Guide to Administration, Programming Environments, and Tools Available on SGI® Altix® XE Systems

007–4901–001

Record of Revision

Version	Description
001	April 2007 Original publication.

Contents

Procedures

About This Guide

This guide is a reference document for people who manage the operation of SGI computer systems running SGI ProPack 5 for Linux operating system on Altix XE systems. It explains how to perform general system configuration and operations and describes programming environments and tools available for SGI Altix XE systems.

This manual contains the following chapters:

Related Publications

For a list of manuals supporting SGI ProPack for Linux releases covering the following topics, see the *SGI ProPack 5 for Linux Service Pack 1 Start Here*:

- SGI documentation supporting SGI Altix XE systems
- Novell documentation for SUSE Linux Enterprise Server 10 (SLES10)
- Red Hat documentation Red Hat Enterprise Linux 5 (RHEL5)
- Intel Compiler Documentation
- Intel documentation about Itanium and Xeon architecture

Obtaining Publications

You can obtain SGI documentation in the following ways:

- See the SGI Technical Publications Library at: http://docs.sgi.com. Various formats are available. This library contains the most recent and most comprehensive set of online books, release notes, man pages, and other information.
- Online versions of the *SGI ProPack 5 for Linux Service Pack 1 Start Here*, the SGI ProPack 5 SP1 release notes, which contain the latest information about software and documentation in this release, the list of RPMs distributed with SGI ProPack 5 SP1, and a useful migration guide, which contains helpful hints and advice for

customers moving from earlier versions of SGI ProPack to SGI ProPack 5, can be found in the `/docs` directory on the SGI ProPack 5 Open/Free Source CD.

The SGI ProPack 5 for Linux SP1 release notes get installed to the following location on a system running SGI ProPack 5: `/usr/share/doc/sgi-propack-5/README.txt`.

- You can view man pages by typing man *title* on a command line.

Conventions

The following conventions are used throughout this document:

Convention	Meaning
`command`	This fixed-space font denotes literal items such as commands, files, routines, path names, signals, messages, and programming language structures.
variable	Italic typeface denotes variable entries and words or concepts being defined.
`user input`	This bold, fixed-space font denotes literal items that the user enters in interactive sessions. (Output is shown in nonbold, fixed-space font.)
[ ]	Brackets enclose optional portions of a command or directive line.
...	Ellipses indicate that a preceding element can be repeated.

Reader Comments

If you have comments about the technical accuracy, content, or organization of this publication, contact SGI. Be sure to include the title and document number of the publication with your comments. (Online, the document number is located in the front matter of the publication. In printed publications, the document number is located at the bottom of each page.)

You can contact SGI in any of the following ways:

- Send e-mail to the following address:

  techpubs@sgi.com

- Contact your customer service representative and ask that an incident be filed in the SGI incident tracking system.

- Send mail to the following address:

  SGI
  Technical Publications
  1140 East Arques Avenue
  Sunnyvale, CA 94085–4602

SGI values your comments and will respond to them promptly.

Configuring and Operating Your System

This chapter provides information on configuring and operating your system and covers the following topics:

Getting the Latest Firmware/Software for Your SGI Altix XE System

For information on the basic input/output system (BIOS), Baseboard Management Controller (BMC) firmware, Intelligent Platform Management Interface (IPMI) control utility, Field Replaceable Unit/Sensor Data Record (FRUSDR) software, Hierarchical Storage Controller (HSC) software, LSA RAID firmware, Scali Manage system management software, Infiniband drivers, INTEL MPI Runtime Environment Kit for Linux, and Java Runtime Environment (JRE), see the **SGI ALtix XE Firmware/Software** page on SGI Supportfolio at:https://support.sgi.com/content_request/691346/index.html

Setting Up a Serial Over LAN Connection

This section describes how to set up a Serial over LAN (SOL) connection to an Altix XE system so you can use the `ipmitool`(1) utility for console access, power control, and sensor queries.

Procedure 1-1 Setting Up a Serial Over LAN Connection

To set up the Serial Over LAN Connection to an Altix XE system, perform the following steps:

1. Make sure your system has BMC firmware v48 or v55 (or later) installed.

   Make sure you have `ipmitool` v1.8.9-rc1 software (or later) installed.

