Integrator Administrator Guide

Contents

Notice

Legal

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi posuere ut est ut ullamcorper. Aenean quis sagittis libero, et iaculis eros. Nunc quis dolor elit. Nulla nisi lorem, placerat ac ante nec, elementum pharetra ex. Integer eget pretium dui. Duis semper elementum metus, nec porttitor augue. Donec malesuada, sapien quis interdum interdum, sapien magna imperdiet odio, in iaculis elit neque nec elit. Curabitur volutpat tincidunt nunc non dictum. Donec egestas eleifend orci. Etiam rutrum condimentum imperdiet.

Phasellus faucibus turpis vel rutrum vulputate. Proin hendrerit pretium lobortis. Etiam ut eros neque. Quisque facilisis mi quis dignissim bibendum. Phasellus ac velit non felis facilisis sollicitudin. Donec massa ante, dignissim dapibus odio eu, pretium mattis orci. Etiam sodales vestibulum mi a dapibus. Curabitur arcu tortor, gravida quis venenatis quis, laoreet efficitur nisl. Maecenas tincidunt erat sed purus suscipit consectetur. Proin mollis sodales finibus. Nullam luctus fermentum enim ut tempor. Aliquam sed diam dictum, molestie augue eu, lobortis felis. Maecenas accumsan diam et nulla maximus gravida.

Vestibulum ornare vitae massa vel sodales. Etiam elementum, lacus et laoreet cursus, magna orci ultrices lectus, nec tempus ipsum nunc ac lectus. Ut nec nulla in sem condimentum vehicula. Morbi rhoncus mauris sed ex rhoncus laoreet. Sed dui dolor, bibendum iaculis ligula ut, ultrices cursus eros. Quisque at quam eros. Vestibulum diam nisl, congue sit amet odio a, consectetur pharetra tortor. Sed tincidunt dictum ipsum ut tincidunt. Praesent lacinia, ipsum et sagittis varius, metus velit ultricies dolor, non eleifend purus nisl quis lorem. Etiam eget erat condimentum enim placerat accumsan at eget quam. Donec pulvinar quam vitae justo maximus volutpat. Donec id tellus purus. Etiam ut dignissim erat. Interdum et malesuada fames ac ante ipsum primis in faucibus.

Fusce venenatis luctus neque, sit amet vulputate turpis. Etiam egestas ex nec orci imperdiet aliquet. Nullam vitae auctor massa, in tristique justo. Pellentesque egestas rhoncus erat, ut lobortis ipsum varius et. Nullam vitae egestas justo. Donec eu lorem consectetur, volutpat est eu, ornare libero. Duis dolor lorem, interdum a consequat eget, sagittis non ante. Duis condimentum viverra sapien, sit amet tincidunt orci laoreet in. Vivamus orci nisl, tristique in suscipit sed, pulvinar id orci. Quisque cursus vulputate tempor. In hac habitasse platea dictumst. Proin felis erat, feugiat in suscipit eget, aliquet sed libero. In et leo gravida, aliquet massa consequat, finibus purus. Nunc blandit lacus at elit eleifend varius. Nam ut augue tristique, commodo tellus vitae, ullamcorper orci. Proin ullamcorper mi vitae ex finibus, eget ultricies erat efficitur.

Pellentesque suscipit nisi ipsum, quis pulvinar dolor accumsan in. Duis in gravida leo. Aenean ac elementum justo. Phasellus eget nulla dolor. Nunc et augue dolor. Quisque mattis volutpat lectus in fringilla. Sed non porttitor odio. Morbi elementum felis orci, a ornare elit accumsan id. Morbi vel mollis metus.

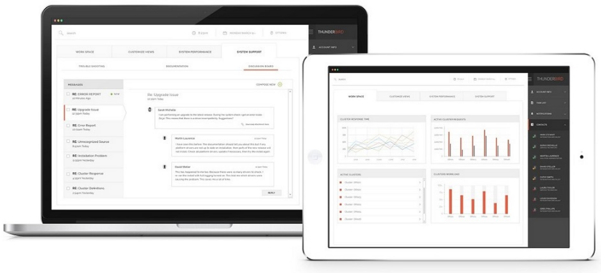
Chapter 1

Thunderbird StormCluster features and benefits

Topics:

StormCluster components work together to give you the advanced cluster management and reporting capabilities that you need within a large computing environment.

The *Thunderbird StormCluster* computing management solution combines powerful cluster management services with a simple-to-use, intuitive *MobileView* front-end application. The combination places control of the full resources of your computing infrastructure into the hands of your team members and it does so in a way the establishes, and sustains, the highest levels of security.



Key *StormCluster* benefits

- Maximum flexibility
- Designed for extensibility
- Ease of management
- Extreme scalability
- Intuitive reports and data visualizations
- Industry-leading support

Key *StormCluster* features

- Easy to use and manage for onsite administrators
- High performing, scalable architecture
- Comprehensive, intelligent scheduling policies
- Complete customization flexibility for integrators
- Heterogeneous platform support
- Continuous live data reporting services
- Robust security services

Component architecture delivers maximum scalability and flexibility

StormCluster calls upon the *ClusterView*, *ClusterAnalyzer*, *ClusterBalance* and *ClusterControl* components become a powerful workload manager for demanding, distributed high-performance computing environments. Not only is a complete set of workload management capabilities available, but the reporting benefits of *ClusterControl*, *ClusterView* and *ClusterAnalyzer* work together to reduce cycle times and maximize productivity in mission critical environments. Equipped with *MobileView* mobile interface, your system administration team and designated system users can collaborate to coordinate their business priorities so that the overall effect is improved even more.

StormCluster Solution architecture overview

Introduction to the solution architecture of the *StormCluster* product line.

StormCluster Solution General Architecture

The main solution components of *StormCluster* have been designed to operate with maximum independence in order to facilitate unlimited scalability. These components operate under the management of *ClusterControl* in order to distribute computing jobs to an unlimited number of cluster nodes each running an unlimited number of components.

The *StormCluster* provides organizations with the ability to deploy and manage a massively distributed computing infrastructure as is needed to control the operations of a wide range of equipment systems and business operations.

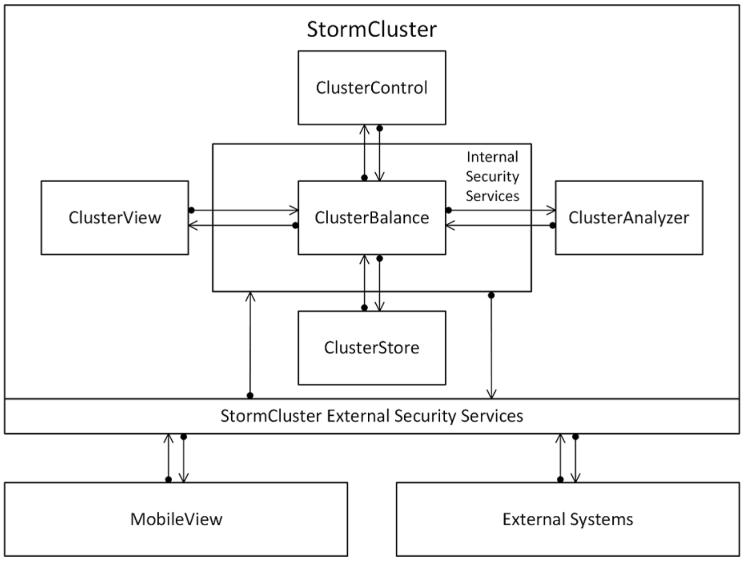


Figure 1: *StormCluster* General Solution Architecture

StormCluster Technology Management Solution

StormCluster from *Thunderbird* provides a comprehensive technology management solution that enables large-scale enterprises to proactively manage their technology infrastructure. In particular, *StormCluster* faciltiates the efficient utilization of sophisticated computing services and does so in such a way that enables almost unlimited scalability. For organizations managing large networks of systems that need to interact on a near real-time basis, *StormCluster* is an ideal solution. *StormCluster* assembles together a variety of specialized components that together provide an integrated solution.

StormCluster ClusterControl Server

StormCluster ClusterControl server handles the communication between the *ClusterBalance* instances and the supporting nodes. It manages the data which the deployed analysis nodes collect from the set of cluster node servers and which are used to coordinate server activities. The *StormCluster ClusterControl* server receives event notifications from nodes and other components, and then sends out instructions according to the rules that have been configured.

StormCluster ClusterView Reporting Server

StormCluster ClusterView reporting server is a web-based reporting tool consisting of *Workbooks*. It collects data from the database and allows the publishing of *Dashboard* views or individual *Worksheets*. These *Worksheets* can be accessed by authorized users through the *MobileView* environment or through the system administration tools.

StormCluster ClusterAnalyzer Analytics Server

StormCluster ClusterAnalyzer is an analytics server that generates a number of reports and visualizations of the data being aggregated by the *StormCluster* deployment. It provides an overall statistical view of the entire cluster. *ClusterAnalyzer* provides dynamic views of the historical behaviour of hosts, resources, and workload across the cluster so that administrators can get an overall picture of cluster's performance.

StormCluster ClusterBalance

A *ClusterBalance* instance is installed on each cluster. This component synchronizes data storage and processing across the different cluster nodes. Each *ClusterBalance* instance collects data from nodes receiving the initial updates and distributes the data resources to other nodes according to configuration rules specified.

StormCluster ClusterStore

A virtual *ClusterStore* is associated with each cluster grid. This components provides a secure data persistence service available to all processes running on all cluster nodes. The *ClusterStore* service collects not only operational data resources but also all administrative and maintenance data resources produced as a byproduct of all *StormCluster* transactions.

StormCluster MobileView

The *MobileView* application provides the User Interface (UI) for the *StormCluster* product environment. The *MobileView* application provides a mobile interface through which users can, in accordance with their security rights, access and interact with various components of the *StormCluster* system.

Analytical reporting with *ClusterView*

ClusterView is an advanced analysis tool for analyzing massive amounts of workload data.

Data-drive decision making

ClusterView enables managers, planners and administrators to easily correlate job and resource data from one or multiple clusters for data-driven decision making. With better insight into datacenter environments, organizations can identify and quickly remove bottlenecks, spot emerging trends, and plan capacity allocations more effectively. *ClusterView* relies heavily upon the analytics provided by the *ClusterAnalyzer*. *ClusterView* then provides the real-time insights that inform and guide *ClusterBalance* services that actively manage the operation of the distributed computing clusters.

Data visualization tools

Unlike traditional business intelligence solutions that require significant time and multiple steps to translate raw data into usable information, *ClusterView* incorporates innovative visualization tools that are built on top of a powerful analytics engine for quick and easy results. Users can utilize the pre-configured dashboards or construct their own, quickly answer questions about their high performance computing infrastructure and the applications running within it so that they can use that information to optimize computing resource utilization.

