IBM Management Processor

Command-Line Interface Version 2.0 User's Guide

SC90-P290-00

Note: Before using this information and the product it supports, read the general information in Appendix C, “Notices”, on page 73.

First Edition (July 2003)

Contents

Tables

Preface

This book provides information about installing and using IBM® Management Processor Command-Line Interface, version 2.0 with the following operating systems:

- Microsoft® Windows® 2000 Server
- Microsoft Windows 2000 Professional
- Red Hat Linux 7.3
- SuSE Linux 8.0

This book is organized into the following chapters:

Chapter 1, “Introducing IBM Management Processor Command-Line Interface”, on page 1 provides an overview of the product.

Chapter 2, “Installing IBM Management Processor Command-Line Interface”, on page 3 contains installation considerations and instructions.

Chapter 3, “Using IBM Management Processor Command-Line Interface”, on page 9 contains usage guidelines and details of each command.

Appendix A, “Sample scripts”, on page 67 contains sample command scripts.

Appendix B, “Getting help and technical assistance”, on page 71 contains information about how to get technical help, support, and service.

Appendix C, “Notices”, on page 73 contains IBM notices and trademark information.

Chapter 1. Introducing IBM Management Processor Command-Line Interface

IBM Management Processor Command-Line Interface, version 2.0 is a management tool for IBM systems running Linux or Microsoft Windows. The system management functions are provided from an easy-to-use command-line interface that connects to a service processor. Using this command-line interface, you can access and set a wide range of information about the health, configuration, communication, and state of your system. These functions are immediately available after you install the command-line interface and make a connection to a service processor.

There are three supported methods that you can use to communicate with a service processor:

- In-band communication using a device driver
- Out-of-band communication using an IP connection
- Out-of-band communication using an RS-485 interconnect

Not all configurations of server and service processor support all functions. See “Supported service processor configurations” on page 3 for more details.

After you have established your communication pathway, you can begin using the following functions provided by the command-line interface:

- Query vital product data (VPD) of system components:
  - BladeCenter™ units, including switch modules
  - Cache coherency
  - Chassis/enclosure
  - Microprocessor and microprocessor electrically erasable programmable read-only memory (EEPROM)
  - Diagnostics
  - Hard disk drive backplane and system board
  - Dual inline memory module (DIMM)
  - Power supply and backplane
  - Service processor device driver, firmware, and hardware revision
  - System slot
  - POST/BIOS
- Query the component light-emitting diodes (LEDs) using light path diagnostics:
  - BladeCenter units
  - CD-ROM drive and hard disk drive cables
  - Centerplane, memory, and adapters
  - Chip and chip set voltage regulator module (VRM)
  - Microprocessor and microprocessor memory
  - Diagnostics and front panel
  - Expansion device, fan, scalability ports, system board, and VRM
  - Memory, memory hot-plug enable, and memory subsystem
  - Peripheral component interconnect (PCI), PCI bus, and PCI adapters
  - Service processor adapter and slot
- Query and set service processor configurations:

  - Restart
  - Network interfaces: Simple Network Management Protocol (SNMP), point-to-point protocol (PPP), and Internet protocol (IP)
  - Serial port hardware and software
  - General identification
  - Vital product data (VPD)
  - Internal clock
  - Alert dial-out settings
  - Dial-in entries
- Query the service processor for entries in its event log recorded by:
  - BIOS
  - Diagnostics
  - POST
  - Service processor
- Query or set server time-outs for your systems:
  - Operating-system loader
  - Operating-system heartbeat
  - POST
  - Power-off delay
- View health and environment information for your systems:
  - System board voltages and thresholds
  - VRM voltages
  - Power-supply voltages
  - Component temperatures and thresholds
  - Fan speeds
- Turn systems on and off remotely:
  - Turn off immediately and with operating-system shutdown.
  - Restart immediately and with operating-system shutdown.
  - Turn on immediately, after a specified delay, and in *n* seconds.
- Query the general state information for your systems:
  - System state stable or unstable
  - System power on or off
  - Number of times restarted
  - Number of hours turned on
  - Universal unique ID (UUID)
  - Blue indicator light on or off
  - Light path LED functioning
- Create scripts that contain multiple commands for you to use and reuse. When you run a script, each command within it is run.

Chapter 2. Installing IBM Management Processor Command-Line Interface

This chapter contains information about limitations, supported service processor configurations, and installation for IBM Management Processor Command-Line Interface.

Limitations

The command-line interface has the following limitations:

- In Linux, you cannot change the default location of the installation from /opt/IBMmpcli/.
- In Linux, you cannot use the Up Arrow and Down Arrow keys to recall a command after you start the command-line interface.

Supported service processor configurations

The command-line interface is supported only on systems with at least one of the following service processors:

- Advanced System Management (ASM) processor
- ASM PCI Adapter
- Integrated system management processor (ISMP)
- Management module
- Remote Supervisor Adapter
- Remote Supervisor Adapter II

You must also ensure that the combination of service processor and system type is supported. Additionally, there are different levels of support, depending on your service processor configuration:

- Full
  Supports any function available in the server relating to the system-management hardware. The command-line interface will be updated as needed to support any new or unique server functions.
- Compatible
  Supports current functions of the command-line interface. There are no plans to support new functions added to the server or command-line interface.
- Service processor configuration
  Supports service processor configuration only, because the service processor cannot access the system hardware.

The following tables list the supported configurations of servers and service processors. If your server is not in the list, you can assume that it does not support the command-line interface.

Table 1. Server configurations with full support

Server name	Service processor
BladeCenter	Management module
HS20	Management module
xSeries™ 335	ASM processor
xSeries 360	Remote Supervisor Adapter

Table 1. Server configurations with full support (continued)

Server name	Service processor
xSeries 440	Remote Supervisor Adapter

Table 2. Server configurations with compatible support

Server name	Service processor
Netfinity® 3000	ASM processor
Netfinity 3600	Remote Supervisor Adapter
Netfinity 4100R	ASM processor
	Remote Supervisor Adapter
Netfinity 4500R	ASM processor
Netfinity 5000	ASM processor
Netfinity 5100	ASM processor
Netfinity 5600	ASM processor
Netfinity 5500	ASM processor
Netfinity 5500 M10	ASM processor
Netfinity 5500 M20	ASM processor
Netfinity 6000R	ASM processor
Netfinity 7000 M10	ASM PCI adapter
Netfinity 7100	ASM processor
Netfinity 7600	ASM processor
Netfinity 8500	ASM PCI adapter
xSeries 220	Remote Supervisor Adapter
xSeries 225	Remote Supervisor Adapter
xSeries 230	ASM processor
xSeries 232	ISMP
	Remote Supervisor Adapter
xSeries 235	ISMP
	Remote Supervisor Adapter II
xSeries 240	ASM processor
xSeries 242	ASM processor
xSeries 250	ASM processor
xSeries 255	ISMP
	Remote Supervisor Adapter
	Remote Supervisor Adapter II
xSeries 330	ASM PCI adapter
	Remote Supervisor Adapter
xSeries 335	ISMP
xSeries 340	ASM processor
xSeries 342	ISMP
	Remote Supervisor Adapter

Table 2. Server configurations with compatible support (continued)

Server name	Service processor
xSeries 345	ISMP
	Remote Supervisor Adapter II
xSeries 350	ASM processor
xSeries 370	ASM PCI adapter
xSeries 450	Remote Supervisor Adapter

Table 3. Server configurations with service processor configuration support

Server name	Service processor
Netfinity 4100R	ASM PCI adapter
Netfinity 4500R	ASM PCI adapter
Netfinity 5000	ASM PCI adapter
Netfinity 5100	ASM PCI adapter
Netfinity 5500 M10	ASM PCI adapter
Netfinity 5500 M20	ASM PCI adapter
Netfinity 5600	ASM PCI adapter
Netfinity 6000R	ASM PCI adapter
Netfinity 7100	ASM PCI adapter
Netfinity 7600	ASM PCI adapter
xSeries 230	ASM PCI adapter
xSeries 240	ASM PCI adapter
xSeries 250	ASM PCI adapter
xSeries 305	Remote Supervisor Adapter
xSeries 330	Remote Supervisor Adapter
xSeries 335	Remote Supervisor Adapter
	Remote Supervisor Adapter II
xSeries 340	ASM PCI adapter
xSeries 350	ASM PCI adapter

There is also limited support for the IBM RXE-100 Remote Expansion Enclosure that enables you to add other devices. See the *RXE-100 Hardware Maintenance Manual* for more information.

Installing the command-line interface

This section provides instructions for installing the command-line interface on the following operating systems:

- Red Hat Linux 7.3
- SuSE Linux 8.0
- Windows 2000 Server
- Windows 2000 Professional

Installing on a Linux operating system

Before you begin the installation, note the following information:

- There is no upgrade path from IBM Management Processor Command-Line Interface for Linux, version 1.x, to IBM Management Processor Command-Line Interface, version 2.0. You must uninstall any previous installation by typing the following command at a shell prompt:

  ```
  rpm -e mpcli
  ```

- If you are not the root user or a member of the root user group, you might not be able to install or uninstall the command-line interface.

Complete the following steps to install the command-line interface:

1. Download the Linux installation file, mpcli-2.0-1.0.i386.rpm, from http://www-3.ibm.com/pc/support/site.wss/document.do?lndocid=MIGR-52058 to a temporary directory.
2. From a shell prompt, type the following command:

   ```
   rpm -ivh mpcli-2.0-1.0.i386.rpm
   ```

   The installation process begins.
3. (Optional) Delete the mpcli-2.0-1.0.i386.rpm file to free up disk space.

After the installation is completed, begin using the command-line interface by typing the following command at a shell prompt:

```
/opt/IBMmpcli/bin/MPCLI.bsh
```

Installing on a Windows operating system

You can install the command-line interface using the InstallShield wizard, or you can perform an unattended installation.

After the installation is completed, the command-line interface is available on the Windows **Start** menu. Click **Start → Programs → IBM → MPCLI** to start the program.

InstallShield wizard installation

Complete the following steps to install the command-line interface:

1. Download the Windows installation file, 90p4977.exe, from http://www-3.ibm.com/pc/support/site.wss/document.do?lndocid=MIGR-52058 to a temporary directory.
2. Double-click the **90p4977.exe** file. The “Welcome to InstallShield Wizard” window opens.
3. Click **Next**. The “License Agreement” window opens.
4. Click **I accept the terms of this license agreement**, and then click **Next**. The “Customer Information” window opens.
5. Enter the following information:
   a. In the **User Name** field, type the user ID you are logged on as.
   b. In the **Organization** field, type the name of your company or organization.
6. From the **Who can use this product** field, click **Anyone** to allow all users to use the command-line interface. Otherwise, click **Only this user**.
7. Click **Next**. The “Installation type window” opens.
8. Click **Complete** to install the command-line interface in the default location. Otherwise, click **Custom** to select a different installation directory.
9. Click **Next**. The “Ready to install” window opens.

10. Click **Install**. The status bar displays the installation progress. When the installation is completed, the “InstallShield Wizard Completed” window opens.
11. Click **Finish**.

Unattended (silent) installation

Complete the following steps to install the command-line interface:

1. Download the Windows installation file, 90p4977.exe, from http://www-3.ibm.com/pc/support/site.wss/document.do?lndocid=MIGR-52058 to a temporary directory.
2. From a command window, type the following command:

   ```
   d:\tempdir\90p4977.exe /S /v/qn
   ```

   where *d*:*tempdir* is the directory where you downloaded the installation file. In an unattended installation, you do not receive any messages notifying you of the installation status or of its completion.

Chapter 3. Using IBM Management Processor Command-Line Interface

With IBM Management Processor Command-Line Interface you can manage and monitor system health and configuration by logging on to a service processor on a system or connected to a system, query for information about system status, or set parameters for system behavior.

This chapter contains the following information about using the command-line interface:

In the **Examples** following the tables, the items in **bold** type are examples only. The parameters that you choose to use will be different.

Command-line interface guidelines

All commands have the following basic structure:

```
command -option parameter
```

You can add multiple options to a command on one line to avoid repeating the same command, for example:

```
command -option1 parameter -option2 parameter -option3 parameter
```

The information for each option is returned in the order in which it was entered and displayed on separate lines.

Observe the following general guidelines when using the command-line interface:

- Case sensitivity
  - All commands and command options are in lowercase and are case sensitive.

