internal trace and log file	/var/ha/log/hags*	CWS, nodes
hagsglsm internal trace and log file	/var/ha/log/hagsglsm*	CWS, nodes
Event Management activity log	/var/ha/log/em.default.partition-name	CWS, nodes
Trace information for the topology services daemon	/var/ha/log/hats.dd.hhmmss.partition-name	CWS, nodes
Information from the topology services startup script	/var/ha/log/hats.partition-name	CWS, nodes
Hardmon resource monitor messages	/var/ha/run/haem.hostname/IBM.PSSP.hmrmd/ IBM.PSSP.hmrmd_log.julian-date	CWS
filec_config command log, no longer in use	/var/sysman/logs/*	Nodes
SP SNMP Agent messages	/var/tmp/SPlogs/spmgr/spgrd.log	CWS, nodes

Type of Message	Log File Name	Location
SP TaskGuide activity messages (exit values, stdout and stderr from commands run by SP TaskGuides	TaskGuide_name.timestamp.tglog View these logs through the SP TaskGuides themselves.	SDR system files

B

Migration

In this appendix, we discuss migration issues related to PSSP and AIX code on the CWS and nodes.

Migration aspects in the Universal Cluster

This appendix does not cover in detail the migration steps for AIX or PSSP. For a complete list of steps and supported migration paths, consult the *PSSP Installation and Migration Guide*, GA22-7347-02 and Chapter 5.2 *RS/6000 SP Cluster: The Path to Universal Clustering*, SG245374. Instead, we provide some guidance in approaching the migration activities.

From a software management perspective, the Universal Cluster can be viewed as a collection of nodes, sharing a common software repository, located on the boot/install servers (BIS). Each BIS is a network installation manager (NIM) master for the nodes it serves. In an environment with more than one BIS, the CWS makes NIM resources (lppsource) available to the other BIS and initiates NIM operations on them.

In a PSSP environment, the migration aspects refer both to AIX and PSSP. Before planning for a migration, check the product documentation and the Read Me First document, for the supported combinations of PSSP and AIX levels.

When dealing with migration topics, the term upgrade is sometimes used to refer a change in the modification level of the software product. An example would be when migrating from AIX 4.3.2 to 4.3.3, or when applying a new Maintenance Level (ML) to AIX (for example 4330-07_AIX_ML). This operation is performed using the smit update_all menu and does not require a network boot. As opposed to the mentioned change of AIX modification level, a release migration operation (like from AIX 4.2.1 to AIX 4.3.3) involves a network boot via NIM (or from installation media, for standalone systems).

The following topics need to be considered when planning for migration:

- AIX level and ML on the CWS. This needs to be correlated with the minimum AIX level required by the PSSP code.
- AIX level and ML of the lppsource. For each AIX release and modification levels, there is an individual lppsource, for example: /spdata/sys1/install/aix433/lppsource, /spdata/sys1/install/aix432/lppsource. At the creation time, the lppsource may include a certain AIX ML. Each lppsource directory has a corresponding NIM lpp_source resource.
- AIX level of the SPOT. The SPOT location is hard-coded in the setup_server script. There is a SPOT for each AIX level, located in the same directory as its corresponding lppsource directory. The SPOT is created by the setup_server script from the lppsource repository. It is a NIM resource and is maintained using NIM commands. The AIX ML of the SPOT is same as of the lppsource from which it was created. If you upgrade the lppsource, you should do the same with the SPOT.

- PSSP level on the CWS. There is a single PSSP level, which also provides support for the nodes with lower PSSP levels.
- PSSP level in the pssplpp repository. There is an individual directory in the /spdata/sys1/install/pssplpp directory for each PSSP level present in the cluster. For example, if you have PSSP 3.1.1 and 3.2 nodes, you would have two directories, PSSP-3.1.1 and PSSP-3.2, each containing the release-specific software. When a node is installed, customized or migrated, the **pssp_script** selects the right pssplpp subdirectory base on the values in the SDR. If the pssplpp repository contain fixes, the fixes are also installed on the nodes.
- AIX levels on nodes.
- PSSP levels of the nodes.

Important: The CWS needs to be at the latest levels of PSSP and AIX used in the cluster.

