bullx B5xx system

Hardware Management CLI Reference Guide

extrem computing

bullx B5xx system

Hardware Management CLI Reference Guide

Hardware

June 2011

Trademarks and Acknowledgements

We acknowledge the right of proprietors of trademarks mentioned in this manual.

All brand names and software and hardware product names are subject to trademark and/or patent protection.

Quoting of brand and product names is for information purposes only and does not represent trademark misuse.

Table of Contents

Preface

This guide describes the Hardware Management CLI (Command Line Interface) for bullx blade systems.

Note	The Bull Support Web site may be consulted for product information, documentation, updates and service offers:
	http://support.bull.com

Intended Readers

This guide is intended for System Administrators and Operators in charge of managing systems locally and remotely and of writing server management batch scripts.

Highlighting

The following highlighting conventions are used in this guide:

Bold	Identifies the following:
	• Interface objects such as menu names, labels, buttons and icons.
	• File, directory and path names.
	• Keywords to which particular attention must be paid.
Italics	Identifies references such as manuals or URLs.
`monospace`	Identifies portions of program codes, command lines, or messages displayed in command windows.
< >	Identifies parameters to be supplied by the user.

```
Commands entered by the user
-----------------------------------------------------------------------
System messages displayed on the screen
-----------------------------------------------------------------------
```

⚠ **WARNING**
A *Warning* notice indicates an action that could cause damage to a program, device, system, or data.

Related Publications

This list is not exhaustive. Useful documentation is supplied on the Resource & Documentation CD(s) delivered with your system. You are strongly advised to refer carefully to this documentation before proceeding to configure, use, maintain, or update your system.

- *Site Preparation Guide,* 86 A1 40FA
  explains how to prepare a Data Processing Center for Bull Systems, in compliance with the standards in force. This guide is intended for use by all personnel and trade representatives involved in the site preparation process.

- *bullx B500 blade system Installation Guide*, 86 A1 48FB
  explains how to install and start the system for the first time. This guide is intended for use by qualified support personnel.

- *bullx B500 blade system Service Guide*, 86 A7 51FB
  explains how to service the system. This guide is intended for use by qualified support personnel.

- *bullx blade system Chassis Hardware Console User's Guide*, 86 A1 50FB
  explains how to use the Chassis Hardware Console (CHC) to manage the system. This guide is intended for use by blade system administrators and operators.

- *bullx B500 Blade Hardware Console User's Guide*, 86 A1 49FB explains how to use the bullx B500 compute blades. This guide is intended for use by customer administrators and operators.

- *bullx B505 Blade Hardware Console User's Guide*, 86 A1 49FE explains how to use the bullx B505 accelerator blades. This guide is intended for use by customer administrators and operators.

- *bullx B510 Blade Hardware Console User's Guide*, 86 A1 49FG explains how to use the bullx B510 dual-nodes blades. This guide is intended for use by customer administrators and operators.

- *Resource and Documentation CD*
  contains the tools and documentation required to configure, operate and maintain the equipment.

Introduction

The Hardware Management CLI provides an easy Command Line Interface (CLI) for local or remote hardware management and automation scripts. It is designed for:

- NovaScale 3005, 4000, 5005 & 9006 series, NS R400 series, NS T800 series, Express 5800 servers, bullx S series, bullx B series, bullx R series, Blade servers and Escala Privilege Line servers.

- A32 or IA64 architectures,

- Linux systems ONLY.

The Hardware Management CLI uses both Out-of-Band and In-Band connections to the target hosts.

The Out-Of-Band network protocols used for these connections are dependant on server type and are based on IPMIoverLAN, SNMP, web services or SSH associated with identity key files (for silent commands), for examples

- IPMIoverLAN for NovaScale R400 & T800 series, ns bullion series, bullx series and all servers using the ipmilan or lanplus model.

- PAM Web services for NovaScale 5005 & 6000 series.

- SNMP requests to the Chassis Management Module for NS Blade servers, EL Blade servers, bullx B series.

- SNMP requests to the SNMP agent of Power Distribution Units (PDU).

- SSH connection to Harware Management Console (HMC) for Escala PL servers. Escala PL servers remote management with HMC requires:

    - identity key files to be installed on the HMC

    - the presence of a SSH client configuration file to perform non-prompt ssh connection to the HMC

    - locally, key files and configuration file (if any) must be in the same directory.

Most of the commands may also use an InBand connection for local Hardware Management of NovaScale servers running Linux and managed via IPMI using the ipmidrv model.

Most of the commands may also use authentication file. This is why the following packages have to be installed:

- perl-XML-LibXML,

- perl-XML-LibXML-Common,

- perl-XML-SAX,

- perl-XML-NamespaceSupport.

There are installed in the the **bin** subdirectory of the **BSMHW** package install directory (Default: **/opt/BSMHW/bin**).

Password specification

All password arguments may be either specified by **-p password** or extracted from the internal authentication file. This file is created with the **bsmcfg_auth_cmd.sh** command, detailed in Appendix B.

Model requirements

The Hardware Management CLI is designed for range of server types, as listed previously. The protocol used to connect the target hosts is given via **-m model** parameter.

Although very similar, the some models have some mandatory arguments.

ipmilan model

- Model used for Cool Cabinet Door and novascale server using IMPI.
- The **user** and **password** arguments are NOT mandatory.
- If **-p password** IS NOT specified and **-u user** IS specified, the internal authentication file is searched for the password, which may be there.

lanplus model

- Model used for Cool Cabinet Door and the novascale server that uses IPMI.
- The **user** argument is mandatory.
- If **-p password** is not specified, the internal authentication file is searched for the password, which may be there.
- IPMI connection is established with the lanplus interface using cipher suite IDs for authentication, integrity, and encryption algorithms. The default cipher suite ID is 3 and specifies the RAKP-HMAC-SHA1 authentication, HMAC-SHA1-96 integrity, and AES-CBC-128 encryption algorithms.

ipmidrv model

- Model used for the server with the ipmidrv driver.
- IPMI management of a local system requires the installation and configuration of the Linux OpenIPMI driver and the installation of the OpenIPMI library.
- The **user**, **password** and **host** arguments MUST NOT be specified for local hardware management via IPMI.

CMC model

- Model used for bullx chassis.
- None specific arguments.

blade_CMC model

- Model used for bullx blade servers.
- None specific arguments.

IPMITOOL Configuration

It is possible to change some IPMITOOL configuration parameters:

- CYPHER_SUITE_ID: this parameter specifies the authentication, integrity and confidentiality algorithms used when connecting via the lanplus interface. The default value is 3.

- ME_CHANNEL: this parameter specifies the destination channel for IPMI requests. The default value is 0x06.

- ME_ADDRESS: this parameter specifies the IPMB address for ipmi requests. The default value is 0x2c.

The parameters to change are in the bsmcli.cfg.custom.sh.tmpl file.

1. Rename the file, as shown, so that it can be called by the BSM CLI scripts.

```
$ cd /opt/BSMHW/bin
$ mv bsmcli.cfg.custom.sh.tmpl bsmcli.cfg.custom.sh
```

2. Edit the bsmcli.cfg.custom.sh file, modifying the values you want to change.

3. Save the file.

Chapter 1. CLI command for bullx blades

This chapter describes Hardware Management CLI available for bullx blade systems:

1.1. bsmBladepower

bsmBladepower.sh is used to perform bullx blades POWER ON/OFF operations and to check the power state by using the Chassis Controller Manager.

```
bsmBladepower.sh -a on
bsmBladepower.sh -a off
bsmBladepower.sh -a soft
bsmBladepower.sh -a awake
bsmBladepower.sh -a asleep
bsmBladepower.sh -a status
```

This command is asynchronous and returns the prompt before the end of command completion.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

A power ON command on an already powered ON system and/or a power OFF command on an already powered OFF system may return 0 (command accepted).

1.1.1. Notes

- In order to power on/off a bullx blade, the system running bsmBladepower must be declared on the target server as a member of a SNMPv1 read-write community. The default read-write community name used by bsmBladepower is **private**.

- For the Power Off command to be taken into account on a remote host running Windows 2000 / 2003 Server, the **Shutdown: Allow system to be shut down without having to log on security** option must be enabled on the remote host. To configure this security option:

    a. Click **Start > Run**.

    b. In the **Open** box, type **gpedit.msc** and click OK.

    c. In the **Group Policy** window, expand **Computer Configuration\Windows\ Settings\Security Settings\Local\Policies\Security Options**.

    d. Set the **Shutdown Security** option to **enabled**.

1.1.2. Options

-h	Displays command usage.
-l level	Specifies the severity of the messages to be logged in syslog. 0: no message is logged (default). 1: FAILED completion only is logged. 3: FAILED and SUCCESSFUL completion are logged.
-q	Quiet option: stops all outputs (stdout, sdterr, or outfile if -o option is specified).
-a action	Action: on, off, soft, awake, asleep, status, restart (Escala PL servers).
-a action	Action: on, off, off_force, status.
-H host	bullx chassis CMC IP address if model is blade CMC.
-m model	Target system model: blade_CMC
-I bladeIndex	Blade index (blade slot).
-S SNMPv	SNMP version: 1 (default value), 2, 3.
-C community	SNMP community (default value is **public** for the **status** action and **private** for the other actions).
-P port	SNMP port (default value is 161).
-T timeout	SNMP timeout (default value is 20s). SSH connection timeout (default value is 30s).
-o outfile	Output file name to which stdout and stderr outputs are redirected.

1.1.3. Usage

1.1.3.1. Power On

Command

```
$ bsmBladepower.sh -a on -m blade_CMC -H 192.168.200.10 -I 7 -U community
```

Output
```
-------------------------------------------------------------------------------
Server blade powered ON
-------------------------------------------------------------------------------
```

1.1.3.2. Soft Power Off

Command

```
$ bsmBladepower.sh -a soft -m blade_CMC -H 192.168.200.10 -I 7 -U community
```

Output
```
-------------------------------------------------------------------------------
Server blade soft powered OFF
-------------------------------------------------------------------------------
```

1.1.3.3. Power Off

Command

```
$ bsmBladepower.sh -a off -m blade_CMC -H 192.168.200.10 -I 7 -U community
```

Output
```
-------------------------------------------------------------------------------
Server blade powered OFF
-------------------------------------------------------------------------------
```

1.1.3.4. Soft Power Awake

Command

```
$ bsmBladepower.sh -a awake -m blade_CMC -H 192.168.200.10 -I 7 -U community
```

Output
```
-------------------------------------------------------------------------------
Server blade powered awake
-------------------------------------------------------------------------------
```

1.1.3.5. Power Asleep

Command

```
$ bsmBladepower.sh -a asleep -m blade_CMC -H 192.168.200.10 -I 7 -U community
```

Output
```
-------------------------------------------------------------------------------
Server blade powered asleep
-------------------------------------------------------------------------------
```

1.1.3.6. Display Power Status

The user can check the server blade power state using the status option.

Note	Power status represents the electrical status and not the functional status of the system. For example, a system may be powered ON but not RUNNING, due to a BIOS initialization error.

Command

```
$ bsmBladepower.sh -a status -m blade_CMC -H hwtest2 -I 7
```

Output

```
-------------------------------------------------------------------------
Server blade power is ON
-------------------------------------------------------------------------
```

Command

```
$ bsmBladepower.sh -a status -m blade_CMC -H hwtest2 -I 7
```

Output

```
-------------------------------------------------------------------------
Server blade power is standby-on
-------------------------------------------------------------------------
```

Return Codes

0	Blade power is ON.
1	Blade power is OFF (OFF, standby-off, standby-on).
-1	Command failed.

1.2.   bsmevent

bsmevent.sh is used to add the events attached to a sensor to the SEL repository.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

1.2.1.   Options

–h	Displays command usage.
–l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
-a action	Action: **setEvent**.
–m model	Target server model: **lanplus**, **ipmilan**.
–H host	BMC target IP address.
–u user	BMC user.
–p password	User password.
-s sensorID	Sensor identifier (put between double-quote).
-b state	State for the sensor identifier (default is list). Possible states are: **list, lnr, lcr, lnc, unc, ucr, unr**.
-e eventDir	Direction taken by the state (default is assert).
-i i2csaAdr	Address of the device generating the event (default is the BMC address).
–Y cypher	Cipher suite ID to be used by lanplus interface (default is 3).
–o outfile	Output file name to which **stdout** and **stderr** outputs are redirected.

1.2.2.   Usage: Server using ipmilan or lanplus model

Syntax
```
bsmevent.sh [-h] [-l level] -a action -m model -H host [-u user]
[-p password] -s sensorID [-b state] [-e eventDir] [-i i2csaAdr]
[-Y cipher] [-o outfile]
```

1.2.2.1.   Insert events attached to a sensor to the SEL repository

The **setEvent** action is used to insert events attached to a sensor to the SEL repository.

Command
```
$ bsmEvent.sh -m ipmilan -H 129.182.6.151 -u MWA -p guest
-a setEvent -s "Rear Fan" -b ucr
```

Output
```
-----------------------------------------------------------------------
Finding sensor Rear Fan... ok
0x0000 | Pre-Init Time-stamp   | Fan #0200#40 | Upper Critical going high
#590100
-----------------------------------------------------------------------
```

1.3. bsmfru

bsmfru displays the FRU contents using the standard output.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

The FRU is displayed in stdout, as follows:

<fru tag>: <fru content>

1.3.1. Notes

- Servers are identified by their DNS name or IP address (-H parameter).

1.3.2. Options

–h	Displays command usage.
–l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
–H host	DNS name or target server IP address, CMC IP address for bullx chassis, or bullx blade IP address
–D domain	Target server domain name (NovaScale 5005 & 6000 series only).
–m model	Target server model: fame, nec, ipmilan, lanplus, tiger, ipmidrv, blade, HMC, CMC, blade_CMC.
–M hwmanager	DNS name or PAP Manager IP address (NovaScale 5005 or 6000 series) or NS and EL Blade server manager IP address, or HMC IP address for PL servers.
–u user	PAP administrator for NovaScale 5005 or 6000 series. SMU user on the target server for NovaScale 4000 series.
	BMC user for NovaScale target servers using the ipmilan model, bullx chassis, bullx blades.
	User for ssh connection to the HMC for PL servers.
–I idfile	Identity key file used to perform connection to the HMC with ssh.
-C configfile	ssh client configuration file used to perform non-prompt connection to the HMC.
-d cfgdir	Directory where identity key file and ssh client configuration file are installed (default value is../etc/ssh).
-p password	User password. (Authentication key for Express5800 servers).
–Y cipher	Cipher suite ID to be used by lanplus interface (default is 3).
–B targetname	Blade name (for Blade servers), or managed system name (CECname) for PL servers.
–S SNMPv	SNMP version: 1 (default value), 2, 3.
-C community	SNMP community (default value **public**).
–P port	SNMP port (default value is 161).
-T timeout	SNMP timeout (default value is 20).
	SSH connection timeout (default value is 30s).
–o outfile	Output file name where stdout and stderr outputs are redirected.
–e ALL	Displays ALL the FRUs linked to the modules on which the domain is defined.

