## Contents

Document Scope ................................................................................................................................. 8

Record of Revision ................................................................................................................................. 8

### Cray Advanced Cluster Engine™ (ACE™) Features

- Linux Clusters ................................................................................................................................. 9
- Scalable Unit ................................................................................................................................. 9
- Networks ..................................................................................................................................... 9
- Cray ACE Management Tool ....................................................................................................... 10
- Flexible/Fast Multi-Cluster Provisioning ..................................................................................... 10
- Flexible ACE Configuration .......................................................................................................... 10

### Advanced Cluster Engine (ACE) Architecture

- Management Servers ...................................................................................................................... 11
- Sub-management Servers ............................................................................................................. 12
- Compute Servers .......................................................................................................................... 12
- Understanding Host Images ............................................................................................................ 13
- Revision Management .................................................................................................................... 13
- Interconnect Schemes ....................................................................................................................... 15

### Systems Administration Guide

- ACE GUI Interface ......................................................................................................................... 16
  - Preserved GUI Preferences ........................................................................................................ 16
  - Installing the ACE GUI ............................................................................................................. 16
  - Starting the ACE GUI ................................................................................................................ 16
  - Logging In .................................................................................................................................. 16
  - ACE Menu Bar ............................................................................................................................ 19
  - Monitor .................................................................................................................................... 19
  - Views ..................................................................................................................................... 19
  - Temps View ............................................................................................................................ 20
  - Actions ................................................................................................................................... 22
  - View/Edit Global Parameters ..................................................................................................... 23
  - ACE Views .............................................................................................................................. 24
    - The Switches Main Panel ......................................................................................................... 25
    - The Switch View/Edit Pop-up ................................................................................................. 25
    - The Ports Panel ....................................................................................................................... 26
    - The Port View/Edit Pop-up ..................................................................................................... 27
    - The Servers Main Panel ......................................................................................................... 28
List of Figures

Figure 1: Advanced Cluster Engine (ACE) Architecture .............................................................. 11
Figure 2: Cluster Revision Management .................................................................................. 14
Figure 3: Example Fat Tree Topology .................................................................................. 15
Figure 4: Example 2D Grid Topology .................................................................................. 15
Figure 5: Example 3D Grid Topology .................................................................................. 15
Figure 6: ACE Login Window .................................................................................................. 17
Figure 7: ACE Login Dialog - SSH Tunnel Detail ............................................................... 17
Figure 8: ACE Main Window - Servers Tab View ............................................................... 18
Figure 9: ACE Main Window - Servers Tab View ............................................................... 19
Figure 10: Views Menu ........................................................................................................ 20
Figure 11: Temps View Temperature Levels and Colors .................................................. 21
Figure 12: ACE GUI Temps View ........................................................................................ 21
Figure 13: Action Menu ......................................................................................................... 23
Figure 14: View/Edit Switch Dialog .................................................................................. 26
Figure 15: View/Edit Port Dialog ....................................................................................... 27
Figure 16: ACE Cluster Window .......................................................................................... 31
Figure 17: Cluster - View/Edit Dialog ................................................................................ 32
Figure 18: Checked Out Icon .............................................................................................. 33
Figure 19: Cluster - Revision Popup ................................................................................... 34
Figure 20: ACE Monitor Console ......................................................................................... 37
Figure 21: ACE Log Console ................................................................................................ 37
Figure 22: ACE Plugin Console ........................................................................................... 37
Figure 23: Progress View .................................................................................................... 38
Figure 24: Start Cluster .......................................................................................................... 38
List of Tables

Table 1: ACE Configurations ................................................................................................................... 10
Document Scope

This document describes the

Record of Revision

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>10/7/2013</td>
<td>Initial release of the document.</td>
</tr>
<tr>
<td>1.0</td>
<td>10/9/2013</td>
<td>ACE 2.0.1 revision with emsd slurm additions.</td>
</tr>
<tr>
<td>2.0</td>
<td>10/10/2013</td>
<td>Additional backup info, sysad info, and typos.</td>
</tr>
<tr>
<td>3.0</td>
<td>10/25/2013</td>
<td>Multiple cluster checkout notes added</td>
</tr>
<tr>
<td>4.0</td>
<td>2/2/2014</td>
<td>Added Default IP’s and tmpfs description</td>
</tr>
<tr>
<td>5.0</td>
<td>2/18/2014</td>
<td>Typos and Minor Additions to CLI commands</td>
</tr>
<tr>
<td>6.0</td>
<td>4/2/2014</td>
<td>ACE CLI updates and removed Ethernet only details</td>
</tr>
<tr>
<td></td>
<td>10/20/2014</td>
<td>Revision 2.1 and was imported to the Cray publications system content management systems as publication number S-2549-A.</td>
</tr>
</tbody>
</table>
Cray Advanced Cluster Engine™ (ACE™) Features

The ACE architecture provides a balanced framework where commodity hardware can be used in a production HPC environment. ACE is designed to maximize job throughput while at the same time reduce management overhead using a balanced, scalable, and redundant interconnect design.

Linux Clusters

The ACE architecture is based on the Linux operating system. Take advantage of the open nature and flexibility of Linux to create the cluster environment that suits your needs. ACE-based systems support instant provisioning of multiple concurrent operating system environments.

Scalable Unit

The ACE architecture uses a scalable unit concept. That is, the cluster is grown in increments that have been designed to scale. Addition of a new unit preserves the interconnect performance between nodes while increasing the computing power of the cluster. A single pair of redundant management servers can handle up to 144 nodes. For larger clusters, a scalable unit includes up to 144 compute nodes (diskless), one or two sub-management nodes (two for active-active HA operation), and interconnect cabling and switches (dual rail for HA).

Networks

There are a minimum of three networks employed in the ACE architecture. See Figure 1. The first network is for user compute program communication over the Mellanox IB network and will be used by the MPI (Message Passing Interface). It can also be used to access parallel file systems.

A Gigabit Ethernet (GbE) operational network is available to the compute nodes and is used by the sub-management node(s) in each scalable unit to provision, boot, and control the nodes in each scalable unit. It can also be used to connect to an optional 10 Gigabit Ethernet (10GbE) global file system and for external high bandwidth communications to users and external file systems.

The third network is a GbE management network, which is used by the two management nodes to provision, boot, and control the sub-management nodes.

The two management units are redundant so that if one fails the other will continue providing management of scalable unit. Each sub-management unit reports to the master management node. The master management node is also redundant thus allowing a fault tolerant management system.

All networks support redundant operation. In multi-rail configurations, if one network has a failure (Interface card, cable, or switch) the other network will ensure proper operation.
Cray ACE Management Tool

Cray Advanced Cluster Engine™ (ACE™) Software Management features a complete management solution that is pre-installed by Cray Manufacturing. This software suite offers an easy to use web-based management interface making it possible to control the cluster hardware from any location. The management modules include Network Management, Server Management, Cluster Management and Storage Management.

In addition, ACE supports multiple networking topologies, diskless configurations and network fail-over to achieve maximum reliability, performance and high availability. It supports multiple root file systems offering instant provisioning for rapid, standard Linux deployment on large diskless systems allowing them to boot 10 to 10,000 servers nearly simultaneously.

Flexible/Fast Multi-Cluster Provisioning

The ACE management tool enables you to separate the cluster software environment from the underlying hardware. That is, there is no need to "pin" a specific version of Linux to a particular node. The ACE tool allows for provisioning multiple clusters within the single physical cluster. Depending on the users' needs, the entire system can be provisioned as one large cluster utilizing all the nodes or it can be provisioned into any number of smaller clusters. Clusters can be defined, but not used, so that a department or group can boot their cluster environment only when needed. Booting a specific cluster environment is fast and virtually independent of the number of node instances created. Revisions (software updates) for a given cluster instance can also be tracked with the ability to "roll back" cluster environments.

Flexible ACE Configuration

ACE supports flexible configurations based on customer needs. For environments, which require extremely high up-times, ACE provides redundant management servers and redundant networks. In environments, which are less demanding, ACE, can support non-redundant networks and management servers. Example hardware configurations are listed below.

Table 1: ACE Configurations

<table>
<thead>
<tr>
<th></th>
<th>ACE Lite</th>
<th>ACE Standard</th>
<th>ACE Scalable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Number of Nodes</td>
<td>34</td>
<td>144</td>
<td>20736</td>
</tr>
<tr>
<td>Multi-SU Support*</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>HA Management</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>HA Ethernet Network</td>
<td>No</td>
<td>Yes (Optional)</td>
<td>Yes</td>
</tr>
<tr>
<td>Dual Rail Infiniband</td>
<td>No</td>
<td>Yes (Optional)</td>
<td>Yes</td>
</tr>
</tbody>
</table>

*Multi-Scalable Unit Support requires sub-management node per SU.
Advanced Cluster Engine (ACE) Architecture

Illustrates the scalable ACE architecture. HA (High Availability) configured components are described below.