    Note: If you receive a `Command not found` error make sure that you are typing the commands in the correct case; they are case sensitive. For a list of valid commands, type `help-cmd`.
  - Boolean and string parameters are not case-sensitive, so you can use lowercase, uppercase, or mixed case. For example, a Boolean parameter set to true could be entered as `TRUE`, `True`, `true`, or any other case variation.
- Data types
  The following data types are predefined:
  - boolean - A string with the value of either true or false.
  - ipaddress - A formatted string of *xxx.xxx.xxx.xxx*, where *xxx* is a number from 0 to 255.
- Delimiters
  - Options are delimited with a minus sign.
  - In a command that requires parameters, a single space is expected between the option and the parameter. Any additional spaces are ignored.

- Multiple parameters
  - Multiple parameters must be separated by commas.
  - Comma-separated parameters may be separated by additional spaces.
- Output format
  - Failed commands generate failure messages consisting of a single line of text in the format `FAILURE: failure_message`.
  - Successful commands are indicated by the message `SUCCESS: command_results`.
- Strings
  - Strings containing spaces must be enclosed in quotation marks, such as in `snmp -contactname "John B. Doe"`; in a string with spaces and no quotation marks, only the first word is recognized.
  - Unless otherwise specified, the maximum string length is 15 characters, including spaces. String values over 15 characters long are truncated.
  - String parameters can be mixed case.

Known problems

The following problems of the command-line interface are known:

- If you are logged on to a remote service processor using an RS-485 interconnect, your connection might be lost after a period of inactivity.
- When using the setmpclock -gmtoffset command, if you attempt to use an invalid parameter, outside the range of -12 to +12, a message indicating that the command was successful might be displayed, even though the command was unsuccessful. To avoid this, enter a valid parameter.
- If you are logged on to a Remote Supervisor Adapter and running the command-line interface, and then fail at logging on to an integrated system management processor on an RS-485 interconnect, the command-line interface might lose all functionality. You must restart the command-line interface and log on again.
- In using the setsmnetwork command, if you make changes using any of the options, they remain in a pending state even if the setsmnetwork -enable command is set to true, which should apply the pending changes. Instead, typing `setsmnetwork -enable true` might return a message indicating that there was a problem sending the command.

Starting the command-line interface

To start the command-line interface in Linux, type the following command at a shell prompt:

```
/opt/IBMmpcli/bin/MPCLI.bsh
```

Before you start the command-line interface, make sure that you are either the root user or a member of the root user group. If an error is returned when you start the command-line interface, the script file might not have execute permissions. To add execute permissions, type `chmod +x MPCLI.bsh` at a shell prompt.

To start the command-line interface in Windows, click **Start → Programs → IBM → MPCLI**.

Logging on to a service processor

Use the following commands to log on to or log off from a service processor.

Table 4. Logon commands

Function	What it does	Command
Log on in-band (locally)	Log on to the local service processor.	logonlocal
Log on out-of-band using IP	Log on with a host name or IP address.	logonip -hostname *IPhost* -userid *IPuserid* -password *IPpassword* where: • *IPhost* is the domain name system (DNS) or IP address of the host • *IPuserid* is a user ID with administrative privileges • *IPpassword* is the password for the user ID
Log on out-of-band using RS-485	Log on to an RS-485 interconnect. Before you can log on to a system on the RS-485 interconnect, you must: 1. Log on to a gateway service processor using either the logonip or logonlocal command. Make sure that the system is connected to this service processor with an RS-485 connection. 2. Detect the systems using the RS-485 interconnect network by typing the following command: `logonrs485 -query`	There are two command options: • logonrs485 -index *RSindex* -userid *RSuserid* -password *RSpassword* • logonrs485 -name *RSname* -userid *RSuserid* -password *RSpassword* where: • *RSindex* is the number of the system the -query command returned • *RSname* is the name for the service processor • *RSuserid* is a user ID with administrative privileges • *RSpassword* is the password for the user ID
Log onto an RXE-100 expansion unit	Log on to an RXE-100 expansion unit.	logonrxe -index
Get service processors on RS-485 bus	Returns a list of service processors that share the RS-485 bus with the current service processor.	logonrs485 -query
Get RXE-100 expansion units	Returns a list of RXE-100 expansion units attached to the system.	logonrxe -query

Table 4. Logon commands (continued)

Function	What it does	Command
Log off from a service processor	Log off from the current session and disconnect from the service processor. If multiple sessions were initiated, any previous sessions are still active after the current session is terminated.	logoff
Find service processor ID	Display the text ID of the current service processor.	whoami

Notes:

1. To log on to a service processor using the logonrs485 command, you must first log on to a gateway service processor using either the logonip or logonlocal command.
2. The logonrs485 command options -index and -name specify which service processor to log on to. Use only one of these options for each logon. Also, if you are logging on to an ISMP service processor, the -userid and -password options are not necessary.
3. Multiple log ons to other service processors are allowed; however, all commands that are issued affect the most recently accessed service processor until another service processor is accessed or the logoff command is issued.

Examples:

- To log on out-of-band with an RS-485 interconnect using a local gateway service processor, type

  ```
  logonlocal

  logonrs485 -query

  logonrs485 -name batman -userid GayleM -password acegEGBDF
  ```

- To log on out-of-band with an IP connection, type

  ```
  logonip -hostname 168.1.101.100 -userid italia -password teramo
  ```

Managing the command-line interface

Use these commands to control the behavior of the command-line interface. They affect the output and function of the application but do not directly affect the service processor.

Table 5. Application-control commands

Function	What it does	Command
Get help	Returns the available help commands.	help
	Returns the application control, logon, and logoff commands.	help-cli
	Returns all the commands you can use after logging onto a service processor.	help-cmd
	Returns all the options for the specified command.	help-cmd *commandname* where *commandname* is the name of a command.
Get debugging information	Provides additional debugging information, such as more detail on a command success or failure. By default, verbose is off.	verbose Typing `verbose` toggles debugging information on or off.
Enter sleep mode	Allows the main execution thread to enter sleep mode.	sleep *milliseconds* where *milliseconds* is the number of milliseconds the application is in sleep mode.
Exit command-line interface	Closes the connection to the service processor and exits the program.	exit

Examples:

- To query the commands you can use after logging into a service processor, type
  ```
  help-cmd
  ```
- To query the options for a specific command, type
  ```
  help-cmd setalertcommon
  ```

Using command-line interface commands

This section contains command function and usage information and examples. It is divided into the following subsections:

BladeCenter unit commands

Use these commands to read and configure BladeCenter-unit-supported functions.

Management module failover commands

These commands enable or disable failover of the management module.

Table 6. Management module failover commands

Function	What it does	Command
Get failover information	Returns whether management module failover is enabled.	getmmfo
Enable failover	Enables management module failover immediately.	setmmfo -enable
Disable failover	Disables management module failover for the specified number of seconds.	setmmfo -disable *seconds* where *seconds* is the number of seconds for disabling failover.

Example:

To disable management module failover for 4 minutes, type

```
setmmfo -disable 240
```

Universal serial bus (USB) configuration commands

These commands display and set USB configuration.

Table 7. USB configuration commands

Function	What it does	Command
Get USB configuration	Returns the USB configuration.	getusb
	Returns the blade server that owns the USB.	getusb -owner
	Returns the last time USB ownership was assigned.	getusb -assigntime
Set USB owner	Assigns USB ownership to a blade server.	setusb -owner *bladeserver* where *bladeserver* is the number (1 - 14) of the blade server you want to own the USB.
Park USB	Prevents any blade server from owning the USB.	setusb -park

Examples:

- To query for the blade server that owns the USB and the last time an owner was assigned, type

  ```
  getusb -owner -assigntime
  ```

- To assign ownership of the USB to the third blade server, type

  ```
  setusb -owner 3
  ```

Keyboard, video, and mouse configuration commands

These commands display and set the keyboard, video, and mouse (KVM) configuration.

Table 8. KVM configuration commands

Function	What it does	Command
Get KVM configuration	Returns the KVM configuration.	getkvm
	Returns the blade server that owns the KVM.	getkvm -owner
	Returns the last time KVM ownership was assigned.	getkvm -assigntime
Assign KVM owner	Assigns KVM ownership to a blade server.	setkvm -owner *bladeserver* where *bladeserver* is the number (1 - 14) of the blade server you want to own the KVM.
Park KVM	Prevents any blade server from owning the KVM.	setkvm -park

Examples:

- To query for the blade server that owns the KVM and the last time an owner was assigned, type

  ```
  getkvm -owner -assigntime
  ```

- To assign ownership of the KVM to the third blade server, type

  ```
  setkvm -owner 3
  ```

Blade server start-option configuration commands

These commands display and set the blade server start-option configuration.

Table 9. Blade server start-option configuration commands

Function	What it does	Command
Get start options	Returns the start sequence for the specified blade server.	getbootoptions -blade *bladenum* where *bladenum* is the number of the blade server.
Set start sequence	Sets the start sequence for the specified blade server.	setbootoptions -options *bladenum*, *bootsequence* where • *bladenum* is the number of the blade server. • *bootsequence* can be any of the following and in any order, separated by commas and enclosed in double quotation marks: – hdd1 – hdd2 – cdrom – floppy – pxe
Delete start sequence	Delete the start sequence for the specified blade server.	setbootoptions -clear *bladenum* where *bladenum* is the number of the blade server.

Examples:

- To set the start sequence for the seventh blade server, type

  ```
  setbootoptions -options -blade 7, "hdd1,cdrom,floppy"
  ```

- To delete the start sequence for the eighth blade server, type

  ```
  setbootoptions -clear 8
  ```

Blade server policy commands

These commands display and set the blade server policy settings.

Table 10. Blade server policy commands

Function	What it does	Command
Get blade server policies	Returns the policy settings for all blade servers.	getpbpolicy
	Returns whether the specified blade server is enabled for local power control.	getpbpolicy -localpower *bladenum* where *bladenum* is the number of the blade server.
	Returns whether the specified blade server is enabled for local KVM control.	getpbpolicy -localkvm *bladenum* where *bladenum* is the number of the blade server.
	Returns whether the specified blade server is enabled for local USB control.	getpbpolicy -localusb *bladenum* where *bladenum* is the number of the blade server.
	Returns whether the specified blade server is enabled for the Wake on LAN® feature.	getpbpolicy -wol *bladenum* where *bladenum* is the number of the blade server.
Enable or disable KVM control	Enables or disables local KVM control for the specified blade server.	setpbpolicy -localkvm *bladenum*, *enableflag* where • *bladenum* is the number of the blade server. • *enableflag* is true to enable local KVM control or false to disable.
	Enables or disables local KVM control for all blade servers.	setpbpolicy -localkvmall *enableflag* where *enableflag* is true to enable local KVM control or false to disable.
Enable or disable power control	Enables or disables local power control for the specified blade server.	setpbpolicy -localpower *bladenum*, *enableflag* where • *bladenum* is the number of the blade server. • *enableflag* is true to enable local power control or false to disable.
	Enables or disables local power control for all blade servers.	setpbpolicy -localpowerall *enableflag* where *enableflag* is true to enable local power control or false to disable.

Table 10. Blade server policy commands (continued)

Function	What it does	Command
Enable or disable USB control	Enables or disables local USB control for the specified blade server.	setpbpolicy -localusb *bladenum*, *enableflag* where • *bladenum* is the number of the blade server. • *enableflag* is true to enable local USB control or false to disable.
	Enables or disables local USB control for all blade servers.	setpbpolicy -localusball *enableflag* where *enableflag* is true to enable local USB control or false to disable.
Enable or disable Wake on LAN	Enables or disables the Wake on LAN feature for the specified blade server.	setpbpolicy -wol *bladenum*, *enableflag* where • *bladenum* is the number of the blade server. • *enableflag* is true to enable Wake on LAN or false to disable.

Examples:

- To enable local power control on all blade servers, type

  ```
  setpbpolicy -localpowerall true
  ```

- To query the power policy for the eighth blade server, type

  ```
  getpbpolicy -localpower 8
  ```

- To enable the third blade server for the Wake on LAN feature, type

  ```
  setpbpolicy -wol 3, true
  ```

Switch module configuration commands

These commands display and set switch module configuration settings.

Table 11. Switch module configuration commands

Function	What it does	Command
Get configuration	Returns the current configuration of the specified switch module.	getsmnetwork -currentconfig *switch* where *switch* is the number of the switch module.
Get configuration method	Returns the current configuration method of the specified switch module. Valid values are static, dhcp, or bootip.	getsmnetwork -currentmethod *switch* where *switch* is the number of the switch module.
Get pending configuration	Returns the pending configuration of the specified switch module.	getsmnetwork -pendingconfig *switch* where *switch* is the number of the switch module.