1.3.3. Usage: bullx blade chassis

Syntax

```
bsmfru.sh [-h] -m CMC -H host [-u user] [-p password] [-o outfile]        [-l
level]
```

1.3.3.1. Display FRUs

Command

```
$ bsmfru.sh -m CMC -H 129.184.238.254 -u root -p pass
```

Output

```
--------------------------------------------------------------------------
FRU System Description :

FRU System Description : CMB_Chassis (ID 2 on MC 0x20)
Board Mfg Date   : Sat Mar 14 11:57:00 2009
Board Mfg        : Sanmina-SCI
Board Product    : LCP chassis
Board Serial     : E86C0910014
Board Part Number: WP11011500-C
Board Extra      : 0
Board Extra      : 0
Board Extra      : DVT
Board Extra      : 0
Board Extra      : WP11011500-C
Board Extra      : BULL


--------------------------------------------------------------------------
```

1.3.4. Usage: bullx blade servers

Syntax

```
bsmfru.sh [-h] -m blade_CMC -H host [-u user] [-p password] [-o outfile]  [-l
level]
```

1.3.5. Usage: Server using ipmidrv model

Syntax

```
bsmfru.sh [-h] -m ipmidrv [-o outfile] [-l level]
```

1.4. bsmFWupg

bsmFWupg is used to upgrade, using an external tool, the firmware of Remote Management Modules. Operations that can be done:

```
bsmFWupg -a bulloemver
bsmFWupg -a bulloemupg
bsmFWupg -a bulloemlisthw
```

bulloemver: display the component firmware version.

bulloemupg: upgrade the the component firmware.

bulloemlisthw: display the list of components which can be upgraded (currently limited to module 0).

This command is available only for ipmilan model NovaScale servers with Raritan Remote Management Modules that can be manipulated with the **KiraTool** command line tool, and on Bull servers which use the **bulloem** command extension for **ipmitool**.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

1.4.1. Notes

- The **KiraTool** software, that allows you to manipulate the module's firmware, must be installed on the system running **bsmFWupg.sh**. Refer to the *KiraTool User Manual* for the installation procedure.

1.4.2. Options

–h	Displays command usage.
–D tooldir	Installation directory of the KiraTool command line tool (full path).
	Default value is **C:\Program files\KiraTool** for Windows, and **/usr/local/bin** for Linux.
–F fw_binfile	Name of the firmware binary file used during the upgrade.
–d fw_bindir	Directory where the firmware binary file is (full path).
	Default value is **/tmp**.
–H host	DNS name or IP address to identify the module over the LAN.
–t tool	Command line tool: kiratool, ipmitool.
–a action	Action for a component (identified by component_id) on the module (specified by module_id): bulloemver: display the firmware version of the component bulloemupg: upgrade the firmware of the component bulloemlisthw: display the list of components which can be upgraded (presently limited to module 0).
–E component_id	Component name
–M module_id	Module ID, from 0 to 3.
–m model	Target server model: ipmilan, lanplus, ipmidrv. If not present, kcs driver will be used.
–u user	Administration login for the Remote Management Module.
–p password	Administration password for the Remote Management Module.
–o outfile	Output file name to which **stdout** and **stderr** outputs are redirected.

1.4.3. Usage: NovaScale servers updated via kiratool

Syntax

```
bsmFWupg.sh [-h] -t kiratool -H host –F fw_binfile [-u user] [-p password]
[-D tooldir] [-d fw_bindir] [-o outfile]
```

Command to upgrade the firmware

```
$ bsmFWupg.sh -t kiratool –H 129.182.6.110 –u super -p pass
–F "fw-wub-MESCB_010100_0026.bin"
```

Output

```
------------------------------------------------------------------------
Starting Firmware Upgrade
Uploading Firmware File
0% ------------------ 50% ------------------ 100%
*************************************************
Upload complete.
Flashing Firmware (takes about 1min)
Successfully upgraded firmware.


------------------------------------------------------------------------
```

1.4.4. Usage: Bull servers updated via ipmitool bulloem extension

Syntax
```
bsmFWupg.sh [-h] -t ipmitool -H host –F [fw_binfile] [-u user]
[-p password] -a action [-E component] [-M module] [-m model]
[-d fw_bindir] [-o outfile]
```

1.4.4.1. Display the firmware version

Command
```
$ bsmFWupg.sh -t ipmitool –a bulloemver –H 129.182.6.110 –u super
-p pass –m ipmilan –E LCP –M 1
```

Output
```
-------------------------------------------------------------------------------
firmware version of LCP : 20
-------------------------------------------------------------------------------
```

1.4.4.2. Upgrade BMC firmware

Command
```
$ bsmFWupg.sh -t ipmitool –a bulloemupg –H 129.182.6.110 –u super
-p pass –m ipmilan –F "fw_xxx.bin" –E BMC
```

Output
```
-------------------------------------------------------------------------------


-------------------------------------------------------------------------------
```

1.4.4.3. Display the list of the components that can be upgraded

Command
```
$ bsmFWupg.sh -t ipmitool –a bulloemlisthw –H 129.182.6.110 –u super
-p pass –m ipmilan
```

Output
```
-------------------------------------------------------------------------------
BMC
LCP
ILBC
MXBC
ADM1069_4_A_98
ADM1069_4_A_9C
ADM1069_3_B_98
ADM1069_3_B_9A
-------------------------------------------------------------------------------
```

Command
```
$ bsmFWupg.sh -t ipmitool –a bulloemlisthw –H 129.182.6.110
```

Output
```
-------------------------------------------------------------------------------
BMC
LCP
ILBC
MXBC
ADM1069_4_A_98
ADM1069_4_A_9C
ADM1069_3_B_98
ADM1069_3_B_9A
-------------------------------------------------------------------------------
```

1.4.4.4. Display all firmware versions

Command

```
$ bsmFWupg.sh -t ipmitool –a bulloemver –H 129.182.6.110 –u super -p pass –E all -M 0
```

Output

```
--------------------------------------------------------------------------------
firmware version of LCP : 20
firmware version of ILBC_BOOT : A.4.4.0.01.00
firmware version of ILBC_FUNC : B.4.5.0.F2.07
firmware version of MXBC_FUNC : C.4.5.0.F2.00
firmware version of ILBC : B.4.5.0.F2.07
firmware version of MXBC : C.4.5.0.F2.00
firmware version of ADM1069_4_A_98 : 12.01.10
firmware version of ADM1069_4_A_9A : 12.02.10
firmware version of ADM1069_4_A_9C : 12.03.10
firmware version of ADM1069_4_A_9E : 12.04.10
firmware version of ADM1069_3_B_98 : 44.01.10
firmware version of ADM1069_3_B_9A : 44.02.10
firmware version of ADM1069_3_A_98 : 44.05.10
firmware version of ADM1069_3_A_9A : 44.06.10
firmware version of ADM1069_3_A_9C : 44.07.10
firmware version of ADM1069_3_A_9E : 44.08.10
firmware version of ADM1069_3_B_9C : 44.03.10
firmware version of ADM1069_3_B_9E : 44.04.10
firmware version of ADM1062_4_A_28 : 20.01.10
firmware version of ADM1062_4_A_2A : 20.02.10
BMC Firmware version: 11.9.0
    Build number:     1146
    Hardware ID:      0x30
    Firmware tag:     Final Edition 20110215
    OEM:              bull
--------------------------------------------------------------------------------
```

1.5.   bsmHWinfocmd

bsmHWinfocmd is used to get various IPMI information such as BMC parameters, LAN
configuration parameters, users list, and Alerting parameters:

```
bsmHWinfocmd -a bmc_info
bsmHWinfocmd -a lan_print
bsmHWinfocmd -a user_list
bsmHWinfocmd -a get_lan_alert_dest
bsmHWinfocmd -a get_pef_policy
bsmHWinfocmd -a get_pef_list
```

bmc_info is used to get IPMI information such as BMC Firmware revision, Manufacturer,
BMC Device support

lan_print is used to get the configuration parameters of the LAN channel used to access the
BMC with an OutOfBand connection

user_list is used to get information about the users configured for the LAN channel used to
access the BMC with an OutOfBand connection

get_lan_alert_dest is used to get information about the LAN alert destinations configured for
the LAN channel used to access the BMC with an OutOfBand connection

get_pef_policy is used to get information about the Alert Policy Table used by the BMC for
Alert processing.

get_pef_list is used to get information about the Platform Event Filter Table used by the BMC
for Alert processing.

When the command is accepted, the return code is 0. If the command fails, the return code
is -1.

1.5.1.   Options

–h	Displays command usage.
-l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
-a param	Type of information to be read:
	bmc_info, lan_print, user_list, get_lan_alert_dest, get_pef_list, get_pef_policy.
–H host	DNS name or IP address of the target BMC.
-m model	Target server model: nec, ipmilan, lanplus, tiger.
–u user	BMC user name.
–p password	BMC user password. (Authentication key for Express5800 servers).
–Y cipher	Cipher suite ID to be used by lanplus interface (default is 3).
–o outfile	Output file name to which **stdout** and **stderr** outputs are redirected.

1.6. bsmGetConfParam

bsmGetConfParam.sh is used to obtain the values of various configuration parameters.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

1.6.1. Notes

- This command includes the following actions, according to the server

```
bsmGetConfParam –a getPowerRestorePolicy
bsmGetConfParam –a getTemperatureControlPoint
bsmGetConfParam –a getPrimaryNTP
bsmGetConfParam –a getSecondaryNTP
```

1.6.2. Options

-h	Displays command usage.
–l level	Specifies the severity of the messages to be logged by syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
–a action	Action: getPowerRestorePolicy, getTemperatureControlPoint, getPrimaryNTP, getSecondaryNTP, available according to the server.
–m model	Target server model: **lanplus**, **ipmilan**.
–H host	BMC target IP address.
–u user	BMC user.
–p password	User password.
–Y cypher	Cipher suite ID to be used by lanplus interface (default is 3)
–o outfile	Output file name to which stdout and stderr outputs are redirected.

1.6.3. Usage

Syntax
```
bsmGetConfParam.sh [-h] [-l level] -a action -m model -H host [-u user] [-p
password] [-Y cipher] [-o outfile]
```

1.6.3.1.   Get the power restore policy

The **getPowerRestorePolicy** action is used to get the power restore policy (On, Off or Restore) which is applied to the system after an AC power loss.

Syntax

```
bsmGetConfParam.sh [-h] -a getPowerRestorePolicy -m model -H host
[-u user] [-p password] [-Y cipher] [-o outfile] [-l level]
```

Command

```
$ bsmGetConfParam.sh -m lanplus -H 172.31.90.21 -u super -p pass
-a getPowerRestorePolicy
```

Output

```
---------------------------------------------------------------------
Power Restore Policy is: on
---------------------------------------------------------------------
```

1.6.4. Usage: Server using ipmilan or lanplus model

Syntax

```
bsmHWinfocmd.sh [-h] -m ipmilan -H host —a param [-u user] [-p password]
[-o outfile] [-l level]
bsmHWinfocmd.sh [-h] -m lanplus -H host —a param -u user [-p password]
[-o outfile] [-l level] [-Y cipher]
```

1.6.4.1. Get IPMI information for a server using ipmilan

Command

```
$ bsmHWinfocmd.sh -a bmc_info -m ipmilan —H 129.182.6.151 —u MWA -p guest
```

Output

```
-------------------------------------------------------------------------------
 Device ID                  : 34
Device revision             : 1
Firmware revision           : 0.18
IPMI version                : 2.0
Manufacturer name           : NEC
....
Additionnal Device Support:
    Sensor Device
    SDR Repository Device
    SEL Device
    FRU Inventory Device
    ......
-------------------------------------------------------------------------------
```

1.6.4.2. Get IPMI information for a server using lanplus

Command

```
$ bsmHWinfocmd.sh -a bmc_info -m lanplus —H 129.182.6.151 —u MWA -p guest
```

Output

```
-------------------------------------------------------------------------------
 Device ID                  : 34
Device revision             : 1
.....
-------------------------------------------------------------------------------
```

1.6.4.3. Get configuration parameters of the LAN channel

Command

```
$ bsmHWinfocmd.sh -a lan_print -m ipmilan –H 129.182.6.151 –u MWA -p guest
```

Output

```
-------------------------------------------------------------------------------
Set in Progress           : Set Complete
Auth Type Support         : NONE MD2 MD5 PASSWORD
Auth Type Enable          : .....
IP Address Source         : Static Address
IP Address                : 129.182.6.151
Subnet Mask               : 255.255.255.0
MAC Address               : 00:12:34:56:78:90
SNMP Community String     : public
IP Header                 : ...
BMC ARP Control           : ....
....
-------------------------------------------------------------------------------
```

1.6.4.4. Get information about users configured for the LAN channel

Command

```
$ bsmHWinfocmd.sh -a user_list -m ipmilan –H 129.182.6.151 –u MWA -p guest
```

Output

```
-------------------------------------------------------------------------------
Channel Number            : 0x7
Maximum User Ids          : 4
Enabled User IDs          : 3

User ID                   : 1
User name                 :
Fixed Name                : Yes
Access Available          : call-in / callback
Link Authentication       : enabled
IPMI Messaging            : enabled
Privilege Level           : ADMINISTRATOR

User ID                   : 2
User name                 : MWA
Fixed Name                : No
Access Available          : call-in / callback
Link Authentication       : enabled
IPMI Messaging            : enabled
Privilege Level           : ADMINISTRATOR

......
-------------------------------------------------------------------------------
```

1.6.4.5. Get information about LAN alert destinations for the LAN channel

Command

```
$ bsmHWinfocmd.sh -a get_lan_alert_dest -m ipmilan –H 129.182.6.151 –u MWA -p guest
```

Output

```
--------------------------------------------------------------------------------
Alert Destination       : 0
Alert Acknowledge       : Unacknowledged
Destination type        : PET Trap
Retry Interval          : 0
Number of Retries       : 0
Alert Gateway           : Default
Alert IP Address        : 0.0.0.0
Alert MAC Address       : 00:00:00:00:00:00

Alert Destination       : 1
Alert Acknowledge       : Unacknowledged
Destination type        : PET Trap
Retry Interval          : 5
Number of Retries       : 3
Alert Gateway           : Default
Alert IP Address        : 129.182.6.22
Alert MAC Address       : 00:11:22:33:44:55
......
--------------------------------------------------------------------------------
```

1.6.4.6. Get information about the Alert Policy Table

Command

```
$ bsmHWinfocmd.sh -a get_pef_policy -m ipmilan –H 129.182.6.151 –u MWA -p guest
```

Output

```
--------------------------------------------------------------------------------
Alert policy table entry  : 1
Policy set                : 1
Policy entry rule         : Try-next-entry
Channel number            : 7
Channel medium            : 802.3 LAN
Destination number        : 2
Alert destination type    : PET
ACK timeout / retry       : 5
Retries                   : 3
IP address                : 129.182.6.22
MAC address               : 00:11:2:33:44:55
Status                    : Enabled

Alert policy table entry  : 2
.....
--------------------------------------------------------------------------------
```

1.6.4.7. Get information about the Platform Event Filter Table

Command

```
$ bsmHWinfocmd.sh -a get_pef_list -m ipmilan –H 129.182.6.151 –u MWA -p guest
```

Output

```
--------------------------------------------------------------------------------
PEF table entry        : 1
Status                 : active
Version                : 0x11
Sensor type            : Any
Sensor number          : Any
Event severity         : Warning
Event class            : Threshold
Event trigger(s)       : (0x01/0x0081)<LNC,<UNC
Action                 : Alert
Policy set             : 1

PEF table entry        : 2
Status                 : inactive

PEF table entry        : 3
Status                 : active
Version                : 0x11
Sensor type            : Power Supply
Sensor number          : Any
Event severity         : Critical
Event class            : Discrete
Event trigger(s)       : Sensor-specific
Action                 : Alert
Policy set             : 1

....
--------------------------------------------------------------------------------
```

1.7. bsmHWpm

The **bsmHWpm** command is available for servers with Manageability Engine. This command is used to control power consumption of platform using power and thermal policies.

```
bsmHWpm.sh -a getPowerBudget
bsmHWpm.sh -a setPowerBudget
bsmHWpm.sh -a getPowerConsumption
bsmHWpm.sh -a getNMVersion
bsmHWpm.sh -a getNMCapabilities
bsmHWpm.sh -a endisNMPolicyControl
bsmHWpm.sh -a getNMPolicy
bsmHWpm.sh -a setNMPolicy
bsmHWpm.sh -a removeNMPolicy
bsmHWpm.sh -a getPowerLimit
bsmHWpm.sh -a setPowerLimit
bsmHWpm.sh -a getTempTrigger
bsmHWpm.sh -a setTempTrigger
bsmHWpm.sh -a setNMPowerDrawRange
bsmHWpm.sh -a getNMStatistics
bsmHWpm.sh -a resetNMStatistics
bsmHWpm.sh -a getNumberOfCPUStates
bsmHWpm.sh -a getMaxAllowedCPUStates
bsmHWpm.sh -a setMaxAllowedCPUPState
bsmHWpm.sh -a setMaxAllowedCPUTState
bsmHWpm.sh -a getPowerBudget —M F
bsmHWpm.sh -a setPowerBudget —M F
bsmHWpm.sh -a getPowerConsumption —M F
bsmHWpm.sh -a getPowerConsumptionHistory —M F
bsmHWpm.sh -a enablePower —M F
bsmHWpm.sh -a disablePower —M F
bsmHWpm.sh -a getLan —M F
bsmHWpm.sh -a getActiveLan —M F
```

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

1.7.1. Notes

- This command only applies to servers that use the ipmilan or lanplus Model
- For NovaScale servers whose name ends with an 'Fx', the model generation argument is mandatory.