Figure 1: Advanced Cluster Engine (ACE) Architecture

The compute nodes are connected together by an InfiniBand (IB) network. This network can be single rail or multi-rail HA (High Availability) where there is more than one IB connection to each compute node. The multi-rail design provides both high performance and fail-over.

Fat-Tree, 2D, or 3D Mesh InfiniBand network configurations are supported and the IB interconnect fabric supports direct connectivity to global storage.

A hierarchical scalable management architecture is used for scalability. Sub-management servers are grouped with up to 144 compute servers. In the HA configuration, a pair of sub-management nodes is responsible for handling the management of a group of compute nodes. An Ethernet network connects the sub-management servers to the compute nodes.

The HA sub-management nodes operate in a stateless active-active mode where they essentially act as surrogates for the head management node. All management nodes (head and sub-management) and the Ethernet management network in HA configuration are redundant.

A fully redundant management network (dual Ethernet networks and dual sub-management servers) can tolerate a failed management node, switch, or cable and continue full management of the system.
The management functions provided by the Sub-Management nodes are:

- Relaying commands from the management server to the compute servers.
- Collecting and merging status from the compute servers and sending it to the main management servers.
- Monitoring the heartbeat signals from the compute servers and alerting the management servers when a compute server fails.
- Supporting DHCP PXE boot operations for the compute servers that are connected directly to a sub-management server group and the groups’ Ethernet switches.

Management Servers

Management servers are used to run the ACE software and can also provide file I/O to the compute nodes. Management servers can be redundant. If a management server or network fails the backup server will take over and the administrator will be notified of the failure.

System files and programs for diskless compute servers are stored on the Management Storage subsystem. This storage is configured to be both fault-tolerant and to have sufficient bandwidth to support the cluster. For small systems the Management Storage system is part of the redundant management servers.

The management network consists of Gigabit Ethernet (GbE) subnets connecting the compute servers in each group and the 10GbE networks that act as a backbone connecting the groups. The sub-management nodes are connected to the local operational (user) network segment and to the Management Storage by a pair of 10 GbE connections.

The operational network and its local subnets can have dual-rails and be fault-tolerant. A separate private network connects the redundant Management Server to the Sub-Management nodes and supports dual-rail GbE connections as well.

Sub-management Servers

The master management server controls sub-management servers. Sub-management servers control a sub-set of up to 144 compute nodes and are a proxy for the head management server. Sub-management servers allow the ACE management system to scale over large numbers of compute servers. Additionally, they provide scalable I/O support to the compute nodes.

Compute Servers

Compute servers are the heart of the cluster. They are the servers where the "number crunching" takes place. The compute servers do not require hard disks but they can be installed.
The operating system is made available to the compute servers by using a Network Block Device (Linux NBD allows you to remotely mount a hard disk from another computer at the device level.) Specific cluster images are then made available to the compute servers through the NBD allowing rapid booting and flexible provisioning of each node. The compute servers share a multi-rail InfiniBand network (explained below). The multi-rail InfiniBand network provides a balanced and redundant interconnect.

Understanding Host Images

ACE supports diskless compute nodes. On boot-up the management servers load host images for the compute nodes (using NBD mounts). Changes to the host images are managed using the ACE management tool. While many of the files are identical for a given cluster, there is some "state" information that is unique to each node (e.g. hostname, IP address, and other important files). This information allows each node to function as an independent Linux host. While much of the read-only files are shared among cluster nodes (provided by the management servers), there is a need for local state information while the node is running. To accommodate the nodes, there are two areas of state storage available. The first area is a ram-disk (a section of memory that acts like a hard disk). The ram-disk provides storage for /etc, /var, and /tmp.

The contents of these directories are initially provided by the read-only root file system for each cluster. As the node runs some of the files in these directories will change (e.g. log files, temporary files). Because changes to these files exist in RAM, all changes will be lost during a reboot of the node. Making changes to these files on a node is not recommended. In addition, because /tmp is located in a ram-disk, this is not a suitable location for users to store temporary files (i.e. the ram-disk may fill up quickly).

The second area is a shared administration file space available to each node. Changes can be made in the /root and /etc/ssh directories and will remain after a reboot and are part of the individual host image. When a host boots, these directories are initialized, by copying from the read only root file system. These files are stored on a shared per-host file system on the management nodes (i.e. each node will maintain these directories on the management server). Information written to these directories will not be lost on reboot.

Finally, users have access to their files in the /home (or similar) file system. There are several options for providing both global and parallel file system access to the nodes.

Revision Management

ACE provides a cluster revision system that enables you to make changes to a cluster software environment, track the changes, and "roll back" changes if necessary. Each cluster may have up to ten revisions. The following figure illustrates how ACE can manage multiple clusters and revisions. Each Cluster and subsequent revisions can be started and stopped using ACE.
Figure 2: Cluster Revision Management
Interconnect Schemes

ACE supports three types of interconnect topologies.

**Figure 3: Example Fat Tree Topology**

**Figure 4: Example 2D Grid Topology**

**Figure 5: Example 3D Grid Topology**
The ACE management system provides a Graphical User Interface (GUI) that enables you to manage the system. Changes to ACE are displayed in real time. Changes can be invoked from both the GUI interface and the command line interface (CLI). The GUI reflects changes made from the CLI after each ACE database update.

Preserved GUI Preferences

If a session is terminated normally, either by exiting from the Monitor menu, or clicking the dismiss icon at the top right corner of the screen, the GUI will save its current state prior to exiting and restore that state the next time it is started. Preserved states include user name, connection parameters, SSH tunnel parameters, passwords (on request only), the visible views and their locations, and the orientations and dividing bar locations of multi-panel views (e.g., switches and cluster views). Column order, sort order, widths and shown/hidden status are also preserved.

Installing the ACE GUI

The zip files containing the ACE GUI for a variety of architectures should be available on the management servers following installation of the ACE system, in /opt/ace/gui.

The files are named according to hardware, operating system and windowing system architecture. For instance, the ACE GUI zip file for Linux running on a 64 bit x86 CPU is named acegui-ace_version-linux-gtk-x86_64.zip.

Execute the following command to install: /opt/ace/gui/gui_install.sh. If the ACE GUI has already been installed in the same location, it is best to remove the original installation first.

Starting the ACE GUI

There are several different ways to start the ACE GUI, depending upon the architecture of the computer on which it is installed. In all cases, the name of the program to run is called eclipse. Eclipse is the name of the Java-based application platform on which the ACE GUI is built.

Logs and run-time preference files will be placed in the directory in which the ACE GUI is started. Most often, that should be the installation directory, but on many systems, it is possible for programs to start in different directories than their installation. This directory is at the user’s discretion.

Logging In

After the Cray splash screen the login window displays.
Figure 6: ACE Login Window

Enter the User name and Password to login. Enter the of your management server pair (typically mgmt). By default, the port to connect to is 9777, but if necessary, change this setting to the port number to which your management server is listening.

If your cluster is behind a firewall, administrators can SSH into any host inside the firewall that can directly reach the management server, administrators may want to create an SSH tunnel to create a secure connection to the management server through the firewall. Clicking on Details will expand the dialog to give access to those parameters.

Figure 7: ACE Login Dialog - SSH Tunnel Detail
Select **Create SSH Tunnel** to enable SSH tunneling login. Enter the name of the SSH server and the SSH port number (22 by default). If your user name and password are the same on the SSH server as on the management server, administrators can save some typing by selecting the **Use ACE user name** and **Use ACE user password** options, otherwise, fill in your user name and password for the SSH server in the SSH Tunnel Parameters area. Note that the ACE server name in the **ACE Connection Parameters** should now be the name of the management server as known by the SSH server. If the SSH server and management server are one and the same, the ACE server name can be set to localhost.

For convenience, you can select **Remember password** to keep the password for the next log in. If the ACE GUI remembers the password for you, it will store it in a preferences file in the current workspace in a form that would be difficult, but not impossible, for someone else to decode. Use with caution.

If the login is successful, the main window should be active as shown below. At the bottom left of the window, "connected" is displayed showing to which server the GUI has connected.

![ACE Main Window - Servers Tab View](image)

**Figure 8: ACE Main Window - Servers Tab View**

Note that there may be a brief delay before any content appears in the window. The duration of this delay depends upon the size of the cluster database and how busy the connected server is at the time. Users of smaller clusters may not notice any delay at all, while larger cluster users may experience a delay of several seconds during which the contents of the screen may show, for instance, that none of the switch ports are connected. This is due to the amount of time it takes to load and interpret the contents of the ACE database, and is completely normal.