   You must have `ipmitool` v1.8.9-rc1 or later installed on the machine you wish to use as the controlling node (headnode) in order to fully operate the INTEL BMC.

2. Make sure the basic input/output (BIOS) system settings are, as follows:

```
Advanced -> Serial Port
Serial B Enable     [Enabled]
Address             [2F8]
IRQ                 [3]


Server Management -> Console Redirection
Console Redirection     [Serial B]
Flow Control            [RTS/CTS]
Baud Rate               [38.4K]
Terminal Type           [VT100]
Legacy OS Redirection   [Enabled]
```

3. Make sure the Deployment CD settings are, as follows:

   - Configure a Server

```
<check> Server Management Settings
<next>
```

   - Communication Options

```
<check> LAN Channel 1 (onboard NIC 1)
<check>  Direct Serial Connection
<next>
```

   - LAN Channel 1

```
<check> Enable LAN Channel 1

choose either DHCP or static IP
if static IP, fill in IP address, netmask and gateway appropriately

<check> Enable Serial Over LAN
<next>
```

- Serial Channel

  ```
  <uncheck>  Enable Serial Channel
  <next>
  ```

- SetUp Users

  ```
  add + enable any users you want to give access to BMC. Make sure you know wh
  <next>

  <Apply>
  ```

4. Build and configure the `ipmitool`.

   When `ipmitool` is built on any system, you must make sure the OpenSLL library is installed prior to configuring the tool before `make`. When the tool is configured correctly, you will see the following displayed at the completion:

   ```
   configure:

   ipmitool 1.8.9-rc1

   Interfaces

         lan     : yes
         lanplus : yes
         open    : yes
         imb     : yes
         bmc     : no
         lipmi   : no

   Extra tools
         ipmievd : yes
   ```

   For more information on using the `ipmitool` utility, see "`ipmitool`(1) Utility Overview" on page 5.

5. It is essential that `lanplus` is available (yes) or the IPMI v2.0 protocol required to communicate with these BMC functions will not be available. Once `ipmitool` is completely built, you can verify that `lanplus` is available, as follows:

   ```
   ipmitool -h
   Interfaces:
   ```

```
open          Linux OpenIPMI Interface [default]
imb           Intel IMB Interface
lan           IPMI v1.5 LAN Interface
lanplus       IPMI v2.0 RMCP+ LAN Interfac
```

6. Make sure that the Serial Over LAN (SOL) console connection baud rate matches the BIOS, `grub`, and `getty` settings

```
ipmitool -I lanplus -o intelplus -H <bmc_ip> -U <user> -P <pass> sol set non-volatile-bit-rate 38.4

ipmitool -I lanplus -o intelplus -H <bmc_ip> -U <user> -P <pass> sol set volatile-bit-rate 38.4
```

Note: You may get an error message when you issue these commands. You can ignore the error messages.

---

7. Edit the `/boot/grub/menu.lst` configuration file so only the following content is at the top:

   ```
   default 0
   timeout 15
   ```

   Remove or comment out any `gfxmenu` and `color` statements that appear in the `grub` configuration file. You will want a timeout of at least fifteen seconds. If it is any shorter, the delay in drawing the `grub` menu and transferring it over the 38.4K SOL connection means that you only have about three seconds to make your operating system boot selection before the default is taken.

   Enter the following at each kernel command line to enable the serial console on Serial B (which is now redirected over the LAN).

   ```
   console=ttyS1,38400
   ```

8. Edit the `getty` entry in the `/etc/inittab` configuration file to read as follows:

   ```
   S1:12345:respawn:/sbin/agetty -h -L 38400 ttyS1 vt102
   ```

   The `-h` option is required to enable hardware flow control which does work with the `getty` .

9. Initiate a SOL connection, as follows:

   ```
   ipmitool -I lanplus -o intelplus -H  -U  -P  sol activate
   ```

Using `ipmitool`(1) for System Administration

This section describes the `ipmitool`(1) utility and covers the following topics:

`ipmitool`(1) Utility Overview

You can use the `ipmitool`(1) utility to manage the Intelligent Platform Management Interface (IPMI) functions of either the local system, using a kernel device driver, or a remote system. These functions include printing field replaceable unit (FRU) information, the local area network (LAN) configuration, sensor readings, and remote chassis power control. IPMI management of a local system interface requires a compatible IPMI kernel driver to be installed and configured. For more detailed information, see the `ipmitool`(1) man page.