Workload intelligence

ClusterView is a workload intelligence solution for cluster management. *ClusterView* collects data, then assembles it into reports for analysis. *ClusterView* provides all the tools you need to collect the data, load it into persistent stores, then convert it into reports for further analysis using a ROLAP (Relational Online Analytical Processing) tool.

Security architecture

The *StormCluster* solution provides a scalable and robust security architecture.

Security Services

The security architecture of the *Thunderbird StormCluster* solution relies on two independent sub-systems that continuously coordinate their operation in order to achieve and sustain a very high level of system security. These two independent sub-systems are:

- Internal Security Services
- External Security Services

These two independent sub-systems manage access to the *ClusterBalance* services by internal nodes and access to the entire *StormCluster* solution by external parties, including the *MobileView* client interface.

Security Threats

By default, *ClusterControl* refuses to accept client requests from hosts not listed in the cluster. If the system is started on the unlisted host, the daemons continue to retry the connection. The master host rejects these requests, but if there are many unlisted hosts doing the same thing, it may become overloaded and be unable to respond to valid requests.

Since *ClusterControl* can handle large clusters (several thousand hosts in a cluster) and is designed to be resistant to this type of attack, a malicious attack needs to simulate a larger scale of false hosts in order to be successful, but theoretically *ClusterControl* still remains potentially vulnerable to a very large-scale attack. Administrators must still plan their security architecture and establish continuous monitoring processes.

With effective security administration procedures in place, the security of *StormCluster* is formidable. The scalability of the security regime is derived from the fact that administrative controls are applied in a top-down fashion while rights to resources are defined in a bottom-up fashion. Clusters will have specific rights assigned regarding the resources that they can request and process.

Ease of management

The *StormCluster* solution has been designed for ease of management.

Delegation of administrative rights

Rather than relying on a single cluster administrator for your *Thunderbird StormCluster* deployment, you can now delegate administrative rights within *ClusterControl* to trusted people throughout your organization. With fine-grained control, you can easily establish limits for administrators to slowly gain more responsibility throughout the *StormCluster* system. By enabling project managers and business owners to control their own workloads and resource allocation policies, users enjoy better service and the burden on cluster administrators is substantially reduced.

An administrator can delegate specific sub-sets of rights to sub-administrators, with the ability to manage other users within their domain or to project owners with limited control over data resources within their domain.

Live customization

Administrators can quickly and easily make changes to a wide range of *StormCluster* system parameters and *ClusterView* reporting definitions and they can do so at any time “on the fly” without the need to restart cluster daemons. This means that you no longer have to wait for scheduled maintenance periods to make configuration changes to your cluster resources. This “live” reconfiguration capability boosts productivity, and minimizes downtime while reacting more swiftly to changing business priorities.

Efficient use of resources

Use scheduling capabilities to facilitate efficient use of cluster resources.

Schedule resources as needed

Guaranteed resource access provides flexible scheduling capabilities within *StormCluster ClusterControl*. Scheduling that works the way you need it to, ensuring that resources are allocated to users and jobs in a fashion consistent with your business needs. With extended on-demand scheduling policies, you now have simplified administration and an optimal alignment of business requirements with available infrastructure.

Business needs are established through an analysis of performance demands using *ClusterView* and *ClusterAnalyzer*, projections of expected demands, and forecasts of future workloads. The business needs in turn will determine the resources that will be required, the scheduling rules that will need to be put into place, and the job management framework that is put into place.

Enhanced fairshare and pre-emptive scheduling

The true scheduling features of *StormCluster* provide the ability to fine-tune the defintions that determine user priority and enable different sharing policies by project, team, or department. Job preemption controls help maximize productivity and asset use by preempting only the jobs that should be preempted.

Unparalleled scalability

You can easily add more than 148,000 cores and 500,000 queued jobs, ensuring that your *StormCluster* environment never runs out of head room.

Improved productivity

Features provided to *StormCluster* by *ClusterBalance* include: bulk job submissions, dynamically adjustable swap space estimates, flexible data handling and the smart handling of dependencies in job arrays. By leveraging these capabilities, *ClusterControl* administrators can ensure that users will spend less time waiting for the cluster, and more time focused on their work.

Chapter

2

Activating *StormCluster*

Topics:

After installing, you must activate your product.

- You have a valid license key for *StormCluster*.
- Your local host (the computer that you will use to access the *StormCluster* Console) is running Firefox 3.6 or later and Internet Explorer (IE) 8 or later.

You must perform some initial post-installation and configuration steps for *StormCluster* to function properly.

1. Obtain a valid license code from *Thunderbird*.
2. Log into your local host.
3. Add the license to your *StormCluster* installation.

   a) Use your web browser to launch the *StormCluster* Console.

   The URL for the *StormCluster* Console is `http://host/console` where `host` is either the *StormCluster* host name or IP address.

   The URL for the *ClusterStore* database is `http://host/clusterstore` where `host` is either the *ClusterStore* host name or IP address.

   For example, if your *StormCluster* host name is `hostA.example.com` with IP address 192.168.1.5, you can use either of the following URLs to launch the *StormCluster* Console:

   - `http://hostA.example.com/console`
   - `http://192.168.1.5/console`

   - `http://hostA.example.com/clusterstore`
   - `http://192.168.1.5/clusterstore`

   b) Specify the *StormCluster* administrator name and password.

   The default administrator name is `admin`, and the default password for this account is `admin`.

   To see the **License Info** page, click **Admin** > **License**.

   c) Add the license file to *StormCluster*.

   You can add the license file to *StormCluster* using one of the following methods:

   - Click **Browse** and navigate to your license file.
   - Open your license file with a text editor and copy the text to the clipboard, then paste this text to the **License Text** field in the **License Info** page.

   d) Click **Save** to save the license file to *StormCluster*.
4. Add clusters for *StormCluster* to monitor.

Follow the steps described in the *StormCluster* Administration guide for every cluster that you want *StormCluster* to monitor.

5. Set up the *ClusterStore* database on a remote host.

   By default, the *StormCluster* host runs using a single *ClusterStore* database host. You may choose to set up multiple *ClusterStore* database hosts to distribute activity across the cluster.

   Follow the steps in the *StormCluster* Administration guide to set up the database on a remote host.

6. Optional. Click the **Admin** tab to verify and configure the date and time of your *StormCluster* host.

Hardware and software requirements

List of hardware and software requirements.

Table 1: Hardware and software requirements for *StormCluster*

Supported hardware platforms for all *StormCluster* components	• Linux® on x64 architectures including RHEL 2.1, 3, 4, 5.x, 6, SUSE Linux Enterprise Server including SLES 8, 9, 10, 11 and generic Linux distributions using 2.6 or greater kernels with glibc 2.3 (Debian, CentOS, Ubuntu, Scientific Linux and others). • Linux on ia64 systems including RHEL 4,5, SLES 9,10,11 and generic Linux distributions using 2.6 kernels and glibc 2.3 and later. • Microsoft Windows® on x32 & x64 platforms including Windows 8, Windows 7, Windows Vista, Windows XP, Windows Server 2003 & 2008 standard & enterprise editions, Windows HPC server 2008 • MacOS X 10.4.x, 10.5.x on Apple hardware • HP-UX 11i1, 11i2, and 11i3 on HP hardware • IBM AIX 5.3,6 & 7 on IBM hardware • Sun/Oracle Solaris 7,8,9 on SPARC 32 & 64 bit, Solaris 10 on SPARC 64 bit and x86_64 systems • Cray Unicos/Ic 2.x
StormCluster Master host conifguration	• Minimum 5 GB of physical memory (RAM) recommended • Available SWAP space three times physical memory • 100 GB of free disk space • Minimum of two high-speed network interfaces • Secondary master host recommended in large clusters (more than 2000 hosts)
StormCluster Computation host conifguration	• 2 GB of physical memory (RAM) recommended • 80 GB of free disk space • Minimum one high-speed network interface

Adding or editing clusters

Clusters can be monitored once they are added to the console.

Use the **Console** to add a cluster.

1. Click **Console**.
2. Under the Grid Management section of the Console menu bar, click **Clusters**.

   The **Clusters Console** page displays.
3. Add or edit a cluster.

   - To add a new cluster, click **Add**.
   - To edit an existing cluster, click the name of the cluster that you want to edit.

   The **Cluster Edit** page displays.
4. Specify (or update) the required fields describing your cluster.

   At a minimum, the following fields must be specified to add the cluster: **Cluster Name**, **Master LIM Hostname**, **Master LIM Port**, **Grid Poller**, and **Primary Administrator Username**.

If you want to monitor multiple clusters and if the host is a server, make sure the clusters use different LIM Ports. However, it is recommended to install on the client host.

For the **Grid Poller** field, select the appropriate poller for your version of the cluster.

5. Specify a host for primary job submission.

   Submit a job using `bsub -m` *`host_name`*.

   For example:

   ```
   bsub -m "hostA"
   ```

   The submission cluster forwards the job to the cluster containing hostA.
6. Click **Create** (or **Save**) to save the settings for your cluster.

- If you edited a cluster that was already in the *StormCluster* Console, you do not have to do anything else.
- If you added a cluster to the *StormCluster* Console, you must add the *StormCluster* host to the cluster.

Adding a host to the cluster as a client

Add any cluster so that it can be monitored.

For any cluster that *StormCluster* monitors, you need to add the host to the cluster as a client to give access to cluster data.

Since you can enable the installer to automatically add the host to the cluster, you can normally skip this task.

However, if you did not have the installer add the host to the cluster, and the host is not a server or a client, you need to manually add the host to the cluster as a client to give access to cluster data. You also need to do this to any cluster that you did not use the installer to add.

1. Log into the master host.
2. If the master host cannot resolve the host name to an IP address, edit the `/etc/hosts` file and add the IP address and host name of your host.

   Note:

   If you can successfully ping the host name from the master host, you can skip this step.
3. Edit the `host.cluster.`*cluster_name* file and add the host to the Host section.
4. Reconfigure the daemon and restart `daemon` to apply your changes to the cluster.
5. Test that you added the host successfully to the cluster.
   a) Log into the host.
   b) From the host, use `telnet` to log into the port of your host.

      The default port is 6879.

      For example, for 7.x clusters,

      `telnet<`*`Master IP`*`>6879`

      If you connect to the IP address of the master host, you added the host successfully.

StormCluster can now monitor the cluster.

If you want *StormCluster* to be able to monitor all individual hosts in the cluster, add all hosts in the cluster to *StormCluster*.