Each view has several icons associated with it. To find out the meaning of the icon, place the mouse on the icon (do not click) and a short descriptive text will appear. If the icon is non-functional in the current context, it will be "grayed out" and no description will be available.
Windows can be arranged in many ways. Right click on a windows main tab to Detach, Move, and Resize the window. When a window is detached, it can be moved anywhere on the screen. Recall all windows to their default settings by clicking on the Views/Default Layout menu item (see below for menus).

Each view also has a series of small icons on the upper right corner like one of the two images below.

The small window with the padlock icon can be used to freeze the window output, i.e. updates from the ACE daemon will be held until the view is unfrozen.

Only the display is frozen, not the cluster. The inverted triangle icon, if present, can be used to pop up a menu controlling the content of the view. Minimizing and maximizing the view is controlled by the conventional small, large and overlapped rectangles.

ACE Menu Bar

There is a menu bar across the top with the following pull down menus:

**Monitor** – GUI control functions

**Views** – View control functions

**Action** – Component operations

**Help** - Information about the GUI

---

**Figure 9: ACE Main Window - Servers Tab View**

The following are available under the menu commands:

**Monitor**

**View/Edit Global Parameters** – start global parameters viewer/editor

**Exit** - Exit the GUI

**Views**

**Switches** - show the switches view

**Servers** - show the servers view

**Clusters** - show the clusters view

---

**SGE Hosts** - show the SGE hosts view

**SGE Queues** - show the SGE queues view
**SGE Jobs** - show the SGE jobs view

**Plugins** – show the available ACE Plugins

**Console** - displays the ACE messages console view  
**Progress** - displays the ACE task progress view

**Default Layout** – reset view layouts to their factory defaults

---

**Figure 10: Views Menu**

**Temps View**

The temperature ranges displayed in Temps view (see Figure 6) are listed in the following table.
Figure 11: Temps View Temperature Levels and Colors

Figure 12: ACE GUI Temps View
**Actions**

- **View/Edit Global Parameters** – start global parameters viewer/editor
- **Restart Subnet** – restart a subnet manager
- **Power On Switches** – power on one or more IB switches
- **Power Cycle Switches** – power cycle one or more IB switches
- **Power Off Switches** – power off one or more IB switches
- **Reboot Servers** – reboot one or more servers
- **Shutdown Servers** – halt one or more servers
- **Power On Servers** – power on one or more servers
- **Power Cycle Servers** – power cycle one or more servers
- **Power Off Servers** – power off one or more servers
- **Change Server State** – set/clear repair/maintenance state of one or more servers
- **Identify Servers** – identify one or more servers in the cabinet
- **Start Cluster** – start one or more hosts in a cluster instance
- **Stop Cluster** – stop one or more hosts in a cluster instance
- **Clone Cluster** - copy a cluster instance
- **Delete Cluster** - delete a cluster instance
- **Reboot Hosts** – reboot one or more hosts
- **Shutdown Hosts** – halt one or more host
- **Power on Hosts** – power on one or more hosts
- **Power Cycle Hosts** – power cycle one or more hosts
- **Power off Hosts** – power off one or more hosts
- **Change Host State** – set/clear repair/maintenance state off one or more servers
- **Identify Hosts** - Identify one or more servers
- **Check out Revision** – check out a cluster revision instance
- **Check in Revision** – check in a cluster revision instance
- **Activate Revision** – make a cluster revision instance bootable
- **Release Revision** – un-check out a cluster revision instance
- **Remove Revision** – delete a cluster revision instance
- **Run Plugin** – Run selected plugin.
Note that the actions are context sensitive and only those that apply to the current view/selection will be active. Here we see only server actions available for selection. Inactive actions will be grayed out. The action icons also appear in the top of a window (to the far right of the tab) when appropriate for a given context. Place the mouse on the icon (hover, do not click) to get context sensitive help.

![Action Menu](image)

**Figure 13: Action Menu**

**View/Edit Global Parameters**

From the Menu Bar Action menu, select View/Edit Global Parameters to bring up the view/edit dialog for the global parameters of the cluster:
The previous dialog shows detailed information about the global cluster parameters, and provides a means of altering the modifiable global properties. If any of the modifiable properties are changed, the Commit button will be enabled. Click it to submit the changes to ACE and dismiss the dialog. Note that there will be a several second delay before the changes are sent back from ACE to the GUI.

The All Properties section of this dialog shows all of the known global parameters in a tree format similar to how they are stored in the ACE database. This section is provided for completeness, but may change from version to version – make no assumptions about what should or should not be in this property tree.

**ACE Views**

There are six main views in the GUI: Switches, Servers, Clusters, SGE Hosts, SGE Queues and SGE Jobs. Additionally, there are two status-monitoring views, Console and Progress. It is possible to view/hide windows using the View menu items. If a view extends beyond the GUI window size, scroll bars will be used to help navigate the window. In addition, column names may be shortened due to space limitations and to see the full name of the column mouse over the column header.
The Switches Main Panel

To view the InfiniBand switches, select the switches tab in the top window. Switch information will be displayed in the window.

The first column is the switch name. If the IB topology is a fat tree, the L1, L2, L3 columns will show the position of the switch in that level of the fat tree. The L2 and L3 columns will only show if the depth of the fat tree requires it.

If the IB topology is a mesh or torus with the end switches along any axis connected to each other), instead of L1, L2 and L3 columns, there will be X and Y columns, for a 2D mesh or torus, and a Z column for a 3D torus. The X, Y and Z (if needed) columns show switch co-ordinates within the mesh or torus.

Subnet identifies which subnet the switch is in. Rack and Slot identify where the switch is physically located. The GUID is the InfiniBand identification number for the switch. The State column indicates the state of the switch (up or down). The ports column contains a summary of the states of the ports on the switch, with green indicating an up condition; red, down; yellow, up but slow port speed detected; and gray, not connected. The organization of the port status lights in the ports column should be the same as on the back-panel of the switch itself. On a 24 port switch, from left to right, the bottom 12 lights are ports 1-12, and the top 12 are ports 13-24. On a 36 port switch, there will be 18 lights per row, numbered 1-18 and 19-36 respectively.

The Switch View/Edit Pop-up

Right-click on any switch, to display a context menu similar to this.
Select View/Edit Switch to bring up the view/edit dialog for the selected switch:

![View/Edit Switch Dialog](image)

**Figure 14: View/Edit Switch Dialog**

This dialog shows the detailed information about the selected switch, and if there were any modifiable properties would provide the means for making changes to those properties.

The **All Properties** section of this dialog shows all of the known global parameters in a tree format similar to how they are stored in the ACE database. This section is provided for completeness, but may change from version to version – make no assumptions about what should or should not be in this property tree.

**The Ports Panel**

There is also a Ports panel in the Switches view, which shows more detailed information about each port on the currently selected switch. It will be blank if either 0 or more than one switch is currently selected in the main panel.

The Port column shows the number of the port. The Rate column shows the expected transfer rate of the port in gigabits per second (Gb/s). The MRate column shows the most recently measured rate of the port in Gb/s. The state column shows the state of the port, which should match the corresponding port in the main panel’s Ports column.
The “Connected To” columns show to what the port is connected, either to another switch, or to a server, and the location of that component in both the IB topology (L1, L2, L3/X, Y, Z) and the physical layout of the cluster (Rack and U#). Note: Ports that are not connected will not be shown.

**The Port View/Edit Pop-up**

Right-click on any port in the Ports panel, display a context menu similar to this:

![Context Menu](image)

Select View/Edit Port to bring up the view/edit dialog for the selected port. This shows detailed information about the selected port, any modifiable properties, and provides the means for changing those properties.

![View/Edit Port Dialog](image)

*Figure 15: View/Edit Port Dialog*

The *All Properties* section shows all of the known properties in a tree format similar to how they are stored in the ACE database. This section is provided for completeness, but the contents are subject to change from version to version.
The Servers Main Panel

![ACE Servers Window](image)

**Figure 16: ACE Servers Window**

To view the list of servers in your cluster, click the servers tab in the top window.

In the above example, there are many servers named "server-0001 – server-XXXX" (You may need to use the scroll bar to see all the servers.) As mentioned previously, servers perform different tasks in the Cluster. The Server Type column identifies what type the server is, either management, group (sub-management), or compute. In this example, server-0001 and server-0002 are the dual/redundant management servers.

You can also search for specific strings within this window using the “search” input boxes at the top of the pane and use the generic search input or search by group or rack number.

In an IB network with a fat tree topology, the L1 column shows the L1 switch that the server is connected to. In a mesh or torus topology, the X and Y columns (2D) and Z column (3D) show the co-ordinates of the switch that the server is connected to. The network node consists of a pair of switches (one on each of the dual InfiniBand networks), and the associated servers connected to those switches. The N column shows the logical index of the server within the network node.