Table 11. Switch module configuration commands (continued)

Function	What it does	Command
Get pending configuration method	Returns the pending configuration method of the specified switch module.	getsmnetwork -pendingmethod *switch* where *switch* is the number of the switch module.
Set IP address	Sets the IP address of the specified switch module.	setsmnetwork -ipaddress *switch*, *ipaddress* where: • *switch* is the number of the switch module. • *ipaddress* is the IP address of the switch module.
Set gateway	Sets the gateway of the specified switch module.	setsmnetwork -gateway *switch*, *ipaddress* where: • *switch* is the number of the switch module. • *ipaddress* is the IP address of the gateway.
Set subnet mask	Sets the subnet mask of the specified switch module.	setsmnetwork -subnet *switch*, *ipaddress* where: • *switch* is the number of the switch module. • *ipaddress* is the IP address of the subnet mask.
Set configuration method	Sets the configuration method of the specified switch module.	setsmnetwork -method *switch*, *configmethod* where: • *switch* is the number of the switch module. • *configmethod* is one of the following configuration methods: – static – dhcp – bootp

Table 11. Switch module configuration commands (continued)

Function	What it does	Command
Apply pending	Sets the pending configuration changes for the specified switch module. It is valid only with other options.	setsmnetwork -pending *switch*, *enableflag* where: • *switch* is the number of the switch module. • *enableflag* true to enable pending changes or false to disable.
	Sets the specified switch module to enable or disable applying changes made to other setsmnetwork options. **Note:** Before making any changes to the switch module configuration, you must disable applying changes. Then, after making configuration changes, you must enable to apply those changes.	setsmnetwork -enable *switch*, *enableflag* where: • *switch* is the number of the switch module. • *enableflag* true to enable applying changes or false to disable.
Get power status	Returns whether the specified switch module is on.	getswitchmodule -poweron *switch* where *switch* is the number of the switch module.
Get memory diagnostics status	Returns whether the memory diagnostics for the specified switch module are on.	getswitchmodule -memdiagson *switch* where *switch* is the number of the switch module.
Get port status	Returns whether other ports are enabled for the specified switch module.	getswitchmodule -cfgotherports *switch* where *switch* is the number of the switch module.
	Returns whether external ports are enabled for the specified switch module.	getswitchmodule -extportson *switch* where *switch* is the number of the switch module.
Get switch presence	Returns whether the specified switch module is present.	getswitchmodule -ping *switch* where *switch* is the number of the switch module.
Get POST status	Returns the results of the last POST for the specified switch module.	getswitchmodule -postresults *switch* where *switch* is the number of the switch module.
Set power enablement	Turns on or off the specified switch module.	setswitchmodule -poweron *switch*, *enableflag* where: • *switch* is the number of the switch module. • *enableflag* is true to enable turning on the switch module and false to disable.

Table 11. Switch module configuration commands (continued)

Function	What it does	Command
Memory diagnostics	Enables or disables the memory diagnostics for the specified switch module.	setswitchmodule -memdiagson *switch*, *enableflag* where: • *switch* is the number of the switch module. • *enableflag* is true to enable memory diagnostics and false to disable.
Other ports	Enables or disables other ports for the specified switch module.	setswitchmodule -cfgotherports *switch*, *enableflag* where: • *switch* is the number of the switch module. • *enableflag* is true to enable other ports and false to disable.
External ports	Enables or disables external ports for the specified switch module.	setswitchmodule -extportson *switch*, *enableflag* where: • *switch* is the number of the switch module. • *enableflag* is true to enable external ports and false to disable.
Restart	Restarts the specified switch module.	setswitchmodule -reboot *switch* where *switch* is the number of the switch module.
Reset configuration	Resets the configuration of the specified switch module.	setswitchmodule -resetconfig *switch* where *switch* is the number of the switch module.

Examples:

- To set the IP address, gateway, and subnet mask and to choose a configuration method for the third switch module, type the following (all on one line)

  ```
  setsmnetwork -ipaddress 3,203.1.168.9 -gateway 3,203.1.168.10
  -subnet 3,255.255.255.0 -method 3,static
  ```

- To enable memory diagnostics and enable other ports for the third switch module, type

  ```
  setswitchmodule -memdiagson 3,true -cfgotherports 3,true
  ```

BladeCenter unit topology command

This command displays the topology of the BladeCenter unit.

Table 12. BladeCenter unit topology command

Function	What it does	Command
Get all component information	Returns all BladeCenter unit component information.	gettopology
Get supported components	Returns the maximum number of supported blade servers.	gettopology -pbsupported
	Returns the maximum number of supported switch modules.	gettopology -smsupported
	Returns the maximum number of supported management modules.	gettopology -mmsupported
	Returns the maximum number of supported power supplies.	gettopology -pssupported
	Returns the maximum number of supported fans.	gettopology -bmsupported
Get how many components are installed	Returns the number of installed blade servers.	gettopology -pbcount
	Returns the number of installed switch modules.	gettopology -smcount
	Returns the number of installed management modules.	gettopology -mmcount
	Returns the number of installed power supplies.	gettopology -pscount
	Returns the number of installed fans.	gettopology -bmcount
Get which components are installed	Returns which blade servers are installed.	gettopology -pbinstalled
	Returns which switch modules are installed.	gettopology -sminstalled
	Returns which management modules are installed.	gettopology -mminstalled
	Returns which power supplies are installed.	gettopology -psinstalled
	Returns which fans are installed.	gettopology -bminstalled

Examples:

- To query how many blade servers, switch modules, and management modules are installed in a BladeCenter unit, type

  ```
  gettopology -pbcount -smcount -mmcount
  ```

- To query how many power supplies are supported in the BladeCenter unit and how many are installed, type

  ```
  gettopology -pssupported -psinstalled
  ```

BladeCenter unit VPD command

This command displays the BladeCenter unit VPD.

Table 13. BladeCenter unit VPD command

Function	What it does	Command
Get all VPD	Returns all VPD for the BladeCenter unit.	getbladecentervpd
Get component VPD	Returns the VPD for the specified switch module.	getbladecentervpd -switchmodule *switchnum* where *switchnum* is the number of the switch.
	Returns the VPD for the midplane.	getbladecentervpd -midplane
	Returns the VPD for the front panel.	getbladecentervpd -frontpanel
	Returns the VPD for the specified power supply.	getbladecentervpd -powersupply *powernum* where *powernum* is the number of the power supply.
	Returns the VPD for the specified blade server.	getbladecentervpd -processorblade *bladenum* where *bladenum* is the number of the blade server.
	Returns the VPD for the specified management module.	getbladecentervpd -mgmtmodule *mm* where *mm* is the number of the management module.

Examples:

- To get the VPD for the eighth blade server and the second management module, type the following (all on one line)

  ```
  getbladecentervpd -processorblade 8 -mgmtmodule 2
  ```

- To get the VPD for the midplane, front panel, and the third power supply, type

  ```
  getbladecentervpd -midplane -frontpanel -powersupply 3
  ```

Network interface commands

The network interface commands are used to configure network protocols and hardware such as SNMP, PPP, IP, network hardware, Dynamic Host Configuration Protocol (DHCP), and DNS.

SNMP configuration commands

These commands read and configure the SNMP interface of the service processor. For more information about service processor SNMP configuration, see the applicable xSeries *User's Guides*.

Table 14. SNMP configuration commands

Function	What it does	Command
Get basic SNMP configuration	Returns the SNMP configuration.	getsnmp If no option is added, the command returns whether agent or traps are enabled and the contact name and location.
	Returns the ID for the system location.	getsnmp -contactlocation
	Returns the ID for the system contact name.	getsnmp -contactname
	Returns whether SNMP traps are enabled.	getsnmp -traps
	Returns whether the SNMP agent is enabled.	getsnmp -agent
	Returns the name for the specified community. There are three user-defined communities available.	getsnmp -communityname *commnum* where *commnum* indicates which community.
Get IP address	Returns the IP address for the specified community.	getsnmp -ipaddress *commnum, commIPnum* where: • *commnum* indicates which community. • *commIPnum* indicates whether this is the first, second, or third IP address for the community.
Get host name	Returns the host name for the specified community.	getsnmp -hostname *commnum, commIPnum* where: • *commnum* indicates which community. • *commIPnum* indicates whether this is the first, second, or third IP address for the community.

Table 14. SNMP configuration commands (continued)

Function	What it does	Command
Set basic SNMP configuration	Sets the system location.	setsnmp -contactlocation *location* where *location* is the system location.
	Sets the system contact name.	setsnmp -contactname *name* where *name* is the system contact name.
	Enables or disables the SNMP traps sent by the service processor. Valid values are true and false.	setsnmp -traps *enableflag* where *enableflag* is true to enable sending SNMP traps or false to disable sending SNMP traps.
	Enables or disables the service processor SNMP agent. Valid values are true and false.	setsnmp -agent *enableflag* where *enableflag* is true to enable the SNMP agent or false to disable the SNMP agent.
Set community name	Sets the specified community name.	setsnmp -communityname *commnum, commname* where: • *commnum* indicates which community. • *commname* is the community name.
Set IP address	Sets an IP address for the specified SNMP community.	setsnmp -ipaddress *commnum*, *commIPnum*, *xxx.xxx.xxx.xxx* where: • *commnum* indicates which community. • *commIPnum* indicates whether this is the first, second, or third IP address for the community. • *xxx.xxx.xxx.xxx* is the IP address you want to assign.

Table 14. SNMP configuration commands (continued)

Function	What it does	Command
Set host name	Sets the host name for the specified SNMP community.	setsnmp -hostname *commnum, commIPnum, hostname* where: • *commnum* indicates which community. • *commIPnum* indicates whether this is the first, second, or third IP address for the community. • *hostname* is the host name you want to assign.

Examples:

- To return the first SNMP community name, type

  ```
  getsnmp -communityname 1
  ```

- To enable the SNMP agent and sending of SNMP traps, type

  ```
  setsnmp -traps true -agent true
  ```

- There are three user-defined SNMP communities, and each community can have three user-defined locations (IP address or host name). To set the host name and IP address for the second location of the second SNMP community, type

  ```
  setsnmp -hostname 2,2,batman -ipaddress 2,2,198.1.100.101
  ```

PPP configuration commands

These commands are supported only on the xSeries 220 server. These commands read and configure the PPP interface of the service processor. For more information about service processor PPP configuration, see the applicable xSeries *User's Guides*.

Table 15. PPP configuration commands

Function	What it does	Command
Get PPP configuration	Returns all PPP configuration information.	getppp
	Returns whether PPP configuration is enabled.	getppp -enabled
	Returns the IP address assigned to the client-side of the network connection.	getppp -remoteip
	Returns the IP address assigned to the service processor-side of the network connection.	getppp -serverip
	Returns the IP address assigned as the subnet mask.	getppp -subnet
	Returns the PPP protocol type, as represented by integers: • 1 = PAP • 2 = CHAP • 3 = CHAP then PAP	getppp -protocol
Set PPP configuration	Enables or disables PPP configuration.	setppp -enabled *enableflag* where *enableflag* is true to enable PPP configuration or false to disable PPP configuration.
	Sets the IP address for the client-side of the network connection.	setppp -remoteip *xxx.xxx.xxx.xxx* where *xxx.xxx.xxx.xxx* is the client-side IP address.
	Sets the IP address for the service processor side of the network connection.	setppp -serverip *xxx.xxx.xxx.xxx* where *xxx.xxx.xxx.xxx* is the server-side IP address.
	Sets the PPP subnet mask.	setppp -subnet *xxx.xxx.xxx.xxx* where *xxx.xxx.xxx.xxx* is the subnet mask.
	Sets the PPP protocol type, as represented by an integer.	setppp -protocol *protocoltype* where *protocoltype* is 1, 2, or 3: • 1 = PAP • 2 = CHAP • 3 = CHAP then PAP

Examples:

- To query the IP address, subnet mask, and protocol type of the PPP configuration, type

```
getppp -serverip -subnet -protocol
```

- To enable PPP configuration, set the client-side and service-processor-side IP addresses, and set CHAP then PAP as the protocol, type

  ```
  setppp -enabled true -remoteip 198.1.100.135 -serverip 168.1.100.101 -protocol 3
  ```

Static IP configuration commands

These commands read and configure the static IP protocol of the service processor. For more information about service processor IP configuration, see the applicable xSeries *User's Guides*.

Table 16. Static IP configuration commands

Function	What it does	Command
Get static IP configuration	Returns the IP configuration, such as host name, IP address, subnet mask, gateway address, and connection type (static or DHCP).	getip If an option is not added, the command returns the static IP configuration of the primary interface.
	For ASM PCI systems only. Returns the IP configuration of specified interface.	getip -interface *index* where *index* is the index number of the interface.
	Cannot be used on ASM PCI systems. Returns the IP configuration of the active interface, such as host name, IP address, subnet mask, gateway address, connection type (static or DHCP), and DHCP configuration if DHCP is the connection type.	getip -active
Set static IP configuration	Sets the host name for the service processor.	setip -interface *index* -hostname *hostname* where: • *index* is the index number of the interface. • *hostname* is name you want to assign to the service processor.
	Set IP address of the service processor.	setip -interface *index* -ipaddress *xxx.xxx.xxx.xxx* where: • *index* is the index number of the interface. • *xxx.xxx.xxx.xxx* is the IP address.
	Sets the IP address of the gateway used by the service processor.	setip -interface *index* -gateway *xxx.xxx.xxx.xxx* where: • *index* is the index number of the interface. • *xxx.xxx.xxx.xxx* is the IP address.