Setting host preference

Select the host(s) you want to run your jobs during job submission.

The enhanced `jobsub -m` option supports setting remote host and cluster preference for individual jobs. Host preference can be set for remote hosts in any cluster.

Submit a job using `jobsub -m` *`cluster_name`* or `jobsub -m` *`host_name@cluster_name`*.

Note:

Cluster preference specified at job submission using `jobsub -m` *`host_name@cluster_name`* or `jobsub -m` *`cluster_name`* takes precedence over cluster preference set in SNDJOBS_TO.

For example:

```
jobsub -m "cluster3 cluster2+1"
```

In this example the submission cluster first considers cluster2, with cluster3 as the second choice.

```
jobsub -m "hostA hostB+1"
```

The submission cluster first considers the cluster containing hostB, with the cluster containing hostA as the second choice.

```
jobsub -m "cluster3 cluster4+1"
```

The submission cluster first considers cluster4 , with cluster3 as the second choice.

```
jobsub -m "hostC@cluster1+1 hostD@cluster1+2 cluster1"
```

All hosts in this example are in cluster1. The cluster selection is combined with the host preference; the submission cluster forwards the job to cluster1 with the filtered and merged host preference `"others hostC+1 hostD+2"`.

Diverting to host destinations

To divert to a host destination when your packets need to be backed up on a separate server.

1. Select the **Open the resource destination** icon.

   The **resource destinations** view opens.
2. To add a resource destination, click the **Add** button (+).
3. Enter a name for the resource destination.

   Fastpath:

   Press **Enter** to browse the directory of registered resource destinations.
4. For the Route option, select **With headers**.

   This is the default selection.
5. Select a destination type.

   The displayed header size changes based on the type selected.
6. To indicate how much of the original message to include in the payload, drag the **Payload Slider**.
7. On the payload diagram, click the header type.

The input fields displayed below the payload diagram vary with the header selected.

8. Complete the fields for the header selected.
9. On the **Description tab**, enter a description for the resource destination.
10. Click **OK**.

Feedback from designated host servers will confirm that the diversion configuration setting have been loaded.

Setting cluster metadata

Cluster services profiled with metadata can be more easily discovered and managed.

Enabling the metadata settings in **Realm Permissions** allows an administrator to view and edit metadata settings for the *StormCluster* deployment. Adding metadata services available across the cluster to be efficiently discovered and invoked.

1. Click the **Console** tab.
2. Under the Utilities section of the **Console** menu bar, click **User Management**.
3. On the user Management page, click the administrator who will be granted rights to apply object metadata.
4. Click **Realm Permissions**.
5. Click **View Metadata** and **Edit Metadata** checkboxes to enable these settings.
6. Click **Save**.

   The **Metadata Settings** button becomes available under the Configuration section of the **Console** menu bar.
7. Apply metadata to the node objects.

   From the **Console** menu bar, you can select the option to define custom metadata fields using the *SetMetadataProperty* variable.

   ```
   $AdminTask SetMetadataProperty  {- nodeName node1 - propertyName}
   ```

Setting common resource definitions

Setting consistent host definitions across clusters promotes resource sharing.

For resource sharing to work between clusters, the clusters should have common definitions of host types, host models, and resources. The submission cluster uses the local configuration files to make job forwarding decisions, so it is important these definitions remain consistent across all clusters. Common resources that are usually defined include host types, host models and host services. Once defined, these common resources are inherited by all subordinate clusters.

1. Ensure that the `resources.shared` file is identical in all clusters. The order of resource definitions as well as the definitions themselves must not change.
2. Ensure that the `cluster.resources` file ReservationUsage section is identical for all clusters.
3. Ensure that the `set.cluster.`*`cluster_name`* file ResourceMap section is consistent for all clusters.

   Since the resources defined in the ResourceMap section vary from cluster to cluster, this section should not be identical across clusters. However, resources with the same name should have the same behavior (such as shared or non-shared) in all clusters.

   Warning:

   The local ResourceMap definition is used within each cluster. Inconsistent definitions can result in jobs being forwarded to the wrong execution cluster.
4. Define targets for alarms where desired. Targets identify the threshold point for alarm notifications. Targets can be associated with any resource.

a) From the **Targets** list, select a target.
b) Use the arrow buttons to move elements to the **Selected Targets** list.
c) Save changes by clicking **OK**, or save and deploy the changes by clicking **OK and Deploy**.

Registering a resource

Add a new resource file to encapsulate and copy packets.

1. Select the **Open the resource destination** icon.

   The **resource destinations** view opens.
2. To add a resource destination, click the **Add** button (+).
3. Enter a name for the resource destination.

   Fastpath:

   Press **Enter** to browse the directory of registered resource destinations.
4. For the Route option, select **To file Route**.

   Current directory (".") and parent directory ("..") indicators are not allowed in the relative path definition.
5. In the **File name base** field, enter the file name prefix.

   This will be used to generate the file names.
6. If required, in the **Max file size**, change the maximum file size.
7. If required, in the **Number of files** field, change number of files.
8. If required, in the **Flush interval** field, enter a new flush interval.
9. Enable/disable **Allow overwrite**.
10. On the **Description tab**, enter a description for the resource destination.
11. If required, on the **Targets** tab, select nodes for the policy deployment.

    ⚠ **Caution:**

    Check the details associated with the policies being provided by a selected node. Unexpected behavior can result when there is a mismatch between the selected node policies and the resources being registered.
12. Click **OK**.

Resource file parameters

Use resource files parameters to customize options.

Table 2: Resource file parameters

Parameter	Description
Name	Name of the system destination.
File name base	Used as a prefix.
Max file size	Maximum size of each capture file in bytes.
Number of files	Maximum number of capture files of the maximum file size to keep on disk.

Parameter	Description
Allow overwrite	Indicates whether capture should stop when the maximum number of capture files is reached. If enabled, when the maximum number capture files have been created, any subsequent captures will cause the oldest capture file to be overwritten. If disabled and the maximum number of capture files have been created, no additional information is captured.
Flush interval	Number of seconds to wait until an automatic flush is triggered. The flush_interval ensures that either when the file is full or after a specific period of time, the capture file is closed and renamed so that the contents can be read.

Enabling grid control

Enable grid control on a cluster to grant access to that cluster. This allows you to run commands on that cluster.

Enable grid control to control clusters using the *ClusterControl* **Console**.

1. Click the **Console** tab.
2. Enable grid control for each applicable user in the host.
   a) Under the Utilities section of the **Console** menu bar, click **User Management**.
   b) Click the name of the user for which you want to enable grid control.
   c) In the **Realm Permissions** section, select the **Cluster Management** field, if it is currently unchecked.
3. Under the Grid Management section of the **Console** menu bar, click **Clusters**.
4. Enable grid control on clusters.

   Perform the following for each cluster that you want to control:

   a) Click the name of the cluster that you want to control.

      The **Cluster Edit** page displays.
   b) Click **Control** tab.

      The **User Authentication settings** section appears.
   c) In the **User Authentication settings** section, specify the settings for the Primary administrator account in the master host.

      To ensure that the *ClusterControl* component of *StormCluster* has access to the appropriate commands, you must consider the following:

      - The specified Primary administrator user name is the name of the adminsitrator account in the cluster for which you are enabling grid control. You must specify the username of the Primary Administrator for the machine. You need to set the cluster username before executing cluster via advocate, otherwise, invalid credentials/no username specified error appears.

        This account is used by the Host-, Queue-, and Job-level controls using `eauth` in the master host to invoke the control actions. After saving these settings, this user name is created as a disabled Unix local account in the host.
      - If you are connecting to the master host using `ssh` private key authentication, you need to provide the private key path pointing to the private key file. As shown in the prerequisites, the public key of this file is added to the `authorized_keys` file of the master host root user.
      - The server top directory is the top-level installation directory (`TOP`).
5. Test grid control configuration by specifying a cluster at job submission.
   a) Submit your job using `jobsub -m` *`cluster_name`*.

The job is forwarded to the selected cluster for scheduling.

b) Check the results of the job submission using `hostacct` *`cluster_name`*

6. Click **Save** to commit your changes.

Configuring memory settings

Configure the memory settings to monitor memory usage and control jobs.

1. Click **Console**.
2. Under the Configuration section of the **Console** menu bar, click **Grid Settings**.
3. Click **Memory Exceptions**.
4. Set the values for the Memory RUSAGE Violations fields:
   - **Enable Memory RUSAGE Job Detection**

     Click this checkbox if you want to search for memory violation jobs.
   - **Email Summary Reports**

     Account to whom the email summary report notification should be sent.
   - **Email Schedule**

     Notification frequency for receiving email. Emails are sent weekly on Sundays during database maintenance.
   - **Memory Overusage Filter Name**

     Filter name to be displayed in the job details legend for memory overuse.
   - **Memory Overusage Allocation**

     Memory usage percentage above the RUsage level that is acceptable prior to flagging the job.
   - **Memory Overusage Background Color**

     Color of the legend. This feature is disabled if the color is set to none.
   - **Memory Underusage Filter Name**

     Filter name to be displayed in the legend and in the Job exception filter display.
   - **Memory Underusage Allocation**

     Memory usage percentage below the RUsage level that is acceptable prior to flagging the job.
   - **Memory Underusage Background Color**

     Color of the legend and row display. This feature is disabled if the color is set to none.
   - **Minimum Run Window**

     Set the minimum run time.
   - **Minimum Memory Limit**

     Set the minimum memory limit for your jobs.
   - **Email Subject**

     Define the email subject by using these replacement tags: *<OVERFILTER>*/*<UNDERFILTER>* and *<CLUSTERNAME>*
   - **Email Message**

     Define the email content by using these replacement tags: *<CLUSTERNAME>*, *<OVERFILTER>*, *<OVERSHOOT>*, *<UNDERFILTER>*, *<UNDERSHOOT>*, and *<REPORTTABLE>*
5. Click **Save**.

Setting timeout and wait times

Control job forwarding times.

By default, timeout and wait time are set to 360 and 10 respectively, but can be changed to suit each individual job-forwarding queue.

1. For the job-forwarding queue define MAX_RSCHED_TIME in `host.queues`.

   MAX_RSCHED_TIME=*timeout* [*wait_time*]
2. Run `badmin reconfig` for the changes to take effect.