The Grp column is the sub-management group number. Rack is the number of the rack in which the server is located. Slot/U# is the physical location within the rack. Network state columns Net1 and Net2 show whether heartbeat status has been received recently on each of the 2 Ethernet management networks.
There are multiple sever states. The possible states are as follows:

- **off** – a power off command has been issued to the server
- **on** – a power on command has been issued to the server
- **cycle** – a power cycle command has been issued to the server
- **shutdown** – a shutdown command has been issued to the server
- **reboot** – a reboot command has been issued to the server
- **unknown** – no data
- **boot** – server is booting (first boot stage)
- **boot_host** – server is booting an ACE host (second boot stage)
- **boot_os** – server is booting the O/S (third boot stage)
- **boot_fail** – the server failed to boot for some reason
- **active** – the server is actively running as an ACE host
- **maintenance** – the server is offline for maintenance
- **repair** – the server is offline for repair
- **fail-on** – power on command failed to power on the server
- **fail-off** – power off command failed to power off the server

The BIOS and HCA columns show the respective firmware versions installed on the server.

**The Server View/Edit Pop-up**

Right-click on any server in the main panel to get the context menu below.

Select **View/Edit Server** to bring up the view/edit dialog for the selected server shown below.
This dialog shows detailed information about the server. Under the BIOS properties section, you will find the vendor, version, revision and date of the BIOS installed on the server. If these values are different from the reference value in the global area of the database, the name of the property will be prefixed with an asterisk (*). To set these values as the global reference for this type of server, select **Set as global ... server reference.** Do the same for CMOS and HCA properties, if desired.

The management group and compute servers may have different architectures, thus there are reference values for each of the three different types of server.

If any of these fields are selected, the **Commit** button will be enabled. Clicking on the **Commit** button will submit the requested changes to ACE and dismiss the screen. Note that there may be a delay of a few seconds before any resulting changes in the database get sent back to the GUI.

The **All Properties** section of this dialog shows all of the known global parameters in a tree format similar to how they are stored in the ACE database. This section is provided for completeness, but may change from version to version – make no assumptions about what should or should not be in this property tree.

**The Clusters Main Panel**

Clusters can be viewed by selecting the Clusters tab in the top window.
The first column is the cluster name. The second and third columns are a short description of the cluster and the Kernel version, respectively. Each cluster has 4 networks. IP1 and IP2 are the redundant Ethernet operational networks to the node and IP3 and IP4 are redundant InfiniBand networks. The network base addresses for the cluster on these networks are shown in the four columns labeled IP1, IP2, IP3, and IP4. The Rev column is the currently active cluster revision number below. The Revs column is the number of revisions available for the cluster. The Hosts column shows the number of hosts the cluster has assigned to it. The final column is the cluster state.

There are two values for state:

init - the cluster is initializing. Creating the cluster can take a few minutes.

ready - the cluster is ready to use.

The Cluster View/Edit Pop-up

If you right-click on any server in the main panel, you will get a context menu similar to this.

Select View/Edit Cluster to bring up the view/edit dialog for the selected cluster:
This dialog shows detailed information about the cluster, and provides a means of altering the modifiable properties of a cluster. If any of the modifiable properties are changed, the **Commit** button will be enabled. Click it to submit the changes to ACE and dismiss the dialog. Note that there will be a several second delay before the changes are sent back from ACE to the GUI.

The **All Properties** section shows all of the known properties in a tree format similar to how they are stored in the ACE database. This section is provided for completeness, but the contents are subject to change from version to version.

### The Hosts Panel

To view the cluster hosts, locate the Hosts panel on the Clusters view. Each node has 4 connections. IP1 and IP2 are IP address assigned for the redundant Ethernet connections to the node and IP3 and IP4 are IP address assigned for the redundant InfiniBand connections. Rev is the image revision for the operating system for the host. The C/O (Checkout) column indicates whether the host is being used as a revision host. A “checked out” icon
Figure 18: Checked Out Icon

indicates that the host has been booted with a writable root file system. If the revision has not been checked out, then the root file system is read-only. Finally the Server column indicates on which server the host instance is running.

The Host View/Edit Pop-Up

If you right-click on any server in the main panel, you will get the context menu below.

Figure 20: Cluster Host - View/Edit Dialog

Select View/Edit Host to bring up the view/edit dialog for the selected host:

This dialog shows the detailed information about the selected host, and if there were any modifiable properties would provide the means for making changes to those properties.

The All Properties section shows all of the known properties in a tree format similar to how they are stored in the ACE database. This section is provided for completeness; however, the contents are subject to change from version to version.

The Revisions Panel

To view the cluster’s revisions, locate the Revisions panel on the Clusters view below. This panel shows the revisions of the selected cluster. The date and time the revision was created is shown in the first column. The revision number is listed in the second column and a description of the revision is listed in the third column.
The State value indicates one of three state values for the cluster. The available states are:

- **init** – the revision is initializing
- **ready** – the revision is ready for use
- **active** – the revision is active

The C/O column indicates whether the revision has been checked out. A “checked out” icon

The checked out Icon indicates that the revision is checked out. The Host column indicates on which host the checked out version is running, and the Server column indicates the server that the host instance is running on. Once checked out, the revision can be modified on the indicated host.

Note that checking out a revision does not modify the revision being checked out. When the revision is checked out, a new provisional revision is created as a copy of the original. This provisional revision is not assigned a revision number until the revision is checked in, at which time it is made permanent and un-modifiable.

**Revision View/Edit Pop-up**

If you right-click on any revision in the revision panel, you will get a context menu similar to this.

![Revision View/Edit Pop-up](image)

**Figure 19: Cluster - Revision Popup**
Select View/Edit Revision to bring up the view/edit dialog (below) for the selected revision:

This dialog shows detailed information about the revision, and provides a means of altering the modifiable properties of a revision. If any of the modifiable properties are changed, the Commit button will be enabled. Click it to submit the changes to ACE and dismiss the dialog. Note that there will be a several second delay before the changes are sent back from ACE to the GUI.

![View/Edit Revision](image)

Figure 22: Cluster Revision - View/Edit

The All Properties section shows all of the known properties in a tree format similar to how they are stored in the ACE database. This section is provided for completeness; however, the contents are subject to change from version to version.

**Racks Tab**

The Racks Tab is an easy way to view the current state of individual racks in your cluster. Right-click the indicated issue to jump directly to that tab/component in the GUI.
The ACE Console

The ACE console provides a way to view messages from ACE as it manages the system. Alerts and faults in the other windows will produce a text error message that is viewable in this window.

The Console window shows a subset of the logs generated by the ACE daemon running on the management node. Non-debug logs are sent periodically to the GUI and will be displayed in the console window.

The Console will maintain, at a minimum, the last 64KB of message text sent by the daemon, up to 256KB of message text maintained. When the upper threshold is exceeded, the oldest 192KB of message text will be discarded to bring the total down to the lower threshold. The original messages can be found in the ace daemon log on the ace server.

On GUI startup, the last 100 logs are sent initially to the GUI and then new logs are sent to the GUI as they occur.

Also, the Console View has some additional icons.

The icon with a paper and a small X can be used to clear the contents of the console window. Similar to the other windows, the console can be locked to prevent movement of messages.

There are three console outputs available.

The ACE Log Console: displaying the ACE daemon messages.
The Ace Monitor Console: providing debug output from the ACE GUI.

The ACE Plugin Console: displays progress/output from ACE plugins.

The small monitor icon can be used to select between these two options. The window with a plus in its top right corner will allow you to create another log viewer, so you could watch both ACE daemon and GUI debugging logs side-by-side, if need be.

Figure 20: ACE Monitor Console

Figure 21: ACE Log Console

Figure 22: ACE Plugin Console

Progress Tab

Any ACE action that is not done immediately will show its progress in the progress window under normal operation.
Starting a Cluster Using the ACE GUI

Any number of cluster software environments can be created using ACE. These software environments or cluster instances are not active running clusters until they are started by ACE. Cluster hosts can be started or stopped using either the ACE GUI or the ACE CLI (ace) tools. Using ACE, multiple clusters of the same type can be started at the same time.

To start a cluster using the ACE GUI, start the GUI and click the clusters tab. Select a cluster from the list of available clusters and press the green start button in the upper right corner of the window. Below, the cluster called "prod" will be started. Notice that cluster prod has no hosts assigned (see the Hosts panel below the main panel, on the left). In addition, revision 4 of prod will be started because it is the active revision (see Revisions panel below the main panel, on the right).

Once the start icon is pressed, a dialog will pop up.
Figure 25: Start Cluster: Selecting Hosts

In this example, we are starting 31 hosts. The number of hosts that the user can select is limited by the GUI to the number of hosts not already assigned to a cluster. In addition, the available servers are listed in the scrolling window and are individually selectable as above. The minimum number of hosts to start is limited to the number of hosts selected in the list, and the number to start will be adjusted as necessary as more hosts are selected. ACE will start the selected hosts and will select from the remaining available servers any additional hosts required to make up the requested total.