For Altix XE systems, you need to use IPMI version 1.8.9-rc1 (or later). Use the following command to determine the version of the `ipmitool`(1) installed on your system:

```
$ mhist find -M ipmitool
ipmitool.1.rpm              # ipmitool-1.8.9-0.rc1.1.x86_64.rpm
ipmitool.2.rpm              # ipmitool-1.8.9-0.rc1.1.src.rpm
ipmitool.3.rpm              # ipmitool-1.8.9-0.rc1.2.x86_64.rpm
```

Useful IPMI Commands

This section describes some useful IPMI commands, as follows:

- Reset a system

  ```
  $ ipmitool -I lan -H 128.162.243.208 power reset
  ```

- Print out the BMC LAN configuration

  ```
  $ ipmitool -I lan -H 128.162.243.208 lan print 1
  Password:
  Set in Progress         : Set Complete
  ```

```
Auth Type Support       : NONE MD5 PASSWORD
Auth Type Enable        : Callback :
                        : User     :
                        : Operator :
                        : Admin    : MD5 PASSWORD
                        : OEM      :
IP Address Source       : Static Address
IP Address              : 128.162.243.208
Subnet Mask             : 255.255.255.0
MAC Address             : 00:04:23:cf:2d:a0
SNMP Community String   :
IP Header               : TTL=0x40 Flags=0x40 Precedence=0x00 TOS=0x10
BMC ARP Control         : ARP Responses Disabled, Gratuitous ARP Enabled
Gratituous ARP Intrvl   : 5.0 seconds
Default Gateway IP      : 0.0.0.0
Default Gateway MAC     : 00:00:00:00:00:00
Backup Gateway IP       : 0.0.0.0
Backup Gateway MAC      : 00:00:00:00:00:00
RMCP+ Cipher Suites     : None
Cipher Suite Priv Max   : XXXXXXXXXXXXXXX
                        :     X=Cipher Suite Unused
                        :     c=CALLBACK
                        :     u=USER
                        :     o=OPERATOR
                        :     a=ADMIN
                        :     O=OEM
```

The system's BMC software baseboard management controller (BMC) is typically available at <*hostname*>-bmc.<*domainname*>.

- Obtain information on an IPMI channel.

```
$ ipmitool -I lan -H 128.162.243.208 channel info 1
Password:
Channel 0x1 info:
  Channel Medium Type   : 802.3 LAN
  Channel Protocol Type : IPMB-1.0
  Session Support       : multi-session
  Active Session Count  : 1
  Protocol Vendor ID    : 7154
  Volatile(active) Settings
    Alerting            : disabled
```

```
    Per-message Auth    : enabled
    User Level Auth     : enabled
    Access Mode         : always available
  Non-Volatile Settings
    Alerting            : disabled
    Per-message Auth    : enabled
    User Level Auth     : enabled
    Access Mode         : always available
```

- Clear the password for user 1, from the local host system (IPMI kernel modules must be loaded)

```
basil:~ # ipmitool lan set 1 password
Password cleared for user 1
```

- Remotely query the system power state

```
ipmitool -I lanplus -o intelplus -H <bmc_ip> -U <user> -P <pass> power status
```

- Remotely power on the system

```
ipmitool -I lanplus -o intelplus -H <bmc_ip> -U <user> -P <pass> power on
```

- Remotely power off the system

```
ipmitool -I lanplus -o intelplus -H <bmc_ip> -U <user> -P <pass> power off
```

- Remotely reset the system

```
ipmitool -I lanplus -o intelplus -H <bmc_ip> -U <user> -P <pass> power reset
```

- Identify a chassis (turn on the blue LED for 45 seconds)

```
ipmitool -I lanplus -o intelplus -H <bmc_ip> -U <user> -P <pass> chassis identi
```

- Read the available sensors

```
ipmitool -I lanplus -o intelplus -H <bmc_ip> -U <user> -P <pass> sensor
```

IPMI Kernel Modules

To use the `ipmitool` utility directly from the host system, as opposed to IPMI over LAN, you need load the following kernel modules:

```
# modprobe ipmi_msghandler
# modprobe ipmi_devintf
```

```
# modprobe ipmi_si
```

If you want these modules loaded automatically at boot time, define them in `MODULES_LOADED_ON_BOOT` in `/etc/sysconfig/kernel` (systems running SLES).