Jobs submitted to queue pend in an execution cluster for up to the specified number of timeout seconds before returning to the submission cluster for rescheduling. Jobs can be forwarded to the same cluster once again after wait time, in seconds.

```
Begin
QueueQUEUE_NAME=Q1PRIORITY=40NICE=10SNDJOBS_TO=cluster_e2+1
cluster_e3MAX_RSCHED_TIME=50 10END QUEUE
```

and in `host.params`, `MBD_SLEEP_TIME=20`

Thus the cluster reselection timeout and wait time are given by:

- timeout = 50x20 seconds = 1000 seconds
- wait time = 10x20 seconds = 200 seconds

Logical drive strength

Provides a notification on the optimal health of logical drives.

Logical drive strength (strong/weak) notifications share these MIB references:

- *StormCluster*sysName
- *StormCluster*svSeverity
- svRaidMonDataLogicalDriveDeviceName
- svRaidMonDataLogicalDriveRaidLevel
- svRaidMonDataPhysicalDriveRaidState

A strong logical drive notification is sent when the state of the logical drive returns to optimal.

Profile	Description
Trap Name	**svRaidMonLogicalDriveStrongStateNotification**
Varbinds	svClusterConfigName

A weak logical drive notification is sent when the state of the logical drive is anything other than optimal.

Severity	Description
Warning	**Drive is rebuilding**
Major	Drive has degraded
Critical	Drive has either failed or is offline

Profile	Description
Trap Name	**svRaidMonLogicalDriveWeakStateNotification**
Varbinds	svClusterConfigName

Physical drive not faulted

Provides a notification on the health of the physical drive.

Faulted/not faulted physical drive notifications share these MIB references:

- *StormCluster*sysName
- *StormCluster*svSeverity
- svRaidMonDataLogicalDriveDeviceName
- svRaidMonDataLogicalDriveRaidLevel
- svRaidMonDataPhysicalDriveRaidState

The physical drive not faulted notification is sent when a physical device enters either the online or hot spare state.

Profile	Description
MIB reference	**ALPHABET- RAIDMON-MIB** **SNMPv2-MIB** • ***StormCluster*sysName** • ***StormCluster*svSeverity** • **svRaidMonDataLogicalDriveDeviceName** • **svRaidMonDataLogicalDriveRaidLevel** • **svRaidMonDataPhysicalDriveRaidState**
Trap Name	**svRaidMonPhysicalDeviceNotFaultedNotification**
Varbinds	svClusterConfigName

Faulty physical drive

Provides a notification on the physical drive.

Faulty/not faulty physical drive notifications share these MIB references:

- *StormCluster*sysName
- *StormCluster*svSeverity
- svRaidMonDataLogicalDriveDeviceName
- svRaidMonDataLogicalDriveRaidLevel
- svRaidMonDataPhysicalDriveRaidState

The faulted physical drive notification is sent when a physical device enters a state other than online or hot spare.

Severity	**Description**
Warning	**Drive is ready**
Minor	**Drive is rebuilding**
Major	Drive has failed

Profile	**Description**
MIB reference	**ALPHABET-RAIDMON-MIB** **SNMPv2-MIB** • ***StormCluster*sysName** • ***StormCluster*svSeverity** • **svRaidMonDataLogicalDriveDeviceName** • **svRaidMonDataLogicalDriveRaidLevel** • **svRaidMonDataPhysicalDriveRaidState**
Trap Name	**svRaidMonPhysicalDeviceFaultedNotification**
Varbinds	svClusterConfigName

Troubleshooting non-reporting clusters

Immediate actions to take when a cluster is not reporting.

If you installed a cluster but it is not reporting in response to the polling sequence, it might be due to the configuration or operation of the cluster hardware or the activation of the cluster environment.

Some of the ways you can tell the cluster is not reporting properly are the following:

- The cluster issues reporting messages with identifiers that do not correspond to a recognized list within *ClusterControl*
- The cluster fails to issue a reporting message in reponse to a scheduled, or a forced, polling sequence
- The cluster only intermittantly issues reporting messages

Reporting schedule is not set to a valid interval

1. In the *ClusterControl* interface to the execution cluster, open `host.params` and locate HOSTS_PENDING_REASON_UPDATE_INTERVAL.
2. Verify the submission cluster reporting schedule is set to a valid interval.
   a) In Configuratoin Tool **Schedule** task, select the applicable submission cluster.
   b) Click the **Intervals** tab.
      - If the **Interval settings** option is set to **Custom** settings, check that all the reporting settings are correct, and then click **Apply**.
      - If the **Interval settings** option is set to **Inherit from Parent**, continue with the next substep.
   c) In the **Relationships** task, select the **Parent**.
   d) Click the **Cluster reporting** tab.
   e) In the **Default settings** section, make sure the **Reporting intervals** is set to **Synchronize**, and that the reporting **Mode** is not set to **Off**.
   f) Specify the reporting interval, in seconds.

      Zero seconds disables the pending reason.

g) Restart the polling sequence.

The issue has been resolved if the cluster successfully responds to the polling sequence. If this does not happen, proceed to the next step.

Reporting mode is not set to dynamic

In the **Interval settings** menu, confirm that the **reporting mode** is set to dynamic:

1. In the **Interval settings** menu, select the applicable submission cluster.
2. Click the **Intervals** tab.
3. Verify the **reporting mode** is set to the dynamic.

   If the **reporting mode** is set to the scheduled, then the cluster will only report at specific times which may not align with the reporting poll schedule.
4. Restart the polling sequence.

   The issue has been resolved if the cluster successfully responds to the polling sequence. If this does not happen, proceed to the next step.

Troubleshooting the inability to add a cluster

Immediate actions to take when a cluster cannot be added to *ClusterControl*.

If you have attempted to target a new cluster but you cannot get the addition acknowledged within *ClusterControl*, then use the following steps to isolate and resolve the problem. This situation can arise for clusters deployed in either execution or submission roles.

Hardware infrastructure problem

If you fail to receive any message during the cluster reporting poll, then try directly polling the unresponsive cluster directly.

1. In Configuration Tool **Connections** task, select the applicable cluster.
2. At the bottom of the **Connections** task, click **Poll**.

   - If there is no reply, the cluster may be offline (the hardware infrastructure could be malfunctioning), or there is a problem with your network connectivity. Contact the approriate system administrators in order to isolate any problems at the physical connectivity layer.
   - If you can poll the cluster, and get a response, then continue with the next remedy.

Restart the cluster

1. Open the cluster's configuration web page by typing its IP address into the **configuration lookup panel**.

   This is also where you can determine if you have the correct credentials for the cluster.
2. Restart the cluster.

   a) In Configuration Tool **Control** task, select the applicable cluster.

   b) At the bottom of the Configuration Tool **Control** task panel, click **Reboot**.

Add the cluster again

Try adding the cluster to *ClusterControl* again.

Verify hardware support

Make sure the hardware configuration deployed for the cluster is supported by *ClusterControl*, and that it is running the certified firmware.

For a list of supported hardware configurations, and for known compatibility issues, see the register of supported hardware.

Wrong credentials

Make sure you are using the correct credentials when trying to add the unit. For some hardware manufacturers, you may have to set the default credentials for the respective hardware before resetting the credentials after the cluster has been added.

1. Consult the hardware manufacturer's reference documentation in order to locate the default credentials for the applicable hardware.
2. Reset the hardware to the default credentials using the applicable hardware configuration interface.
3. In *ClusterControl*, access the Configuration Tool **Set up** task.
4. Click the **Add Clusters** tab.
5. To add the new cluster unit, click **Add a cluster**, select the extension type, and click **Add**.
6. When prompted by the **Enter credentials** dialogue, input the default hardware credentials.
7. Select the new cluster.
8. In the **Default logon** section, enter the new username and password for the cluster.

Cluster is connected to the wrong database

Make sure the Cluster is connected to the correct *ClusterStore* database, as follows:

1. In Configuration Tool **Cluster** task, select the the applicable cluster.
2. Click the **Resources** tab.
   - If the database status is **Connected**, go to the next step.
   - If the database status is **Disconnected** or **Unavailable**, click **Connect to database**.

   ⚠ **Caution:**

   Do not overwrite the disconnected or unavailable database connection, as this can lead to data loss. Be use to save copies of any database connection configurations before deleting or modifying them.

   Note:

   When you create a new database connection, searches or administrative routines that were in place for previous connections will need to be respecified for the new database connection.

Execution clusters not connected to the correct database.

Make sure the associated execution clusters are also connected to the correct *ClusterStore* database.

If there is an issue with associated execution clusters, this may be a sign that there are more serious problems with the configuration of the *ClusterControl* environment or with the network environment.

1. In Configuration Tool **System** task, select the **Job Router**.
2. Click the **Resources** tab.
   - If the Job Router database status is **Connected**, go to the next step.
   - If the Job Router database status is **Disconnected** or **Unavailable**, click **Connect to database** .

Firewall problems

Try adding the unit with the network firewall turned off. This will determine if the problems are related to interference in network connectivity due to secutity settings.

For information about how to disable the network firewall, request assistance from your network administrator.

Important:

Do not turn off the firewall permanently. Reactivate it after your tests are complete.

Improper configuration

This activity should be done in collaboration with your network administrators.

Make sure your networks are configured properly, as follows:

1. In the Configuration Tool **Network view** task, select a network.
2. Click the **Properties** tab, and make sure all the settings are correct (IP prefix, subnet mask, routes, and so on).
3. Initiate a network diagnosis by selecting **Test Network**.
4. Repeat these steps for all the networks on your system.

Incorrect request address

Make sure the Cluster, Job Router, and all redirectors are using the correct addresses in their requests.

1. In the Configuration Tool **System** task, click the **Directory** view.
2. Select the Physical view, and click the **Resources** tab.
3. From the **Resources** drop-down list, select the appropriate resources in sequence and verify the associated network addresses.
4. In the Logical view, select the Job Router role, and click the **Resources** tab.
5. Under the **Servers** section, click **Advanced**.
6. Select the appropriate **Network address** for each server, and click **Validate**.
7. Click the **Properties** tab.
8. Select a **Redirector**, and click **Edit the item**.
9. From the **cluster interface** drop-down list, select the appropriate cluster.
10. Repeat the last two steps for each redirector.
11. Try adding the cluster to *ClusterControl*.

Job router process offline

Make sure the Job Router process is online, as follows:

1. In the Config Tool **System** task, select the Job Router process.
2. At the bottom of the **System** task, click **Diagnose**.
3. If there are issues, try to fix them.

Databse offline

Make sure the *ClusterStore* database is online, as follows:

1. In the Config Tool **System** task, select the Database.

   Confirm that the correct Database has been selected.
2. At the bottom of the **System** task, click **Diagnose**.

   The diagnosis process will identify a series of issues. The most common issues relate to incorrect identifiers having been included in the Database.
3. For each of the issues identified, take corrective action as directed by the diagnosis instructions.