Figure 26: Start Cluster: Process Window

After clicking on OK, a progress window will display while the cluster is starting hosts. Note, however, that deselecting a host from the list will not correspondingly reduce the number to start.

Figure 27: Start Cluster: Confirmation

If you look at the Clusters View you will notice that cluster prod now has 31 hosts running as indicated in the Hosts column. If there are still unassigned hosts, these steps may be repeated to add more hosts to the cluster.
Clicking the Cancel button while the operation is pending will not stop ACE from starting any hosts on the cluster. It will, instead, cancel the GUI’s expectation of a response from the server. This is because the GUI doesn’t actually perform the work required to start hosts in the cluster, but rather requests that operation from ACE. Presently, this is true for all actions performed from the GUI.

### Stopping a Cluster Using the ACE GUI

To stop a cluster, first select the cluster name and press the red stop icon in the upper right hand corner.
**Figure 36: Stop a Cluster**

A popup dialog asking how many nodes to stop will be presented.

**Figure 37: Stop Cluster: Hosts Selection**

The number of hosts that the user can select is limited by the GUI to the number of hosts assigned to the cluster. In addition, the assigned servers are listed in the scrolling window and are individually selectable. The minimum number of hosts to stop is limited to the number of hosts selected in the list, and the number to stop will be adjusted as necessary as more hosts are selected. ACE will stop the selected hosts and will select from the remaining assigned hosts any additional hosts required to make up the requested total.
In this example, we stopped 31 hosts. As with starting a cluster a progress bar will be shown during the actual shutdown. Completion is shown below.

Creating/Cloning a Cluster Using the ACE GUI

Clusters are created by copying (cloning) the installed default cluster images or by importing provided cluster images.

The default cluster images can be used to start a cluster or as the basis for other clusters or revisions. The default cluster images should not be deleted as they are a working baseline cluster.

The default cluster images are usually based on the management server O/S

Contact Cray for the availability of a particular Linux Distribution.

Most importantly, the default cluster images have all the necessary software that is needed for cluster hosts to function properly within the ACE hardware environment.

The ACE management tool allows for a cluster to be copied and used as basis for another cluster. In addition each cluster can support up to ten revisions. A revision is a change to the underlying software environment of the cluster, that is a new package, updated driver, or other common changes that are made in a cluster environment.
A cluster can be cloned using the ACE GUI. To clone a cluster, first select the cluster tab and select cluster image to clone. In this example, we will use the prod cluster. Select the **Clone Cluster** in the actions menu or select the clone icon in the upper right hand corner from the cluster tab.

A dialog will pop up and prompt for the new cloned cluster name.

**Figure 41: Clone Cluster: Description Dialog**

Choose a name that is different from existing clusters and does not contain any whitespace.

**Important!: Cluster names must not contain special characters other than the underscore character “_”.

The initial description in the dialog will be “clone of …”, but you may change it. The Clone Revision will default to the current active revision of the cluster being cloned. An alternate revision may be selected if needed. Once the cluster is cloned, a window will
open indicating the process has been completed. Be aware the cluster will be in the Init state while the image is being built.

Figure 42: Clone Cluster: Status Initializing

Now the cluster is complete and ready for checkout.

Figure 29: Clone Cluster: Status Ready for Checkout

After the cluster is in the Ready state, a version can be checked out. Remember you should have a cluster checkout only once. Multiple checkouts could cause sync issues. To check out a revision, first highlight the Test cluster and then highlight the Initial Revision in the Revisions panel. Select Check Out Revision from the Action menu or select the checkout Revision icon from the Revisions panel.
A dialog will pop up. Select an available server to serve as a revision host from the drop-down list, then click OK to begin the checkout process.

A progress dialog will open while a server is started and the initial revision of Test is loaded onto the server. A Checkout Confirmation follows this.
Figure 32: Clone Cluster: Checkout Confirmation

The name of the new revision host can be found in the Revisions panel.

Figure 33: Clone Cluster: Revisions Panel

Alternatively the name of revision host can be found in the log window in a message similar to the following:

2011-12-05 05:12:29.790 INF Host Test-r001 started on server-0033
2011-12-05 05:12:29.791 INF Cluster Test revision 1 checked out

In our example, the host test-r001 has been started on server-0033.

At this point you can log onto the revision host as root and make changes to the cluster revision (e.g. add or update packages).

Once modifications are complete and you are satisfied with the changes, the revision can be checked back in by highlighting the revision in the Revisions panel and selecting Check in Revision in the Actions menu or using the Check In icon in the upper right hand corner of the Revisions panel.

A confirmation dialog displays.
Once the revision is checked in, you will see a confirmation.

If you inspect the Revisions Pane you will notice a new revision (revision 2). The revision may take some time to become ready i.e. it may be in the init state for a few minutes.

Once it is ready, it must be activated to be used on the cluster “Test”. ACE will automatically increment the revision number. Also, the host that was running the cluster revision is now stopped.

In order to use the new revision, it must be activated (i.e. set as the bootable revision for this particular cluster). A cluster is activated by selecting a revision in the Revisions
panel and clicking the green icon in the upper right hand corner of the pane or selecting Activate Revision in the Actions menu.

Figure 37: Clone Cluster: Activate Revision Dialog

In our example we will activate revision 2. The following confirmation window will appear: After you press OK, the completion window will appear:

Figure 38: Clone Cluster: Activate Revision Success

There can only be one active revision. The revision is marked active as shown below.

Figure 39: Clone Cluster: Revision Active and Running

To start the cluster with your new revision follow the procedure described in the section above. The cluster “Test” (revision2) is shown running on one server as shown below.
Previously created revisions can also be removed using the delete icon in the upper right corner of the Revisions window or from the Action menu. The active revision cannot be removed thus you will need to make another revision active to delete.

![Figure 40: Clone Cluster: Revision Pane - Delete Revision](image1)

After clicking on the Revision you want to delete, click the remove button.

![Figure 41: Clone Cluster: Deletion of a Revision](image2)

After clicking OK to delete.
Deleting a Cluster Using the ACE GUI

Entire clusters can be deleted using the ACE GUI. The cluster must be stopped and cannot be running on any nodes. To delete a cluster, simply stop the cluster, highlight the cluster, and click the red X icon above the cluster window or select Delete Cluster from the Action menu.

A confirmation window will be displayed.

After clicking OK, the completion dialog will be displayed.
The entire cluster and all revisions will be deleted. It is not possible to delete a running cluster.

Controlling Power with the ACE GUI

**CAUTION!**: Changing the power status of servers in a running cluster can cause job interruption and termination.

![ACE GUI screenshot](image)

**Figure 45: Server Power Control: Right Mouse Button Click**

Power to servers can be controlled using the ACE GUI. The power options are power on, power off, power cycle (on/off/on). As an example to power cycle a host, simply select the server or servers in the server window and then select the power cycle icon. The example provided below is an example of a right mouse button click on a specific server.

The Power Cycle Servers option in the Action menu can also be used. A confirmation window will be displayed as shown below. Select OK to power cycle the node. A progress dialog will be displayed.
By default, power cycling (or rebooting, halting, or powering off or on) one or more servers will cause all affected servers not presently running the active revision of their respective clusters to be booted with the new revision the next time they boot. Select Boot current revision to allow such hosts to keep the revision they were last booted with.

After the server has been re-powered, it will reload the cluster OS and a confirmation dialogue displays. Multiple servers can be similarly powered off/on.

---

**How To Find The ACE Version**

To find the version of ACE simply type

```
$ ace version
```

To find out the version of the ACE GUI, simply select the *Help/About ACE* menu item and locate the GUI Version and build date as shown below.
Figure 48: ACE GUI Version Window

ACE GUI Window Column Sorting

All tables shown in an ACE GUI view have a sort order associated with them. When first displayed, the table will be sorted by the first column in ascending order, with a few exceptions.