Table 16. Static IP configuration commands (continued)

Function	What it does	Command
	Sets the IP address of the subnet mask.	setip -interface *index* -subnet *xxx.xxx.xxx.xxx* where: • *index* is the index number of the interface. • *xxx.xxx.xxx.xxx* is the IP address . .
	Sets the IP configuration type.	setip -interface *index* -method *configtype* where: • *index* is the index number of the interface. • *configtype* is: – Static – DHCP – DHCPStatic

Examples:

- To query all IP configuration details, type

  ```
  getip
  ```

- To set the IP host name, IP address, gateway, and subnet mask, type the following (all on one line)

  ```
  setipaddress -interface 1 -hostname batman -ipaddress 186.1.111.6
  -gateway 130.10.115.100 -subnet 168.1.100.101
  ```

Network hardware configuration commands

These commands read or enable attributes of the network hardware configuration of the service processor. For more information about service processor network hardware configuration, see the applicable xSeries *User's Guides*.

Table 17. Network hardware configuration commands

Function	What it does	Command
Get network hardware configuration	Returns the network hardware configuration for the specified interface.	getnethw -interface *index* where *index* is the index number of the interface.
	Returns data for all getnethw options for the specified interface.	getnethw -interface *index* -all where *index* is the index number of the interface.
Get line type	*This is an ASM PCI option only.* Returns the type of hardware media for the specified interface. Valid values are: • ENET for Ethernet • TR for Token Ring	getnethw -interface *index* -linetype where *index* is the index number of the interface.

Table 17. Network hardware configuration commands (continued)

Function	What it does	Command
Get interface status	Returns whether the specified interface is enabled or disabled. Only one interface can be enabled at a time.	getnethw -interface *index* -enabled where *index* is the index number of the interface.
Get data rate	Returns the data rate of the specified interface. Valid values are: • Auto, 10m, or 100m for Ethernet • Auto, 4m, or 16m for Token Ring	getnethw -interface *index* -datarate where *index* is the index number of the interface.
Get duplex mode	Returns the duplex mode of the specified interface. Valid values are: • Auto • Half • Full	getnethw -interface *index* -duplex where *index* is the index number of the interface.
Get MAC address	Returns the network-administrator-assigned MAC address, in the [*xx xx xx xx xx xx*] format, for the specified interface.	getnethw -interface *index* -adminmac where *index* is the index number of the interface.
Get MTU	Returns the maximum transmission unit (MTU) for this network segment.	getnethw -interface *index* -mtu where *index* is the index number of the interface.
Get routing bytes	Returns whether routing bytes for the specified interface are enabled if the line type is Token Ring.	getnethw -interface *index* -routingbytes where *index* is the index number of the interface.
Get burned MAC	Returns the MAC address burned into the ROM of the network interface card (NIC).	getnethw -interface *index* -burnedmac where *index* is the index number of the interface.
Set interface index	Sets the network hardware configuration index entry.	setnethw -interface *index* where *index* is the index number of the interface.
Set line type	*This is an ASM PCI option only.* Sets the type of hardware media for the specified interface.	setnethw -interface *index* -linetype *linetype* where: • *index* is the index number of the interface. • linetype is ENET for Ethernet or TR for Token Ring.

Table 17. Network hardware configuration commands (continued)

Function	What it does	Command
Enable or disable interface	Enables or disables the specified interface. Only one interface can be enabled at a time.	setnethw -interface *index* -enabled *enableflag* where: • *index* is the index number of the interface. • *enableflag* is true to enable the interface or false to disable the interface.
Set data rate	Set the data rate for the specified interface.	setnethw -interface *index* -datarate *datarate* where: • *index* is the index number of the interface. • *datarate* is – Auto, 10m, or 100m for Ethernet – Auto, 4m, or 16m for Token Ring
Set duplex mode	Sets the duplex mode for the specified interface.	setnethw -interface *index* -duplex *duplexmode* where: • *index* is the index number of the interface. • *duplexmode* is Auto, Full, or Half.
Set MAC address	Sets the network-administrator-assigned MAC address for the specified interface. To use the hardware MAC address, specify 00.00.00.00.00.00 for the adminmac value.	setnethw -interface *index* -adminmac *xx xx xx xx xx xx* where: • *index* is the index number of the interface. • *xx xx xx xx xx xx* is the administrator-assigned MAC address.
Set MTU	Sets the MTU for this network segment.	setnethw -interface *index* -mtu *mtunum* where: • *index* is the index number of the interface. • *mtunum* is the MTU for this network segment.

Table 17. Network hardware configuration commands (continued)

Function	What it does	Command
Enable or disable routing bytes	Enables or disables the routing bytes for the specified interface if the line type is Token Ring.	setnethw -interface *index* -routingbytes *rbytes* where: • *index* is the index number of the interface. • *rbytes* is true to enable routing bytes or false to disable them.

Examples:

- To enable interface 1 and set its data rate to Auto, type

  ```
  setnethw -interface 1 -enabled true -datarate Auto
  ```

- To set the network-administrator-assigned MAC address for interface 1, type

  ```
  setnethw -interface 1 -adminmac "00 00 00 00 00 00"
  ```

DHCP configuration commands

These commands read DHCP configuration of the service processor. For more information about service processor DHCP configuration, see the applicable xSeries *User's Guides*.

Table 18. DHCP configuration commands

Function	What it does	Command
Get DHCP configuration	Returns the DHCP configuration of the service processor.	getdhcp
Set DHCP configuration	Sets whether DHCP configuration is enabled.	setdhcp -enabled *enableflag* where *enableflag* is true to enable DHCP or false to disable.

Example:

- To enable DHCP configuration, type

  ```
  setdhcp -enable True
  ```

DNS configuration commands

These commands configure the DNS client of the service processor. You can have up to four static IP address-to-host-name mappings without having a DNS server present.

Table 19. DNS configuration commands

Function	What it does	Command
Get DNS configuration	Returns the DNS client configuration.	getdns
	Returns the static mapping of the specified system.	getdns -mapping *ipaddressorhost* where *ipaddressorhost* is the system IP address or host name.
	Returns whether DNS is configured. Valid values are true and false.	getdns -enabled
	Returns the IP address of the specified DNS server.	getdns -server *ipaddress* where *ipaddress* is the IP address of the DNS server.
Enable or disable DNS client	Enables or disables the DNS client.	setdns -enabled *enableflag* where *enableflag* is true to enable the DNS client or false to disable.
Set DNS IP address	Sets the IP address of the specified DNS server.	setdns -server *servernum*, *ipaddress* where: • *servernum* is: – 1 for the primary server – 2 for the secondary server – 3 for the tertiary server • *ipaddress* is the IP address.
Set IP address to host name mapping	Set the static mapping of an IP address to a host name when there is no DNS server.	setdns -mapping *mappingnum*, *ipaddress*, *hostname* where: • *mappingnum* is the mapping number. • *ipaddress* is the IP address. • *hostname* is the host name of the DNS server.

Examples:

• To clear the third static mapping by specifying blank parameters, type

```
setdns -mapping 3,"","
```

- To set the second static DNS map for an IP address and host name, type

```
setdns -mapping 2,33.12.125.89,batman
```

Serial port configuration commands

These commands enable read and write access to both hardware and software configurations of a serial port. For more information about service processor serial port configurations, see the applicable xSeries *User's Guides*.

Serial port software configuration commands

These commands read and set the software configuration of the serial port.

Table 20. Serial port software configuration commands

Function	What it does	Command
Get serial port software configuration	Returns the software configuration of the serial port.	getcomsw If no option is added, the command returns the configuration of serial port one.
	Returns whether dial-in access using the serial port is enabled. Values are true and false.	getcomsw -dialinenabled
	Returns whether the service processor owns the serial port that it shares with the operating system on startup. Values are true and false.	getcomsw -own
	Returns the number of seconds that must elapse between successive logins.	getcomsw -tamperdelay
Set dial-in access	Enables or disables dial-in access using the serial ports.	setcomsw -dialinenabled *enableflag* where *enableflag* is true to enable dial-in access and false to disable.
Set serial port ownership	Enables or disables the service processor to own the serial port it shares with the operating system on startup.	setcomsw -own *enableflag* where *enableflag* is true to enable ownership and false to disable.
Turn session on or off	Enable or disable the session on serial port one.	setcomsw -sessionon *enableflag* where *enableflag* is true to turn on the session and false to turn off.
Set login delay	Sets the number of seconds that must elapse between successive logins.	setcomsw -tamperdelay *seconds* where *seconds* is the number of seconds that must elapse.

Examples:

- To configure the COM port software to enable the session and to set the ownership of the COM port to the service processor upon restart, type

  ```
  setcomsw -session true -own true
  ```

- To query the number of seconds between successive logins and whether dial-in access for the serial port is enabled, type

  ```
  getcomsw -tamperdelay -dialinenabled
  ```

Serial port hardware configuration commands

These commands read and set the serial port hardware configuration.

Table 21. Serial port hardware configuration commands

Function	What it does	Command
Get serial port hardware configuration	Returns the hardware configuration of the serial port.	getcomhw If no option is added, the command returns the configuration of serial port one.
	Returns the hardware configuration of the specified serial port.	getcomhw -port *index* where *index* is the index number of the serial port.
Set serial port index	Sets the serial port index number.	setcomhw -port *index* where *index* is the index number of the serial port.
Set baud rate	Sets the baud rate for the specified serial port.	setcomhw -port *index* -baudrate *baudrate* where: • *index* is the index number of the serial port. • *baudrate* is a valid baud rate at which the modem should run.
Set parity	Sets the parity on the specified serial port.	setcomhw -port *index* -parity *parity* where: • *index* is the index number of the serial port. • *parity* is one of the following: – None – Odd – Even – Mark – Space
Set stop bit value	Sets the stop bit value for the specified serial port.	setcomhw -port *index* -stopbits *bitvalue* where: • *index* is the index number of the serial port. • *bitvalue* is a stop bit value of 0, 1, or 2.

Table 21. Serial port hardware configuration commands (continued)

Function	What it does	Command
Set initialization string	Sets the initialization string for the modem on the specified serial port.	setcomhw -port *index* -initstring *initstring* where: • *index* is the index number of the serial port. • *initstring* is an initialization string for the modem.
Enable or disable serial port	Enables or disables the specified serial port.	setcomhw -port *index* -enabled *enableflag* where: • *index* is the index number of the serial port. • *enableflag* is true to enable the port and false to disable.
Set caller ID	Sets the caller ID for a discovered phone number on the specified serial port.	setcomhw -port *index* -calleridstring *callerID* where: • *index* is the index number of the serial port. • *callerID* is the caller ID.
Return to default settings	Returns the modem on the specified serial port to its factory-default settings.	setcomhw -port *index* -returnfactorydefault *factorysettings* where: • *index* is the index number of the serial port. • *factorysettings* is the control string in your modem documentation for its default settings.
Set time delay	Sets a time delay before and after the sending of a modem escape string for the specified serial port.	setcomhw -port *index* -escapeguardtime *escape* where: • *index* is the index number of the serial port. • *escape* is the time delay.
Set escape code	Sets an escape code for the specified serial port.	setcomhw -port *index* -escapestring *escape* where: • *index* is the index number of the serial port. • *escape* is the escape code.

Table 21. Serial port hardware configuration commands (continued)

Function	What it does	Command
Set area code	Sets the area code for the modem on the specified serial port.	setcomhw -port *index* -dialprefix *areacode* where: • *index* is the index number of the serial port. • *areacode* is the area code.
Set phone number	Sets the phone number for the modem on the specified serial port.	setcomhw -port *index* -dialpostfix *phonenum* where: • *index* is the index number of the serial port. • *phonenum* is the phone number.
Set auto answer	Sets the auto-answer string for the modem on the specified serial port.	setcomhw -port *index* -autoanswer *answer* where: • *index* is the index number of the serial port. • *answer* is the auto-answer string. The generic value is ATS0=1.
Set auto-answer stop	Sets the auto-answer stop string for the modem on the specified serial port.	setcomhw -port *index* -autoanswerstop *answerstop* where: • *index* is the index number of the serial port. • *answerstop* is the auto-answer stop string. The generic value is ATS0=0.
Set query string	Sets the query string for the modem on the specified serial port.	setcomhw -port *index* -querystring *modemquery* where: • *index* is the index number of the serial port. • *modemquery* is the modem query string. The generic value is AT.
Set exit string	Sets the string used to stop communication with the modem on the specified serial port.	setcomhw -port *index* -hangupstring *hangup* where: • *index* is the index number of the serial port. • *hangup* is the string used to hang up.