Execution clusters offline

1. Make sure the Execution Clusters are online, as follows:
   a) In the Config Tool **System** task, select each of the applicable Clusters.
   b) At the bottom of the **System** task, click **Diagnose**.

c) For each of the issues identified, take corrective action as directed by the diagnosis instructions.

2. Restart the cluster.
3. Try adding the cluster to *ClusterControl*.
4. Test the Cluster connection.
5. If you still cannot add the cluster, contact *Thunderbird* technical support.

Chapter 3

Cluster management controls

Topics:

Using *ClusterControl* for cluster management to maximize job throughput.

Thresholds

The **Thresholds** page under the Management section of the *ClusterControl* **Console** menu bar. This page shows the configured thresholds in your cluster. A threshold triggers an alert if your clusters, hosts, queues, or jobs meet the conditions of the threshold.

- Name. The name of the cluster or host and the threshold.
- Type. The type of threshold (for example, High/Low, Baseline, and Time Based)
- High. The high threshold boundary value. If the current value of the monitored data source item is greater than this value for a specified duration, the threshold triggers an alert.
- Low. The low threshold boundary value. If the current value of the monitored data source item is lower than this value for a specified duration, the threshold triggers an alert.
- Trigger. The amount of time that the data source item must be in breach of the threshold before the threshold triggers an alert.
- Duration. If the data source item is still in breach of the threshold, this is the amount of time from when the alert was first triggered.
- Repeat. The amount of time that the threshold waits before repeating the alert if the data source item is still in breach of the threshold.
- Current. The current value of the monitored data field.
- Triggered. Indicates whether this threshold has trigged an alert.
- Enabled. Indicates whether this threshold is currently active.
- Ack. Indicates whether the threshold alerts have been acknowledged: "on" indicates that the threshold has been acknowledged; "off" indicates that the threshold either has not been acknowledged, or had its acknowledgement reset.

Threshold items

The **Threshold Item** page for a threshold allows you to configure threshold settings and event triggering.

Event triggering behavior is based on re-alert cycle settings. When the threshold first triggers an alert, the event trigger is invoked based on a high or low threshold breach. If the alert stays triggered, the event trigger is invoked again unless the re-alert cycle is set to **Never**. When the alert reverts to normal, the threshold triggers the norm threshold command or script.

You can configure the following items from this page:

- Template propagation enabled: Enable the propagation of changes to the threshold template.
- Threshold name: The name of the threshold as it appears in the Name column in the list of thresholds.

  Note:

  You can use placeholders to customize your threshold name. Placeholders for the threshold name are enclosed by pipe characters (`|`), for example, `|cluster_name|`.
- Threshold enabled.
- Weekend exemption: Disable threshold alerts on weekends.
- Disable restoration email: Disable threshold alerts when the threshold has returned to normal.
- Reset acknowledgement: Reset acknowledgements when the threshold has returned to normal.
- High/low threshold values.
- Threshold type: High/low, baseline, or time based.
- Event triggering (Shell command): Specifies event trigger commands or shell scripts in the event of a breach.
  - High Threshold Trigger Command/Script: If the threshold is breached because the data source exceeds this value, the threshold triggers the specified command or shell script.
  - Low Threshold Trigger Command/Script: If the threshold is breached because the data source drops below this value, the threshold triggers the specified command or shell script.
  - Norm Threshold Trigger Command/Script: If the threshold is breached, then returns to normal, the threshold triggers the specified command or shell script.
- Event triggering (Grid administrator host level triggers): Specifies host-level actions in the event of a breach.
  - Host Level Action (High Threshold): If the threshold is breached because the data source exceeds this value, the threshold triggers the specified action on the host.
  - Host Level Action (Low Threshold): If the threshold is breached because the data source drops below this value, the threshold triggers the specified action on the host.
- Email message body: Email alert message content. This specifies the template that is used in alert email notifications for this threshold.

  Note:

  You can use placeholders to customize your alert emails and provide additional information. Placeholders for the email message body are enclosed by angle brackets (<>), for example, `<cluster_name>`.
- Syslog settings.
- Data type: Special formatting for the given data.
- Re-alert cycle: The amount of time the threshold repeats the alert, if it is still in breach.
- Notify accounts and extra alert emails: Email addresses to be notified when the threshold raises an alert.

Quick Reference: basic cluster management commands

Basic commands to manage the *StormCluster* hosts status.

Table 3: Frequently used commands for cluster management

Action	**Command**
Display resource and activity information about the cluster	`showcluster`
Status of all clusters	`badmin showstatus`
Cluster restart	`badmin restart -p`
Host load information	`hostload`
Accounting information	`hostacct`
List and status of all hosts in cluster	`listhosts`
Cluster limit data across all clusters	`hostlimits`
Status of submission and execution clusters	`statclusters`

Cluster management controls

Using *ClusterControl* for cluster management to maximize job throughput.

Thresholds

The **Thresholds** page under the Management section of the *ClusterControl* **Console** menu bar. This page shows the configured thresholds in your cluster. A threshold triggers an alert if your clusters, hosts, queues, or jobs meet the conditions of the threshold.

- Name. The name of the cluster or host and the threshold.
- Type. The type of threshold (for example, High/Low, Baseline, and Time Based)
- High. The high threshold boundary value. If the current value of the monitored data source item is greater than this value for a specified duration, the threshold triggers an alert.
- Low. The low threshold boundary value. If the current value of the monitored data source item is lower than this value for a specified duration, the threshold triggers an alert.
- Trigger. The amount of time that the data source item must be in breach of the threshold before the threshold triggers an alert.
- Duration. If the data source item is still in breach of the threshold, this is the amount of time from when the alert was first triggered.
- Repeat. The amount of time that the threshold waits before repeating the alert if the data source item is still in breach of the threshold.
- Current. The current value of the monitored data field.
- Triggered. Indicates whether this threshold has trigged an alert.
- Enabled. Indicates whether this threshold is currently active.
- Ack. Indicates whether the threshold alerts have been acknowledged: "on" indicates that the threshold has been acknowledged; "off" indicates that the threshold either has not been acknowledged, or had its acknowledgement reset.

Threshold items

The **Threshold Item** page for a threshold allows you to configure threshold settings and event triggering.

Event triggering behavior is based on re-alert cycle settings. When the threshold first triggers an alert, the event trigger is invoked based on a high or low threshold breach. If the alert stays triggered, the event trigger is invoked again unless the re-alert cycle is set to **Never**. When the alert reverts to normal, the threshold triggers the norm threshold command or script.

You can configure the following items from this page:

- Template propagation enabled: Enable the propagation of changes to the threshold template.
- Threshold name: The name of the threshold as it appears in the Name column in the list of thresholds.

  Note:

  You can use placeholders to customize your threshold name. Placeholders for the threshold name are enclosed by pipe characters (`|`), for example, `|cluster_name|`.
- Threshold enabled.
- Weekend exemption: Disable threshold alerts on weekends.
- Disable restoration email: Disable threshold alerts when the threshold has returned to normal.
- Reset acknowledgement: Reset acknowledgements when the threshold has returned to normal.
- High/low threshold values.
- Threshold type: High/low, baseline, or time based.
- Event triggering (Shell command): Specifies event trigger commands or shell scripts in the event of a breach.
  - High Threshold Trigger Command/Script: If the threshold is breached because the data source exceeds this value, the threshold triggers the specified command or shell script.
  - Low Threshold Trigger Command/Script: If the threshold is breached because the data source drops below this value, the threshold triggers the specified command or shell script.
  - Norm Threshold Trigger Command/Script: If the threshold is breached, then returns to normal, the threshold triggers the specified command or shell script.
- Event triggering (Grid administrator host level triggers): Specifies host-level actions in the event of a breach.
  - Host Level Action (High Threshold): If the threshold is breached because the data source exceeds this value, the threshold triggers the specified action on the host.
  - Host Level Action (Low Threshold): If the threshold is breached because the data source drops below this value, the threshold triggers the specified action on the host.
- Email message body: Email alert message content. This specifies the template that is used in alert email notifications for this threshold.

  Note:

  You can use placeholders to customize your alert emails and provide additional information. Placeholders for the email message body are enclosed by angle brackets (<>), for example, `<cluster_name>`.
- Syslog settings.
- Data type: Special formatting for the given data.
- Re-alert cycle: The amount of time the threshold repeats the alert, if it is still in breach.
- Notify accounts and extra alert emails: Email addresses to be notified when the threshold raises an alert.

Configuring idle job detection

Identify all idle jobs in your cluster.

Idle jobs are configured per cluster in the cluster edit page. Specific queues may be excluded in idle job calculation such as jobs submitted to an interactive queue.

Note:

After installing *StormCluster*, older settings for idle jobs are no longer in effect. The configuration settings must be updated.

1. Click **Console**.
2. Under the Grid Management section of the *ClusterControl* console menu bar, click **Clusters**.
3. Click the cluster name of the cluster in order to set the idle job detection.
4. Select **Enable Idle Job Detection** checkbox to search for idle jobs in the cluster.
5. Set values for the following fields:

   - **Email Notification Type**
   - **Minimum Runtime**
   - **Floating Window**
   - **CPU Time Threshold**
   - **Include Job Types**
   - **Job Commands**
   - **Idle Jobs Exclude Queues**
6. In the execution cluster, open `host.params` and locate HOSTS_PENDING_REASON_UPDATE_INTERVAL.

   Pending reasons are updated every 60 seconds by default. You can modify the update interval to get faster feedback when jobs pend.
7. Specify the update period, in seconds.

   Zero seconds disables the pending reason.
8. Click **Save**.

Acknowledging threshold alerts

Set configurations to acknowledge all threshold alerts.

1. Click the **Threshold** tab.

   If there are several thresholds, use the **Threshold Status** menu bar to filter the threshold view.
2. Under the Management section of the **Console** menu bar, click **Thresholds**.
3. Click the checkbox at the right side of each threshold with triggered alerts that you want to acknowledge.
4. In the **Choose an action** field, select **Acknowledge** and click **Go**.

   If a long period of time elapses after initiating an action, it is possible that there is a circular reference between the associated actions. Check the definitions of all interdependent actions to confirm that there are no circular references if you suspect that there is a problem.
5. In the Acknowledge Message window, specify an acknowledgement reason message (or leave blank for no message) and click **Yes** to acknowledge the triggered alerts for all thresholds.

   This message is recorded in the `cluster.log`, threshold log database table, and syslog files.

   For each selected threshold, the **Acknowledge** icon changes into the **Reset Acknowledge** icon.

You can repeat the above steps, but select **Reset Acknowledgement** in the **Choose an action** field to allow the thresholds to resend future email and system log notifications with each triggered alert for the threshold.