1.7.2. Options

–a action	Action.
–h	Displays command usage.
–m model	Target server model: lanplus, ipmilan.
–H host	BMC target IP address.
–l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
–u user	BMC user.
–p password	User password.
–Y cipher	Cipher suite ID to be used by the lanplus interface (default is 3).
–o outfile	Output file name where **stdout** and **stderr** outputs are redirected.
–M generation	Model generation: for NovaScale servers which name ends with an 'Fx', model generation must be 'F'.
-z value	Hexadecimal code of the manageability engine channel to reach (default is 0x06)
-e value	Hexadecimal code of the manageability engine address to reach (default is 0x2c)
arglist	List of arguments specific to the action.

1.7.3. Usage: Server using ipmilan or lanplus model

Syntax
```
bsmHWpm.sh [-h] [-l level] -m model –H host [–u user –p password]
[-o outfile] [-M generation] –a action arglist
```

1.7.3.1. Get the Total Power Budget

The getPowerBudget action is used to get the Total Power Budget (in Watts) that is maintained by the Power Budget Control Service.

Syntax
```
bsmHWpm.sh [-h] [-l level] -m model –H host [–u user –p password] [-o
outfile] [-z ME channel] [-e ME address] –a getPowerBudget
```

Command

```
$ bsmHWpm.sh -m lanplus –H 199.182.100.160 –u root -p bull –a getPowerBudget
```

Output
```
-------------------------------------------------------------------------
Power Budget = 377 Watts
-------------------------------------------------------------------------
```

1.7.3.2. Set the Total Power Budget

The setPowerBudget action is used to set the Total Power Budget (in Watts) that should be maintained by the Power Budget Control Service.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a setPowerBudget —x value
```

Options

-x value	Total Power Budget to be set. This value must be greater than the min Power Limit value and less than the max Power Limit value returned by the getNMCapabilities action.

Command

```
$ bsmHWpm.sh -m lanplus —H 199.182.100.160 —u root -p bull —a setPowerBudget —x 350
```

Output

```
-------------------------------------------------------------------------------
Total Power Budget has been set
-------------------------------------------------------------------------------
```

1.7.3.3. Obtain global or individual policy Power Consumption Statistics

The getPowerConsumption action is used to get global or per policy Power Consumption Statistics.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a getPowerConsumption
[-i policyID]
```

Options

-i policyID	This argument is used to obtain individual policy power consumption statistics. Without this argument, global power consumption statistics are returned

Command

```
$ bsmHWpm.sh -m lanplus —H 199.182.100.160 —u root -p bull —a getPowerConsumption
```

Output

```
-------------------------------------------------------------------------------
Current consumption: 167 Watts
Minimum consumption: 135 Watts
Maximum consumption: 511 Watts
Average consumption: 203 Watts
-------------------------------------------------------------------------------
```

1.7.3.4. Get Node Manager and firmware version numbers

The getNMVersion action is used to get Node Manager and firmware version numbers.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a getNMVersion
```

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a getNMVersion
```

Output

```
--------------------------------------------------------------------------
Node Manager Version:
  Manufacturers ID: 0x000157 (Intel)
  Node Manager Version: supported Node Manager 1.5
  IPMI interface version: Patch version (binary encoded) 0x00
  IPMI interface version: Major Firmware revision (binary encoded)0x01
  IPMI interface version: Minir Firmware revision (BCD encoded) 0x04
--------------------------------------------------------------------------
```

1.7.3.5. Get Node Manager Capabilities

The getNMCapabilities action is used to get Node Manager Capabilities.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a getNMCapabilities
—g policyTriggerType
```

Options

-g policyTriggerType   Policy Trigger Type:
                                  0: No Policy Trigger
                                  1: Inlet Temperature Policy Trigger Type

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a getNMCapabilities —g 0
```

Output

```
--------------------------------------------------------------------------
Node Manager Capabilities:
  Manufacturers ID: 0x000157 (Intel)
  Max Concurrent Settings: 10
  Max Power/Thermal value to be settable as trigger or max Power Limit to be
maintained  if Policy trigger type is equal to 0: 32767
  Min Power/Thermal value to be settable as trigger or min Power Limit to be
maintained  if Policy trigger type is equal to 0: 1
  Min Correction time: 6000 ms
  Max Correction time: 600000 ms
  Min Statistics Reporting Period: 1 s
  Max statistics reporting Period: 3600 s
  Domain limiting scope: CPU power limiting
  IPMI interface version: Minir Firmware revision (BCD encoded) 0x04
--------------------------------------------------------------------------
```

1.7.3.6. Enable/Disable Node Manager Policy Control

The endisNMPolicyControl action is used to Enable/Disable Node Manager Policy Control.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model –H host [–u user –p password] [-o
outfile] [-z ME channel] [-e ME address] –a endisNMPolicyControl
–f policyControl [-i policyID]
```

Options

–f policyControl	Policy Control value: 0: Global Disable Node Manager policy control 1: Global Enable Node Manager policy control 2: Per Domain Disable Node Manager policies for the specified domain (domain 0) 3: Per Domain Enable Node Manager policies for the specified domain (domain 0) 4: Per Policy Disable Node Manager policy for the specified policy within domain (domain 0) 5: Per Policy Enable Node Manager policy for the specified policy within domain (domain 0)
-i policyID	This argument is mandatory if the policyControl argument is set to 4 or 5

Command

```
$ bsmHWpm.sh -m lanplus –H 172.31.50.98 –u ADMIN -p ADMIN –a endisNMPolicyControl –f 5 –i 0
```

Output

```
-------------------------------------------------------------------------
Node Manager Policy Control is done.
-------------------------------------------------------------------------
```

1.7.3.7. Get the Node Manager policy parameters

The getNMPolicy action is used to get the Node Manager policy parameters.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a getNMPolicy -i policyID
```

Options

-i policyID   Policy ID

Command

- To get the Node Manager policy parameters, for a lanplus target server model, enter:

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a getNMPolicy —i 0
```

Output

```
--------------------------------------------------------------------------
Node Manager Policy:
  Manufacturers ID: 0x000157 (Intel)
   Domain Id: 0
   Policy enabled
   per Domain Node Manager policy control enabled:
   Global Node Manager policy control enabled:
   Policy Trigger Type: No Policy Trigger. Policy will maintain Power Limit
   Policy Type: Power Control Policy
   Policy Exception Actions:
      send alert disabled
      shutdown system disabled
   Power Limit: 32767 [Watts]
   Print Correction Time Limit: 600000 [ms]
   Policy Trigger Limit: 32767
   Statistics Reporting Period: 30 [s]
--------------------------------------------------------------------------
```

1.7.3.8. Create or modify a Node Manager Policy

The setNMPolicy action is used to create or modify a Node Manager Policy.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model –H host [–u user –p password]
[-o outfile] [-z ME channel] [-e ME address] –a setNMPolicy –i policyID
 –g policyTriggerType –x powerLimit [-G policyTriggerLimit]
–c exceptionAction –T correctionTimeLimit –r statisticsReportingPeriod
```

Options

–i policyID	Policy ID

-g policyTriggerType	Policy Trigger Type:
	    0: No Policy Trigger
	    1: Inlet Temperature Policy Trigger type

–x powerLimit	Power Limit to be maintained.
–G policyTriggerLimit	Defines the Inlet temperature (in Celsius) to be used as Policy Trigger value. This argument is mandatory only if the policy trigger type is set to 1

–c exceptionAction	Defines the action to be taken if the maintained given policy power limit is exceeded over the Correction Time Limit.Possible values are:
	    0: no action
	    1: send alert
	    2: hard system shutdown via BMC

–T correctionTimeLimit	Defines the max time (in seconds), in which the Node Manager must take corrective actions in order to bring the platform back within the specified power limit before taking the Exception Action.
-r statisticsReportingPeriod	Defines the Statistics Reporting Period in seconds.

Command

```
$ bsmHWpm.sh -m lanplus –H 172.31.50.98 –u ADMIN -p ADMIN –a setNMPolicy –i 0 –g 0 –x 380 –c 0 –T
500 –r 20
```

Output

```
-----------------------------------------------------------------------
Node Manager Policy 0 has been set
-----------------------------------------------------------------------
```

1.7.3.9. Removes a Node Manager Policy

The removeNMPolicy action is used to remove a Node Manager Policy.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a removeNMPolicy
—i policyID —g policyTriggerType
```

Options

–i policyID   Policy ID

–g policyTriggerType   Policy Trigger Type:
           0: No Policy Trigger
           1: Inlet Temperature Policy Trigger type

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a removeNMPolicy —i 1 —g 0
```

Output

```
-----------------------------------------------------------------------
Node Manager Policy 1 has been removed
-----------------------------------------------------------------------
```

1.7.3.10. Get the Power Limit to be maintained on a policy ID

The getPowerLimit action is used to get the Power Limit to be maintained for a policy ID.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a getPowerLimit —i policyID
```

Options

–i policyID   Policy ID

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a getPowerLimit —i 0
```

Output

```
-----------------------------------------------------------------------
Node Manager Policy 0 Power Limit = 365 Watts
-----------------------------------------------------------------------
```

1.7.3.11. Set the Power Limit to be maintained on a policy ID

The setPowerLimit action is used to set the Power Limit to be maintained for a policy ID.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a setPowerLimit —i policyID
—x powerLimit
```

Options

–i policyID   Policy ID

–x powerLimit   Power Limit to be maintained.

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a setPowerLimit —i 1 —x 365
```

Output

```
------------------------------------------------------------------------
Node Manager Policy 1 Power Limit has been set
------------------------------------------------------------------------
```

1.7.3.12. Get the Inlet temperature

The getTempTrigger action is used to get the Inlet temperature (in Celsius) that defines the Policy Trigger Limit for a policy ID.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a getTempTrigger
—i policyID
```

Options

–i policyID   Policy ID

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a getTempTrigger —i 0
```

Output

```
------------------------------------------------------------------------
Policy 0 Inlet Temperature Trigger Limit = 0x0032 Celsius
------------------------------------------------------------------------
```

1.7.3.13. Set Inlet temperature

The setTempTrigger action is used to set Inlet temperature (in Celsius) that defines the Policy Trigger Limit for a policy ID.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a setTempTrigger
—i policyID —x tempLimit
```

Options

–i policyID   Policy ID

–x tempLimit   Temperature Limit used as Policy Trigger value

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a setTempTrigger —i 1 —x 85
```

Output

```
------------------------------------------------------------------------
Policy 1 Inlet Temperature Trigger Limit has been set
------------------------------------------------------------------------
```

1.7.3.14. Set the Min and Max power consumption range

The setNMPowerDrawrange action is used to set the Min and Max power consumption range.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a setNMPowerDrawRange
 —n minPower —N mawPower
```

Options

–n minPower   Minimum Power Draw in Watts

–N maxPower   Maximum Power Draw in Watts

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a setNMPowerDrawRange —n 200 —N 400
```

Output

```
------------------------------------------------------------------------
Power Draw range has been set
------------------------------------------------------------------------
```

1.7.3.15. Get the Node Manager Power Statistics

The getNMStatistics action is used to get the Node Manager Power Statistics.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a getNMStatistics —O mode
[—i policyID]
```

Options

–O mode	Specifies the type of statistics to be collected: 1: global power statistics (in Watts) 2: global inlet temperature statistics (in Celsius) 17: per policy power statistics (in Watts) 18: per policy trigger statistics (in Celsius)
–i policyID	This argument is mandatory only if **mode** argument is set to 17 or 18

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a getNMStatistics —O 1
```

Output

```
-------------------------------------------------------------------------------
Node Manager Statistics:
   Manufacturers ID: 0x000157 (Intel)
   Current: 683
   Minimum: 620
   Maximum: 752
   Average: 682
   Timestamp: 1285366705
   Statistics Reporting Period: 1 s
   domain Id: 0
   Measurement state: Measurements in progress(host CPU in S0 state)?
-------------------------------------------------------------------------------
```

1.7.3.16. Reset the Node Manager Power Statistics

The resetNMStatistics action is used to reset the Node Manager Power Statistics.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a resetNMStatistics —O mode
[—i policyID]
```

Options

–O mode	Specifies the type of statistics to be reset: 0: global statistics including power statistics and inlet temperature statistics 1: per policy statistics including power and trigger statistics
–i policyID	This argument is mandatory only if **mode** argument is set to 1

Command

- To reset the global Node Manager Power Statistics, for a lanplus target server model, enter:

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a resetNMStatistics —O 0
```

Output

```
-------------------------------------------------------------------------------
Node Manager Statistics have been reset
-------------------------------------------------------------------------------
```

1.7.3.17. Get the number of P-States and T-States available on the target platform

The getNumberOfCPUStates action is used to get the number of P-States and T-States available on the target platform.

Syntax
```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a getNumberOfCPUStates
```

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a getNumberOfCPUStates
```

Output
```
--------------------------------------------------------------------------
Number of P-States available on the platform: 9
Number of T-States available on the platform: 8
--------------------------------------------------------------------------
```

1.7.3.18. Get the maximum P-State and T-State

The getMaxAllowedCPUStates action is used to get the maximum P-State and T-State.

Syntax
```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a getMaxAllowedCPUStates
```

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a getMaxAllowedCPUStates
```

Output
```
--------------------------------------------------------------------------
Current maximum P-State: 5
Current maximum T-State: 4
--------------------------------------------------------------------------
```

1.7.3.19. Set the maximum P-State

The **setMaxAllowedCPUPState** action is used to set the current maximum P-State.