Usually there is a single version of AIX and PSSP used across the cluster at the time of the initial system installation. The CWS, lppsource, SPOT, and nodes are at the same maintenance level of AIX and the PSSP level is common on CWS and nodes. During the system exploitation phase, when the nodes host production applications, things are no longer so straightforward. If the Universal Cluster is used for server consolidation, chances are that the different applications in use are not certified for the same release or maintenance levels of AIX and PSSP. Under those circumstances, software maintenance becomes a more challenging tasks, because it has to address the following problems:

- Some nodes in the cluster need to run the latest AIX and PSSP levels, thus requiring the CWS to be updated at these levels.
- A production node should be able to be restored at any time with the exact levels of AIX and PSSP used for the initial installation. If during the time elapsed since the initial installation, software is updated in the pssplpp and lppsource repositories, the node will be reinstalled with newer levels of PSSP and AIX. This could lead to application failures due to the different system code.
- A node customization is performed, for example, to change an adapter IP address. During the node customization process, the **pssp_script** picks up the latest version of PSSP in the pssplpp repository, as well as the latest releases for some AIX software, like:
    - perfagent.server or perfagent.tools
    - bos.rte.tty
    - devices.sio.sa.diag

- perl.rte

Each computing site defines its own policy to address the described issues. Here are only a few recommendations:

- Before making changes to production systems, test them on non-production machines. If possible, create a testing environment similar to the one used in production.
- Before you place the PSSP and RSCT updates in the pssplpp repository, keep them in a separate directory and do the updates from the CWS and on a test node. Only apply the fixes, do not commit them.
- Maintain a single standard, minimal install image (mksysb) per release/modification level of software. Example: bos.obj.ssp.432, bos.obj.ssp.433. Use customization scripts to configure the nodes and to install additional software.
- Any AIX corrective service (PTFs) applied to the mksysb image must also be placed in the lppsource directory and the Shared Product Object Tree (SPOT) must be updated.

Debugging migration problems

There are two instances when the node migration is performed with the help of PSSP software:

1. When AIX is migrated from one release to another (e.g., from 4.2.1 to 4.3.3). Usually, in such cases the PSSP level changes too. This task is performed by setting the node response to migrate and uses the NIM to network boot the node and to perform the AIX migration. The PSSP migration is done by the pssp_script, allocated by the NIM to the node as invoked by NIM as the script resource.
2. When only PSSP level is migrated. The node response is set to customize, using the **`spbootins`** command. The PSSP migration is accomplished by the pssp_script. Note that you need to copy the **`pssp_script`** from the CWS to the node and execute it.

See Example B-1 on page 281 for a list of resources allocated for two sample nodes. The node response set with the **`spbootins`** command is:

Node sp3n12	customize
Node sp3n11	migrate

Example: B-1 Allocated NIM resources

```
sp3en0 > lsnim -l sp3n12
sp3n12:
   class          = machines
   type           = standalone
   platform       = rs6k
   netboot_kernel = up
   if1            = spnet_en0 sp3n12 10005AFA0AB5 ent
   cable_type1    = bnc
   Cstate         = ready for a NIM operation
   prev_state     = ready for a NIM operation
   Mstate         = currently running
sp3en0 > lsnim -l sp3n11
sp3n11:
   class          = machines
   type           = standalone
   platform       = rs6k
   netboot_kernel = up
   if1            = spnet_en0 sp3n11 10005AFA147C ent
   cable_type1    = bnc
   Cstate         = BOS installation has been enabled
   prev_state     = ready for a NIM operation
   Mstate         = currently running
   boot           = boot
   bosinst_data   = 11_migrate
   lpp_source     = lppsource_aix433
   nim_script     = nim_script
   script         = psspscript
   spot           = spot_aix433
   control        = master
```

When NIM is involved, debugging the migration process is similar to debugging a node installation. See the Chapter 3, “Installation” on page 89 for details. Debugging the node customization process starts with a look at the **pssp_script** and **psspfb_script** logs, located in the /var/adm/SPlogs/sysman directory.

PSSP level migration debugging example

Scenario: A node is migrated from PSSP 3.1.1 to PSSP 3.2. We follow the procedure described in *PSSP: Installation and Migration Guide*, GA22-7347, starting with step 4. For the sake of simplicity, we assume that AIX is already at the desired level, so we skip steps 1 to 3. When running the **pssp_script** on the node, we notice that the command hangs and the node’s LED shows c69.