Click **Reset Acknowledge** next to the threshold to allow the threshold to resend future email and syslog notifications with each triggered alert for the threshold.

Cluster reselection rules

Set rules on when jobs are forwarded to clusters that have available resources.

Jobs are forwarded to the execution cluster based on swp, mem, type, ncpu, model, boolean resources, cpu factors, and user-defined shared resource requirements. In some cases, however, jobs may reach the execution cluster and find other required resources are not available. The cluster reselection policy sets the length of time jobs pend, and allows returned jobs to be forwarded to other execution clusters that may have different resources available.

Intelligence is built into execution clusters so that in the event that required resources are no longer available on a given execution cluster, the job will be automatically forwarded to an execution cluster that does have the required resources.

The time forwarded jobs spend in being forwarded between execution clusters before returning to the submission cluster for rescheduling can be configured for each queue.

The wait time before the same job is returned to the same execution cluster can also be set. Both values are defined in MAX_RSCHED_TIME in `host.queues` and are multiplied by MBD_SLEEP_TIME (`host.params`):

```
MAX_RSCHED_TIME = timeout [wait_time]
```

timeout

timeout*MBD_SLEEP_TIME determines how long a job stays pending in the execution cluster before returning to the submission cluster.

The timeout value can be customized based on average job runtime and XL_RES_RATIO for the cluster. For example:

timeout = MAX(job_runtime, (XL_RES_RATIO - 1) * job_runtime) * 2 / MBD_SLEEP_TIME

wait time

wait_time*MBD_SLEEP_TIME determines how long the submission cluster waits for other execution clusters to become available before returning to the same execution cluster. The wait time only applies when there are execution clusters the job has not yet tried.

Interface bypass

Interface bypass notification is sent if the bypass group table operational status sets to bypass mode (value=0).

Minor Condition

A `bypass` notification is sent if the sum of `ifInDiscards` and `ifOutDiscards` exceeds 10 within an interval of 3600 seconds.

Profile	Description
Frequency	3600 seconds
Severity	Minor
Condition	DELTA (IF-MIB::ifInDiscards + IF-MIB::ifOutDiscards) > 10

Major Condition

Profile	Description
Frequency	0 seconds (Immediate)

Severity	Major
Condition	ALPHABET-MIB::svBypassGroupGroupTableOperStatus == 0 (eBypassWatchdogOperStatus_BYPASS_OPER_STATUS_BYPASS)

Interface bypass group is in active mode

This alarm is cleared when the bypass group table operational status sets to active (value=1).

Profile	Description
Frequency	0 seconds (Immediate)
Severity	Cleared
Condition	ALPHABET-MIB::svBypassGroupGroupTableOperStatus == 1 (eBypassWatchdogOperStatus_BYPASS_OPER_STATUS_ACTIVE)

Cluster management basic commands

Lists the basic commands used for managing clusters. The basic commands are: `showcluster`, `badmin showstatus`, `badmin restart -p`, `hostload`, `hostacct`, `listhosts`, `hostlimits`, and `statclusters`.

showcluster command

Displays execution cluster resource provider and consumer information, resource flow information, and connection status between the submission cluster and execution cluster.

```
% showcluster report
```

Use `-app` to view available application profiles in remote clusters.

Information related to *StormCluster* is displayed under the heading Job Forwarding Information.

local_queue

Name of a *StormCluster* queue.

job_flow

Indicates direction of job flow.

- send: The local queue is a submission cluster send-jobs queue (SNDJOBS_TO is defined in the local queue).
- recv: The local queue is an execution cluster receive-jobs queue (RCVJOBS_FROM is defined in the local queue).

remote

Shows the name of the remote queue, always the same as the local queue name.

For receive-jobs queues, always “-”.

cluster

For send-jobs queues, shows the name of the execution cluster containing the receive-jobs queue.

For receive-jobs queues, shows the name of the submission cluster that can send jobs to the local queue.

status

Indicates the connection status between the local queue and remote queue.

- ok: The submission cluster and execution cluster can exchange information and the system is properly configured.
- disc: Communication between the two clusters has not been established. This could occur because there are no jobs waiting to be dispatched, or because the remote master cannot be located.
- reject: The remote queue rejects jobs from the send-jobs queue. The local queue and remote queue are connected and the clusters communicate, but the queue-level configuration is not correct.

badmin showstatus command

Displays current cluster information. The last two lines only appear during a parallel `daemon` restart initiated with the `badmin restart -p` command.

```
% badmin showstatus
```

badmin restart -p command

Command to start daemon in parallel.

```
% badmin restart -p
```

Starts `batchd` in parallel, leaving the existing `batchd` free to respond to queries and run commands while the new `batchd` restarts, reading configuration files and replaying events. Once all events have been read `batchd` daemons merge, replaying new events and leaving only the new `batchd` running.

During a parallel `batchd` restart, new `badmin restart` and `badmin reconfig` commands are not accepted. Parallel `batchd` restart using `badmin mbdrestart -p` does not work with duplicate event logging (LSB_LOCALDIR in `host.conf`).

The existing command `badmin restart` remains unchanged.

hostload command

This command runs locally (in the execution cluster) when submitted by a running job.

```
% hostload
```

-cname

The option `-cname` includes the cluster name for execution cluster hosts and host groups in output for `hostload`.

```
% hostload -cname
```

hostacct command

Displays accounting statistics for finished jobs within all clusters connected by *StormCluster*.

```
% hostacct
```

This command runs locally (in the execution cluster) when submitted by a running job.

Accounting statistics include:

- Date and time of submission.
- Date and time of execution.
- Time the job took to run.
- Host the job ran on.

- Any suspensions or resumptions of the job.
- The submission user account name.

listhost command

This command runs locally (in the execution cluster) when submitted by a running job.

```
% listhosts
```

-cname

The option `-cname` includes the cluster name for execution cluster hosts and host groups in output for `listhosts`.

hostlimits command

Displays cluster limit data across all clusters, including forward limits and aggregated execution cluster limits.

All clusters are shown by default.

To specify a remote limit with the `-n` option, include the cluster name as well as the limit name: `hostlimits -n` *`limit_name`*`@`*`cluster_name`*.

-fwd

Displays forward slot allocation limits.

Use `-fwd` with `-c` to display forward slot limit configuration.

-fwd -C *cluster_name...*

Displays forward slot allocation limits for one or more specific clusters. `-C` cannot be used without `-fwd`.

Use `-fwd -C` with `-c` to display forward slot limit configuration for the specified cluster.

hostpart command

Displays information about host partitions. By default, hostpart displays information about all host partitions. Host partitions are used to configure host-partition fairshare scheduling.

ClusterView provides commands for running tasks on remote hosts and ports (LIM_PORT and RES_PORT) for communication. Therefore, even if your cluster restricts users from directly logging into or running commands on remote hosts (therefore restricting your users to using commands to access remote hosts), users can still run some commands to run tasks on remote systems under certain circumstances.

The `hostpart -x` command where `x` defines the host range can be directed to any cluster host. It will be propagated in order to return partition reports from the designated host range.

statclusters command

This command runs locally (in the execution cluster) when submitted by a running job.

```
% statclusters
```

Sample command response:

```
CLUSTER_NAME STATUS MASTER_HOST ADMIN HOSTS SERVERS

sub_cluster ok hostA admin 1 1

cluster2 ok hostB admin 1 1

cluster3 ok hostC admin 2 2
```

Cluster capacity reports

Reports the usage of all slots in the cluster.

With the *cluster capacity* capability provided by *ClusterView* and *ClusterAnalyzer*, implementation teams can:

- Analyze activity by project and user
- Understand and improve service levels
- Monitor and improve scheduling policies
- Alleviate bottlenecks and boost productivity
- Manage the performance of the *StormCluster* environment
- Analyze capability in order to:
    - Analyze capacity
    - Tune cluster service level definitions
    - Refine job scheduling rules

Worksheets in the *ClusterAnalyzer Dashboard*:

- **Capacity Summary**: Shows a map of the clusters, which you can customize to show locations within a data center. You can view and manage efficiently many years of data.
- **Cluster Usage**: View the number of slots in each state that varies over the time period.
- **Host Usage**: Drill down to the host level and see what servers are used by individuals working on a particular project.
- **Cluster Workload**: Slot status for a particular cluster based on the selected dimension.
- **Data**: Visualize average slots, average CPU usage, and average memory usage data by changing the values on the right.

Adding classifiers

Filter data sources by class.

A classifier creates conditions for filtering and classifying data sources. If no conditions are specified, all of the data sources are considered for classification. You can filter data sources by an expression (email, Peer to Peer), protocol or subscriber attribute, for example. These filters are also called "classifier conditions".

1. Click the **Classifiers** icon.

   All **Classifiers** that have been defined will be accessible for selection in the **Classifiers** view.

   The **Classifiers** view opens.
2. To add a classifier, click **Classify**.
3. Enter a name and description for the classifier.
4. Select the classifier type:
   a) Enumerated: All possible values of the classifier must be specified in the definition
   b) Integer: Classifier can be assigned any integer value
   c) String: Classifier can be assigned any string value.
5. Click **Next**.
6. Click **Create a new rule**.
7. Set up one or more rules for the classifer, then click **Next**.

   Rules are conditions that, if met, set values for the classifers.
8. (Optional) Set target(s) for the classifier (which nodes this classifer is available for).

   If no targets are set, then the classifier is available for all elements on all nodes.

9. Define additional **Classifiers** by selecting **Customize Classifiers** option.

   ```
   $AdminTask SetMetadataProperty  {- nodeName node1 - propertyName}
   ```

   ```
   $AdminTask SetMetadataProperty  {- classifiername valueString -f%}
   ```

10. Remove old alarms and filter match items.

    It is a best practice to remove old alarms or filter match items when they are no longer needed. Failure to regularly remove old alarms or filter match items may result in inconsistent reporting results.

    a) Click **Filter match items**.
    b) Select **Filter match items** that no longer apply and that you wish to remove.
    c) Click the **Delete** button.

       The old alarms and filter match items will be removed.

11. Click **Finish.**

Classifiers have been added to data sources. They can be used to filter and analyze data received from these sources.

Note:

You can follow the same procedures to change existing Classifiers or to add new ones.

Creating dynamic reporting tables

Dynamic reporting tables display real-time throughput information.

1. Click the **Dynamic tables** icon.

   The **Dynamic tables** view opens.

2. To create a dynamic table, click **Table**.

   For advanced table management, use the whole toolbar.

3. Enter a name and description for the table.
4. (Optional) Set a timeout value.

   a) In the "Time out" field, enter the number of units.
   b) Select a unit of time.
   c) Set additional control parameters using `hostlimits` command.