Syntax
```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password]
[-o outfile] [-z ME channel] [-e ME address] —a setMaxAllowedCPUPState
—s stateNumber
```

Options

-s stateNumber	P-State number to be set; from 0 to n-1, where **n** is the Number of P-States (returned with the getNumberOfCPUStates action) available on the platform.

Command

```
$ bsmHWpm.sh -m lanplus —H 172.31.50.98 —u ADMIN -p ADMIN —a setMaxAllowedCPUPState —s 3
```

Output
```
--------------------------------------------------------------------------
CPU P-State/T-State has been set
--------------------------------------------------------------------------
```

1.7.3.20. Set the current maximum T-State

The **setMaxAllowedCPUTState** action is used to set the current maximum T-State.

Syntax
```
bsmHWpm.sh [-h] [-l level] -m model -H host [-u user -p password]
[-o outfile] [-z ME channel] [-e ME address] -a setMaxAllowedCPUTState
-s stateNumber
```

Options

-s stateNumber	T-State number to be set; from 0 to n-1, where **n** is the Number of T-States (returned with the getNumberOfCPUStates action) available on the platform.

Command

```
$ bsmHWpm.sh -m lanplus -H 172.31.50.98 -u ADMIN -p ADMIN -a setMaxAllowedCPUTState -s 3
```

Output
```
-------------------------------------------------------------------------
CPU P-State/T-State has been set
-------------------------------------------------------------------------
```

1.7.3.21. Get the maximum power, the minimum power and the power budget cap

The **getPowerBudget** action is used to get the maximum power, the minimum power and the power budget cap (in Watts or BTU/h).

Syntax
```
bsmHWpm.sh [-h] [-l level] -m model -H host [-u user -p password] [-o
outfile] -M generation -w unit -a getPowerBudget
```

Options

-M generation	Model generation: For novascale NS4xxF2 model, model generation must be F.
-w unit	Unit in which is expressed the returned result. Possible values are watt or btuphr.

Command for a Novascale NS4xxF2 (result in Watt)

```
$ bsmHWpm.sh -m lanplus -H 199.182.100.160 -u root -p bull -a getPowerBudget -M F -w watt
```

Output
```
-------------------------------------------------------------------------
Maximum power: 506 Watt
Minimum power: 139 Watt
Power cap    : 506 Watt
-------------------------------------------------------------------------
```

Command for a Novascale NS4xxF2 (result in BTU/hr)

```
$ bsmHWpm.sh -m lanplus -H 199.182.100.160 -u root -p bull -a getPowerBudget -M F -w btuphr
```

Output
```
-------------------------------------------------------------------------
Maximum power: 2027224563  BTU/hr
Minimum power: 309237645  BTU/hr
Power cap    : 2027224563  BTU/hr
-------------------------------------------------------------------------
```

1.7.3.22. Set the power budget cap

The **setPowerBudget** action is used to set the power budget cap (in Watts, BTU/hr or as a percentage).

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model –H host [-u user –p password] [-o
outfile] –M generation -w unit –a setPowerBudget –x value
```

Options

-M generation	Model generation: For novascale NS4xxF2 model, model generation must be F.
–w unit	Unit for the result returned. Possible values are watt, btuphr or percent.
–x value	Total Power Budget to be set. This value must be greater than the min Power Limit value and less than the max Power Limit value returned with the getNMCapibilities action.

Command for a novascale NS4xxF2 (result in Watts)

```
$ bsmHWpm.sh -m lanplus –H 199.182.100.160 –u root -p bull –a setPowerBudget –x 350 –M F –w watt
```

Output

```
-------------------------------------------------------------------------------
Power budget is set.
-------------------------------------------------------------------------------
```

Command for a novascale NS4xxF2 (result in BTU/hr)

```
$ bsmHWpm.sh -m lanplus –H 199.182.100.160 –u root -p bull –a setPowerBudget –x 350 –M F –w btuphr
```

Output

```
-------------------------------------------------------------------------------
Power budget is set.
-------------------------------------------------------------------------------
```

1.7.3.23. Get warning threshold, failure threshold, system instantaneous headroom, system peak headroom and instantaneous power consumption

The **getPowerConsumption** action is used to get warning threshold, failure threshold, system instantaneous headroom, and system peak headroom (in watts or BTU/h) and also instantaneous power consumption (in Ampere).

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model –H host [–u user –p password] [-o
outfile] –M generation -w unit –a getPowerConsumption
```

Options

-M generation	Model generation: For novascale NS4xxF2 model, model generation must be F.
-w unit	Unit in which is expressed the returned result. Possible values are watt or btuphr.

Command for a novascale NS4xxF2 (power consumption value in Watts)

```
$ bsmHWpm.sh -m lanplus –H 199.182.100.160 –u root -p bull –M F –w watt –a getPowerConsumption
```

Output

```
-----------------------------------------------------------------------
Power consumption information
System Board System Level
Reading                : 140 W
Warning threshold      : 917 W
Failure threshold      : 966 W

Amperage value: 0.6 A
Headroom
Statistic                      Reading
System Instantaneous Headroom : 1264 W
System Peak Headroom          : 1230 W
-----------------------------------------------------------------------
```

Command for a novascale NS4xxF2 (power consumption value in BTU/h)

```
$ bsmHWpm.sh -m lanplus –H 199.182.100.160 –u root -p bull –M F –w btuphr –a getPowerConsumption
```

```
-----------------------------------------------------------------------
Power consumption information
System Board System Level
Reading                : 477 BTU/hr
Warning threshold      : 3129 BTU/hr
Failure threshold      : 3296 BTU/hr

Amperage value: 0.6 A
Headroom
Statistic                      Reading
System Instantaneous Headroom : 824633721 BTU/hr
System Peak Headroom          : 1889785610 BTU/hr
-----------------------------------------------------------------------
```

1.7.3.24. Get get the average and the peak power consumption history

The **getPowerConsumptionHistory** action is used to get the average and the peak power consumption history (in watts or BTU/hr) for the last minute, last hour, last day, and the last week.

Syntax
```
bsmHWpm.sh [-h] [-l level] -m model –H host [–u user –p password] [-o
outfile] –M generation -w unit –a getPowerConsumptionHistory
```

Options

-M generation	Model generation: For novascale NS4xxF2 model, model generation must be F.
-w unit	Unit for the result returned. Possible values are watt or btuphr.

Command for a novascale NS4xxF2 (power consumption history in Watts)

```
$ bsmHWpm.sh -m lanplus –H 199.182.100.160 –u root -p bull –M F –w watt –a getPowerConsumptionHistory
```

Output

```
--------------------------------------------------------------------------
Power Consumption History
Statistic                   Last Minute  Last Hour  Last Day  Last Week

Average Power Consumption      0 W          0 W        0 W       0 W
Max Power Consumption          0 W          0 W        0 W       0 W
Min Power Consumption          0 W          0 W        0 W       0 W

Max Power Time
Last Minute     : Tue Apr 06 13:46:58 2010
Last Hour       : Tue Apr 06 13:24:33 2010
Last Day        : Mon Apr 05 14:37:09 2010
Last Week       : Sat Apr 03 19:55:13 2010
Min Power Time
Last Minute     : Tue Apr 06 13:46:58 2010
Last Hour       : Tue Apr 06 12:34:19 2010
Last Day        : Mon Apr 05 13:34:19 2010
Last Week       : Tue Apr 06 13:46:58 2010
--------------------------------------------------------------------------
```

Command for a novascale NS4xxF2 (power consumption history in BTU/h)

```
$ bsmHWpm.sh -m lanplus –H 199.182.100.160 –u root -p bull –M F –w btuphr –a
getPowerConsumptionHistory
```

Output

```
-------------------------------------------------------------------------------
Power Consumption History

Statistic                    Last Minute      Last Hour      Last Day      Last
Week

Average Power Consumption  549755813 BTU/hr    549755813 BTU/hr    549755813
BTU/hr   549755813 BTU/hr
Max Power Consumption      549755813 BTU/hr      2095944040 BTU/hr
2095944040 BTU/hr  1340029796 BTU/hr
Min Power Consumption      549755813 BTU/hr      549755813 BTU/hr    549755813
BTU/hr   549755813 BTU/hr

Max Power Time
Last Minute       : Tue Apr 06 13:47:58 2010
Last Hour         : Tue Apr 06 13:24:33 2010
Last Day          : Mon Apr 05 14:37:09 2010
Last Week         : Sat Apr 03 19:55:13 2010
Min Power Time
Last Minute       : Tue Apr 06 13:47:58 2010
Last Hour         : Tue Apr 06 12:34:19 2010
Last Day          : Mon Apr 05 13:34:19 2010
Last Week         : Tue Apr 06 13:47:58 2010
-------------------------------------------------------------------------------
```

1.7.3.25.   Enable power capping

The **enablePower** action is used to enable power capping.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model –H host [–u user –p password] [-o
outfile] –M generation –a enablePower
```

Options

–M generation   Model generation: For novascale NS4xxF2 model, model
                             generation must be F.

Command

```
$ bsmHWpm.sh -m lanplus –H 199.182.100.160 –u root -p bull –M F –a enablePower
```

Output

```
-------------------------------------------------------------------------------
Power cap is enabled.
-------------------------------------------------------------------------------
```

1.7.3.26. Disable power capping

The **disablePower** action is used to disable power capping.

Syntax
```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password] [-o
outfile] —M generation—a disablePower
```

Options

-M generation   Model generation: For novascale NS4xxF2 model, model generation must be F.

Command

```
$ bsmHWpm.sh -m lanplus —H 199.182.100.160 —u root -p bull —M F —a disablePower
```

Output
```
-------------------------------------------------------------------------------
Power cap is disabled.
-------------------------------------------------------------------------------
```

1.7.3.27. Get information about the share type for Ethernet port used by the BMC

The **getLan** action displays the share type for the Ethernet port used by the BMC.

Syntax
```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password] [-o
outfile] —M generation—a getLan
```

Options

–M generation   Model generation: For novascale NS4xxF2 model, model generation must be F.

Command

```
$ bsmHWpm.sh -m lanplus —H 199.182.100.160 —u root -p bull —M F —a getLan
```

Output
```
-------------------------------------------------------------------------------
dedicated
-------------------------------------------------------------------------------
```

1.7.3.28. Obtain the share type for the active Ethernet port used by the BMC

The **getActivLan** action displays the share type for the active Ethernet port used by the BMC. The possible value are dedicated, shared, shared with failover lom2, shared with failover all loms.

Syntax
```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password] [-o
outfile] —M generation —a getActiveLan
```

Options

–M generation   Model generation: For novascale NS4xxF2 model, model generation must be F.

Command

```
$ bsmHWpm.sh -m lanplus —H 199.182.100.160 —u root -p bull —M F —a getActiveLan
```

Output
```
-------------------------------------------------------------------------------
dedicated
-------------------------------------------------------------------------------
```

1.7.3.29. Get servers ethernet port's MAC addresses

The **getMac** action is used to obtain the MAC addresses of the server Ethernet ports, and in particular the BMC ones. This action can be used only if an iDRAC card is present.

Syntax

```
bsmHWpm.sh [-h] [-l level] -m model —H host [—u user —p password] [-o
outfile] —M generation—a getMac
```

Options

–M generation	Model generation: For novascale NS4xxF2 model, model generation must be F.

Command

```
$ bsmHWpm.sh -m lanplus —H 199.182.100.160 —u root -p bull —M F —a getMac
```

Output

The output is as follows:

```
-----------------------------------------------------------------------
System LOMs
NIC Number      MAC Address             Status

0               64:f9:b8:cf:00:00       Enabled
1               64:f9:b8:d1:00:00       Enabled
2               64:f9:b8:d3:00:00       Enabled
3               64:f9:b8:d5:00:00       Enabled
-----------------------------------------------------------------------
```

1.8.   bsmidentify

The **bsmidentify** command is used to identify physically a remote host via user-visible lights or beeps for a defined interval.

This command is available for NovaScale servers using the ipmilan, lanplus or ipmidrv model, and for bullx chassis and blades.

1.8.1.   Options

-h	Displays command usage.
-l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
–m model	Target server model: ipmilan, lanplus, ipmidrv, CMC.
–H host	DNS name or target server IP address.
–C value	Specifies the duration of identification (in seconds).
	This value may be set from 0 to 250.
	This value set to 0 stops the identification.
	For bullx chassis and blades, this parameter specifies the switching on(1)/off(0) of the LED indicator of the component.
–B targetname	bullx blade name or bullx chassis component name.
	Possible bullx chassis components are:
	LCPCMM : for CMM and Local Control Panel
	IBS : for IB Switch blade panel
	ESM : for Ethernet Switch Module panel
	UCM : for Ultra-capacitor Module
-I index	bullx blade index (bullx blade slot)
–M hwmanager	bullx chassis CMC IP address.
–u user	BMC user.
–p password	User password.
–Y cipher	Cipher suite ID to be used by lanplus interface (default is 3).
–o outfile	Output file name where **stdout** and **stderr** outputs are redirected.
–S SNMPv	SNMP version: 1 (default value), 2, 3.
–U community	SNMP community for SNMP writing.
–P port	SNMP port (default value is 161).
-T timeout	SNMP timeout (default value is 20).

1.8.2. Usage: bullx blade chassis & bullx blade servers

Notes

- For bullx chassis and blades, if the **–B targetname** option is not set, then the **-B LCPCMM** option is used.

Syntax
```
bsmidentify.sh [-h] -m CMC -M hwmanager [-B targetname | -I index] -C value
-U community [-S SNMPv] [-P port] [-T timeout] [-o outfile]
```

1.8.3. Usage: Server using ipmidrv model

Syntax
```
bsmidentify.sh [-h] -m ipmidrv [-l level] -C value [-o outfile]
```

1.9. bsminfo

bsminfo.sh is used to direct, to the standard output, the configuration information specified in the -i information parameter.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

1.9.1. Notes

- This command concerns NovaScale 5005 & 6000 series servers, and NovaScale servers using the ipmilan, lanplus or ipmidrv model for *hardware status and BMCinfo* information.

1.9.2. Usage: Server using ipmilan, lanplus or ipmidrv model

Syntax

```
bsminfo.sh [-h] -i information -m ipmilan -H host [-u user] [-p password] [-o
outfile] [-l level]

bsminfo.sh [-h] -i information -m lanplus -H host -u user [-p password] [-o
outfile] [-l level] [-Y cipher]

bsminfo.sh [-h] -i information -m ipmidrv [-o outfile] [-l level]
```

Notes

- Servers are identified by their DNS name or IP address (-H parameter).

Options

–h	Displays command usage.
–l	Logs in syslog the successful completion of the command (OK). By default, only FAILED completion is logged.
-i information	Information type.
	Currently, only *hardwarestatus and BMCinfo* information is available.
–H host	DNS name or target server IP address.
–m model	Target server model: ipmilan, lanplus, ipmidrv
–u user	BMC user.
–p password	User password.
-Y cipher	Cipher suite ID to be used by lanplus interface (default is 3).
–o outfile	Output file name to which **stdout** and **stderr** outputs are redirected.

-i Option	Hardware Component Level (-e option specified)
hardwarestatus	Returns the power status ON/OFF of the chassis
BMCinfo	Returns the BMC IPMI version, FW revision,...

Table 1-1.   bsminfo options and parameters for NovaScale servers using ipmilan/ipmidrv

1.9.2.1. Display the chassis power status

hardwarestatus is used to display `chassis power status` info.

Notes

- −e option is ignored for the ipmilan model.
- -H host and -D domain parameters are not used (ignored if present).

Command