The Example B-2 shows the output of the psspfb_script.

Example: B-2   psspfb_script output

```
=====================================================
psspfb_script: = Setting up authentication environment...          =
               =====================================================
+ OIFS=

+ IFS=
+ /etc/methods/showled 0xc69
+ K4FILES=/spdata/sys1/k4srvtabs
+ [[ ! -d /spdata/sys1/k4srvtabs ]]
+ sname=sp3n12
+ typeset -l sname
+ /bin/rm /spdata/sys1/k4srvtabs/sp3n12-new-srvtab
+ 1> /dev/null 2>& 1
+ [[ -f /usr/lpp/ssp/bin/get_keyfiles ]]
+ /usr/lpp/ssp/bin/get_keyfiles sp3n12-new-srvtab 192.168.3.130
Parsing operands
Removing cons (getty) entry from inittab
Finding getty process
Terminating process Name=/usr/sbin/getty        PID=7590
Getting my node number
Sending request for keyfile
```

The **psspfb_script** log file shows that the **get_keyfiles** script is causing the problems. The output of the **get_keyfiles** script is in the /var/adm/SPlogs/get_keyfiles/get_keyfiles.log file and is shown in Example B-3.

Example: B-3   get_keyfiles output

```
************************************************
*   Beginning of logging for -- get_keyfiles
***  Wed Jul 18 13:22:31 EDT 2001
Parsing operands
Removing cons (getty) entry from inittab
Finding getty process
Getting my node number
Sending request for keyfile
Setting up the socket
Opening temporary keyfile for writing
Connecting to port 32801
Opening /dev/tty0
```

We also look on the CWS for the kfserver log file, located in the /var/adm/SPlogs/kfserver directory, and shown in Example B-4. From the server side, it appears that the key files are sent to the node. Actually, the files do not get to the node, and the node keeps waiting for them.

Example: B-4 kfserver output

```
************************************************
*   Beginning of logging for -- kfserver
***  Wed Jul 18 13:22:42 EDT 2001

Recieved request from node 12
Uuencoding keyfile /spdata/sys1/k4srvtabs/sp3n12-new-srvtab
Sending keyfile to /dev/tty0 on sp3n12

************************************************
***  End of logging for -- kfserver
```

A good starting point is to kill the hanging process, in order to allow the **setup_server** to complete as described in the Example B-5 on page 283.

Example: B-5 Finding the get_keyfiles PID

```
sp3n12 > ps -ef |grep key
    root 122082  51244   2 17:03:49  pts/0  0:00 grep key
    root 126232 130574   0 16:44:08  pts/0  0:00 perl
/usr/lpp/ssp/bin/get_keyfiles sp3n12-new-srvtab 192.168.3.130
sp3n12 > kill 126232
```

Next we run the **get_keyfiles** command manually, but the process hangs again. We then run it in debug mode (**perl -d**). We find that the script hangs while executing the *kfcli* subroutine, which creates a socket connection to the CWS and waits to receive the srvtab file. We interogate the sockets on the node and CWS and get the output shown in Example B-6 on page 283.

Example: B-6 netstat output

```
sp3n12 > netstat -an |grep 32801
tcp4      79      0  192.168.3.12.38345     192.168.3.130.32801    CLOSE_WAIT
sp3en0 > netstat -an |grep 32801
tcp4       0      0  192.168.3.130.32801    192.168.3.12.38345     FIN_WAIT_2
tcp4       0      0  *.32801                *.*                    LISTEN
```

We also have a look at the kfserver program code, running on the CWS as a inetd subserver. We notice that, for Kerberos 4 authentication, the kfserver uses the **s1term** command to send the srvtab file to the node, as root.SPbgAdm principal. We know that, in order to use **s1term**, a principal needs to be in the ACL of the hardmon service. The existing ACL is shown in Example B-7.

Example: B-7   hardmon ACL

```
sp3en0 > cat hmacls
sp3en0 root.admin a
sp3en0 rcmd.sp3en0 a
sp3en0 hardmon.sp3en0 a
1 root.admin vsm
1 rcmd.sp3en0 vsm
1 hardmon.sp3en0 vsm
sp3en0 root.SPbgAdm a
```

We test the permissions of the root.SPbgAdm principal. We notice the lack of authorization, so we add the appropiate line to the ACL file. Now the **s1term** command works. See Example B-8 for the commands output.