Examples:

- To configure the first serial port and load the initial carrier string to the host, type

  ```
  setcomhw -port 1 -initstring "AT OT #2"
  ```

- To enable the first serial port and set the parity to Even, type

  ```
  setcomhw -port 1 -enabled true -parity Even
  ```

Service processor configuration commands

The following commands allow you to read and set the service processor clock and IDs, update the firmware, reset to the default configuration, and restart the service processor.

Service processor ID commands

Use the service processor ID commands to query and set service processor identification strings.

Table 22. Service processor ID commands

Function	What it does	Command
Get all ID strings	Returns the identification strings stored in the service processor ID table. If no parameter is specified, all IDs are returned.	getmpid
Get numeric ID	Returns the numeric identification string of the service processor.	getmpid -numeric
Get text ID	Returns the text identification string of the service processor.	getmpid -text
Get asset tag	Returns the asset tag of the service processor.	getmpid -assettag
Set numeric ID	Sets the numeric ID string of the service processor.	setmpid -numeric
Set text ID	Sets the text ID string of the service processor.	setmpid -text
Set asset tag	Sets the asset tag of the service processor.	setmpid -assettag

Example:

- To set the service processor ID tag to SP Batman #13, type

  ```
  setmpid -text "SP Batman #13"
  ```

Service processor clock commands

These commands are used to read and write to the internal clock on the service processor. Time is interpreted relative to the current time on your service processor. On systems where a Greenwich mean time (GMT) offset is supported, the service processor clock is converted to GMT using the offset and then converted to local time for the system.

Table 23. Service processor clock commands

Function	What it does	Command
Get all service processor clock information	Returns the time and date, whether daylight saving time (DST) is used, and GMT offset information as recorded in the service processor.	getmpclock

Table 23. Service processor clock commands (continued)

Function	What it does	Command
Get time and date	Returns the current time of the internal clock on the service processor using the 24-hour clock format and the calendar format of *mm*/*dd*/*yyyy*.	getmpclock -timeanddate
Get DST	Reports whether the service processor uses DST.	getmpclock -dst
Get GMT offset	Returns the GMT offset, in hours. This value ranges from -12 to +12.	getmpclock -gmtoffset
Set service processor time	Sets the time for the internal clock on the service processor using the 24-hour clock.	setmpclock -time *hh:mm* where *hh*:*mm* is the hour and minutes.
Set service processor date	Sets the date for the internal clock of the service processor using the format of *mm*/*dd*/*yyyy*.	setmpclock -date *mm*/*dd*/*yyyy* where *mm*/*dd*/*yyyy* is the month, day, and year.
Enable/disable DST	Enables or disables DST.	setmpclock -dst *enableflag* where *enableflag* is true to enable DST or false to disable.
Set GMT offset	Sets GMT offset, in hours. Valid values range from -12 to +12.	setmpclock -gmtoffset *offsetnum* where *offsetnum* is the GMT offset.

Examples:

- To set the service processor time to 9:00 p.m. and the date to 3 November 2003, and enable daylight saving time, type

  ```
  setmpclock -time 21:00 -date 11/03/2003 -dst TRUE
  ```

- To query the time and date on the service processor and whether it is using DST, type

  ```
  getmpclock -timeanddate -dst
  ```

Service processor reset command

This command resets the service processor configuration to its factory-default settings.

Table 24. Service processor configuration command

Function	What it does	Command
Reset service processor	Resets the service processor configuration to its factory-default settings.	resetmp

Service processor firmware update command

This command updates service processor firmware. The -usb option is valid for the management module only.

Note: You must obtain the applicable firmware update file from http://www.ibm.com/pc/support/

Table 25. Service processor firmware update command

Function	What it does	Command
Update firmware	Updates the firmware for the main application image.	fwupdate -mn *filename* where *filename* is the fully-qualified path name for the file.
	Updates the firmware for the boot ROM image.	fwupdate -br *filename* where *filename* is the fully-qualified path name for the file.
	Updates the firmware for the remote graphics image.	fwupdate -vnc *filename* where *filename* is the fully-qualified path name for the file.
	Updates the firmware for the USB images; however, each USB image must be updated individually.	fwupdate -usb *filename* where *filename* is the fully-qualified path name for the file.

Examples:

- To update a boot ROM using a file you have downloaded from a support site, type

  ```
  fwupdate -br d:\firmware\x220\batman\CNETBRUS.PKT
  ```

- To update a main application firmware image using a file you have downloaded from a support site, type

  ```
  fwupdate -mn d:\firmware\x220\batman\CNETMNUS.PKT
  ```

Service processor restart command

The restart command restarts the current service processor hardware and closes the command-line-interface connection to the service processor. If another logon session with a different service processor began before this session, the application returns to that session.

Table 26. Service processor restart command

Function	What it does	Command
Restart service processor	Restarts the current service processor hardware and close the command-line-interface connection to the service processor.	restartmp

Service processor event commands

The following commands can be used to read and change dial-out and dial-in alert configuration settings; read and clear the event log; and read, enable, or disable alert triggers for the service processor.

Event log commands

These commands display or clear entries in the service processor event log. When a query is sent to the event log, events are listed in a last-in-first-out order.

Table 27. Service processor event log

Function	What it does	Command
Get event log	Returns the first service processor event log entry.	getmplog getmplog -first
	Returns the next service processor event log entry.	getmplog -next
	Returns all service processor event log entries in last-in-first-out order.	getmplog -all
Clear event log	Deletes all entries from the service processor event log.	clearmplog

Example:

- To query the service processor event log for the next event log entry, type

```
getmplog -next
```

Alert trigger commands

The alert trigger commands read, enable, or disable supported dial-out triggers that are monitored by the service processor. The following table lists all available parameters and the alert triggers they control. For more information about service processor alert triggers, see the applicable xSeries *User's Guide*.

Table 28. Alert trigger commands

Function	What it does	Command
Get software configuration of COM port	Returns a list of all the events that are currently enabled.	getalerttrigger -enabled
	Returns a list of all the events that are currently disabled.	getalerttrigger -disabled
	Returns a superset of events that can be forwarded by any service processor.	getalerttrigger -possible
	Returns a list of all the events that can be forwarded by the target service processor.	getalerttrigger -supported
Enable alert triggers	Enables an alert or set of alerts. When you enable an alert, it is set to true.	setalerttrigger -enabled *alert* where *alert* is the name of an alert trigger parameter. You can specify more than one parameter.
Disable alert triggers	Disables an alert or set of alerts. When you disable an alert, it is set to false.	setalerttrigger -disabled *alert* where *alert* is the name of an alert trigger parameter. You can specify more than one parameter.

The following table contains the alert trigger parameters that you can use with the setalerttrigger command.

Parameter	Enables or disables alert triggering for
all	All supported events
critical.all	All supported critical events
critical.temp	A critical temperature event
critical.voltage	A critical voltage event
critical.tamper	A critical tampering event
critical.multiple_fan	A critical multiple fan-failure event
critical.power_supply	A critical power-failure event
critical.dasd	A critical hard disk backplane event
critical.vrm	A critical VRM failure event
noncritical.all	All noncritical events
noncritical.rps	A redundant power supply (RPS) event
noncritical.single_fan	An event of alerts for a single fan failure
noncritical.temperature	A temperature event
noncritical.voltage	A voltage event
noncritical.log_full	A log-full event
noncritical.log_75%	A log-75%-full event
noncritical.secondary	A secondary event
system.all	All supported system-level events
system.post	When an operating-system POST timeout expires
system.os	When an operating-system heartbeat timeout expires
system.application	A system-application alert passed to the service processor
system.power_off	When the system is powered off
system.power_on	When the system is powered on
system.boot	When the system failed to start
system.loader	When an operating system loader timeout expired
system.pfa	An alert received through a Predictive Failure Alert® (PFA)
system.inventory	A system-inventory event
system.netstack	A system-network-stack event
complex.ping	A complex discovery ping event
info.remote_login	An informational remote logon event
info.chipkill	An informational Chipkill™ memory event
info.fuelgauge	An informational fuel-gauge event
info.component	An informational component-activity event
normal.single_fan	A normal single-fan event
normal.voltage	A normal voltage event
normal.temperature	A normal temperature event
normal.component	A normal component-activity event

Parameter	Enables or disables alert triggering for
expdev.crit_temp	An expansion-device critical temperature event
expdev.noncrit_temp	An expansion-device non-critical temperature event
expdev.crit_voltage	An expansion-device critical voltage event
expdev.noncrit_voltage	An expansion-device noncritical voltage event
expdev.fan_failure	An expansion-device fan-failure event
expdev.pfa	An expansion-device PFA event
expdev.power_failure	An expansion-device power-failure event
expdev.crit_voltfault	An expansion-device voltage-fault event
expdev.fuelgauge	An expansion-device fuel-gauge event
blade.multswitch	A BladeCenter unit multiple switch event
blade.incompat_config	A BladeCenter unit incompatible-configuration event
blade.redund_mm	A BladeCenter unit redundant management-module event
blade.kvm_usb_fail	A BladeCenter unit KVM/USB switching failure event

Examples:

- To enable an event to be sent when a temperature, voltage, or power supply is in a critical state, type

  ```
  setalerttrigger -enabled critical.temp, critical.voltage, critical.powersupply
  ```

- To query the alert triggers that are supported by the service processor, type

  ```
  getalerttriggers -supported
  ```

Alert dial-out common configuration commands

These commands display and set the alert dial-out configuration that is common (with the exception of triggers) to all dial-out entries.

Table 29. Alert dial-out common configuration commands

Function	What it does	Command
Get all dial-out alerts	Returns the status of all common dial-out alerts.	getalertcommon If parameters are not specified, all common dial-out alert status is returned.
Get polling delay	Reports the polling delay, in seconds, between entries sent by the service processor.	getalertcommon -entrydelay
Get retry delay	Reports the retry delay, in seconds, before the service processor attempts to resend an alert.	getalertcommon -retrydelay
Get retry limit	Reports the number of cycles a service processor attempts through the dial-out entry table before failing.	getalertcommon -retrylimit
Get SMTP address	Reports the SMTP server IP address.	getalertcommon -smtpserver
Log sent as e-mail	Reports whether the log is sent as an e-mail attachment.	getalertcommon -sendlog

Table 29. Alert dial-out common configuration commands (continued)

Function	What it does	Command
Set polling delay	Sets the polling delay, in seconds, between entries reported by the service processor.	setalertcommon -entrydelay *seconds* where *seconds* is the number of seconds for the polling delay.
Set retry delay	Sets the retry delay, in seconds, before the service processor attempts to poll the entry table.	setalertcommon -retrydelay *seconds* where *seconds* is the number of seconds for the retry delay.
Set retry limit	Sets the number of cycles a service processor attempts through the dial-out entry table before failing.	setalertcommon -retrylimit *cycletime* where *cycletime* is the number of cycles through the table before failing.
Set SMTP address	Sets the SMTP server IP address.	setalertcommon -smtpserver
Stop sending alerts	Cancels all pending notifications and clears the queue.	setalertcommon -stopsending
Send log as e-mail	Specifies whether to send the log as an e-mail attachment.	setalertcommon -sendlog *emailflag* where *emailflag* is true to send logs by e-mail or false to not send e-mail logs.

Examples:

- To query the SMTP IP address and whether the log is sent as an e-mail attachment, type

  ```
  getalertcommon -smtpserver -sendlog
  ```

- To set the SMTP IP address and enable sending the log by e-mail, type

  ```
  setalertcommon -smtpserver 260.1.100.101 -sendlog true
  ```

Alert dial-out entry configuration commands

These commands display and set the alert dial-out entry configuration to allow access to profiles of recipients of the remote alerts. For more information about remote alert recipients, see the applicable xSeries *User's Guides*.