GNU GRUB Boot Loader

The boot loader is the first software program to run on your system and it loads the operating system and then transfers control to it. GNU GRUB is the boot loader used on your SGI Altix XE system. For more information on the GRUB boot loader and the grub shell utility, see http://www.gnu.org/software/grub/manual/grub.html.

Setting up MPI

This section describes how to run Intel MPI on your system.

Procedure 1-2 Setting up MPI

If you use environment modules, to see what version of Intel MPI is installed on your machine, use the following command:

```
system 51% module show intel-mpi
-------------------------------------------------------------------
/sw/com/modulefiles/intel-mpi/3.0.021:

prepend-path     LIBRARY_PATH /sw/sdev/intel-mpi/v3.0.021/ia64/lib
prepend-path     LD_LIBRARY_PATH /sw/sdev/intel-mpi/v3.0.021/ia64/lib
prepend-path     PATH /sw/sdev/intel-mpi/v3.0.021/ia64/bin
prepend-path     C_INCLUDE_PATH /sw/sdev/intel-mpi/v3.0.021/ia64/include
prepend-path     FPATH /sw/sdev/intel-mpi/v3.0.021/ia64/include
prepend-path     CPLUS_INCLUDE_PATH /sw/sdev/intel-mpi/v3.0.021/ia64/include
prepend-path     INTEL_MPI_PATH /sw/sdev/intel-mpi/v3.0.021/ia64
-------------------------------------------------------------------
```

To run Intel MPI on your SGI Altix XE system, starting the `mpd` daemons:

```
mpdallexit
rm -f mpd.hosts
echo cl1node006 >> mpd.hosts
echo cl1node007 >> mpd.hosts
```

```
mpdboot -n 3
mpdtrace -l
```

Run Intel MPI, as follows:

```
mpiexec -genv I_MPI_DEBUG 3 \
-genv I_MPI_DEVICE rdma:InfiniHost_III_Lx0 -np 2  a.out
```

Run Voltaire MPI, as follows:

```
mpirun_rsh -np 2 cl1node001 cl1node002 a.out
```

Network Services on SGI Altix XE Systems

Currently, the only InfiniBand (IB) optimized services, that is, services that have been accelerated to IB native performance are, Message Passing Interface (MPI) and network file system (NFS).

Non-optimized services include the following:

- file transfer protocol (FTP)
- telnet
- Secure Shell (SSH)
- Station Call Processor (SCP)
- rsync
- Any utility using sockets interface, for example, a backup utility

Identifying Which HCA Is Installed

SGI supports several kinds of single data rate (SDR) host channel adapters (HCAs) and a double data rate (DDR) HCA. SGI currently supports the following models:

- MT23108 InfiniHost - 2-port SDR PCIX HCA 400 with 128 MB of memory on card
- MT25204 [InfiniHost III Lx HCA] - 1-port SDR or DDR PCIe HCA 410 memFree

To find out what model of HCA is installed on your system, use the following command:

```
% lspci -v | grep -i Mellanox000:02:02.0 PCI bridge: Mellanox Technologies MT23108 PCI Bridge (rev a1) (
0000:d0:00.0 InfiniBand: Mellanox Technologies MT23108 InfiniHost (rev a1)
        Subsystem: Mellanox Technologies MT23108 InfiniHost
```

You can ignore the first and the last lines of the command output. The middle line indicates there is an HCA 400 installed in your system. On a system with HCA 410, the `lspci -v` command output alone will not distinguish between SDR and DDR HCAs.

The output of the `ibstatus` command with GridStack 4.1 will tell you the speed of the HCA (DDR or not).

You can use DDR switches with SDR HCAs and DDR HCAs with SDR switches. Each port will auto-negotiate its link based on what is connected on the other end. The speed will the lower of the two, so if the HCA is SDR or the switch port is SDR the link will be SDR. There is a potential latency issue anytime a packet moves from an inbound SDR port to an outbound DDR port on the same switch ASIC. This is because the switch needs to buffer part of the packet before it transmits it to the outbound port.