Example: B-8   Adding hardmon ACL authorization

```
sp3en0 > /bin/ksrvtgt root SPbgAdm
sp3en0 > hmgetacls 1:12
  frame1/slot12     -  -  -  -
sp3en0 > s1term -w 1 12
s1term: 0026-614 You do not have authorization to access the Hardware Monitor.
sp3en0 > echo "1 root.SPbgAdm vsm" >> /spdata/sys1/spmon/hmacls
sp3en0 > hmadm setacls
hmadm: 0026-641I The ADMIN command "setacls" was sent.
sp3en0 > hmgetacls 1:12
  frame1/slot12     v  s  m  -
```

We now retry the **get_keyfiles** command on the node and this time it runs well. The problem was with the authorization for the root.SPbgAdm to access the serial line and was caused by the missing entry in the hardmon ACL.

C

Software maintenance strategy

This appendix describes the philosophy of, and the differences between, AIX maintenance levels and PSSP PTF sets. We introduce the following topics:

- The process for obtaining PTFs.
- Find the problems that PTFs address.
- Practical system administration.

Overview

PSSP and AIX packaging follows the convention of dividing the system software into *filesets*, each of which can contain a group of logically related files. Each fileset can be separately installed and updated. Changes to fileset levels are tracked by version, release, maintenance, and fix levels, known as VRMF (e.g., ssp.basic 3.2.0.11 - Version 3, Release 2, Maintenance level 0, Fix level 11). Application of filesets will increment a component of VRMF. This maintenance strategy can be summarized as “upward only,” and ensures a fix cannot be created for a lower level that is not already contained in a higher level.

Maintenance levels (sometimes called Modification levels) increment the (M) portion of VRMF. When this is done, the fix level (F) portion is reset to zero for the affected filesets. Maintenance levels contain an accumulation of changes from all previous fix and maintenance levels. Binary compatibility is preserved within a version and release.

Program Temporary Fixes (PTFs) are a fileset update, at a specific VRMF, to correct or prevent a particular problem. They increment the (F) portion of VRMF. This is known as corrective service. Corrective service is the application of one or more PTFs to correct a specific problem.

PTFs apply to all previous maintenance levels of the same version and release. They are cumulative, in other words, if we apply the fileset ssp.basic at fileset level 3.2.0.10, we encompass all fixes made for the ssp.basic fileset at levels 3.2.0.9, 3.2.0.8, and below.

A PTF may exist at a later maintenance level than the customer has on their system. It is part of the design of AIX maintenance that customers can install PTFs from higher maintenance levels. Remember, it is not required that a system be updated to the latest maintenance level just to install a PTF. Prerequisites may exist that ensure that additional fileset updates for a particular PTF are included.

An Authorized Program Analysis Report (APAR) is a description of a problem and it’s resolution. An APAR will consist of one or more PTFs and it will have a status of Open or Closed on the IBM Support databases.

- Open: The problem has been identified and described but the defect code for resolution is still being developed or tested.
- Closed: The defect resolution code is available, there may be a short delay while the code is packaged and shipped to distribution sites.

Why should you bother with PTFs

If your system is stable, performance is good and customers are happy, then there may be no reason to install new filesets. However, as system workloads change, new applications are installed, new users added, or extra nodes installed, circumstances may change. Some threshold or timing tolerance may be crossed that, seemingly, no amount of tuning or administration can rectify. It may be that this is a known problem that is already resolved. Often, problems discovered internally by IBM, are never experienced by customers because the circumstances are so unique. But testing, which does not stop just because a product is shipped, has shown there is potential for the problem to occur.

PTFs are developed either because there has been a problem found or there is extra functionality, such as extra trace facilities or better debug output (among other things). A request for a PTF to be included in a PSSP set or AIX maintenance level can come from IBM support on behalf of customers or from IBM development.

Preventative maintenance is of a benefit to customers and IBM. It reduces the possibility of rediscovering and diagnosing problems that have already been fixed, and reduces the number of fileset updates that may have to be installed for *corrective service*. The larger the size of an update, the greater the amount of time will be required to apply the corrective service update. Thus, more exposure to unforeseen problems with the application may exist.