The sort order of a table can be altered by clicking on the column you wish to sort. For example, when first displayed, the servers view might have looked as below:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>N</th>
<th>Grp</th>
<th>Rack</th>
<th>Slot/Ch</th>
<th>Blade</th>
<th>Host</th>
<th>Net1</th>
<th>Net2</th>
<th>State</th>
<th>BOS</th>
<th>HCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>server-0001</td>
<td>management</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>falcon1</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>5.5.0.0.468...0020101005</td>
<td>2.7.700</td>
</tr>
<tr>
<td>server-0002</td>
<td>management</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>falcon2</td>
<td></td>
<td></td>
<td></td>
<td>unknown</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server-0003</td>
<td>compute</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>pre001</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0004</td>
<td>compute</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>pre002</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0005</td>
<td>compute</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td>pre003</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0006</td>
<td>compute</td>
<td>6</td>
<td>1</td>
<td>1</td>
<td>6</td>
<td>pre004</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0007</td>
<td>compute</td>
<td>7</td>
<td>1</td>
<td>1</td>
<td>7</td>
<td>pre005</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0008</td>
<td>compute</td>
<td>8</td>
<td>1</td>
<td>1</td>
<td>8</td>
<td>pre006</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0009</td>
<td>compute</td>
<td>9</td>
<td>1</td>
<td>1</td>
<td>9</td>
<td>pre007</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0010</td>
<td>compute</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>10</td>
<td>pre008</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0011</td>
<td>compute</td>
<td>11</td>
<td>1</td>
<td>1</td>
<td>11</td>
<td>pre009</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0012</td>
<td>compute</td>
<td>12</td>
<td>1</td>
<td>1</td>
<td>12</td>
<td>pre010</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0013</td>
<td>compute</td>
<td>13</td>
<td>1</td>
<td>1</td>
<td>13</td>
<td>pre011</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0014</td>
<td>compute</td>
<td>14</td>
<td>1</td>
<td>1</td>
<td>14</td>
<td>pre012</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0015</td>
<td>compute</td>
<td>15</td>
<td>1</td>
<td>1</td>
<td>15</td>
<td>pre013</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0016</td>
<td>compute</td>
<td>16</td>
<td>1</td>
<td>1</td>
<td>16</td>
<td>pre014</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
<tr>
<td>server-0017</td>
<td>compute</td>
<td>17</td>
<td>1</td>
<td>1</td>
<td>17</td>
<td>pre015</td>
<td></td>
<td></td>
<td></td>
<td>active</td>
<td>080016</td>
<td>2.7.626</td>
</tr>
</tbody>
</table>

Figure 49: Default Sort Order

Notice the pale triangle to the right of the Name column header, pointed upward. This indicates that the Name column is sorted in ascending order. To change the sort order to descending, click on the column header, and the table will be re-sorted and look something as below:
Figure 50: Descending Sort Order

The pale triangle is now pointed downward, indicating that the column is sorted in descending order. Another click would remove the sort criterion from the column altogether. The actual order that the servers would appear in this case is undefined, essentially random.

It is also possible to sort by more than one column. For instance, to sort by Name and Host, first click on the Name column header, then hold down the Ctrl key while clicking on the Host column header. This will re-sort the table and look something like the below:

Figure 51: Multi-column Sort Order

Notice that now instead of simply a triangle pointing in the direction of the sort, there is also a number. This number shows what order the columns are used to sort the table. They only show if there are two or more columns in the sort criteria for the table.

ACE GUI Column Ordering

Columns can be re-ordered by dragging their column headers to new positions in the table. For instance, the Type column can be moved to be column 1 by clicking it and dragging it to the left.
ACE GUI Resizing Columns

The columns can be resized by clicking and dragging the divider between two column headers. Each column has a built-in minimum size based on the expected contents of the column, and cannot be resized below this minimum width.

Resizing a column will resize and/or move adjacent columns to maintain the built-in minimum width for all columns. When a view is narrower than the total of the minimum widths of the visible columns in it, the view will show a left/right scroll bar, the columns will be set to their minimum widths, and resizing for the columns will be disabled.

ACE GUI Hiding Columns

Columns can be hidden from view on request by the user. Clicking the inverted triangle will bring up a menu of several choices controlling the contents of the view. For each pane in the view, there will be a Show/Hide Columns selection, which when selected will display a list of all defined columns for the panel, with checks beside those columns that are shown.

The rest are hidden. Note that on a few views, there are some columns that are disabled due to the topology of the IB network. Columns that are not needed because the topology doesn’t require it are disabled and cannot be shown.

The rest are hidden. Note that on a few views, there are some columns that are disabled due to the topology of the IB network. Columns that are not needed because the topology doesn’t require it are disabled and cannot be shown.
ACE GUI Moving Views

A view can be moved to another location in the main window, or completely detached from the window. Click on the view’s tab and drag it. After you begin the drag, an indication of where the window will be placed when dropped will be drawn on the screen. Below, the SGE Queues view was dragged to the right to place it below the SGE Hosts view.

Dragging the view out of the window entirely will detach the view from the window. Unneeded views can be dismissed by clicking the X in the view’s tab. The view can always be restored later by selecting it from the View menu at the top left of the window.
**Figure 54: ACE GUI Moving Views – Detach Window**
ACE Command Line Reference (CLI)

The Cray Advanced Cluster Engine (ACE) command line interface is designed for administration when a GUI is not available or desired. The ACE command line interface (CLI) command is `ace` and is installed in `/opt/ace/bin`, which should be added to your execution $PATH variable.

All ACE commands are entered by first typing `ace` then the command. As in:

```
$ ace <command> <optional-arguments>
```

If you need an example of arguments then type “ace <command>” the usage will be provided.

```
$ ace start
usage: ace start <cluster> <num-hosts> [ <server> ... ] [ -- <host-id> ... ]
```

Command Line Arguments

Normally typed commands as well as wild card processing has been added to the host expansion of `ace <command> -h <cluster-compute-nodes>` for ACE versions greater than 2.0-20140306-XXXX as follows:

The following rules exist (in order of processing):

- Expansion according to ranges: `prod-0[1-2][3-4]0` gets expanded to "prod-0130 prod-0140 prod-0230 prod-0240". Previously only a single range was expanded.

- Then wild card processing (*) - just like bash shell – match any number of characters (?) - just like bash shell – match a single character

- You may escape the wild cards so that they are not inadvertently interpreted by the user's shell of choice:
  - `ace cycle -h prod*` Cycle all nodes that have a cluster name starting with the string "prod"
  - `ace cycle -h prod-000\`? Cycle all nodes that have a cluster name starting with "prod-000" and *one* trailing character. This will not match "prod-001a"
  - `ace cycle -h prod-00[0-1]\`? Cycle all nodes that have a cluster name starting with "prod-000" and/or "prod-001" and have a single trailing character.
  - `ace cycle -h \*[0-1][0-1]\`* Cycle all nodes that have a cluster name starting with ANYSTRING00ANYSTRING, ANYSTRING01ANYSTRING, ANYSTRING10ANYSTRING, ANYSTRING11ANYSTRING Of course, just like most shells, "ANYSTRING" could be ""
General Commands

ACE supports the following CLI commands:

- **ace help**: Print this help message
- **ace ping**: Check connectivity
- **ace date**: Show UTC date
- **ace time**: Show time
- **ace log_flush**: Flush logs to disk
- **ace debug_on**: Turn debug logs on
- **ace debug_off**: Turn debug logs off
- **ace version**: Show ACE version

Status Commands

- **ace switches**: Show switch status
- **ace ports**: Show detailed switch port status
- **ace ethswitches**: Show Ethernet switch status (If your cluster is Ethernet only)
- **ace servers**: Show server status
- **ace clusters**: Show cluster status
- **ace hosts**: Show host status
- **ace revisions**: Show revision status
- **ace get_logs**: Show logs
- **ace loads**: Show load status per server
- **ace memory**: Show memory usage status per server
- **ace temps**: Show CPU temp status per server
- **ace mics**: Show MIC node status
For example, to see specific servers you can enter the command:

\$ ace servers

In the following example, there are 12 servers and as mentioned, servers perform different tasks in the cluster. The next column is server type. Server-0001 and server-0002 are the dual/redundant management servers. The server type Group is a sub-management server. There is a pair of sub-management servers for approximately every 160 compute nodes. This is called a sub-management group.

The sub-management servers boot from the management servers and they serve as a boot server and caching server for their compute nodes. The heartbeat status from compute servers is aggregated by the sub-management server and sent to the management server. The root file systems are exported to the sub-management server and from there to the compute servers.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>N</th>
<th>Grp</th>
<th>Rack</th>
<th>Slot</th>
<th>Host</th>
<th>Net1</th>
<th>Net2</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>server-0001</td>
<td>management</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-</td>
<td>up</td>
<td>up</td>
<td>active</td>
</tr>
<tr>
<td>server-0002</td>
<td>management</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>-</td>
<td>down</td>
<td>down</td>
<td>unknown</td>
</tr>
<tr>
<td>server-0003</td>
<td>group</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>sysgrp-0001</td>
<td>up</td>
<td>up</td>
<td>active</td>
</tr>
<tr>
<td>server-0004</td>
<td>group</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>sysgrp-0002</td>
<td>up</td>
<td>up</td>
<td>active</td>
</tr>
<tr>
<td>server-0005</td>
<td>compute</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td>prod-0001</td>
<td>up</td>
<td>up</td>
<td>active</td>
</tr>
<tr>
<td>server-0006</td>
<td>compute</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>1</td>
<td>6</td>
<td>prod-0002</td>
<td>up</td>
<td>up</td>
<td>active</td>
</tr>
<tr>
<td>server-0007</td>
<td>compute</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>1</td>
<td>7</td>
<td>prod-0003</td>
<td>down</td>
<td>down</td>
<td>fail-on</td>
</tr>
<tr>
<td>server-0008</td>
<td>group</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>8</td>
<td>1</td>
<td>1</td>
<td>8</td>
<td>sysgrp-0003</td>
<td>up</td>
<td>down</td>
<td>active</td>
</tr>
<tr>
<td>server-0009</td>
<td>group</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>9</td>
<td>1</td>
<td>1</td>
<td>9</td>
<td>sysgrp-0004</td>
<td>up</td>
<td>down</td>
<td>active</td>
</tr>
<tr>
<td>server-0010</td>
<td>compute</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>10</td>
<td>prod-0004</td>
<td>up</td>
<td>down</td>
<td>active</td>
</tr>
<tr>
<td>server-0011</td>
<td>compute</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>11</td>
<td>1</td>
<td>1</td>
<td>11</td>
<td>prod-0005</td>
<td>up</td>
<td>down</td>
<td>active</td>
</tr>
<tr>
<td>server-0012</td>
<td>compute</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>12</td>
<td>1</td>
<td>1</td>
<td>12</td>
<td>-</td>
<td>down</td>
<td>down</td>
<td>off</td>
</tr>
</tbody>
</table>