```
$ bsminfo.sh -i hardwarestatus -m ipmilan -H nsmaster -u MWA -p guest
```

Output
```
--------------------------------------------------------------------------------
Chassis Power is on
--------------------------------------------------------------------------------
```

Command

```
$ bsminfo.sh -i hardwarestatus -m lanplus -H nsmaster -u MWA -p guest
```

Output
```
--------------------------------------------------------------------------------
Chassis Power is on
--------------------------------------------------------------------------------
```

1.9.2.2. Display BMC information

BMCinfo is used to display BMC information such as IPMI version, firmware revision, etc.

Command

```
$ bsminfo.sh -i BMCinfo -m ipmilan -H 129.182.6.50 -u MWA -p guest
```

Output
```
--------------------------------------------------------------------------------
Device ID          : 32
Device Revision    : 1
Firmware Revision  : 3.26
IPMI Version       : 2.0
Manufacturer ID    : 116
--------------------------------------------------------------------------------
```

Command

```
$ bsminfo.sh -i BMCinfo -m lanplus -H 129.182.6.50 -u MWA -p guest
```

Output
```
--------------------------------------------------------------------------------
Device ID          : 32
Device Revision    : 1
Firmware Revision  : 3.26
IPMI Version       : 2.0
Manufacturer ID    : 116
--------------------------------------------------------------------------------
```

1.10. bsmlog

bsmlog is used to direct system log contents (SEL) to the standard output.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

If –v option is not specified, the output format is:

<rank> | <record id> | <local date> | <local time> | <event target> | <severity> - <event description>

If –v option is specified, one record is displayed by field (on several rows according to the format):

: <value>

1.10.1. Notes

- Servers are identified by their DNS name or IP address (-H parameter).
- **–n range** is ignored if the **–c option** is specified.
- By default, the **-n range** parameter is set to **-n 10**.

1.10.2. Options

–h	Displays command usage.
-l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
–v	Displays record details. See below.
-n range	Range of records to be displayed.
	May be specified as a number or as an interval:
	-n n: displays the n most recent records. -n r1-r2: displays records from r1 to r2.
	Events are displayed in chronological order beginning with the most recent ones.
–e element	Hardware element concerned (in PAM PUID format).
–H host	DNS name or target server IP address, CMC IP address for bullx chassis, or bullx blade IP address.
–m model	Target server model: fame, nec, ipmilan, lanplus, tiger, ipmidrv, blade, HMC, CMC, blade_CMC.
-M hwmanager	DNS name or PAP Manager IP address (NovaScale 5005 & 6000 series) or NS and EL Blade servers manager IP address, or HMC IP address for PL servers.
–u user	PAP administrator for NovaScale 5005 & 6000 series.
	SMU user on the target server for NovaScale 4000 series.
	BMC user for NovaScale target servers using the ipmilan model, bullx chassis, bullx blades.
	User for ssh connection to the HMC for PL servers.
–I idfile	Identity key file used to perform connection to the HMC with ssh.
–C configfile	ssh client configuration file used to perform non-prompt connection to the HMC.
–d cfgdir	Directory where identity key file and ssh client configuration file are installed (default value is**../etc/ssh**).
–t event_type	Type of events to list. Valid values are **console** or **hardware** (default value).
–p password	User password. (Authentication key for Express5800 servers).
–Y cipher	Cipher suite ID to be used by lanplus interface (default is 3).
–B targetname	Blade name (for Blade servers), or managed system name (CECname) for PL servers.
–S SNMPv	SNMP version: 1(default value), 2, 3.
–C community	SNMP community: (default value is **public** for reading the log, and **private** for clearing the log).
-P port	SNMP port: default value is 161.

-T timeout	SNMP timeout: default value is 20.
	SSH connection timeout (default value is 120s).
-o outfile	Output file to which stdout and **stderr** outputs are redirected.
–c	Clears the SEL (for NovaScale servers using the *ipmilan or ipmidrv* model).

1.10.3. Usage: bullx blade chassis

bsmlog is also used to clear the system log contents (SEL) on NovaScale and Evolutive Line Blade servers, NovaScale servers using the ipmilan or ipmidrv model, bullx chassis and bullx blades.

Syntax
```
bsmlog.sh [-h] [-v] -m CMC -H host [—u user] [—p password] [-n range]   [-c]
[-o outfile] [-l level]
```
Command to display the most recent SELs

```
$ bsmlog.sh -n 2 -m CMC -H 129.184.238.254 -u root -p pass
```

Output
```
--------------------------------------------------------------------------
2 | 0x011d | 05/13/2009 | 17:35:55 | 0x80 | Temperature (IBSW_TEMP1) #2000#06
| Lower Non-critical going low #501414 | Deasserted
1 | 0x011c | 05/13/2009 | 17:35:22 | 0x80 | Temperature (IBSW_TEMP1) #2000#06
| Lower Non-critical going low #501314 | Reading 19 < Threshold 20 degrees C
$
--------------------------------------------------------------------------
```

1.10.4. Usage: bullx blade servers

bsmlog is also used to clear the system log contents (SEL) on NovaScale and Evolutive Line Blade servers, NovaScale servers using the ipmilan or ipmidrv model, bullx chassis and bullx blades.

Syntax
```
bsmlog.sh [-h] [-v] -m blade_CMC -H host [—u user] [—p password] [-c]
[-n range] [-o outfile]  [-l level]
```
Command
TBC
Output
```
--------------------------------------------------------------------------
```
TBC
```
--------------------------------------------------------------------------
```

1.10.5. Usage: Server using ipmidrv model

bsmlog is also used to clear the system log contents (SEL) on NovaScale and Evolutive Line Blade servers, NovaScale servers using the ipmilan or ipmidrv model, bullx chassis and bullx blades.

Syntax
```
bsmlog.sh [-h] [-v] -m ipmidrv [-n range] [-o outfile] [-l level] [-c]
```

1.11. bsmOSpm

The **bsmOSpm** command, is used to control the CPU policy and frequency of platform via cpufrequtils package.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

1.11.1. Options

–a action	Action.
–h	Displays command usage.
-m model	Remote host OS: linux.
–H host	Remote target name or IP address.
–l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
–u user	Username for remote connection via ssh.
–C configfile	Config file used for remote connection (default is "").
–I idfile	Identity file used for remote connection (default is *id_dsa*).
-d cfgdir	Full path directory containing config file and_identity file.
-D pathdir	Full path directory containing script bsmOSpm_local on remote host.
–o outfile	Output file name where **stdout** and **stderr** outputs are redirected.
arglist	List of arguments specific to the action.

1.11.2. Usage

Syntax
```
bsmOSpm.sh [-h] [-l level] –m model –H host [–u user] [–o outfile]
[-I idfile] [-d cfgdir] [-C configfile] [-D pathdir] –a action arglist
```

1.11.2.1. Get the governor and the frequency range of one or all CPUs

The **getPolicy** action is used to get the governor and the frequency range for one or all CPUs.

Syntax

```
bsmOSpm.sh [-h] [-l level] -m model –H host [–u user] [-o outfile]
[-I idfile] [-d cfgdir] [-C configfile] [-D pathdir] –a getPolicy [-c CPU]
```

Options

-c cpu   CPU concerned by the action (number, all, synth (default value)).

Command to get a synthesis

```
$ bsmOSpm.sh -m linux –H 199.184.238.37 –u root –a getPolicy –c synth
```

Output

```
----------------------------------------------------------------------------
CPUs ( 0 to 7) 2000000 kHz ( 68 %) – 2560000 kHz ( 87 %) — userspace
or
CPUs 0,1,2,4,5,6,7 2000000 kHz ( 68 %) – 2560000 kHz ( 87 %) – userspace
CPUs 3 2560000 kHz ( 87 %) – 2940000 kHz ( 100 %) – userspace
----------------------------------------------------------------------------
```

Command to get results for all CPUs

```
$ bsmOSpm.sh -m linux –H 199.184.238.37 –u root –a getPolicy –c all
```

Output

```
----------------------------------------------------------------------------
CPU 0 2000000 kHz ( 68 %) – 2560000 kHz ( 87 %) – userspace
CPU 1 2000000 kHz ( 68 %) – 2560000 kHz ( 87 %) – userspace
CPU 2 2000000 kHz ( 68 %) – 2560000 kHz ( 87 %) – userspace
CPU 3 2000000 kHz ( 68 %) – 2560000 kHz ( 87 %) – userspace
CPU 4 2000000 kHz ( 68 %) – 2560000 kHz ( 87 %) – userspace
CPU 5 2000000 kHz ( 68 %) – 2560000 kHz ( 87 %) – userspace
CPU 6 2000000 kHz ( 68 %) – 2560000 kHz ( 87 %) – userspace
CPU 7 2000000 kHz ( 68 %) – 2560000 kHz ( 87 %) – userspace
----------------------------------------------------------------------------
```

Command to get reults for one CPU

```
$ bsmOSpm.sh -m linux –H 199.184.238.37 –u root –a getPolicy –c 5
```

Output

```
----------------------------------------------------------------------------
CPU 5 2000000 kHz ( 68 %) – 2560000 kHz ( 87 %) – userspace
----------------------------------------------------------------------------
```

1.11.2.2. Set the CPU governor and the frequency range

The **setPolicy** action is used to set the CPU governor and to set the frequency range for all CPUs for the **usespace** governor.

Syntax

```
bsmOSpm.sh [-h] [-l level] -m model –H host [–u user] [-o outfile]
[-I idfile] [-d cfgdir] [-C configfile] [-D pathdir] –a setPolicy
–g governor [-y freq_min –x freq_max]
```

Options

–g governor	cpufreq governor. Values available are **userspace**, **ondemand**, **performance**.
-y freq_min	minimum CPU frequency the governor may select (only if userspace governor is specified). The frequency may be specified in kHz or as a percent of the maximum frequency possible (hardware limit).
-x freq_max	maximum CPU frequency the governor may select (only if userspace governor is specified). The frequency may be specified in kHz or as a percent of the maximum frequency possible (hardware limit).

Command to set the CPU frequency governor to ondemand

```
$ bsmOSpm.sh -m linux –H 199.184.238.37 –u root –a setPolicy –g ondemand
```

Output

```
-----------------------------------------------------------------------
OK CPUs (0 to 15)  ondemand
-----------------------------------------------------------------------
```

Command to set the CPU frequency governor to userspace, with frequency range in percent

```
$ bsmOSpm.sh -m linux –H 199.184.238.37 –u root –a setPolicy –g userspace –y 77% -x 87%
```

Output

```
-----------------------------------------------------------------------
OK CPUs (0 to 15)  77%  -  87% userspace
-----------------------------------------------------------------------
```

Command to set the CPU frequency governor to userspace, with frequency range in kHertz

```
$ bsmOSpm.sh -m linux –H 199.184.238.37 –u root –a setPolicy –g userspace –y 2000000 -x 2560000
```

Output

```
-----------------------------------------------------------------------
OK CPUs (0 to 15)  2000000 kHz (68%)  -  2560000 kHz (87%)  userspace
-----------------------------------------------------------------------
```

1.11.2.3. Get the current frequency of one or all CPUs

The **getCurrentFreq** action is used to get the current frequency of one or all CPUs.

Syntax

```
bsmOSpm.sh [-h] [-l level] -m model —H host [—u user] [-o outfile]
[-I idfile] [-d cfgdir] [-C configfile] [-D pathdir] —a getCurrentFreq [-c
CPU]
```

Options

—c cpu	CPU concerned by the action (number, all, synth (default value)).

Command to get a synthesis

```
$ bsmOSpm.sh -m linux —H 199.184.238.37 —u root —a getCurrentFreq —c synth
```

Output

```
--------------------------------------------------------------------------------
CPUs ( 0 to 7) 2128000 kHz
--------------------------------------------------------------------------------
```

Command to get the frequency of CPU 6

```
$ bsmOSpm.sh -m linux —H 199.184.238.37 —u root —a getCurrentFreq —c 6
```

Output

```
--------------------------------------------------------------------------------
CPU 6 2394000 kHz
--------------------------------------------------------------------------------
```

1.11.2.4. Set the current frequency for one or all CPUs

The **setCurrentFreq** action is used to set the current frequency for one or all CPUs.

Syntax

```
bsmOSpm.sh [-h] [-l level] -m model —H host [—u user] [-o outfile]
[-I idfile] [-d cfgdir] [-C configfile] [-D pathdir] —a setCurrentFreq
[-c CPU]
```

Options

-c cpu	CPU concerned by the action (number, all, synth (default value)).
—f freq	CPU frequency to be set. The userspace governor must have been set and loaded. The frequency may be specified in kHz or as a percent of the maximum frequency possible (hardware limit).

Command to set the frequency for all CPUs

```
$ bsmOSpm.sh -m linux —H 199.184.238.37 —u root —a setCurrentFreq —f 2394000 —c all
```

Output

```
--------------------------------------------------------------------------------
OK CPUs ( 0 to 7) 2394000 kHz
--------------------------------------------------------------------------------
```

Command to set the frequency for CPU 6

```
$ bsmOSpm.sh -m linux —H 199.184.238.37 —u root —a setCurrentFreq —f 2394000 —c 6
```

Output

```
--------------------------------------------------------------------------------
OK CPU 6 2394000 kHz
--------------------------------------------------------------------------------
```

1.11.2.5. Get the minimum and maximum CPU frequency allowed

The **getHWlimits** action is used to obtain the minimum and maximum CPU frequency allowed.

Syntax

```
bsmOSpm.sh [-h] [-l level] -m model —H host [—u user] [-o outfile]
[-I idfile] [-d cfgdir] [-C configfile] [-D pathdir] —a getHWlimits
[-c CPU]
```

Options

–c cpu   CPU concerned by the action (number, all, synth (default value)).

Command to get the frequency allowed for each CPU

```
$ bsmOSpm.sh -m linux –H 199.184.238.37 –u root –a getHWlimits –c all
```

Output

```
-----------------------------------------------------------------------
CPU 0 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 1 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 2 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 3 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 4 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 5 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 6 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 7 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 8 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 9 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 10 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 11 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 12 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 13 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 14 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
CPU 15 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
-----------------------------------------------------------------------
```

Command to get the frequency allowed for CPU 3

```
$ bsmOSpm.sh -m linux –H 199.184.238.37 –u root –a getHWlimits –c 3
```

Output

```
-----------------------------------------------------------------------
CPU 3 1596000 kHz ( 54 %) – 2927000 kHz (100 %)
-----------------------------------------------------------------------
```

1.12. bsmPDUpower

The **bsmPDUpower** command is used for monitoring and the administration of Power Distribution Units (PDUs). This includes obtaining the total power consumption for all the outlets of the PDU (and also the warning and critical consumption thresholds) and for a PDU, poweron/poweroff of the PDU outlets, and obtaining their status. The command uses SNMP get and set requests to the SNMP agent of the PDU:

```
bsmPDUpower.sh -a on/off
bsmPDUpower.sh -a status
bsmPDUpower.sh -a getPowerConsumption
```

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

1.12.1. Notes

- Actually, only PDUs from APC are supported. For APC metered rack PDUs, only the **getPowerConsumption** action is available.
- The Firmware Revision of APC PDUs must be equal to or higher than v3.

1.12.2. Options

-a action	Action.
-h	Displays command usage.
-m model	PDU model: apc.
-H host	DNS name or IP address of the PDU.
-l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
-o outfile	Output file name where **stdout** and **stderr** outputs are redirected.
arglist	list of arguments specific to the action.

1.12.3. Usage

Syntax
```
bsmPDUpower.sh [-h] [-l level] -m model –H host [-o outfile] –a action
arglist
```

1.12.3.1. Set on or off one or all outlets of a PDU

The **on** and **off** actions are used to set on, or off, one, or all, outlets for a PDU.

Syntax
```
bsmPDUpower.sh [-h] [-l level] -m model –H host [-o outfile] –a on/off
[-p SNMP port] [-S SNMPv] [[-C community]|[-u user –L seclevel
–A authPass –P authPro –X privPass –x privPro]] [-c outlet] [-T timeout]
```

Options

–p port	SNMP port (default value: 161)
–S SNMPv	SNMP version: 1 (default value), 2c, 3
–C community	SNMPv1 community (default value 'private' pour actions on and off
–L seclevel	SNMPv3 SecurityLevel (noAuthNoPriv | authNoPriv | authPriv)
–u user	SNMPv3 user.
–A authPass	SNMPv3 authentication passphrase (15 to 32 ASCII characters)
–P authPro	SNMPv3 authentication protocol (default value 'MD5')
–X privPass	SNMPv3 privacy passphrase (15 to 32 ASCII characters, different than –authPass)
–x privPro	SNMPv3 privacy protocol (default value 'DES')
–c outlet	number or name of one outlet for a PDU (default value 'all': all outlets are switched on or off)
–T timeout	SNMP timeout (default value: 30 seconds)

Command to set all outlets off

```
$ bsmPDUpower.sh –H 129.182.6.177 –a off –m apc –S 3 –L authPriv –u $user –A $authpass –P md5 –X
$privpass –x des –c all
```

Output
```
-------------------------------------------------------------------------
Outlet1 power: Off(2)
Outlet2 power: Off(2)
Outlet3 power: Off(2)
Outlet4 power: Off(2)
Outlet5 power: Off(2)
Outlet6 power: Off(2)
Outlet7 power: Off(2)
Outlet8 power: Off(2)
-------------------------------------------------------------------------
```

Command to set outlet 5 on

```
$ bsmPDUpower.sh –H 129.182.6.177 –a on –m apc –S 3 –L authPriv –u $user –A $authpass –P md5 –X
$privpass –x des –c 5
```

Output
```
-------------------------------------------------------------------------
Outlet5 power: On(1)
-------------------------------------------------------------------------
```

1.12.3.2.   Get the status for one or all outlets of a PDU

The **status** action is used to obtain the status for one, or all outlets, of a PDU.

Syntax

```
bsmPDUpower.sh [-h] [-l level] -m model —H host [-o outfile] —a status
[-p SNMP port] [-S SNMPv] [[-C community]|[-u user —L seclevel
—A authPass —P authPro —X privPass —x privPro]] [-c outlet] [-T timeout]
```

Options

–p port	SNMP port (default value: 161)
–S SNMPv	SNMP version: 1 (default value), 2c, 3
–C community	SNMPv1 community (default value 'public' for status action
–L seclevel	SNMPv3 SecurityLevel (noAuthNoPriv | authNoPriv | authPriv)
-u user	SNMPv3 user.
–A authPass	SNMPv3 authentication passphrase (15 to 32 ASCII characters)
–P authPro	SNMPv3 authentication protocol (default value 'MD5')
–X privPass	SNMPv3 privacy passphrase (15 to 32 ASCII characters, different than –authPass)
–x privPro	SNMPv3 privacy protocol (default value 'DES')
–c outlet	number or name of one outlet of PDU (default value 'all': all outlets are switched on or off)
–T timeout	SNMP timeout (default value: 30 seconds)

Command to obtain the status for all outlets

```
$ bsmPDUpower.sh —H 129.182.6.177 —a status —m apc —S 3 —L authPriv —u $user —A $authpass —P md5 —X
$privpass —x des —c all
```

Output

```
-------------------------------------------------------------------------
Power Distribution Unit: 129.182.6.177, MODEL: "AP7922", Serial Nb:
"ZA0904000484", Firm Rev: "v3.5.7"
Outlet1 power: Off(2)
Outlet2 power: Off(2)
Outlet3 power: Off(2)
Outlet4 power: Off(2)
Outlet5 power: On(1)
Outlet6 power: Off(2)
Outlet7 power: Off(2)
Outlet8 power: Off(2)
-------------------------------------------------------------------------
```

Command to obtain the status for outlet 4

```
$ bsmPDUpower.sh —H 129.182.6.177 —a status —m apc —S 1 —C public —c 4
```

Output

```
-------------------------------------------------------------------------
Power Distribution Unit: 129.182.6.177, MODEL: "AP7922", Serial Nb:
"ZA0904000484", Firm Rev: "v3.5.7"
Outlet4 power: Off(2)
-------------------------------------------------------------------------
```

1.12.3.3. Get the power consumption

The **getPowerConsumption** action is used to get the power consumption (in Watts) for all outlets of a PDU.

Syntax
```
bsmPDUpower.sh [-h] [-l level] -m model —H host [-o outfile]
—a getPowerConsumption [-p SNMP port] [-S SNMPv] [[-C community]|
[-u user —L seclevel —A authPass —P authPro —X privPass —x privPro]]
[-T timeout]
```

Options

–p port	SNMP port (default value: 161)
-S SNMPv	SNMP version: 1 (default value), 2c, 3
–C community	SNMPv1 community (default value 'public' for getPowerConsumption action
–L seclevel	SNMPv3 SecurityLevel (noAuthNoPriv | authNoPriv | authPriv)
–u user	SNMPv3 user.
–A authPass	SNMPv3 authentication passphrase (15 to 32 ASCII characters)
-P authPro	SNMPv3 authentication protocol (default value 'MD5')
–X privPass	SNMPv3 privacy passphrase (15 to 32 ASCII characters, different than –authPass)
–x privPro	SNMPv3 privacy protocol (default value 'DES')
–T timeout	SNMP timeout (default value: 30 seconds)

Command

```
$ bsmPDUpower.sh –H 129.182.6.177 –a getPowerConsumption –m apc
```

Output
```
--------------------------------------------------------------------------
Power OK: Reading 5 Watts|power=5;5520;7360;0
--------------------------------------------------------------------------
```

Or should be:

```
power=10;5000;7000;0

power consumption:10
high power level:      5000
critical power level:  7000
minimal consumption:      0
```

Output OK:
```
Power OK: 10|power=10;5000;7000;0
```

Output WARNING:
```
Power WARNING: High power level detected: 5500|power=5500;5000;7000;0
```

Output CRITICAL:
```
Power CRITICAL: Critical power level detected: 7500|power=7500;5000;7000;0
```

1.13. bsmpower

bsmpower.sh is used to perform system POWER ON/OFF and FORCE POWER OFF operations and to check the power state:

```
bsmpower.sh -a on
bsmpower.sh -a off
bsmpower.sh -a off_force
bsmpower.sh -a status
```

This command is asynchronous and returns the prompt before the end of command completion.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

A power ON command on an already powered ON system and/or a power OFF command on an already powered OFF system may return 0 (command accepted).

An acknowledgement message is displayed in **stdout** (except if the **-q option** is specified).

1.13.1. Notes

- For the Power Off command to be taken into account on a remote host running Windows 2000 / 2003 Server, the **Shutdown: Allow system to be shut down without having to log on security** option must be enabled on the remote host. To configure this security option:

  a. Click **Start > Run**.

  b. In the **Open** box, type **gpedit.msc** and click **OK**.

  c. In the **Group Policy** window, expand **Computer Configuration\Windows\ Settings\Security Settings\Local\Policies\Security Options**.

  d. Set the **Shutdown Security** option to **enabled**.

- Servers are identified by their DNS name or IP address (-H parameter).

- In order to power on/off a chassis, the system executing bsmpower must be declared on the target server as a member of a SNMPv1 read-write community. The default read-write community name used by bsmpower is **private**.

1.13.2. Options

-h	Displays command usage.
-l level	Specifies the severity of the messages to be logged in syslog. 0: no message is logged (default). 1: FAILED completion only is logged. 3: FAILED and SUCCESSFUL completion are logged.
-q	Quiet option: stops all outputs (stdout, sdterr, or outfile if the **-o option** is specified).
-a action	Action: on, off, off_force, status, restart. The restart action is only available for Escala PL servers and logical partitions.
-a action	Action: on, off, off_force, status.
-H host	DNS name or target system IP address, bullx blade IP address.
-D domain	Target system domain name.
-m model	Target system model.
-M hwmanager	DNS name or Hardware Manager IP address.
-u user	User identification.
-Y cipher	Cipher suite ID used by the lanplus interface (default is 3).
-I idfile	Identity key file for ssh connection.
-C configfile	ssh client configuration file for non-prompt connection.
-d cfgdir	Directory in which the identity key file and ssh client configuration file are installed (default value is ../etc/ssh).
-L lpar	Logical partition name.
-f profilename	Name of the partition profile to be used when activating a partition. This argument is required unless the managed system is in the manufacturing default configuration.
-p password	User password or authentication key.
-B targetname	Managed system name.
-S SNMPv	SNMP version: 1 (default value), 2, 3.
-C community	SNMP community (default value is **public** for the **status** action and **private** for the **on** and **off** actions).
-P port	SNMP port (default value is 161).
-T timeout	SNMP timeout (default value is 20s). SSH connection timeout (default value is 30s).
-o outfile	Output file name to which stdout and stderr outputs are redirected.

1.13.3.   Usage: bullx blade chassis

Notes

- In order to power on/off a chassis, the system executing bsmpower must be declared on the target server as a member of a SNMPv1 read-write community. The default read-write community name used by bsmpower is "private".

- The bsmpower.sh -a status command displays system power ON/OFF status. System power ON/OFF status represents the electrical status and not the functional status of the system. For example, a system may be powered ON but not RUNNING, due to a BIOS initialization error.

Syntax

```
bsmpower.sh [-h] [-q] -a action -m CMC -M hwmanager [-S SNMPv] [-C community]
[-P port] [-T timeout] [-o outfile] [-l level]
```

1.13.3.1.   Power On

Command

```
$ bsmpower.sh -a on -m CMC -M 129.184.238.254 -C write_com
```

1.13.3.2.   Power Off

Command

```
$ bsmpower.sh -a off -m CMC -M 129.184.238.254 -C write_com
```

1.13.3.3.   Display Power Status

Command

```
$ bsmpower.sh -a status -m CMC -M 129.184.238.254
```

Output Example

```
--------------------------------------------------------------------------
Chassis Power is ON
--------------------------------------------------------------------------
```

Return Codes

0	Chassis power is ON.
1	Chassis power is OFF (Deep Standby, Light Standby).
-1	Command failed.

1.13.4. Usage: bullx blade servers

Notes

- The bsmpower.sh -a status command displays system power ON/OFF status. System power ON/OFF status represents the electrical status and not the functional status of the system. For example, a system may be powered ON but not RUNNING, due to a BIOS initialization error.

Syntax

```
bsmpower.sh [-h] [-q] -a action -m blade_CMC -H host [-u user] [-p password]
[-o outfile] [-l level]
```

1.13.4.1. Power On

Command

```
$ bsmpower.sh -a on -m blade_CMC -H 129.184.238.110 -u root –p pass
```

Output

```
-------------------------------------------------------------------------------
Blade Power Control: on
-------------------------------------------------------------------------------
```

1.13.4.2. Power Off

Command

```
$ bsmpower.sh -a off -m blade_CMC -H 129.184.238.110 -u root –p pass
```

Output

```
-------------------------------------------------------------------------------
Blade Power Control: off
-------------------------------------------------------------------------------
```

1.13.4.3. Display Power Status

Command

```
$ bsmpower.sh -a status -m blade_CMC -H 129.184.238.110 -u root –p pass
```

Output Example

```
-------------------------------------------------------------------------------
Blade power is ON
-------------------------------------------------------------------------------
```

Return Codes

0	Blade power is ON.
1	Blade power is OFF.
-1	Command failed.

1.13.5. Usage: Server using ipmidrv model

Notes

- The bsmpower.sh -a status command displays system power ON/OFF status. System power ON/OFF status represents the electrical status and not the functional status of the system. For example, a system may be powered ON but not RUNNING, due to a BIOS initialization error.

Syntax

```
bsmpower.sh [-h] [-q] -a action -m ipmidrv [-o outfile] [-l level]
```

1.14. bsmreset

bsmreset.sh is used to forcibly Power Reset a server.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

If the server is not in the right state to perform the reset command, an error is displayed.

1.14.1. Notes

- Servers are identified by their DNS name or IP address (-H parameter).

1.14.2. Options

–h	Displays command usage.
–l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
–q	Quiet option: stops all outputs (**stdout**, **sdterr**, or **outfile** if **-o** option is specified).
–H host	DNS name or target server IP address, bullx blade IP address.
–D domain	Target server domain name (NovaScale 5005 and 6000 series only).
–m model	Target server model: fame, nec, ipmilan, lanplus, tiger, blade_CMC.
–M hwmanager	DNS name or PAP Manager IP address (NovaScale 5005 & 6000 series only).
–u user	PAP administrator for NovaScale 5005 & 6000 series.
	SMU user on the target server for NovaScale 4000 series.
	BMC user for NovaScale target servers using the ipmilan model, and for bullx blades.
–p password	User password.(Authentication key for Express5800 servers).
–Y cipher	Cipher suite ID to be used by lanplus interface (default is 3).
–o outfile	Output file name to which **stdout** and **stderr** outputs are redirected.

1.14.3. Usage: bullx blade servers

Syntax
```
bsmreset.sh [-h] [-q] -m blade_CMC -H host [-u user] [-p password]
[-o outfile] [-l level]
```

1.15. bsmsensor

bsmsensor is used to direct sensor values to the standard output.

This command is not available for NovaScale 5005 & 6000 series servers.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

The sensor values are displayed in stdout, as follows:

<sensor addr> | <sensor id> | <sensor value> | <sensor unit> | <status> | <lower non recoverable> | <lower critical> | <lower non critical> | <upper non critical> | <upper critical> | <supper non recoverable>

If the **-v option** is specified, one sensor is displayed per field, over several rows depending on the format.

<label>: <value>

The fields and labels display may change according to servers and to sensors.

1.15.1. Notes

- Servers are identified by their DNS name or IP address (-H parameter).
- Sensor type (see definition in IPMI specification) may be specified either with the sensor type code in hexadecimal (i.e. -t 0x01 for temperature) or with the sensor type string (i.e. -t **Power Unit**).
- **-s sensorid** and **–t type** parameters (for **–m ipmilan** model) are exclusive.
- **-B bladename** and **–c** parameters (for **–m blade** model) are exclusive.

1.15.2. Options

–h	Displays command usage.
–l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
–v	Displays details of each sensor. See output section below.
–H host	DNS name or target server IP address, CMC IP address for bullx chassis, or bullx blade IP address..
-m model	Target server model: nec, ipmilan, lanplus, tiger, ipmidrv, blade, CMC, blade_CMC.
–M hwmanager	NS and EL Blade server manager IP address.
–u user	SMU user on the target server for NovaScale 4000 series.
	BMC user for NovaScale target servers using the ipmilan model, bullx chassis, bullx blades.
–p password	User password. (Authentication key for Express5800 servers).
–Y cipher	Cipher suite ID to be used by lanplus interface (default is 3).
–c	On blade servers, used In order to get sensors of the blade chassis only.
	On IPMI servers, used in conjunction to -v in order to get sensors information in a more compact form.
–B bladename	Blade name.
	For Blade servers only.
–S SNMPv	SNMP version: 1 (default value), 2, 3.
	For Blade servers only.
–C community	SNMP community (default value is **public**).
	For Blade servers only.
-P port	SNMP port (default value is 161).
	For Blade servers only.
–T timeout	SNMP timeout (default value is 20).
	For Blade servers only.
–o outfile	Output file name to which **stdout** and **stderr** outputs are redirected.
-s sensorid	Specify a list of sensor(s), specified between quotes "" and separated by comma(s), to be read.
–t type	Process only sensors of the specified type (as defined in IPMI specification).
–a	Process only analog sensors of the type specified with **-t type**.

1.15.3. Usage: bullx blade chassis

Syntax

```
bsmsensor.sh [-h] [-v] -m CMC -H host [—u user] [—p password] [-o outfile]
[-l level] [-s sensorid_list] [-t type [-a]] [-c]
```

1.15.3.1. Display sensors

Command

```
$ bsmsensor.sh -m CMC -H 129.184.238.254 —u root -p pass
```

Output

```
-------------------------------------------------------------------------------
0x00200001 | CMB_Temp         | 0x20 | 31.000    | degrees C | ok      |
0.000      | 10.000 | 20.000    | 40.000   | 50.000    | 60.000
...
0x00200006 | IBSW_Temp1       | 0x20 | 22.000    | degrees C | ok      |
0.000      | 10.000 | 20.000    | 40.000   | 50.000    | 60.000
...
0x00200009 | FAN2A            | 0x20 | 12500.000 | RPM       | ok      |
3200.000   | 6900.000 | 8000.000 | 17800.000 | 18400.000 | 19000.000
...