Table 30. Alert dial-out entry configuration commands

Function	What it does	Command
Get dial-out alert entry	Returns the configuration of the specified dial-out alert entry.	getalertentry -index *entrynumber* where *entrynumber* is the number of the entry, which ranges from 1-12.
Set a dial-out alert entry	Specifies which dial-out alert entry to update. This command always uses the -index option and must be used with one or more of the other setalertentry options.	setalertentry -index *entrynumber* where *entrynumber* is the number of the entry, which ranges from 1-12.
Enable a dial-out entry	Enables or disables the dial-out alert entry without first clearing the configuration.	setalertentry -index *entrynumber* -enabled *enableflag* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *enableflag* is true to enable clearing the configuration or false to disable clearing the configuration.
Delete a dial-out alert entry	Deletes the dial-out alert entry specified by the -index option.	deletealertentry -index *entrynumber* where *entrynumber* is the number of the entry, which ranges from 1-12.
Set sending of critical events	Enables or disables sending only critical events.	setalertentry -index *entrynumber* -criticaleventseverity *enableflag* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *enableflag* is true to enable sending critical events or false to disable sending critical events.

Table 30. Alert dial-out entry configuration commands (continued)

Function	What it does	Command
Add entry description	Adds a description for the specified dial-out alert entry.	setalertentry -index *entrynumber* -description *descriptiontext* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *descriptiontext* is a string that identifies the dial-out alert entry.
Set connection type	Specifies the connection type for the dial-out alert entry. Valid parameters are: • snmp.ppp • snmp.lan • director.lan • director.modem • pager.numeric • pager.alphanumeric • email.lan • email.ppp • director.all	setalertentry -index *entrynumber* -type *connection* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *connection* is one of the valid parameters for a dial-out entry.
Set phone number	Sets the phone number of the specified dial-out alert entry for the service processor to dial.	setalertentry -index *entrynumber* -number *phonenum* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *phonenum* is the phone number.
Set IP address	Sets the IP address for the service processor to use if required by the specified dial-out alert entry.	setalertentry -index *entrynumber* -ipaddress *ipaddress* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *ipaddress* is the IP address you want to add.
Set PIN	Sets a personal identification number (PIN) code if required by the specified dial-out alert entry.	setalertentry -index *entrynumber* -pin *pincode* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *pincode* is the PIN.

Table 30. Alert dial-out entry configuration commands (continued)

Function	What it does	Command
Set e-mail address	Sets an e-mail address if required by the specified dial-out alert entry.	setalertentry -index *entrynumber* -emailaddress *email* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *email* is the e-mail address.
Set PPP login	Sets a PPP logon ID if required by the specified dial-out alert entry.	setalertentry -index *entrynumber* -pppid *ppplogin* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *ppplogin* is the PPP logon ID.
Set PPP Password	Sets a PPP password if required by the specified dial-out entry.	setalertentry -index *entrynumber* -ppppassword *password* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *password* is the PPP password.

Examples:

- To configure the first alert entry to send only critical events and to dial out to a phone number, type

  ```
  setalertentry -index 1 -critcaleventseverity true -number "1-919-555-1212"
  ```

- To configure the seventh alert entry to use a director.all connection type and send all events, type

  ```
  setalertentry -index 7 -type director.all -criticaleventseverity false
  ```

Dial-in configuration commands

These commands display and set the dial-in (serial port) configuration, which provide information about who is allowed to dial in to the service processor.

Table 31. Dial-in configuration commands

Function	What it does	Command
Get dial-in entry configuration	Returns the configuration for the specified dial-in entry.	getdialinentry -index *entrynumber* where *entrynumber* is the number of the entry, which ranges from 1-12.
Set a login ID	Sets the logon ID for the specified dial-in entry.	setdialinentry -index *entrynumber* -id *loginid* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *loginid* is the login ID.
Set a login password	Sets the logon password for the specified dial-in entry.	setdialinentry -index *entrynumber* -password *loginpassword* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *loginpassword* is the password login ID.
Enable dial-back	Enables or disables the dial-back feature for the specified entry. Valid values are true and false.	setdialinentry -index *entrynumber* -dialback *enableflag* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *enableflag* is true to enable dial-back or false to disable dial-back.
Set a phone number	Sets the phone number for the specified entry for the service processor to dial back.	setdialinentry -index *entrynumber* -number *phonenum* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *phonenum* is dial-back phone number.

Table 31. Dial-in configuration commands (continued)

Function	What it does	Command
Enable write permissions	Enables or disables write permissions for the specified dial-in entry. Valid values are true and false.	setdialinentry -index *entrynumber* -readonly *enableflag* where: • *entrynumber* is the number of the entry, which ranges from 1-12. • *enableflag* is true to enable write permissions or false to disable write permissions.

Examples:

- To enable the dial-back feature and set the logon ID and password for the fourth alert entry, type

  ```
  setdialentry -index 4 -dialback true -id GayleM -password acegEGBDF
  ```

- To specify the dial-back phone number for the seventh alert entry, type

  ```
  setdialentry -index 7 -number "919-555-1212"
  ```

System power commands

Use these commands to read or set system power options such as server time-outs, remote power control, and preboot execution environment (PXE) reboot.

PXE reboot commands

These commands read and set the PXE boot flag on the target system.

Table 32. PXE reboot commands

Function	What it does	Command
Get PXE reboot state	Returns the PXE boot flag value in the BIOS code of the target system.	getpxe
Set PXE reboot state	Sets the PXE boot flag in the BIOS code of the target system.	setpxe -enabled *bootflag* where *bootflag* is true to enable PXE reboot or false to disable PXE reboot.

Example:

- To disable the PXE bootstrap, which enables the system to start locally, type

  ```
  setpxe -enabled false
  ```

Server timeout commands

Use these commands to read and set four server timeouts. Values for all timeout queries are in seconds.

Table 33. Server timeout commands

Function	What it does	Command
Get all server timeout	Returns information for all server timeouts.	getservertimeout
Get load timeout	Returns the number of 30-second intervals the service processor allows for the operating system to load. For example, if this command returns a value of 10, there are 10 intervals of 30 seconds in length. In this case, the operating system would have 300 seconds (5 minutes) to load.	getservertimeout -loader
Get heart beat timeout	Returns the timeout value for the heartbeat between the device driver and the service processor.	getservertimeout -os
Get POST timeout	Returns the number of 30-second intervals the service processor allows for POST to be completed.	getservertimeout -boot
Get shutdown timeout	Returns the number of seconds the service processor allows for the operating system to shut down.	getservertimeout -shutdown
Set load timeout	Sets the number of seconds the service processor allows for the operating system to load.	setservertimeout -loader *loadtime* where *loadtime* is the number of seconds. Valid value range is 0 to 7650 and must be a multiple of 30.
Set heartbeat timeout	Sets the timeout value, in seconds, for the heartbeat between the device driver and the service processor.	setservertimeout -os *heartbeattime* where *heartbeattime* is the number of seconds. Valid values are 0 or 10 to 255.
Set POST timeout	Sets the number of 30-second intervals the service processor allows for POST to be completed.	setservertimeout -boot *POSTtime* where *POSTtime* is the number of seconds. Valid values are 0 or 7 to 255.
Set shutdown timeout	Sets the number of seconds the service processor allows for the operating system to shut down.	setservertimeout -shutdown *shutdowntime* where *shutdowntime* is the number of seconds. Valid values are 0 or 30 to 36000.

Examples:

- To query the POST, load and heartbeat timeout values, type

  ```
  getservertimeout -post -loader -os
  ```

- To set the POST timeout to 5 minutes, the load timeout to 5 minutes, and the heartbeat timeout to 30 seconds, type

  ```
  setservertimeout -post 10 -loader 300 -os 30
  ```

Remote power control commands

These commands control remote power management to the system where the target service processor is located.

Table 34. Remote power control commands

Function	What it does	Command
Shutdown	Shuts down the system immediately.	poweroff -now
	Shuts down the system after the operating system shutdown timer has expired.	poweroff -shutdown
Restart	Shuts down the operating system and restarts the system immediately.	restart -now
	Shuts down the operating system and restarts the system after the operating system shutdown timer has expired.	restart -shutdown
Get start time and date	Returns the current power-on time and date of the system in the format of *xx*:*xx mm*/*dd*/*yyyy*.	getpowerontime or getpowerontime -query

Table 34. Remote power control commands (continued)

Function	What it does	Command
Set start options	Starts the system immediately.	setpowerontime -now
	Starts the system in the specified number of minutes.	setpowerontime -delay *minutes* where *minutes* is the number of minutes after the command is sent that the service processor starts the system (1 to 35791).
	Starts the system on the specified month, day, and year. The date is referred to by the internal clock of the service processor.	setpowerontime -date *mm/dd/yyyy* where *mm/dd/yyyy* is the month, day, and year that the service processor starts that system and is in the format *mm*/*dd*/*yyyy*
	Starts the system at the specified hour and minutes. Time is referenced by the internal clock of the service processor.	setrpowerontime -time *xx:xx* where *xx:xx* is the time in hours and minutes after the command is sent that the service processor starts that system in the 24-hour clock format of *xx*:*xx* .
	Clears the current date and time for the system to start.	setrpowerontime -clear

Examples:

- To query the current time and date that the system was turned on, type
  ```
  getpowerontime
  ```
- To start the system at 4:00 p.m. on 3 November 2003, type
  ```
  setpowerontime -time 16:00 -date 11/03/2003
  ```

System statistics commands

These commands display the system statistics, which are maintained by the service processor, set the blue indicator light function, and display light path diagnostics states.

Table 35. System statistics commands

Function	What it does	Command
Get all system statistics	Returns all available system statistics of the target service processor.	getsysstat
Get system operating status	Returns the system state. Valid values are: • In POST • Off/Unknown • Stopped • Booting • OS is running	getsysstat -state
	Returns the number of times the system has been restarted.	getsysstat -restart
	Returns whether the system power is on or off.	getsysstat -power
	Returns, in hours, how long the system has been powered on.	getsysstat -uptime
Get system UUID	Returns the system UUID.	getsysstat -uuid
Get blue indicator light status	Returns whether the blue indicator light state is on, off, or blink.	getsysstat -bluelight
Set blue indicator light status	Returns the state for the blue indicator light. These values are not case sensitive.	setsysstat -bluelight *state* where *state* is: • On • Off • Blink

Table 35. System statistics commands (continued)

Function	What it does	Command
Get light path diagnostics LEDs status	Returns all LEDs that are supported.	getlightpath -all
	Returns which LEDs are on.	getlightpath -on
	Returns which LEDs are off.	getlightpath -off
	Returns which LEDs are blinking.	getlightpath -blink
	Returns which LEDs are on the front panel.	getlightpath -frontpanel
	Returns which LEDs are diagnostic.	getlightpath -diags
	Returns which LEDs are in the specified state.	getlightpath -state *LEDstate* where *LEDstate* is the state of the LED: • On • Off • Blinking • All
	Returns whether the specified LED is on, off, or blinking. See Table 36 for a list of LEDnames.	getlightpath -location *"LEDname"* where *"LEDname"* is name of an LED. Make sure that the LED name is enclosed in quotation marks.

Table 36. LED names

cdrom cable location	pci bus location
center plane card location	pci card location
chip set vrm location	pci slot location
cpu location	processor blade location
cpu memory location	rear panel location
diagnostics panel location	scalibility port location
dmc backplane location	smp expansion module 1, chip set vrm location
fan location	smp expansion module 1, memory card location
add cable location	smp expansion module 1, memory hot plug location
front panel location	smp expansion module 1 location
memory card a location	smp expansion module 2 location
memory card b location	smp expansion module 3 location
memory location	smp expansion module 4 location
memsubsystem locationnative i/o	sp card location
card location	sp slot location
pci backplane a location	system board location
pci backplane b location	vrm location

Examples:

- To return statistics on the current state of the system, how many times the system has been restarted, and the number of hours the system has been running, type

  ```
  getsysstat -state -restart -uptime
  ```

- To get the location of the diagnostics LED panel, type

```
getlightpath -location "diagnostics panel location"
```

- To turn the blue indicator light on for a system, type

```
setsysstat -bluelight On
```

System component commands

These commands provide information about system components that the service processor is monitoring, such as memory, power supplies, hard disk backplanes, and microprocessors.

Power supply command

This command is supported only on xSeries 370 servers. This command monitors the power supplies.