The X, Y, Z, N columns describe the location of the server in the InfiniBand network topology. In a mesh or torus InfiniBand network, X, Y, Z are the position of the network node to which the servers are connected.

The network node consists of a pair of switches (for dual InfiniBand networks), and the associated servers connected to those switches. In a fat-tree InfiniBand network, X, Y, Z are the one-relative index of the level 1 (X), level 2 (Y), or level 3 (Z) network node. N is the logical index of the server within the network node.

The Grp column indicates the group number to which the server belongs. Slot is the physical slot in the rack. Network states show connectivity from the management server to the particular server and are based on the heartbeat status.

Finally there are multiple server states. The states are as follows:

- **off** – a power off command has been issued to the server.
- **on** – a power on command has been issued to the server.
- **cycle** – a power cycle command has been issued to the server.
- **shutdown** – a shutdown command has been issued to the server.
- **reboot** – a reboot command has been issued to the server.
- **unknown** – no data.
boot – server is booting (first boot stage).

boot_host – server is booting an ACE host (second boot stage).

boot_os – server is booting the O/S (third boot stage).

boot_fail – the server failed to boot for some reason.

active – the server is actively running as an ACE host.

maintenance – the server is being repaired (power control commands will still work).

repair – the server is being repaired. Treated as if it doesn't exist (no power control).

fail-on – tried to turn the server on, but it didn't come on.

fail-off – tried to turn the server off, but it didn't turn off.

- – no data of state (cleared state)

uptodate – connected secondary mgmt. node.

synching – drbd syncing state between primary and secondary mgmt. node.

disconnect – lost secondary mgmt. node connection.

To see status and location of the InfiniBand switches.

$ ace switches

The values of X, Y, Z, N, Rack, Slot are similar to the ace servers command.

The Type value indicates one of four type values for the switch. The available type are:

core - core switch, typically a 36port used ad level1 or level2 switch connected to other switches.

spine- spine module in a director class switch.

leaf - leaf module in a director class switch (connect to node or edge switch).

drive - edge switch, the switch is connect from edge to node.
The GUID is the InfiniBand identification number for the switch. The State column indicates the state of the switch. The Ports value indicates one of three ports values for the switch. The available ports are:

- underscore “_” is not configured in the ACE database
- 0 IB link not connected
- 1 IB link connected
- X Misconnected IB link

<table>
<thead>
<tr>
<th>Name</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>N</th>
<th>Rack</th>
<th>Slot</th>
<th>Type</th>
<th>GUID</th>
<th>State</th>
<th>Ports</th>
</tr>
</thead>
<tbody>
<tr>
<td>switch-0001</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td></td>
<td>edge</td>
<td>0x0002c9020040c2c8</td>
<td>up</td>
<td>11111111111111</td>
</tr>
</tbody>
</table>
Enter the command below and the following will be displayed.

```
$ ace ports
```

<table>
<thead>
<tr>
<th>Switch</th>
<th>Port</th>
<th>Rate</th>
<th>MRate</th>
<th>State</th>
<th>Destination</th>
<th>DPort</th>
<th>Rack</th>
<th>Slot</th>
</tr>
</thead>
<tbody>
<tr>
<td>switch-0001</td>
<td>1</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0005</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>switch-0001</td>
<td>2</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0006</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>switch-0001</td>
<td>3</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0007</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>switch-0001</td>
<td>4</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0008</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>switch-0001</td>
<td>5</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0009</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>switch-0001</td>
<td>6</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0010</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>switch-0001</td>
<td>7</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0011</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>switch-0001</td>
<td>8</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0012</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>switch-0001</td>
<td>33</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0004</td>
<td>1</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>switch-0001</td>
<td>34</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0003</td>
<td>1</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>switch-0001</td>
<td>35</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0002</td>
<td>1</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>switch-0001</td>
<td>36</td>
<td>56</td>
<td>56</td>
<td>up</td>
<td>server-0001</td>
<td>1</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

Entering the below command will display the following.

```
$ ace clusters
```

Similar to the `ace host` command above, IP1-IP4 are the networks used by the cluster. These are base addresses which are automatically assigned when the cluster is created/cloned. The IP addresses of the hosts in that cluster will start at those base addresses.

The `Rev` column is the cluster revision number. The `Revs` value is the number of revisions available for the cluster.

Hosts is the number of hosts the cluster is running and a 0 means the cluster is defined, but it is not running on any nodes.

The final column is the cluster state. There are two values for state:

- **init** - the cluster is initializing. Creating the cluster can take a few minutes.
- **ready** - the cluster is ready to use.

```
$ ace revisions
```

The revision number is listed in the second column. A description of the revision is listed in the third column. The host and server fields indicate whether the revision has been checked out. Once checked out, the revision can be modified / updated on that host. The `State` value indicates one of three state values for the cluster. The available states are:

- **init** – the revision is initializing
- **ready** – the revision is ready for use
- **active** – the revision is active (the revision that will be used when the cluster is started)

If the column `Checkout` has a yes this indicates the revision has been checked out.

```
<table>
<thead>
<tr>
<th>Name</th>
<th>Revision</th>
<th>Description</th>
<th>Kernel</th>
<th>State</th>
<th>Checkout</th>
<th>Host</th>
<th>Server</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>prod</td>
<td>1</td>
<td>Initial revision</td>
<td>2.6.32-431.el6.x86_64</td>
<td>active</td>
<td>no</td>
<td>-</td>
<td>-</td>
<td>201410150639.20</td>
</tr>
<tr>
<td>sysgrp</td>
<td>1</td>
<td>Initial revision</td>
<td>2.6.32-431.el6.x86_64</td>
<td>active</td>
<td>no</td>
<td>-</td>
<td>-</td>
<td>201410150528.33</td>
</tr>
</tbody>
</table>
```
Each host node has 4 connections. IP1 and IP2 are the redundant Ethernet connections to the node and IP3 and IP4 are redundant InfiniBand connections.

Rev is the image revision for the operating system for the host. The Checkout (CO) column indicates whether the revision has been checked out of the repository. A "1" indicates that the host has been booted with a writable root file system.

If the revision has not been checked out, then the root file system is read-only. Finally the Server column indicates on which server the host instance is running.

The following example displays the server load information.

The following example displays server temperature information.
To show MIC status:

```bash
$ ace mics
```

### Cluster Commands

The following commands can be used to create and modify a cluster.

- **start**
  
  Start hosts in a cluster

- **stop**
  
  Stop hosts in a cluster

- **clone**
  
  Clone a cluster

- **checkout**
  
  Check out a cluster revision

- **release**
  
  Release a cluster revision

- **checkin**
  
  Check in a cluster revision

- **activate**
  
  Activate a cluster revision

- **update**
  
  Update the boot images for a cluster

- **remove**
  
  Remove a cluster revision

- **delete**
  
  Delete a cluster

- **export_image**
  
  Export cluster image to File

- **import_image**
  
  Import File(Cluster Image) to ACE

- **get_logs**
  
  Show logs

### Start Command

```bash
$ ace start <cluster> <num-hosts> [<server> ...] [ -- <host-id> ... ]
```

The start command starts hosts in a cluster. The number of hosts to start is required. If no servers are specified, the ACE daemon will start the hosts on available servers.