-------------------------------------------------------------------------------
```

1.15.4. Usage: bullx blade servers

Syntax

```
bsmsensor.sh [-h] [-v] -m blade_CMC -H host [—u user] [—p password] [-o
outfile]  [-l level] [-s sensorid_list] [-t type [-a]] [-c]
```

1.15.5. Usage: Server using ipmidrv model

Syntax

```
bsmsensor.sh [-h] [-v] -m ipmidrv [-o outfile] [-l level] [-s sensorid_list]
[-t type [-a]] [-c]
```

1.16. bsmSetConfParam

bsmSetConfParam.sh is used to set the values of various configuration parameters.

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

1.16.1. Options

–h	Displays command usage.
–l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
–a action	Action: setPowerRestorePolicy.
–m model	Target server model: lanplus, ipmilan.
–H host	BMC target IP address.
–u user	BMC user.
–p password	User password.
–Y cypher	Cipher suite ID to be used by lanplus interface (default is 3)
–o outfile	Output file name to which stdout and stderr outputs are redirected.

1.16.2. Usage

Syntax

```
bsmSetConfParam.sh [-h] [-l level] -a action -m model -H host [-u user] [-p
password] [-Y cipher] [-o outfile]
```

1.16.2.1. Set the power restore policy

The **setPowerRestorePolicy** action is used to set the power restore policy (On, Off or Restore) which will be apply to the system after an AC power loss.

Syntax
```
bsmSetConfParam.sh [-h] -a setPowerRestorePolicy -m model -H host
-x value [-u user] [-p password] [-Y cipher] [-o outfile] [-l level]
```

Options

–x value	on: chassis always powers up after AC main electicity is switched on or returns following an interruption.
	off: chassis always stays powered off after AC mains electricity is switched on, power pushbutton or command required to power on system.
	restore: after AC main electricity is switched on or returns, power is restored to the state that was in effect when AC main electricity was switched off or lost.

Command

```
$ bsmSetConfParam.sh -m lanplus -H 172.31.90.21 -u super -p pass
-a setPowerRestorePolicy –x on
```

1.17. bsmsol

The **bsmsol** command is used to establish a Serial-Over-Lan session.

This command is available only for NovaScale servers using the ipmilan or lanplus model.

1.17.1. Notes

- Enter "~." to close the session.
- Only one Serial-Over-Lan session can be opened on a given target server.
- When the target server is a NovaScale 3005 with a BMC firmware including encryption, the cipher suite parameter must be set to "3".

1.17.2. Options

–h	Displays command usage.
-l level	Specifies the severity of the messages to be logged in syslog.
	0: no message is logged (default).
	1: only the FAILED completion of the command is logged.
	3: FAILED and successful completion of the command are logged.
–m model	Target server model: ipmilan, lanplus.
–H host	DNS name or target server IP address.
–C crypt	Cipher suite to be used by ipmilan model for lanplus encryption (default value "0" means no encryption).
–L priv	Remote session privilege level (ADMINISTRATOR (default), OPERATOR, USER).
–u user	BMC user.
-p password	User password.
–Y cipher	Cipher suite ID to be used by lanplus interface (default is 3).

1.17.3. Usage

Syntax
```
bsmsol.sh [-h] -m ipmilan -H host [-u user] [-p password] [-l level] [-C
crypt] [-L priv]
bsmsol.sh [-h] -m lanplus -H host -u user [-p password] [-l level] [-L priv]
[-Y cipher]
```

1.17.3.1. Establish SOL session with ipmilan

Command
```
$ bsmsol.sh -m ipmilan -H 129.182.6.50 -u root -p root
```

Output
```
------------------------------------------------------------------------
 [SOL Session operational. Use ~? For help]
Shell>
......
Shell>~. [terminated ipmitool]

------------------------------------------------------------------------
```

1.17.3.2. Establish SOL session with lanplus

Command

```
$ bsmsol.sh -m lanplus -H 129.182.6.50 -u root -p root
```

Output

```
-----------------------------------------------------------------------------
 [SOL Session operational. Use ~? For help]
Shell>
......
Shell>~. [terminated ipmitool]
-----------------------------------------------------------------------------
```

Appendix A. Server ON / OFF Status Codes

This appendix lists the interconnection between the ON / RUNNING / TRANSIENT / OFF states returned by the bsmpower.sh -a status command and the server states.

OFF corresponds to server states:

> INACTIVABLE
>
> INACTIVE

TRANSIENT corresponds to server states:

> POWERING ON
>
> POWERING ON FAILED
>
> TIMEOUT DURING POWER ON
>
> POWERING ON SUSPENDED
>
> POWERING DOWN
>
> POWER DOWN FAILED
>
> TIMEOUT DURING POWER DOWN

ON corresponds to server states:

> DOMAIN HALTED
>
> POWERED ON - LOADING BIOS
>
> BIOS READY - STARTING EFI
>
> BIOS LOADING TIMEOUT
>
> RECOVERING BIOS
>
> TIMEOUT DURING START EFI

RUNNING corresponds to server states:

> EFI STARTED - BOOTING OS
>
> RUNNING

Appendix B. User/Password Internal Authentication File

B.1. User/Password Authentication

Most of the Hardware Management Commands need a user/password authentication in order to access a target identified by:

- the PAP manager (specified by **-M hwmanager**) for NovaScale 5005 & 6000 Series servers,

- the platform manager (specified by **-H host**) for NovaScale 4000 Series servers, Express 5800 servers and NovaScale servers using the ipmilan model,

- the FTP server site (specified by **-S ftpserver**) for the bsmpamcmd command,

- or any generic device identified by its IP address.

The **password** information used by the Hardware Management Commands may be either specified in the command, or extracted from an authentication internal file if no **password** is specified in the command.

This internal authentication file contains the list of targets (PAP managers, platform managers, FTP servers, or generic devices), identified by their role (manager, FTP server, or device) and their DNS name or IP address, that can be accessed with dedicated users/passwords.

This file is created and managed using the **bsmcfg_auth_cmd** command.

The **bsmcfg_auth_cmd** command allows to:

- add an entry defined by: `user + password + target`

- delete an entry defined by: `user + target`

- get the encoded password for an entry defined by: `user + target`

- get the decoded password for an entry defined by: `user + target`

- ist the entries.

The **password** information is stored in the authentication internal file as a base64 encoded string.

B.2. bsmcfg_auth_cmd command

bsmcfg_auth_cmd command manages the internal authentication file:
```
bsmcfg_auth_cmd.sh -a add
bsmcfg_auth_cmd.sh -a getE
bsmcfg_auth_cmd.sh -a getD
bsmcfg_auth_cmd.sh -a del
bsmcfg_auth_cmd.sh -a list
```

Syntax

```
bsmcfg_auth_cmd.sh [-h] [-a action] [-M manager] [-H manager] [-S ftpserver]
[-D ipaddr] [-u user] [-p password]
```

When the command is accepted, the return code is 0. If the command fails, the return code is -1.

B.2.1. Notes

- -M manager, -H manager, -D ipaddr and -S ftpserver parameters are exclusive.
- For NovaScale servers using the ipmilan model, it is possible to use the same user/password authentication for all the servers available. This can be done by setting "*" as the manager parameter.

B.2.2. Options

-h	Displays command usage.
-a action	Action on the internal authentication file: `add, del, getE, getD, list.`
-M manager	DNS name or PAP Manager IP address (NovaScale 5005 and 6000 Series servers).
-H manager	DNS name or platform Manager IP address (NovaScale servers using the tiger, nec, or ipmilan model).
-S ftpserver	DNS name or FTP server IP address
-D ipaddr	Generic Device IP address
-u user	Manager or FTP server user.
-p password	Manager or FTP server user password.

B.2.2.1. Add a target

add is used to add in the internal authentication file, a target to the list of targets that can be accessed with a dedicated user/password.

Syntax

```
bsmcfg_auth_cmd.sh –a add –u user -p password [-M manager] [–H manager] [–S ftpserver] [–D ipaddr]
```

Notes

- -M or -H or -S or -D parameter is mandatory.

Command to add a PAP Manager

```
$ bsmcfg_auth_cmd.sh -a add -u pap -p pappwd -M paptest2
```

Output

```
-------------------------------------------------------------------------------
Key entry successfully added
-------------------------------------------------------------------------------
```

Command to add a platform Manager

```
$ bsmcfg_auth_cmd.sh -a add -u MWA -p mwapwd -H 129.182.6.151
```

Output

```
-------------------------------------------------------------------------------
Key entry successfully added
-------------------------------------------------------------------------------
```

Command to add a ftp server

```
$ bsmcfg_auth_cmd.sh -a add -u ftpuser -p ftppwd -S MyFtpServer
```

Output

```
-------------------------------------------------------------------------------
Key entry successfully added
-------------------------------------------------------------------------------
```

Command to add a generic device

```
$ bsmcfg_auth_cmd.sh -a add -u devuser -p devpwd -D 129.182.3.3
```

Output

```
-------------------------------------------------------------------------------
Key entry successfully added
-------------------------------------------------------------------------------
```

Command

```
$ bsmcfg_auth_cmd.sh -a add -u MWA2 -p mwa2pwd -H "*"
```

B.2.2.2. Extract the encoded password

getE is used to extract, from the internal authentication file, the encoded password needed to access a target with a dedicated user.

Syntax
```
bsmcfg_auth_cmd.sh -a getE -u user [-M manager] [-H manager] [-S ftpserver]
[-D ipaddr]
```

Note

- -M or -H or -S or -D parameter is mandatory.

Command to get the encoded password to access a PAP Manager

```
$ bsmcfg_auth_cmd.sh -a getE -u pap -M paptest2
```

Output
```
-------------------------------------------------------------------------
xxxxxx
-------------------------------------------------------------------------
```

Command to get the encoded password to access a platform Manager

```
$ bsmcfg_auth_cmd.sh -a getE -u MWA -H 129.182.6.151
```

Output
```
-------------------------------------------------------------------------
yyyyyy
-------------------------------------------------------------------------
```

Command to get the encoded password to access a ftp server

```
$ bsmcfg_auth_cmd.sh -a getE -u ftpuser -S MyFtpServer
```

Output
```
-------------------------------------------------------------------------
zzzzzz
-------------------------------------------------------------------------
```

Command to get the encoded password to access a generic device

```
$ bsmcfg_auth_cmd.sh -a getE -u devuser -D 129.182.3.3
```

Output
```
-------------------------------------------------------------------------
yyyyyy
-------------------------------------------------------------------------
```

```
$ bsmcfg_auth_cmd.sh -a getE -u MWA2 -H "*"
```

Output
```
-------------------------------------------------------------------------
vvvvvv
-------------------------------------------------------------------------
```

B.2.2.3.   Extract the password needed to access a target

getD is used to extract, from the internal authentication file, the password needed to access a target for a user.

Syntax
```
bsmcfg_auth_cmd.sh –a getD –u user [–M manager] [–H manager]
[–S ftpserver] [–D ipaddr]
```

Notes

- -M or -H or -S or -D parameter is mandatory.

Command to get the password to access a PAP Manager

```
$ bsmcfg_auth_cmd.sh -a getD -u pap -M paptest2
```

Output
```
--------------------------------------------------------------------
pappwd
--------------------------------------------------------------------
```

Command to get the password to access a platform Manager

```
$ bsmcfg_auth_cmd.sh -a getD -u MWA -H 129.182.6.151
```

Output
```
--------------------------------------------------------------------
mwapwd
--------------------------------------------------------------------
```

Command to get the password to access a ftp server

```
$ bsmcfg_auth_cmd.sh -a getD -u ftpuser -S MyFtpServer
```

Output
```
--------------------------------------------------------------------
ftppwd
--------------------------------------------------------------------
```

Command to get the password to access a generic device

```
$ bsmcfg_auth_cmd.sh -a getD -u devuser -D 129.182.3.3
```

Output
```
--------------------------------------------------------------------
devpwd
--------------------------------------------------------------------
```

```
$ bsmcfg_auth_cmd.sh -a getD -u MWA2 -H "*"
```

Output
```
--------------------------------------------------------------------
mwa2pwd
--------------------------------------------------------------------
```

B.2.2.4. Delete a target

del is used to delete, in the internal authentication file, a target from the list of targets that can be accessed by a user.

Syntax

```
bsmcfg_auth_cmd.sh -a del -u user [-M manager] [-H manager]
[-S ftpserver] [-D ipaddr]
```

Notes

- -M or -H or -S or -D parameter is mandatory.

Command to delete a PAP Manager

```
$ bsmcfg_auth_cmd.sh -a del -u pap -M paptest2
```

Output

```
--------------------------------------------------------------------------
Key entry successfully deleted
--------------------------------------------------------------------------
```

Command to delete a platform Manager

```
$ bsmcfg_auth_cmd.sh -a del -u MWA -H 129.182.6.151
```

Output

```
--------------------------------------------------------------------------
Key entry successfully deleted
--------------------------------------------------------------------------
```

Command to delete a ftp server

```
$ bsmcfg_auth_cmd.sh -a del -u ftpuser -S MyFtpServer
```

Output

```
--------------------------------------------------------------------------
Key entry successfully deleted
--------------------------------------------------------------------------
```

Command to delete a generic device

```
$ bsmcfg_auth_cmd.sh -a del -u devuser -D 129.182.3.3
```

Output

```
--------------------------------------------------------------------------
Key entry successfully deleted
--------------------------------------------------------------------------
```