5. (Optional) Set a unique-by option. Then, click **Add**.
6. (Optional) Set target(s) for the table (which nodes this table is available for).

   If no targets are set, then the classifier is available for all elements on all nodes.

7. Click **OK**.

You can now publish the table.

Adding columns to a dynamic reporting table

Enhance your reports by adding new columns to reporting tables.

Expand dynamic reporting tables to show the columns you need. Once a column is added, reports may be run against all of the fields represented.

1. Go to the Columns tab of the **Dynamic tables view**.
2. Select **Add a column**.

   The Columns tab expands.

3. In the **Name** field, select a column.
4. Select a data type.
5. Select a default value:

- **Null:** no value
- **Specified:** enter a default value.
- **Expression:** to allow the expression to be reused by other columns, select **Reusable**. For reusable expressions, you can select an existing expression or click the ellipsis (...) and create a new expression. Otherwise, select **One time** (expression can be used only by this column).

6. Select the radial button to the left of the **Reusable** or **One time** option to indicate whether the expression is exclusively for this column or can be reused in other columns.

   While working with expressions, the following toolbar options are available:

   - Click the **Undo** button to undo the last action.
   - Click the **Redo** button to redo the last undone action.
   - Click the **Search for text** button to locate text within the **Description** box.
   - Click the **Change panel fonts and/or colors** button to change the panel fonts and/or colors.
7. Click **OK**.

Creating a limiter

A new limiter acts as a buffer between aggregate views.

A limiter controls the value of a threshold, providing control over when actions are executed. They can be used to apply actions after a threshold (like number of connections) is exceeded, or can be used to define a control system to manage the amount of bandwidth on the network.

Limiters are used in:

- limit threshold condition
- published expressions template

To specify a remote limit with the `-n` option, include the cluster name as well as the limit name: `hostlimits -n` *`limit_name@cluster_name`*.

-fwd

Displays forward slot allocation limits.

Use `-fwd` with `-c` to display forward slot limit configuration.

-fwd -C *cluster_name...*

Displays forward slot allocation limits for one or more specific clusters. `-C` cannot be used without `-fwd`.

Use `-fwd -C` with `-c` to display forward slot limit configuration for the specified cluster.

1. Click the **Limiter view** toolbar button. The **Limiters** view opens.
2. To create a limiter, click the **Add** button (+).
3. Enter a name and description for the limiter.
4. In the **Threshold** fields, enter a threshold value and unit of measurement.
5. If the threshold unit of measurement is Connections, select a **Unique by** option.
6. For all other threshold units of measurement, select a flow direction.
7. You can set up conditions on the Conditions tab and define targets on the Targets tab.

Defining subscriber attributes

Define new subscriber attributes to accurately maintain your database.

1. Click the **Subscriber attribute definitions** icon.

   The **Subscriber attributes** view opens.
2. To add a subscriber attribute definition, click the **Add** button (+).

3. Enter a name and description for the subscriber attribute.
4. In the **Execution scope** area, select a scope.
5. (Optional) In the **Policy** area, check an option:
   a) Suppress attribute from policy files when unused in policy.
   b) Notify the system when this attribute is set.
   c) Make this attribute reportable by Network Demographics.
6. Select an attribute type:
   a) **Accept any value.** Select from the list.
   b) **Specified values.** Enter a value and click **Add**.
   c) Add or remove attributes on the list record (contained attribute list).

   A Record attribute contains a group of attributes that can be a mix of any of the simple-types (strings, integers, Booleans and so on). For example, MyRecordAttribute, type 'record' is a list that declares other attributes:

```
MySubString1  type is string
          mySubString2  type is string
          someIntAttr   type is integer
          andABoolAttr  type is boolean
```

7. Click **OK**.

Analysis dimensions

Table of dimensions you can run reports from.

Table 4: Dimensions or filters for running reports

Measure	**Description**
Date	Jobs finish date (shows local time zone).
ISO Week Number	Jobs finish week.
Clusters	Hosts running jobs that work together as a single unit.
Project	Jobs assigned for specified project. This is the -P *project_name* option in jobsub.
Queue	Jobs submitted to one of the specified queues. This is the -Q *queue_name* option in jobsub.
Users	One who submits jobs.
Host Type	The combination of operating system version and host CPU architecture.
Host Model	The combination of host type and CPU speed (CPU factor) of the computer.
Host Group	A group of hosts running jobs.
Application	Jobs submitted to the specified application profile. This is the -app *application_profile_name* option in jobsub.
SLA Tag	Service class where the job runs. This is the -sla *servi ce_class_name* option in jobsub.

Measure	Description
Job Exit Status	Traditionally jobs finishing normally report a status of `Done`, which usually means the job has finished normally. `Exit` status means that the job has exited abnormally.
Memory Ranking	Memory reservation of jobs.
Run Time Ranking	Ranking of run time jobs.
Pending Time Ranking	Ranking of pending jobs.
Number of Processors	Number of slots used by a job.

Analysis pane columns

The results of your investigation or maintenance query are listed in the *ClusterView* **Analysis Pane**. You can choose what information to view by showing or hiding columns. This section lists the columns available. This list is organized alphabetically by column.

Column	Associated reports	Description
Access rules	• Access rule configuration	Name of the access rule.
Acknowledged by	• Alarm report	User who acknowledged the alarm. When the alarm is acknowledge automatically by the system, **Service** is indicated.
Acknowledged on	• Alarm report	Time the alarm was acknowledged.
Action	• Inventory report	The change in state: added, removed, or no change.
Alarm	• Alarm monitoring • Alarm report	Alarm entity name.
Calculation status	• Health statistics	If health statistics area unavailable, the reason is shown here.
Created by	• Incidents	User who originally reported the incident.
Creation time	• Incidents	Time the incident was reported.
Custom fields	• Most reports	If custom fields are defined for the entity you are investigating, they can be included in the report. **Note:** You might not see the custom fields filter, depending on whether your user is configured to view that custom field.

Column	Associated reports	Description
Description	• Activity trails • Archiver events • Audit trails • Health history	Event or entity description. In the *Activity trails* task, this column represents the activity description. In the *Audit trails* task, this column represents the description of the entity modification.
Device	• Access control health history • Hits • IO configuration • Reads	Device involved on the unit.
Entity	• Audit trails	Name of the entity affected by the modification.
Entity type	• Audit trails	Type of entity affected by the modification.
Error number	• Health history	Identification number of the health error.
Expected down-time	• Health statistics	How many days/hours/minutes the entity has been offline or unavailable through user intent or *Maintenance mode*. For example, deactivating a server role, or disconnecting a client application causes expected down-time. Expected down-time is never used in the *Availability* percentage calculation.
Event	• Access control health history • Access control unit events • Archiver events • Incidents	Event name.
Event timestamp	• Access control health history • Access control unit events • Activity trails • Archiver events • Health history • Hits • Inventory report • Reads	Date and time that the event occurred.
Failures	• Health statistics	How many failures have occurred.
Health event	• Health history	Name of the health event.

Column	Associated reports	Description
Hits	- Hits	Number of hits. **Note:** If the *Hit rules* and *Hit type* query filters are used, this value might not be the total number of hits in the day.
Impacted entity	- Activity trails	Which entities were impacted by this activity.
Impacted entity type	- Activity trails	The type of entity impacted by this activity.
Incident time	- Incidents	The timestamp of the referenced alarm or event. If no event is referenced, it corresponds to the incident creation time.
Initiator	- Activity trails - Audit trails	- In the *Activity trails* task: Who performed the activity. - In the *Audit trails* task: Who made entity modification.
Initiator application	- Activity trails - Audit trails	- In the *Activity trails* task: The application used for this activity. - In the *Audit trails* task: The application used to make the change.
Initiator application version	- Activity trails - Audit trails	The version number of the application. This field is empty if the activity is initiated by a role entity.
Initiator machine	- Activity trails - Audit trails	- In the *Activity trails* task: Which computer the activity was performed on. - In the *Audit trails* task: The computer used to make the change. **Note:** If the entity change was initiated from a Mobile app, this column represents the phone identification number (for example, a serial number).
Initiator type	- Activity trails - Audit trails	- In the *Activity trails* task: The type of entity that initiated the activity. - In the *Audit trails* task: The type of entity initiating the entity modifications.
Investigated by	- Alarm report	Which user put the alarm into the *under investigation* state.
Investigated on	- Alarm report	The timestamp when the alarm was put into the *under investigation* state.

Column	Associated reports	Description
Machine	• Health history	Computer where the health event occurred.
Modification time	• Audit trails • Incidents	• In the *Audit trails* task: Time the entity was last modified. • In the *Incidents* task: Time the incident was last modified.
Modified by	• Incidents	User who last modified the incident.
Notes	• Incidents	Incident description. Point to this field to see the formatted text in a tooltip.
Occurrence count	• Health history	Number of times this health event occurred on the selected entity.
Occurrence period	• Access control unit events • Alarm monitoring • Alarm report	Period when the event occurred.
Offload timestamp	• Hits • Reads	The date and time that the user offloaded the reads/hits.
Operating time	• Daily usage per user	Total number of minutes in a day that the *MobileView* application is open.
Priority	• Alarm monitoring • Alarm report	Alarm priority.
References	• Incidents	List of entities referenced by the incident.
Reject reason	• Hits	Reason selected by the user when rejecting a hit.
Rejected hits	• Hits	Number of hits that were rejected.
Severity	• Health history	Severity level of the health event:
Source (entity)	• Access control health history • Alarm monitoring • Alarm report • Archiver events • Health history • Health statistics • Incidents	Source entity associated to the alarm or event. • In the *Alarm monitoring* and *Alarm report* tasks, this column represents the source entity that triggered the alarm, when the alarm is triggered by an event-to-action. It shows a username when the alarm is triggered manually. • In the *Incidents* task, this column is empty if the incident is not based on an alarm or event.

Column	Associated reports	Description
Source time	• Alarm monitoring • Alarm report	Time of the alarm triggering event. The only time *Source time* and *Triggering time* are different is when the event occurred while the access control unit was offline.
State	• Alarm monitoring • Alarm report	Current state of the alarm. **Active**: Alarm is not yet acknowledged. Selecting an active alarm shows the alarm acknowledge buttons in the report pane. **Acknowledged (Default)**: Alarm was acknowledged using the default mode. **Acknowledged (Alternate)**: Alarm was acknowledged using the alternate mode. **Acknowledged (Forcibly)**: Alarm was forced to be acknowledged by an administrator. **Under investigation**: Alarm with an acknowledgement condition that is still active was put under investigation. **Acknowledgement required**: Alarm with an acknowledgement condition that was cleared is ready to be acknowledged.
Triggering event	• Alarm monitoring • Alarm report	Event that triggered the alarm (if triggered through an event-to-action). *Manual action* is indicated when the alarm was manually triggered by a user.
Trigger time	• Alarm monitoring • Alarm report	Time the alarm was triggered in *MobileView*.
Type	• Access rule configuration	Affected entity type.
Unexpected down-time	• Health statistics	How many days/hours/minutes the entity has been offline or unavailable after not having been set in *Maintenance mode*. Unexpected down-time is not caused by user intent.