When a host is started, it is assigned to a server and the server is powered on. The host remains assigned to that server until it is unassigned with a stop command.

```
# ace start prod 3
starting host prod-0001 on server-0005
starting host prod-0002 on server-0006
starting host prod-0003 on server-0007
server-0005: Chassis Power Control: Up/On
server-0006: Chassis Power Control: Up/On
server-0007: Chassis Power Control: Up/On

# ace start prod 1 server-0008
starting host prod-0004 on server-0008
server-0008: Chassis Power Control: Up/
```
\texttt{# ace start prod 1 server-0009 -- prod-0005}
starting host prod-0005 on server-0009
server-0009: Chassis Power Control: Up/On

\textbf{Stop Command}

\texttt{$ \textbackslash{} ace \ stop \ <\text{cluster}> \ <\text{num-hosts}> \ [\text{<host-id>} \ ...]$}

The stop command stops hosts in a cluster. The number of hosts to stop is required. If no hosts are specified, the ACE daemon will stop the higher number hosts. When a host is stopped, it is powered off and unassigned from the server. Once a host is stopped, the server is available as a candidate for starting other hosts. To power off or shutdown a host and leave it assigned to the server, the server power off and shutdown commands should be used.

\texttt{# ace stop prod 3}
stopping host prod-0003 on server-0007
stopping host prod-0004 on server-0008
stopping host prod-0005 on server-0009
server-0007: Chassis Power Control: Down/Off
server-0008: Chassis Power Control: Down/Off
server-0009: Chassis Power Control: Down/Off

\texttt{# ace stop prod 1 prod-0002}
stopping host prod-0002 on server-0006
server-0006: Chassis Power Control: Down/Off

\textbf{Clone Command}

\texttt{$ \textbackslash{} ace \ clone \ <\text{new\_cluster}> \ <\text{old\_cluster}> \ <\text{revision}> \ [ \text{attr=value} \ ... \ ]$}

The clone command creates a new cluster by making an identical copy of the root file system represented by the specified cluster revision. The new cluster will have new base IP addresses and associated hosts. Once a cluster is cloned, cluster hosts may be started and stopped on the servers.

\textbf{Important!}: Cluster names may have no special characters other than the underscore “\_” character.

\texttt{# ace clone test prod 1}
created cluster test

\texttt{# tail -f /tmp/cluster_clone.out}

\texttt{$ \textbackslash{} ace \ checkout \ <\text{cluster}> \ <\text{revision}> \ [ \text{<server}> \ ]$}

The checkout command makes a writable copy of the cluster’s root file system and starts a revision host on an available server. An administrator can login to the server and modify the root file system. Typical modifications might include installing or removing
software packages or updating configuration files. Once the changes have been made, the cluster revision should be checked back in using the `ace checkin` command.

**Note:** Do not checkout multiple revisions of clusters as it may lead to cluster file system sync issues. Checkouts should be done in a serial incremental fashion. There are a maximum of 9 revisions per cluster. Standard procedure for the system administrator should be to export every 5th revision for backup/cleanup.

**Checkout Command**

```
# ace checkout prod 1
starting host prod-r001 on server-0005
cluster prod revision 1 checked out
server-0005: Chassis Power Control: Up/On

# ace checkout test 1 server-0006
starting host test-r001 on server-0006
cluster test revision 1 checked out
server-0006: Chassis Power Control: Up/On

# tail -f /tmp/cluster_checkout.out
```

**Release Command**

```
$ ace release <cluster> <revision>

This command releases the revision but does not check it in. It is basically an “undo” of a checkout.

# ace release prod 1
stopping host prod-r001 on server-0005
Releasing cluster prod revision 1
```

**Checkin Command**

```
$ ace checkin <cluster> <revision> [description]

The checkin command takes a checked out cluster revision and checks it back in as a new revision. The revision host will be stopped and a new revision will be created containing any changes applied by the revision host.

# ace checkin test 1
stopping host test-r001 on server-0006
cluster test revision 1 checked in as new revision 2
# tail -f /tmp/cluster_checkin.out
```

**Activate Command**

```
$ ace activate <cluster> <revision>

The activate command specifies the revision of the root file system image which should be used by cluster hosts which boot or are restarted.

# ace activate test 2
```
cluster test revision 2 activated

# tail -f /tmp/cluster_activate.out

Update Command

$ ace update <cluster>

Update all booting process files of ACE side and related with kernel for the given cluster.

# ace update test
Cluster update started for cluster test
Progress can be monitored with the command: tail -f /tmp/cluster_update_test.out

# tail -f /tmp/cluster_update_test.out

Remove Command

$ ace remove <cluster> <revision>

The remove command deletes a cluster revision.

# ace remove test 1
cluster test revision 1 removed

Delete Command

$ ace delete <cluster>

The delete command deletes a cluster.

# ace delete test
deleted cluster test

Export Command

$ ace export_image <cluster> <revision> <output file>

This exports a .aci file (ACE cluster image file) for backup.

# ace export_image prod 1 /root/prod_rev1.aci
export cluster prod

# tail -f /tmp/export_image.out

Import Command

$ ace import_image <new cluster> <image file name> [desc]
This imports a .aci file (an ACE cluster image file).

```
# ace import_image test2 /root/prod_rev1.20141018-040827.aci
Import cluster test2

# tail -f /tmp/import_image.out
```

**Get Logs Command**

```
$ ace get_logs
```

This will output the /tmp/ace.log to the screen using a Ctrl+c to escape.

```
# ace get_logs
2014-10-18 04:07:37.192 INF ACE command: put /global/sync_files_seq=32379
2014-10-18 04:07:37.195 INF client ID 130457 has closed connection
2014-10-18 04:07:42.198 INF ACE command: get /global/sync_files_seq
2014-10-18 04:07:42.198 INF client ID 130458 has closed connection
```

**Server Commands**

- `poweron` Power on one or more servers
- `poweroff` Power off one or more servers
- `reset` Power reset one or more servers
- `cycle` Power cycle one or more servers
- `shutdown` Shutdown one or more servers.
- `reboot` Reboot one or more servers
- `identify` Identify one or more servers

The shutdown and reboot commands execute the corresponding Linux command on the node(s). All other server commands are IPMI/iSCB commands used to control the power on servers/sub-racks. They take one or more servers as an argument. The identify command flashes the LED (if it exists).

**MIC Commands**

The following commands are for mics on individual nodes.

- `mic_boot` Boot MIC card(s)
- `mic_reboot` Reboot MIC card(s)
- `mic_reset` Reset MIC card(s)
- `mic_shutdown` Shutdown MIC card(s)
- `mic_status` Get status MIC card(s)
Examples:

$ ace mic_status [--host ] <server> [ mic0 | mic1 | ... ]

# ace mic_status --host prod-0001
mic0: ready
mic1: ready

Wait for the node ACE status to become "active" and check again.

# ace servers | grep prod-0001
server-0005 compute 1 1 1 5 1 1 1 1 prod-0001 up down active

$ ace mic_status --host prod-0001
mic0: online (mode: linux image: /lib/firmware/mic/uos.img)
mic1: online (mode: linux image: /lib/firmware/mic/uos.img)

Plugin Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>plugins</td>
<td>Show the Plugins</td>
</tr>
<tr>
<td>plugin_add</td>
<td>Add a Plugin</td>
</tr>
<tr>
<td>plugin_delete</td>
<td>Delete a Plugin</td>
</tr>
<tr>
<td>run</td>
<td>Run a Plugin</td>
</tr>
</tbody>
</table>

Here is an example of the syntax of each of the new ACE commands using the ACE CLI.

List the plugins.

# ace plugins

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Min-Hosts</th>
<th>Max-Hosts</th>
<th>User</th>
<th>Group</th>
<th>Timeout</th>
</tr>
</thead>
<tbody>
<tr>
<td>basic</td>
<td>check</td>
<td>Simple plug-in example</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>120</td>
</tr>
<tr>
<td>ctcx</td>
<td>test</td>
<td>Simple plug-in example</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>120</td>
</tr>
<tr>
<td>hello</td>
<td>test</td>
<td>Simple plug-in example</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>120</td>
</tr>
<tr>
<td>hpl</td>
<td>test</td>
<td>Simple plug-in example</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>120</td>
</tr>
<tr>
<td>netperf_10g</td>
<td>test</td>
<td>10Gbe Test</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3600</td>
<td></td>
</tr>
<tr>
<td>pingpong</td>
<td>test</td>
<td>Simple plug-in example</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>3600</td>
</tr>
</tbody>
</table>

Delete the pingpong plugin from the database.

# ace plugin_delete pingpong
deleted plugin pingpong

Add a pingpong plugin to the database.

# ace plugin_add pingpong type="'test'" timeout=3600 desc="'Pingpong Test'"

Run the pingpong plugin on 5 of the prod cluster hosts.

# ace run pingpong prod 5

Run the pingpong plugin on 2 prod hosts, prod-001 and prod-0002

# ace run pingpong prod 2 prod-00[01-02]
**Database Commands**

The following commands provide ACE database management:

- **get** - Get database information
- **put** - Put data in to database
- **sync** - Synchronize database with secondary storage

The database commands are low level commands which should not be used directly by the administrator except when requested by Cray technical support personnel.