AIX recommended maintenance levels

A distinction needs to drawn between the maintenance portion of VRMF and *recommended maintenance levels*; the application of a recommended maintenance package does not increment any portion of the VRMF.

Maintenance levels, as described by VRMF, can be determined by the command, **oslevel**, as shown in Example C-1.

Example: C-1 oslevel

```
[8:root@sp3en0:]/ # oslevel
4.3.3.0
```

AIX recommended maintenance packages are a collection of PTFs applied on top of an AIX maintenance level. They have been tested as a unit and have enough exposure to be recommended for release as *preventative maintenance*.

A new flag (-r) has been created for the **oslevel** command and can now be used to show what the complete recommended maintenance level is on a system. The output shows the highest level at which all filesets, comprising a recommended maintenance level, are found as shown in Example C-2.

Example: C-2 oslevel -r

```
[8:root@sp3en0:]/ # oslevel -r
4330-04
```

A more verbose way, to show the same information, and also show all system applied recommended maintenance, is shown in Example C-3.

Example: C-3 instfix -i | grep AIX_ML

```
[8:root@sp3en0:]/ # instfix -i | grep AIX_ML
    All filesets for 4.3.1.0_AIX_ML were found.
    All filesets for 4.3.2.0_AIX_ML were found.
    All filesets for 4.3.1.0_AIX_ML were found.
    All filesets for 4.3.2.0_AIX_ML were found.
    All filesets for 4.3.3.0_AIX_ML were found.
    All filesets for 4330-02_AIX_ML were found.
    All filesets for 4320-02_AIX_ML were found.
    All filesets for 4330-03_AIX_ML were found.
    All filesets for 4330-04_AIX_ML were found.
    Not all filesets for 4330-05_AIX_ML were found.
    All filesets for 4330-01_AIX_ML were found.
    Not all filesets for 4330-06_AIX_ML were found.
    Not all filesets for 4330-07_AIX_ML were found.
    Not all filesets for 4330-08_AIX_ML were found.
```

The **instfix** command can also be used to determine if a specific recommended maintenance package is installed (in this example, 4.3.3.0-08), enter:

```
instfix -ik 4330-08_AIX_ML
```

To determine which filesets need updating for the system to reach the 4.3.3.0-08 level, enter:

```
instfix -ciqk 4330-08_AIX_ML | grep “:-:”
```

AIX recommended maintenance levels may be rolled out approximately every eight to twelve weeks.

PSSP PTF sets

> **Important:** PSSP PTF sets are not the same as AIX recommended maintenance levels.

PSSP PTF sets are collections of fixes. They include the latest level of PSSP filesets as of the build date shown; refer to Table 6-8 for an example.

If your system is stable, you may want to hold the PSSP PTF set level a couple levels behind the most current. As an example, if PSSP PTF set 10 is the latest available, then run your system on PSSP PTF set 8. Of course, this may not be possible, as some later PTFs may be important for stable operations or corrective maintenance.

PSSP PTF sets may be rolled out approximately every six to eight weeks.

Table 6-8 PSSP PTF set 10

PTF	File Set Name	File Set Level
U478551	ssp.basic	3.2.0.10
U478552	ssp.clients	3.2.0.7
U478550	ssp.css	3.2.0.10
U478548	ssp.docs	3.2.0.5
U478549	vsd.cmi	3.2.0.4
U478543	vsd.rvsd.rvsdd	3.2.0.7
U478542	vsd.rvsd.scripts	3.2.0.7
U478546	vsd.vsdd	3.2.0.9
U478558	rsct.basic.rte	1.2.0.10
U478560	ppe.pedb	3.1.0.6
U478547	ppe.poe	3.1.0.10
U478559	LoadL.full	2.2.0.9
U478555	LoadL.msg.En_US	2.2.0.2
U478554	LoadL.msg.en_US	2.2.0.2
U478557	LoadL.so	2.2.0.9
U478553	mmfs.base.cmds	3.2.0.6

PTF	File Set Name	File Set Level
U478545	mmfs.base.rte	3.2.0.9
U478544	mmfs.gpfs.rte	1.3.0.8
U478556	mmfs.msg.en_US	3.2.0.5

Where to get a PTF

Fixes can be ordered by APAR or PTF number. Ordering PTFs on physical media from your support center may incur a fee.