Table 1-1 on page 11 shows which upper layer protocols (ULPs) can take advantage of multiple HCAs per host or 2-port HCAs. GridStack 4.1 is based on OFED 1.0. The number in the cell indicates how many ports are visible through the particular ULP/Protocol. If the cell is blank, it means SGI does not have the information or has not yet tested that combination.

Table 1-1 Protocols Using Multiple Host Channel Adapters (HCAs)

Protocol	2-port HCA/GridStack 3.55	Multiple HCA/GridStack 3.5.5	2-port HCA/OFED 1.0 based stack	Multiple HCA/OFED 1.0 based stack
IPoIB	1	1	2	N
MPT	2	N	2	N
IBverbs				
Voltaire MPI	2	N	2	N
iSER				
SDP				
uDAPL				

Programming Environments and Tools

This chapter describes the programming environments and tools available on an SGI Altix XE system. It covers the following topics:

Running Intel MPI

This section describes how to run Intel Message Passing Interface (MPI) software on your system.

Determining the Current Version of Intel MPI

You can use the `module show intel-mpi` to determine the current version of Intel MPI on your system, as follows:

```
system 51% module show intel-mpi
-------------------------------------------------------------------
/sw/com/modulefiles/intel-mpi/3.0.021:

prepend-path     LIBRARY_PATH /sw/sdev/intel-mpi/v3.0.021/ia64/lib
prepend-path     LD_LIBRARY_PATH /sw/sdev/intel-mpi/v3.0.021/ia64/lib
prepend-path     PATH /sw/sdev/intel-mpi/v3.0.021/ia64/bin
prepend-path     C_INCLUDE_PATH /sw/sdev/intel-mpi/v3.0.021/ia64/include
prepend-path     FPATH /sw/sdev/intel-mpi/v3.0.021/ia64/include
prepend-path     CPLUS_INCLUDE_PATH /sw/sdev/intel-mpi/v3.0.021/ia64/include
prepend-path     INTEL_MPI_PATH /sw/sdev/intel-mpi/v3.0.021/ia64
-------------------------------------------------------------------
```

Running Intel MPI

This section describes how to run Intel MPI on your system.

Procedure 2-1 Running Intel MPI

To run Intel MPI, perform the following steps:

1. The Intel MPI library uses a multipurpose daemon (MPD) job start-up mechanism as described in the *Intel Cluster Toolkit 3.0 Tutorial*. To start the MPD daemons, perform the following commands (change host list as needed):

```
mpdallexit   rm -f mpd.hosts
echo cl1node006 >> mpd.hosts  echo cl1node007 >> mpd.hosts
mpdboot -n 3  mpdtrace -l
```

2. To run Intel MPI, perform the following command:

```
mpiexec -genv I_MPI_DEBUG 3 \ -genv I_MPI_DEVICE rdma:InfiniHost_III_Lx0 -np 2
```

Debugging Intel MPI

The `I_MPI_DEBUG` variable controls output of debugging information that is printed out when an MPI program starts running. Reasonable values for this variable are 2, 3, 10, 20, 30, and 200. The higher the value that is used, the more debug information is provided. A value of 4,000 prints out the near maximum of debugging information available.

Determining What DAPL Version the Intel MPI Library Supports

Intel MPI uses Direct Access Programming Library (DAPL) as a fabric independent API to run on fast interconnects like InfiniBand as described *Intel MPI Library* available under **Software Products** at Intel.com. Currently, Intel MPI supports DAPL version 1.1 as well as DAPL version 1.2. Intel MPI automatically determines the version of the DAPL standard to which the software on your system conforms. To determine the DAPL provider you want to use, look in the `/etc/dat.conf` file on your system.

The `-genv I_MPI_DEVICE rdma` setting uses the first DAPL layer defined in the `/etc/dat.conf` file, that is, “interface adapter” by default.

Running Voltaire MPI

This section describes how to run Voltaire Message Passing Interface (MPI) software on your system.

Procedure 2-2 Running Voltaire MPI

To run Voltaire MPI, perform the following steps:

1. To run Voltaire MPI, perform the following command:

```
mpirun_rsh -np 2 cl1node001 cl1node002 a.out
```

Index