Table 37. Power supply command

Function	What it does	Command
Get all power supply information	Returns values for all options if no option is used.	getfuelgauge
Power supplies supported	Returns the maximum number of power supplies supported by the system.	getps -max
Power supplies installed	Returns the number of power supplies installed in the system.	getps -inst
Available power supplies	Returns the number of available power supplies.	getfuelgauge -available
Failed power supplies	Returns the number of failed power supplies.	getfuelgauge -failed
Bit mask value	Returns a bit mask value for the fuel gauge status: 0 equals off, 1 equals on.	getfuelgauge -status
	Returns a bit mask value for a voltage fault: 0 equals off, 1 equals on.	getfuelgauge -voltagefault
Maximum current	Returns the maximum amount of current, in watts, drawn by the system.	getfuelgauge -maximumcurrent
Average current	Returns the average amount of current, in watts, drawn by the system.	getfuelgauge -averagecurrent
Power supply redundancy	Returns whether the system is operating with a redundant power supply.	getfuelgauge -redundancystatus
	Returns the number of power supplies required to provide power redundancy.	getfuelgauge -powersupplyrequired
	Returns whether the redundant power supply in the system is in use.	getfuelgauge -redundantpowersupply
	Returns whether the service processor is monitoring redundancy in the system.	getfuelgauge -monitorredundancy
Low-fuel threshold	Returns the low-fuel threshold, in watts.	getfuelgauge -lowfuelthreshold

Table 37. Power supply command (continued)

Function	What it does	Command
Power supply capacity	Returns the total number of power supplies that can be installed in the system.	getfuelgauge -powersupplycapacity
Maximum power	Returns the maximum available power, in watts, available to the system.	getfuelgauge -maxpower
Low-fuel state	Returns whether the system is operating in a low-fuel state.	getfuelgauge -lowfuelstate
Over power capacity	Returns whether the drawn current exceeds 100% of the capacity of the power supply in the system.	getfuelgauge -overcurrentstate

Examples:

- To query the number of available power supplies, the number of failed power supplies, and whether the server is in a low-fuel state, type

  ```
  getfuelgauge -available -failed -lowfuelstate
  ```

- To query the number of power supplies the system supports and the number installed, type

  ```
  getps -max -inst
  ```

Memory command

This command provides information about the memory (DIMMs) installed in a system that the service processor is monitoring.

Table 38. Memory command

Function	What it does	Command
Get all DIMM information	Returns all memory information.	getdimm
Get maximum DIMMs	Returns the maximum number of DIMMs supported by the system.	getdimm -max
DIMMs installed	Returns whether the specified DIMM is installed in the system.	getdimm -inst *DIMMnumber* where *DIMMnumber* is the number of the memory module. If a parameter is not specified, all installed DIMMs are returned.
Amount of RAM	Returns the amount of memory as RAM for the specified DIMM.	getdimm -memory *DIMMnumber* where *DIMMnumber* is the number of the memory module. If a parameter is not specified, the total of all DIMMs as RAM is returned.

Table 38. Memory command (continued)

Function	What it does	Command
Memory type	Returns the type of memory of the specified DIMM, such as SDRAM or EDO.	getdimm -type *DIMMnumber* where *DIMMnumber* is the number of the memory module. If a parameter is not specified, the type of all DIMMs is returned.
Memory configuration	Returns the memory configuration of the specified DIMM, such as nonparity, parity, or ECC.	getdimm -cfgtype *DIMMnumber* where *DIMMnumber* is the number of the memory module. If a parameter is not specified, the configuration of all DIMMs is returned.
Memory size	Returns the memory size, in MB, for the specified DIMM.	getdimm -size *DIMMnumber* where *DIMMnumber* is the number of the memory module. If a parameter is not specified, the size of all DIMMs is returned.
Memory frequency	Returns the frequency of the specified DIMM, in MHz.	getdimm -frequency *DIMMnumber* where *DIMMnumber* is the number of the memory module. If a parameter is not specified, the frequency of all DIMMs is returned.

Table 38. Memory command (continued)

Function	What it does	Command
Memory manufacturer	Returns the manufacturer ID for the specified DIMM.	getdimm -mfrid *DIMMnumber* where *DIMMnumber* is the number of the memory module. If a parameter is not specified, the manufacturer of all DIMMs is returned.
	Returns the manufacturer location for the specified DIMM.	getdimm -mfrlocn *DIMMnumber* where *DIMMnumber* is the number of the memory module. If a parameter is not specified, the manufacturer location of all DIMMs is returned.
Memory identification	Returns the part number of the specified DIMM.	getdimm -partnum *DIMMnumber* where *DIMMnumber* is the number of the memory module. If a parameter is not specified, the part number of all DIMMs is returned.
	Returns the serial number of the specified DIMM.	getdimm -serialnum *DIMMnumber* where *DIMMnumber* is the number of the memory module. If a parameter is not specified, the serial number of all DIMMs is returned.
	Returns the revision level for the specified DIMM.	getdimm -revision *DIMMnumber* where *DIMMnumber* is the number of the memory module. If a parameter is not specified, the revision level of all DIMMs is returned.

Examples:

- To query the part number, serial number, and manufacturer ID for the first DIMM, type

  ```
  getdimm -partnum 1 -serialnum 1 mfrid 1
  ```

- To query how many memory modules are installed, the total amount of RAM, and the type of memory, type

  ```
  getdimm -inst -memory -type
  ```

Hard disk command

This command returns information about the hard disk drives.

Table 39. Hard disk backplane command

Function	What it does	Command
SCSI buses supported	Returns the maximum number of SCSI buses supported by the system.	getdasd -buscount
Hard disks installed	Returns hard disk drives, by IDs, installed on the specified SCSI bus.	getdasd -scsiid *scsibus* where *scsibus* is the number of the SCSI bus.
	Returns the hard disk drives that are installed.	getdasd -installed
	Returns the hard disk drives that were installed when the service processor started.	getdasd -expected
Hard disks failed	Returns the hard disks that have failed.	getdasd -failed

Example:

- To query the number of installed hard disk drives and the IDs of the hard disk drives on the first SCSI bus, type

  ```
  getdasd -installed -scsiid 1
  ```

CPU command

This command returns information about each microprocessor (also called a CPU) that the service processor is monitoring.

Table 40. CPU command

Function	What it does	Command
Microprocessors supported	Returns the maximum number of microprocessors supported by the system.	getcpu -maxcpu
VRMs supported	Returns the number of VRMs installed in the system.	getcpu -maxvrm

Example:

- To query the number of installed VRMs and the number of supported microprocessors, type

  ```
  getcpu -maxvrm -maxcpu
  ```

Vital product data command

This command is used to read the vital product data (VPD) of various components of a system. Each command option returns values as specified in your specific system hardware manual or firmware update.

Table 41. VPD command

Function	What it does	Command
Get all VPD information	Returns all the VPD for all system components.	getvpd
Get VPD for system components	Returns VPD for the system POST/BIOS.	getvpd -postbois
	Returns VPD for the backplane installed in the system.	getvpd -planariobackplane
	Returns VPD for the I/O adapter installed in the system.	getvpd -planariocard
	Returns VPD for the system front panel.	getvpd -planarfrontpanel
	Returns VPD for the memory adapter installed in the system.	getvpd -planarmemorycard
	Returns VPD for the midplane.	getvpd -midplane
	Returns VPD for the service processor hardware.	getvpd -mphardware
	Returns VPD for the main ROM on the service processor.	getvpd -mprom
	Returns VPD for the boot code of the service processor.	getvpd -mpboot
	Returns VPD for the service processor device driver.	getvpd -mpdevicedriver
	Returns VPD for the hard disk drives installed in the system.	getvpd -dasd
	Returns VPD for the microprocessors installed on the system board.	getvpd -cpu
	Returns VPD for the system board.	getvpd -planar
	Returns VPD for the power supplies installed in the system.	getvpd -ps
	Returns VPD for the system power backplane.	getvpd -powerbackplane
	Returns VPD for the voltage regulator monitor (VRMs) installed in the system.	getvpd -vrm
	Returns VPD for the drawer management controller installed in the system.	getvpd -dmc

Example:

- To query the VPD for the service processor hardware, the installed hard disk drives, and the installed microprocessors, type

```
getvpd -mphardware -dasd -cpu
```

Component activity log command

This command returns entries that are written in the component activity log.

Table 42. Component activity log command

Function	What it does	Command
Get component activity log entries	Returns the 10 most recent entries in the component activity log.	getcomplog
	Resets the component activity log to the beginning.	getcomplog -reset
	Returns the next 10 entries in the component activity log.	getcomplog -next

Example:

- To query the next 10 entries in the component activity log, type

```
getcomplog -next
```

System health and environment commands

These commands provide system health and environment information, such as voltages, temperatures, and fan speeds. You can access the current values and thresholds for the system hardware that the service processor is monitoring.

Temperature commands

The following commands read the temperatures that are monitored by the service processor and return the values as currently monitored temperatures. Temperature queries are divided into two commands: current temperature values and temperature thresholds.

Table 43. Temperature commands

Function	What it does	Command
Get all temperatures	Returns all the current temperatures the service processor is monitoring. All values are reported in Celsius.	gettemperatures If parameters are not specified, every temperature for which access exists is returned.
Get microprocessor temperature	Returns the current microprocessor temperature.	gettemperatures -cpu *cpunumber* where *cpunumber* is the number of the microprocessor. If a parameter is not specified, current temperatures of all microprocessors are returned.
Get hard disk temperatures	Returns the current hard disk drive temperature.	gettemperatures -dasd *dasdnumber* where *dasdnumber* is the number of the hard disk drive. If a parameter is not specified, current temperatures of all hard disk drives are returned.
Get center card temperatures	Returns the current center card temperature.	gettemperatures -planarcenter

Table 43. Temperature commands (continued)

Function	What it does	Command
Get ambient system temperatures	Returns the current ambient system temperature.	gettemperatures -systemambient
Get service processor temperatures	Returns the current service processor temperature.	gettemperatures -managementprocessor
Get power supply temperatures	Returns the current power supply unit temperature.	gettemperatures -ps *psnumber* where *psnumber* is the number of the power supply unit. If a parameter is not specified, current temperatures of all power supply units are returned.
Get microprocessor area temperatures	Returns the current microprocessor area ambient temperature.	gettemperatures -planarcpu
Get PCI area temperatures	Returns the current PCI area ambient temperature.	gettemperatures -planarpci
Get I/O area temperatures	Returns the current I/O area ambient temperature.	gettemperatures -planario
Get all temperature thresholds	Returns all the thresholds for each temperature the service processor is monitoring.	gettemperaturethesholds If parameters are not specified, every temperature threshold that exists is returned.
Get microprocessor temperature thresholds	Returns the current microprocessor temperature threshold.	gettemperaturethesholds -cpu *cpunumber* where *cpunumber* is the number of the microprocessor. If a parameter is not specified, current thresholds of all microprocessors are returned.
Get hard disk temperature thresholds	Returns the current hard disk drive temperature threshold.	gettemperaturethesholds -dasd
Get center card temperature thresholds	Returns the current center card temperature threshold.	gettemperaturethesholds -center
Get ambient system temperature thresholds	Returns the current system ambient temperature threshold.	gettemperaturethesholds -ambient
Get service processor temperature thresholds	Returns the current service processor temperature threshold.	gettemperaturethesholds -managementprocessor

Examples:

- To query the temperature thresholds for the second service processor and the center card, type

  ```
  gettemperaturethresholds -cpu 2 -planarcenter
  ```

- To query the temperature of the second power supply, the service processor, and the first hard disk, type

  ```
  gettemperatures -ps 2 -managementprocessor -dasd 1
  ```

Voltage commands

Voltages are maintained by the service processor. Querying the voltages returns read-only values in volts. Voltage queries are divided into two commands: current voltage measurements and voltage thresholds.

Table 44. Voltage commands

Function	What it does	Command
Get all voltages	Returns all the current voltages the service processor is monitoring.	getvoltages
Get VRM voltages	Returns the current voltage of a voltage regulator module (VRM).	getvoltages -vrm *vrmnumber* where *vrmnumber* is the number of the VRM. If a parameter is not specified, the current voltages of all VRMs are returned.
Get system board voltages	Returns the current voltages of all system boards. To get a list of supported voltages, run the getvoltages -sbvolts command.	getvoltages -systemboard *sbvoltage* where *sbvoltage* is a voltage value. If a parameter is not specified, the voltages of all system boards are returned.
Get power supply voltages	Returns the current voltage of a power supply.	getvoltages -ps *psnumber* where *psnumber* is the number of the power supply. If a parameter is not specified, current voltages of all power supplies are returned.
Get supported system board voltages	Returns the supported system board voltages.	getvoltages -sbvolts
Get supported power supply voltages	Returns the supported power supply voltages.	getvoltages -psvolts *powersupply* where *powersupply* is the number of the power supply.
Get all voltage thresholds	Returns all the current voltage thresholds the service processor is monitoring.	getvoltagethresholds If a parameter is not specified, every voltage threshold that exists is returned.

Table 44. Voltage commands (continued)

Function	What it does	Command
Get system board voltage thresholds	Returns the system board voltage threshold for the specified voltage setting. To get a list of supported voltages, run the getvoltages -sbvolts command.	getvoltagethresholds -systemboard *sbvoltage* where *sbvoltage* is a voltage value.

Examples:

- To query the values for the first VRM and the system board record for -5 volts, type

  ```
  getvoltages -vrm 1 -systemboard N5V
  ```

- To query the system board threshold for 12 volts, type

  ```
  getvoltagethresholds -systemboard 12V
  ```

Fan commands

Fans monitored by a service processor can be queried for status such as fan speed, installed fans, and fans that are running.