For internet connected customers, the URL listed below is the site for downloading AIX or PSSP fixes without a fee. Here, you can search the database by APAR number, fileset name, PTF number or APAR abstract:

```
http://techsupport.services.ibm.com/rs6k/fixdb.html
```

A good system administration practise is to order the latest AIX recommended maintenance level and PSSP PTF set level and have it available should it be needed.

PTF system administration

Important: Your CWS must be at or above the highest level of any filesets on any of the nodes.

- Always take a current mksysb backup of your CWS and verify it as good. If you have separate volume groups on the CWS do a savevg of these and verify them as well.
- Read the README files shipped with the PTFs. Check for prerequisites or corequisites before applying PTFs.
- Ensure with your application vendor that they have verified support of the applications with these PTF levels.
- Do a preview install before installing and check the output fully for potential issues before actual installation.
- Always apply to the CWS first and test thoroughly before any work on the nodes or attached servers.
- Take another mksysb of the CWS.

- Copy AIX PTFs to the /spdata/sys1/install/<aix_level>/lppsource using the **bffcreate** command. If you use another command, do not forget to run **inutoc** within the directory to rebuild the .toc file.
- Copy PSSP PTFs to the /spdata/sys1/install/pssplpp/<pssp_version> directory and rebuild the .toc file.
- Preview any PTFs updates and then update the SPOT using the /spdata/sys1/install/<aix_level>/lppsource directory. If any maintenance needs to be done to the nodes the new SPOT will be used.
- Take a current mksysb backup of a test node.
- Login to the test node and mount the /spdata/sys1/install/<aix_level>/lppsource directory from the CWS if updating AIX. For PSSP mount the /spdata/sys1/install/pssplpp/<pssp_version>.
- After applying the PTFs take another mksysb backup, storing it on the CWS in the /spdata/sys1/install/images/ directory.
- Thoroughly test the node.
- Store the new mksysb_name of the node into the SDR so the next install of the test node will use the updated image.

Useful URLs

The following list of URLs may provide useful information during problem determination:

- http://techsupport.services.ibm.com/rs6k/sp/status/

  PSSP APAR reports by version detail, the status of reported problems, the contents of PSSP PTF sets, and a description of them.
- http://techsupport.services.ibm.com/rs6000/techKnow

  This will provide information about Open/Closed APARs.
- http://techsupport.services.ibm.com/rs6000/fixes

  Download URL for fixes, maintenance levels, microcode.
- http://www.rs6000.ibm.com/support/sp/sp_secure/readme/

  For the latest README files for the SP subsystems.
- http://techsupport.services.ibm.com/rs6000/notification

  Subscribe to e-mail notifications on product fixes, security alerts, service announcements, and more.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 295.

- *RS/6000 SP Cluster: The Path to Universal Clustering,* SG24-5374
- *SP Perspectives: A New View of Your SP System,* SG24-5180
- *HACMP Enhanced Scalability Handbook, SG24-5328*
- HACMP/ES Customization Examples, SG24-4498
- HACMP Enhanced Scalability: User-Defined Events, SG24-5327
- RSCT Group Services: Programming Cluster Applications, SG24-5523
- Understanding and Using the SP Switch, SG24-5161
- IBM 9077 SP Switch Router: Get Connected to the SP Switch, SG24-5157
- PSSP Version 3 Survival Guide, SG24-5344
- RS/6000 SP Systems Handbook, SG24-5596

Other resources

These publications are also relevant as further information sources:

- *IBM RS/6000 SP: Planning Volume 1, Hardware and Physical Environment,* GA22-7280
- *IBM RS/6000 SP: Planning Volume 2, Control Workstation and Software Environment,* GA22-7281
- *Parallel System Support Programs for AIX: Diagnosis Guide, Version 3 Release 2, GA22-7350*
- Parallel System Support Programs for AIX: Messages Reference, Version 3 Release 2, GA22-7352
- AIX V4.3 Problem Solving Guide and Reference, SC23-4123