```
$ bsmcfg_auth_cmd.sh -a del -u MWA2 -H "*"
```

Output

```
--------------------------------------------------------------------------
Key entry successfully deleted
--------------------------------------------------------------------------
```

B.2.2.5. Display the contents of the internal authentication file

list is used to display the contents of the internal authentication file, this means the list of targets that can be accessed with their user/password details.

Syntax
```
bsmcfg_auth_cmd.sh –a list
```

Command
```
$ bsmcfg_auth_cmd.sh -a list
```

Output
```
---------------------------------------------------------------------
<?xml version="1.0"?>
 <auth_entries>
  <auth_entry keyname="HWMGR_mgr1" password="cHdkMQ%3D%3D" user="usr1" />
  <auth_entry keyname="HWMGR_mgr2" password="d3h5" user="usr2" />
  <auth_entry keyname="FTPSRV_srv1" password="cHdkMQ%3D%3D" user="usr1" />
  <auth_entry keyname="DEV_ipaddr1" password="ZGV2cHdk" user="devuser" />
  <auth_entry keyname="HWMGR_*" password="d3h6" user="MWA2" />
</auth_entries>
---------------------------------------------------------------------
```

Glossary

This glossary may contain entries that are not relevant to your system.

A

ABR
Automatic BIOS Recovery.

ACPI
Advanced Configuration and Power Interface.
An industry specification for the efficient handling of power consumption in desktop and mobile computers. ACPI specifies how a computer's BIOS, operating system, and peripheral devices communicate with each other about power usage.

ADM1069
The ADM1069 Super Sequencer® is a configurable supervisory/ sequencing device that offers a single-chip solution for supply monitoring and sequencing in multiple supply systems.

ARU
Add / Removeable Unit. A hardware logical unit, or a group of logical units, that can be viewed / handled by an Operating System, or the BIOS, or the Platform Management Software. An ARU can be nested and is not necessarily separable from other ARUs. An ARU is also known as a PMU.

ASR
Automatic Server Restart.

ASIC
Application Specific Integrated Circuit.

B

Base Operating System
The Operating System that is booted at initialization.

BCE
Elementary calculation block.

BCEA
ASIC elementary calculation block.

BCEF
FPGA elementary calculation block.

BCS
Bull Coherent Switch. This is the Bull eXternal Node Controller providing SMP upgradeability up to 16 processors. The BCS ensures global memory and cache coherence, with optimized traffic and latencies, in both IPF-preferred and XPF-preferred variants.

BHC
See Blade Hardware Console.

BIOS
Basic Input / Output System. A program stored in flash EPROM or ROM that controls the system startup process.

BIST
Built-In Self-Test. See POST.

Blade Hardware Console
Graphical user interface used to access the management software embedded in the blade module.

BMC

Baseboard Management Controller. See Embedded Management Controller.

BOOTP

Network protocol used by a network client to obtain an IP address from a configuration server.

BSM

Bull System Manager. A software package that allows the management of data centers. BSM is capable of supporting many different types of servers.

BT

Block Transfer. One of the three standardized IPMI System interfaces used by system software for transferring IPMI messages to the BMC. A per-block handshake is used to transfer data (higher performance).

C

Chassis Hardware Console

Graphical user interface used to access the management software embedded in the Chassis Management Module.

CHC

See Chassis Hardware Console.

Clipping

An Event filter criterion. Clipping is defined on a Count / Time basis aimed at routing a pre-defined number of messages only. Identical messages are counted and when the number of messages indicated in the Count field is reached within the period of time indicated in the Time field, no other messages will be selected for routing.

CMB

Chassis Management Board.

CMC

A Corrected Memory Check condition is signaled when hardware corrects a machine check error or when a machine check abort condition is corrected by firmware. See MCA.

CMC

Chassis Management Controller.

CMM

Chassis Management Module.

Core

Core is the short name for the processor execution core implemented on a processor. A core contains one or more threads (logical processors).

CRU

Customer Replaceable Unit. A component (board, module, fan, power supply, etc.) that is replaced or added by the End User as a single entity.

CSE

Customer Service Engineer.

D

DES
Data Encryption Standard.

DHCP
Dynamic Host Configuration Protocol.

DMA
Direct Memory Access. Allows data to be sent directly from a component (e.g. disk drive) to the memory on the motherboard). The microprocessor does not take part in data transfer enhanced system performance.

DNS
Domain Name Server.

DSIB/DSIBL
Dummy System Interface Board. The boards designed by Bull when there is not a BCS in the system.

E

EEPROM
Electrically Erasable Programmable Read-Only Memory. A type of memory device that stores password and configuration data.

EFI
Extensible Firmware Interface. A specification for a firmware-OS interface.

EFI Shell
Simple, interactive user interface that allows EFI device drivers to be loaded, EFI applications to be launched, and operating systems to be booted. In addition, the EFI Shell provides a set of basic commands used to manage files and the system environment variables. See Shell.

Embedded Management Controller
Also known as BMC (Baseboard Management Controller). This controller, embedded on the main system board, provides out-of-band access to platform instrumentation, sensors and effectors.

EMM
Embedded Management Module. Software embedded in the server module to implement management functions and accessible from the Hardware Console graphical interface.

EPROM
Erasable Programmable Read-Only Memory. A type of memory device that is used to store the system BIOS code. This code is not lost when the computer is powered off.

ESB
Ethernet Switch Board.

ESM
Ethernet Switch Module.

F

FC-LGA
Flip-Chip Land Grid Array.

FDB
Fan Distribution Board.

Flash EPROM
Flash Erasable Programmable Read-Only Memory. A type of memory device that is used to store the system firmware code. This code can be replaced by an updated code from a floppy disk, but is not lost when the computer is powered off.

FPGA
Field Programmable Gate Array.

FQDN
Fully Qualified Domain Name.

FRU
Field Replaceable Unit. A component (board, module, fan, power supply, etc.) that is replaced or added by Customer Service Engineers as a single entity.

G

GPU
Graphical Processing Unit.

GUI
Graphical User Interface.

H

HA
High Availability. Refers to a system or component that is continuously operational for a desirably long length of time.

Hardware
The physical parts of a system, including the keyboard, monitor, disk drives, cables and circuit cards.

Hardware Partition
A set of hardware components that can boot and run a Base OS image.

Hard Partitioning
Ability to split a platform into a number of independent smaller hardware partitions or to merge multiple independent hardware partitions to form a single larger hardware partition.

HDD
Hard Disk Drive.

HPC
High Performance Computing.

HPC Cluster
High Performance Computing Cluster. A group of computers linked together to form a single computer.

Host Operating System
The Operating System that is booted at initialization and that is a Virtual Machine Monitor (VMM) and a number of guest OS.

Hot-Plugging
The operation of adding a component without interrupting system activity.

Hot-Swapping
The operation of removing and replacing a faulty component without interrupting system activity.

HT
HyperThreading. See Multi-Threading.

I

I2C
Intra Integrated Circuit. The I2C (Inter-IC) bus is a bi-directional two-wire serial bus that provides a communication link between integrated circuits (ICs). The I2C bus supports 7-bit and 10-bit address space devices and devices that operate under different voltages.

IB
InfiniBand.

iBMC
Integrated Baseboard Management Controller. See Embedded Management Controller.

iCare
The iCare Console (insight Care) is a web-based administration application which provides tools for hardware unit maintenance.

ICH
Input/Output Hub. Provides a connection point between various I/O components and Intel processors.

ICMB
Intelligent Chassis Management Bus. Name for the architecture, specifications, and protocols used to interconnect intelligent chassis via an RS-485-based serial bus for the purpose of platform management.

ILB / ILBL
I/O Legacy Boards.The Bull-designed I/O boards for the MESCA modules.

INCA
INtegrated Cluster Architecture.

IOH
Input/Output Hub. An Intel QPI agent that handles I/O requests for processors.

IPMB
Intelligent Platform Management Bus. Abbreviation for the architecture and protocol used to interconnect intelligent controllers via an I2C based serial bus for the purpose of platform management.

IPMI
Intelligent Platform Management Interface. A specification owned by Intel which describes mechanisms and devices to completely offload the task of managing system hardware from the primary CPU.

J

JOEM
JTAG Over Ethernet Module.

JTAG
Joint Test Action Group.

K

No entries.

L

LAN
Local Area Network.

LCD
Liquid Crystal Display.

LCP
Local Control Panel. Module consisting of a controller, a LCD color display, a green and a blue LED and a Power ON button.

LDAP
Lightweight Directory Access Protocol.

LED
Light Emitting Diode.

Logical Partition
When the Base Operating System is a Virtual Machine Monitor, a logical partition is the software environment used to run a Guest Operating System.

Logical Processor
See Thread.

M

MAC
Media Access Control.

MCA
A Machine Check Abort exception occurs when an error condition has arisen that requires corrective action.

MESCA
Multiple Environments on a Scalable Csi-based Architecture.

MIB
Management Interface Base.

MIMD
Multiple Instruction Multiple Data

MMX
MultiMedia eXtensions.

MTB/MTBC
Memory and Tukwila Board / Memory and Tukwila Board Controller.

MTBF
Mean Time Between Failure.

Multicore
Presence of two or more processors on a single chip.

Multi-Threading
The ability of a single processor core to provide software visibility similar to that of several cores and execute several threads in apparent (to software) simultaneity while using limited additional hardware resources with respect to a core without multi-threading.
Depending on core design, the instructions issued for execution by the core at a given cycle may be either **Hyper-Threading** (HT) - from a single thread, switching to another thread upon occurrence of specific events (e.g. cache misses) or **Simultaneous Multi-Threading** (SMT) - from both threads.

MXB/MXBC
Memory and Xeon Board / Memory and Xeon Board Controller.

N

Nehalem
NEHALEM Intel Xeon Processor (8 cores per die).

NFS
Network File System.

NIC
Network Interface Controller.

NUMA
Non Uniform Memory Access.

NVRAM
Non-Volatile Random Access Memory.

O

Off-Lining
See On-Lining / Off-Lining.

On-Lining / Off-Lining
On-lining and off-lining are dynamic logical operations. On-lining is the non-physical addition of an ARU to the running OS. The on-lined unit already exists in the configuration as an inactive unit (present and connected). Off-lining is the non-physical removal of an ARU from the running OS. The off-lined unit remains in the configuration as an inactive unit, ready to be on-lined.

OOB
Out Of Band. Access to system platform management that does not go through the OS or other software running on the main processors of the managed system.

OPMA
Open Platform Management Architecture.

P

PCI
Peripheral Component Interconnect. Bus architecture supporting high-performance peripherals.

PCIe
PCI Express. Latest standard in PCI expansion cards.

PDB
Power Distribution Board. Sub-assembly of the Power Supply Module.

PDU
Power Distribution Unit. Power bus used for the connection of peripheral system components.

Platform Event
A platform event is an event that originates directly from platform firmware (BIOS) or platform hardware, independently of the state of the Operating System or System Mangement Hardware.

PEF
Platform Event Filtering.
A feature in IPMI that enables the BMC to generate a selectable action (e.g. power on/off, reset, send Alert, etc.) when a configurable event occurs on the management system.

PET
The Platform Event Trap format is used for sending a platform event in an SNMP Trap. See Platform Event.

PIROM
The Processor Information ROM contains information about the specific processor in which it resides. This information includes robust addressing headers to allow for flexible programming and forward compatibility, core and L2 cache electrical specifications, processor part and S-spec numbers, and a 64-bit processor number.

PMU
Physically Manageable Unit. A hardware logical unit, or a group of logical units, that can be viewed / handled by an Operating System, or the BIOS, or the Platform Management Software. A PMU can be nested and is not necessarily separable from other PMUs. A PMU is also known as an ARU.

POST
Power On Self Test.

Processor
Each processor contains one or more dies in a single package. Each die contains one or more cores. Each core contains one or more threads (logical processors). Each processor is housed in a processor socket.

PSMI
Power Supply Management Interface.

PSU
Power Supply Unit. Sub-assembly of the Power Supply Module.

PSWB
PCI SWitch Board.

PSWM
PCI SWitch Module.

PWM
Pulse Width Modulation.

Q

QDR
Quad Data Rate. Communication signalling technique where data is transmitted at four points in the clock cycle.

QPI
Quick Path Interconnect. High-speed point-to-point Intel interface, used to interconnect processors and I/O Hubs, and optionally node controllers (BCS).

QSB
Quad Switch Board.

QSFP
Quad Small Form-factor Pluggable. Low-power interconnect technology.

QSMB
Quad Switch Module. InfiniBand Switch.

R

RADIUS
Remote Authentication Dial-In User Service.

RAS
Reliability, Availability, Serviceability.

RMII
Reduced Media Independent Interface. A standard that reduceds the number of signals/pins required to connect an Ethernet chip to physical layer transceiver. See MII.

RTC
Real Time Clock.

S

SAS
Serial Attached SCSI. A data transfer technology used to move data to and from computer storage devices such as hard drives and tape drives.

SATA
Serial ATA. A computer bus technology for connecting hard disks and other devices.

SEL
System Event Log. A record of system management events. The information stored includes the name of the event, the date and time the event occurred and event data. Event data may include POST error codes that reflect hardware errors or software conflicts within the system.
A non-volatile storage area into the BMC and associated interfaces for storing System platform Event information for later retrieval.

Server Hardware Console
Graphical user interface used to access the management software embedded in the server module.

SHC
See Server Hardware Console.

SIB/SIBL
System Interface Board. The boards designed by BULL which contain the BCS (Bull Coherent Switch).

Simultaneous Multi-Threading
See Multi-Threading.

SMBIOS
System Management BIOS.

SM-BUS
System Management Bus.

SMI
System Management Interrupt.

SMP
Symmetrical Multi Processor. The processing of programs by multiple processors that share a common operating system and memory.

SMT
Simultaneous Multi-Threading.

SMTP
Simple Mail Transfer Protocol.

SNC
Scalable Node Controller. The processor system bus interface and memory controller for the Intel870 chipset. The SNC supports both the Itanium2 processors, DDR SDRAM main memory, a Firmware Hub Interface to support multiple Firmware hubs, and two scalability ports for access to I/O and coherent memory on other nodes, through the FSS.

SNMP
Simple Network Management Protocol.

SoC
System on Chip.

Socket
Central Processing Unit mutlticore interface.

SOL

Serial Over LAN. Mechanism that enables the input and output of the serial port of a managed system to be redirected via an IPMI session over IP.

SO-DIMM

Small Outline Dual In-line Memory.

SR

Scratch Register. Internal registers of both the Tukwila processor and the I/O Hub used as scratch area.

SSD

Solid State Drive.

SSH

Secured Shell.

SSL

Secure Socket Layer.

T

TELNET

TELecommunication NETwork. Protocol used on the Internet or Local Area Networks to provide a bidirectional interactive communications facility.

Thread

A thread or logical processor is the execution context within a single core and the software visibility of multi-threading. A single multi-threaded processor contains two or more threads (or logical processors).

Thresholding

An Event filter criterion. Thresholding is defined on a Count / Time basis aimed at routing significant messages only. Identical messages are counted and when the number of messages indicated in the Count field is reached within the period of time indicated in the Time field, this message is selected for routing.

TKW

TUKWILA Intel Itanium Processor (4 cores per socket).

TSM

Ten Gigabit Ethernet Switch Module.

U

UCM

Ultra Capacitor Module.

V

VMM

Virtual Machine Monitor.

W

WOL

Wake On Lan. A feature that provides the ability to remotely power on a system through a network connection.

X

XCSI
Extended Common System Interface. High-speed point-to-point Bull interface, used to interconnect servers. XCSI ports are located and managed in the BCS (node controller).

XNC
External Node Controller. See BCS.

Y

No entries.

Z

No entries.