Table 45. Fan commands

Function	What it does	Command
Get fan speeds	Returns information for all fans the service processor is monitoring: • Fan speeds • Fan rotations per minute (RPMs) • Number of fans supported • Number of fans installed • List of currently installed • List of fans initially installed	getfan If a parameter is not specified, information for all fans for which access exists is returned.
	Returns the fan speed for each fan the service processor has access to and is monitoring. The values are a percentage of maximum RPM.	getfan -speeds *fannumber* where *fannumber* is the number of the fan. If a parameter is not specified, the current speed of all fans is returned.
	Returns the fan speed for each fan the service processor has access to and is monitoring. The values are raw RPM.	getfan -rpm *fannumber* where *fannumber* is the number of the fan. If a parameter is not specified, the current RPM of all fans is returned.
Get number of fans supported	Returns the maximum number of fans supported by the system.	getfan -num

Table 45. Fan commands (continued)

Function	What it does	Command
Get number of fans installed	Returns the number of fans currently installed in the system.	getfan -inst
	Returns the number of fans initially installed in the system.	getfan -expected
Get number of fans running	Returns a list of operational fans.	getfan -run

Examples:

- To query the speed of the first fan as a percentage of the maximum RPM, type
  `getfan -speeds 1`
- To query how many fans are installed and how many fans are working, type
  `getfan -inst -run`

Using scripts

You can use scripts instead of typing each command, one at a time, to accomplish a specific task or set of tasks. For example, to make logging in to a service processor easier, you can create a logon script to avoid repeatedly typing the host name, user ID, and password.

You can create a script in any word processor application, and it can have any file name you choose. You can name your scripts so you can easily keep track of their usage. For example, a logon script might be named *hostname_myname*.loginSP, where *hostname* is your system name and *myname* is your user ID.

Table 46. Scripting commands

Function	What it does	Command
Write to a script file	Captures all subsequent commands and stores them in the specified output file.	commandfile *outputfilename* where *outputfilename* is the relative path and name of the output file.
Stop capturing commands	Stops capturing commands.	commandfile
Run a script file	Reads the command set from the specified input file.	inputfile *inputfilename* where *inputfilename* is the relative path and name of the input file.
Write command output to a file	Redirects the output of subsequent commands to the specified file rather than the command window. As the commands run, the command window remains blank because all results, even failures, are captured in the output file.	outputfile *outputfilename* where *outputfilename* is the relative path and name of the output file.

Table 46. Scripting commands (continued)

Function	What it does	Command
Reset output to prompt window	Returns command output back to the command window from the output file specified in the outputfile command.	resetoutput

Examples:

- To capture subsequent commands and store them in a specified script file, type

  `commandfile` **`c:\MPCLI\scripts\eventscripts.txt`**
- To run a specified script file, type

  `inputfile` **`c:\MPCLI\scripts\logonscript.txt`**
- To send the output of subsequent commands to a specified file rather than viewing the output in the command window, type

  `outputfile` **`c:\MPCLI\commandresults\VPDquery.txt`**

For script file examples, see Appendix A, “Sample scripts”, on page 67.

Appendix A. Sample scripts

This appendix contains some sample scripts. You can use these scripts, modify them to suit your needs, or refer to them when you are creating your own scripts.

In the following sample scripts, the items in **bold** type are examples only. The parameters that you choose to use will be different.

Get and set network hardware configuration

See “Network hardware configuration commands” on page 29 for command variables and valid values used in the following sample script.

```
outputfile ./enetcfgresults.txt
getmpid
getmpclock
setnethw -interface 1 -enabled false
setdhcp -enabled false
setnethw -interface 1 -linetype ″ENET″ -enabled true
setip -interface 1 -hostname X
setip -interface 1 -ipaddress 9.67.37.00
setip -interface 1 -subnet 255.255.255.128
setnethw -interface 1 -datarate ″AUTO″
setnethw -interface 1 -duplex ″AUTO″
setnethw -interface 1 -adminmac ″00 00 00 00 00 00″
setnethw -interface 1 -gateway 9.67.37.1
setnethw -interface 1 -enabled true
resetoutput
restartmp
```

Log on to and get service processor information

See “Logging on to a service processor” on page 11 for command variables and valid values used in the following sample script.

```
outputfile ./getaccess.txt
logonip -hostname SPbatman -userid gisellem -password s0ngb1rd
getmpid -text
getmpclock -timeanddate
getdialinentry -index 12
logoff
exit
```

Get service processor information and log

See “Service processor event commands” on page 40 for command variables and valid values used in the following sample script.

```
outputfile ./mplog.txt
getmpid
getmpclock
getmplog -first
getmplog -all
resetoutput
```

Get and set various policies and set start options

See “Blade server policy commands” on page 17 for command variables and valid values used in the following sample script.

```
getpbpolicy -localpower 2
setpbpolicy -localpower 2,false
getpbpolicy -localpower 2
setpbpolicy -localpowerall true
getpbpolicy -localpower 2

getpbpolicy -localkvm 2
setpbpolicy -localkvm 2,false
getpbpolicy -localkvm 2
setpbpolicy -localkvmall true
getpbpolicy -localkvm 2

getpbpolicy -localusb 2
setpbpolicy -localusb 2,false
getpbpolicy -localusb 2
setpbpolicy -localusball true
getpbpolicy -localusb 2

bootoptions -get 2
bootoptions -set 2,"pxe,cdrom,floppy"
bootoptions -get 2

getkvm
setkvm -owner 2
getkvm
setkvm -park
getkvm
```

Log on to and flash service processor

See “Vital product data command” on page 60 for command variables and valid values used in the following sample script.

```
outputfile ./rsaflash.txt
logonip -hostname 192.168.1.100 -userid gisellem -password s0ngb1rd
getmpid -text
getvpd -mpboot
getvpd -mprom

fwupdate -mn d:\firmware\x220\batman\CNETMNUS.PKT
logoff
sleep 15000
logonip -hostname 192.168.1.100 -userid gisellem -password s0ngb1rd
fwupdate -br d:\firmware\x220\batman\CNETBRUS.PKT
logoff
sleep 15000
logonip -hostname 192.168.1.100 -userid gisellem -password s0ngb1rd
fwupdate -vnc d:\firmware\x220\batman\CNETRGUS.PKT
logoff
exit
```

Log on to and set dial-in configuration

See “Dial-in configuration commands” on page 48 for command variables and valid values used in the following sample script.

```
outputfile ./setaccess.txt
logonip -hostname svcprocella -userid gisellem -password s0ngb1rd
getmpid -text
getmpclock -timeanddate
setdialinentry -index 12 -id gisellem -password s0ngb1rd -readonly false
logoff
exit
```

Set SNMP configuration

See “SNMP configuration commands” on page 23 for command variables and valid values used in the following sample script.

```
outputfile ./snmpcfgresults.txt
getmpid
getmpclock
setsnmp -agent false
setsnmp -contactname "Mark Oniram"
setsnmp -contactlocation "Service Processor Lab"
setsnmp -communityname 0,Public
setsnmp -ipaddress 1,1,9.37.113.211
setsnmp -ipaddress 1,2,0.0.0.0
setsnmp -ipaddress 1,3,0.0.0.0
setsnmp -communityname 0,0
setsnmp -ipaddress 2,1,0.0.0.0
setsnmp -ipaddress 2,2,0.0.0.0
setsnmp -ipaddress 2,3,0.0.0.0
setsnmp -communityname 0,0
setsnmp -ipaddress 3,1,0.0.0.0
setsnmp -ipaddress 3,2,0.0.0.0
setsnmp -ipaddress 3,3,0.0.0.0
setsnmp -agent true
resetoutput
```

Get and set switch module configuration

See “Switch module configuration commands” on page 18 for command variables and valid values used in the following sample script.

```
getsmnetwork -currentconfig 1
getsmnetwork -currentmethod 1
getsmnetwork -pendingconfig 1
getsmnetwork -pendingmethod 1
switchmodule -getpoweron 1
switchmodule -getmemdiagson 1
switchmodule -getcfgotherports 1
switchmodule -getextportson 1
switchmodule -ping 1
switchmodule -getpostresults 1
setsmnetwork -ipaddress 1, 192.168.1.125
setsmnetwork -gateway 1, 192.168.1.126
```

setsmnetwork -subnet **1,255.255.255.0**
setsmnetwork -method **1**, ″**static**″
setsmnetwork -pending **1**,**false**
setsmnetwork -pending **1**,**true**
switchmodule -setpoweron **1**,**true**
switchmodule -setmemdiagson **1**,**true**
switchmodule -setcfgotherports **1**,**true**
switchmodule -setextportson **1**,**true**
switchmodule -ping **1**
switchmodule -getpostresults **1**

Get VPD

See “Vital product data command” on page 60 for command variables and valid values used in the following sample script.

outputfile **./vpd.txt**
getmpid
getmpclock
getvpd -postbios
getvpd -planariobackplane
getvpd -planariocard
getvpd -planarfrontpanel
getvpd -planarmemorycard
getvpd -planarcpu
getvpd -mphardware
getvpd -mprom
getvpd -mpboot
getvpd -mpdevicedriver
getvpd -dasd
getvpd -cpu
getvpd -ps

Appendix B. Getting help and technical assistance

If you need help, service, or technical assistance or just want more information about IBM products, you will find a wide variety of sources available from IBM to assist you. This appendix contains information about where to go for additional information about IBM and IBM products, what to do if you experience a problem with your xSeries or IntelliStation® system, and whom to call for service, if it is necessary.

Before you call

Before you call, make sure that you have taken these steps to try to solve the problem yourself:

- Check all cables to make sure that they are connected.
- Check the power switches to make sure that the system is turned on.
- Use the troubleshooting information in your system documentation, and use the diagnostic tools that come with your system.
- Go to the IBM Support Web site at http://www.ibm.com/pc/support/ to check for technical information, hints, tips, and new device drivers.
- Use an IBM discussion forum on the IBM Web site to ask questions.

You can solve many problems without outside assistance by following the troubleshooting procedures that IBM provides in the online help or in the publications that are provided with your system and software. The information that comes with your system also describes the diagnostic tests that you can perform. Most xSeries and IntelliStation systems, operating systems, and programs come with information that contains troubleshooting procedures and explanations of error messages and error codes. If you suspect a software problem, see the information for the operating system or program.

Using the documentation

Information about your IBM xSeries or IntelliStation system and preinstalled software, if any, is available in the documentation that comes with your system. That documentation includes printed books, online books, readme files, and help files. See the troubleshooting information in your system documentation for instructions for using the diagnostic programs. The troubleshooting information or the diagnostic programs might tell you that you need additional or updated device drivers or other software. IBM maintains pages on the World Wide Web where you can get the latest technical information and download device drivers and updates. To access these pages, go to http://www.ibm.com/pc/support/ and follow the instructions. Also, you can order publications through the IBM Publications Ordering System at http://www.elink.ibmlink.ibm.com/public/applications/publications/cgibin/pbi.cgi.

Getting help and information from the World Wide Web

On the World Wide Web, the IBM Web site has up-to-date information about IBM xSeries and IntelliStation products, services, and support. The address for IBM xSeries information is http://www.ibm.com/eserver/xseries/. The address for IBM IntelliStation information is http://www.ibm.com/pc/intellistation/.

You can find service information for your IBM products, including supported options, at http://www.ibm.com/pc/support/.

Software service and support

Through IBM Support Line, you can get telephone assistance, for a fee, with usage, configuration, and software problems with xSeries servers, IntelliStation workstations, and appliances. For information about which products are supported by Support Line in your country or region, go to http://www.ibm.com/services/sl/products/.

For more information about Support Line and other IBM services, go to http://www.ibm.com/services/, or go to http://www.ibm.com/planetwide/ for support telephone numbers. In the U.S. and Canada, call 1-800-IBM-SERV (1-800-426-7378).

Appendix C. Notices

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

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

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this publication to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product, and use of those Web sites is at your own risk.

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

Some software may differ from its retail version (if available) and may not include all user manuals or all program functionality.

IBM makes no representations or warranties regarding third-party products or services.

Edition notice

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Trademarks

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

BladeCenter	IntelliStation
Chipkill	Netfinity
e-business logo	Predictive Failure Analysis
@server	xSeries
IBM	Wake on LAN

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

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

Index

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

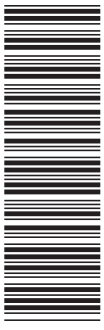
U

V

W

Part Number: 90P2900

Printed in U.S.A.



(1P) P/N: 90P2900

SC90-P290-00