- Parallel System Support Programs for AIX: Installation and Migration Guide, GA22-7347
- Parallel System Support Programs for AIX: Administration Guide, SA22-7348
- Parallel System Support Programs for AIX: Command and Technical Reference, Volume 1, SA22-7351
- IBM DCE for AIX, Version 3.1: Administration Guide - Core Components
- Group Services Programming and Reference Guide, SA22-7355
- Event Management Programming Guide and Reference, SA22-7354
- RS/6000 SP System Service Guide, GA22-7442
- RS/6000 SP: SP Switch Service Guide, GA22-7443
- RS/6000 SP: SP Switch2 Service Guide, GA22-7444
- RS/6000 SP Installation and Relocation Guide, GA22-7441

Referenced Web sites

These Web sites are also relevant as further information sources:

- Web site for the latest installation planning guides

  `http://www.rs6000.ibm.com/aix_resource/sp_books/planning/index.html`
- AIX manuals

  `http://www.rs6000.ibm.com/resource/aix_resource/Pubs/index.html`
- PSSP manuals

  `http://www.rs6000.ibm.com/resource/aix_resources/sp_books/index.html`
- `APARs and PTFs`

  `http://techsupport.services.ibm.com/rs6000/fixes`
- Forums

  `http://techsupport.services.ibm.com/rs6000/forums`
- General AIX technical documents

  `http://techsupport.services.ibm.com/rs6000/techKnow`
- Main entry for IBM support page

  `http://techsupport.services.ibm.com/eserver/support`
- RS/6000 SP support page

  `http://www.rs6000.ibm.com/support/sp`

How to get IBM Redbooks

Search for additional Redbooks or Redpieces, view, download, or order hardcopy from the Redbooks Web site:

ibm.com/redbooks

Also download additional materials (code samples or diskette/CD-ROM images) from this Redbooks site.

Redpieces are Redbooks in progress; not all Redbooks become Redpieces and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

IBM Redbooks collections

Redbooks are also available on CD-ROMs. Click the CD-ROMs button on the Redbooks Web site for information about all the CD-ROMs offered, as well as updates and formats.

Special notices

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

The following terms are trademarks of other companies:

Tivoli, Manage. Anything. Anywhere.,The Power To Manage., Anything. Anywhere.,TME, NetView, Cross-Site, Tivoli Ready, Tivoli Certified, Planet Tivoli, and Tivoli Enterprise are trademarks or registered trademarks of Tivoli Systems Inc., an IBM company, in the United States, other countries, or both. In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli A/S.

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

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

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

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

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

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

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

Abbreviations and acronyms

API	Application Programming Interface
CWS	Control Work Station
DMA	Direct Memory Access
FSD	Fault Service Daemon
GRF	Goes Really Fast
IBM	International Business Machines Corporation
ITSO	International Technical Support Organization
LPP	Licensed Program Product
MCA	Micro Channel Architecture
PCI	Peripheral Component Interface
PID	Process Identifier
PTF	Program Temporary Fixes
rcp	Remote copy
rsh	Remote shell
SDR	System Data Repository
SPOT	Shared Product Object Tree
SPS	SP Switch
SRC	System Resource Controller
TOD	Time of Day

Index

IBM

Redbooks

Universal Clustering Problem Determination Guide

(0.1”spine)
0.1”<->0.169”
53<->89 pages

Universal Clustering Problem Determination Guide

Explains how to identify the components involved in a problem

Offers ideas about how to approach universal cluster problems

Chapters are self-cointained for independent use

Problem determination and problem solving on the RS/6000 SP and clusters can be difficult because the malfunction may imply the movement of several components within AIX and PSSP. Problem determination on the RS/6000 universal cluster is part of the daily tasks of a system administrator. Although the RS/6000 universal cluster is a very stable platform, the manner in which distributed and parallel environments are intermixed may cause certain problems that cannot be easily solved by someone with only AIX experience.

This redbook gives a comprehensive explanation of certain RS/6000 SP and cluster components and provides the reader with tools and procedures that can be used for problem isolation and problem solving in a cluster environment. This redbook is oriented to RS/6000 SP and cluster professionals who install, configure, and administer universal cluster systems. Several procedures are outlined and tested, along with the explanation for the causes of common SP and cluster problems.

This universal clustering guide is the perfect companion for the RS/6000 SP and clusters product manuals when you need to identify and solve system problems.

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

SG24-6602-00 ISBN 0738423394