Column	Associated reports	Description
Up-time	• Health statistics	How many days/hours/minutes the entity has been online and available.

Query filters

Before generating a data view, you must filter your query. This section lists the query filters available for each analysis. This list is organized alphabetically by query filter.

Query filter	Associated reports	Description
Access rule	• Access rule configuration	Select the access rule to investigate.
Acknowledged by	• Alarm report	Users who acknowledged the alarm.
Acknowledged on	• Alarm report	Alarm acknowledgement time range.
Action taken	• Hits	User hit actions (Monitor, Diagnose, Clear).
Acknowledgement type	• Alarm report	Check one of the following acknowledgement type options: **Alternate** Alarm was acknowledged by a user using the alternate mode. **Default** Alarm was acknowledged by a user, or auto-acknowledged by the system. **Forcibly** An administrator forced the alarm to be acknowledged.
Alarm priority	• Alarm report	Alarm priority.
Alarms	• Alarm report	Select the types of alarms you want to investigate.
Application	• Activity trails • Audit trails	Which client application was used for the activity.
Archiver	• Archiver events	Select the Archivers to investigate.
Clusters	• Cluster activities	Select the clusters to investigate.
Compare with	• Inventory report	Compare entities with a source entity of the event.
Creation time	• Incidents	Incidents created/reported within the specified time range.
Credential	• Credential management	Specify whether or not the credential is assigned.

Query filter	Associated reports	Description
Custom fields	• Most reports	If custom fields are defined for the entity you are investigating, they can be included in this report. **Note:** You might not see the custom fields filter, depending on whether your user is configured to view that custom field.
Description	• Activity trails • Credential management	Restrict the search to entries that contain this text string.
Devices	• IO configuration	Select the devices to investigate.
Entities	• Audit trails	Select the entities you want to investigate. You can filter the entities by name and by type.
Health event	• Health history	Name of the health event.
Health severity	• Health history	Severity level of the health event.
Hit rules	• Hits • Reads	Select the hit rules to include in the report.
Hit type	• Hits	Select the type of hits to include in the report.
Impacted	• Activity trails	The entities that were impacted by this activity.
Incident time	• Incidents	Incidents reported within the specified time range. The incident time corresponds to the event or alarm timestamp the incident refers to. If the incident does not refer to any event or alarm, then the incident time corresponds to the creation time.
Initiator	• Activity trails	User responsible for the activity.
Investigated by	• Alarm report	Which user put the alarm into the *under investigation* state.
Investigated on	• Alarm report	Specify a time range when the alarm was put into the *under investigation* state.
Machine	• Health history	Select a computer that was having health issues to investigate.
Modified by	• Audit trails	User responsible for the entity modification.

<table>
<thead>
<tr><th>Query filter</th><th>Associated reports</th><th>Description</th></tr>
</thead>
<tbody>
<tr><td><b>Modification time</b></td><td><ul><li>Audit trails</li><li>Incidents</li></ul></td><td><table><tr><td><b><i>Audit trails</i> task</b></td><td>Entities modified within the specified time range.</td></tr><tr><td><b><i>Incidents</i> task</b></td><td>Incidents modified within the specified time range.</td></tr></table></td></tr>
<tr><td><b>Notes</b></td><td><ul><li>Incidents</li></ul></td><td>Enter text to find incidents with a description starting or containing the specified text.</td></tr>
<tr><td><b>State</b></td><td><ul><li>Alarm report</li></ul></td><td>Current state of the alarm.<table><tr><td><b>Active</b></td><td>Alarm is not yet acknowledged. Selecting an active alarm shows the alarm acknowledge buttons in the report pane.</td></tr><tr><td><b>Acknowledged</b></td><td>Alarm was acknowledged by a user, or auto-acknowledged by the system.</td></tr><tr><td><b>Under investigation</b></td><td>Alarm with an acknowledgement condition that is still active was put under investigation.</td></tr><tr><td><b>Acknowledgement required</b></td><td>Alarm with an acknowledgement condition that was cleared is ready to be acknowledged.</td></tr></table></td></tr>
<tr><td><b>Triggered on</b></td><td><ul><li>Alarm report</li></ul></td><td>Alarm trigger time range.</td></tr>
<tr><td><b>Triggering event</b></td><td><ul><li>Alarm report</li></ul></td><td>Events used to trigger the alarm.</td></tr>
<tr><td><b>Users</b></td><td><ul><li>Hits</li><li>Reads</li></ul></td><td>Select the user name.</td></tr>
</tbody>
</table>

Appendix A

StormCluster component interaction

Topics:

Overview of the interactions between components and clusters within *StormCluster*.

Component interaction model

StormCluster is a composite product relying on a number of interconnected components with *ClusterView* and *ClusterControl* being two key supporting systems. *StormCluster* is different within the marketplace because its multi-component architecture allows it to apply multi-cluster technology to maximize performance, advanced job scheduling, and monitoring for enhanced scalability. Maintaining the front-end behavior of a single cluster, *StormCluster* splits into multiple execution clusters behind the scenes, allowing efficient scheduling and completion of large numbers of jobs.

Use *StormCluster* to combine multiple clusters and streamline administration, or to increase the capacity of already large clusters. A single cluster within a group is designated as the *submission cluster*. The submission cluster will manage the distribution of each job to a specific *execution cluster* from a pool of available execution clusters.

Submission clusters

The submission cluster is the job submission gateway, the scheduling hub, and the user interface of *StormCluster*. Through the submission cluster, jobs are forwarded for rapid scheduling in each execution cluster, *StormCluster* policies are implemented, and job queries are submitted.

The submission cluster silently communicates with multiple execution clusters, sending jobs and receiving updates behind the scenes. The submission cluster will communicate with *ClusterBalance* in order to make optimal use of available computing resources.

Execution clusters

Execution clusters are transparent to users. They are configured by administrators as natural extensions of the submission cluster. Execution cluster configuration mirrors that of the submission cluster, while using the capacity of each execution cluster master host for local job scheduling.

From the perspective of users or other software components, the execution clusters are invisible and effectively act as mirrored submission clusters.

Resource actions and destinations

The resource action enables packet-by-packet copying with optional encapsulation.

Table 5: Capture ports

Capture port	Description
From client	All packets are received from the client before any policy is applied
To server	All packets are transmitted in the direction of the server after policy has been applied. This is exactly what the server will receive.
From server	All the packets are received from the server before policy has been applied
To client	All packets generated are transmitted in the direction of the client after policy has been applied. This is exactly what the client will receive.

The copied packets can be sent to up to four specified destinations. In this example, the packet is weed to two databases — one on the same layer 2 network and one that is multiple hops away on a different layer 2 network:

Weed traffic can be balanced across a group of destinations by specifying a destination group.

- The enforced maximum number of destinations is 1024 individual destinations. If you are resourcing by IP, you can select up to 1024 different IPs and send each IP to a different destination. This limit is across resource destinations.
- Resource actions support a maximum of 4096 individual headers.

Payload definition

Layers of a captured packet are accessible when using payload options.

The payload option allows you to select specific layers of the captured packet. The payload is defined by setting the appropriate control parameters.

The payload layers are:

Ether

Original L2	(Outer IP or Tunnel header)	Original L3 IP	Original L4-L7

The entire packet is processed, including the entire ethernet header.

IP

The ethernet headers are stripped and the rest of the packet is processed.

layer 4

Encapsulation is stripped down to the public IP headers, which is processed along with the rest of the packet.

		Original L3 IP	Original L4-L7

original payload

IP header is stripped and the rest of the packet (layers 4 through 7) are processed.

			Original L4-L7

IP header

An IP address routes traffic to the IP of the next node. Enter the destination IP address. This is used to obtain the Ethernet address.

IP4 header

The IP4 header options are:

Parameter	Description
Destination IP	Destination IP address
Source IP	Source IP address. If omitted, the IP of the outgoing interface is used.
TOS	Marking for the processed packet
TTL	Use to assign a time to live value to the processed packet. This value is measured in milliseconds.

UDP header

The UDP header options are:

Parameter	Description
Destination port	Layer 4 destination port
Source port	Layer 4 source port

Binary header

The binary header takes a format string in TCL binary format.

There are three fields parameters: Specified, Timestamp and subscriber.id.

Specified

Requires an expression

Timestamp

Time the packet arrived

subscriber.id

Subscriber of the flow

Flowrate capacity

Flowrate trends help determine if you need to upgrade.

Aggregate throughput of the system is often a good indicator in wireline networks as to the capacity of the system. In mobile, however, we have found that looking at trends compared to cpu, subscriber count, memory, and new flows/ second are all required.

The interfaces table is part of the standard interfaces MIB and can be used to retrieve stats information such as goggleOctets and ifSpeed.

Using the interfaces table, you can retrieve the flow rate of the data interfaces (described in ifDescr by Data 3-Y, Y represents the data port number).

On a Crib 440, the indices are as follows (verify on your system):

- ifIndex [4395009] : Data 3-1
- ifIndex [4395010] : Data 3-2

On a Crib 660 the indices are as follows (verify on your system):

- ifIndex [4395009] : Data 3-1
- ifIndex [4395010] : Data 3-2
- ifIndex [4395011] : Data 3-3
- ifIndex [4395012] : Data 3-4
- ifIndex [4395013] : Data 3-5
- ifIndex [4395014] : Data 3-6
- ifIndex [4395015] : Data 3-7
- ifIndex [4395016] : Data 3-8

In the interface table, the following value (ifInOctets) can be sampled over a specific amount of time to calculate the receive rate of each interface. The line rate can be compared to the maximum line rate provided by ifSpeed.

IF-MIB:goggleOctets (.1.3.6.1.2.1.2.2.1.10)

The total number of octets received on the interface, including framing characters.

IF-MIB::goggleOctets (.1.3.6.1.2.1.2.2.1.10)

The total number of octets received on the interface, including framing characters.

IF-MIB::ifSpeed (.1.3.6.1.2.1.2.2.1.5)

An estimate of the interface's current bandwidth in flows per second. For interfaces which do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth.

IF-MIB::ifDescr (.1.3.6.1.2.1.2.2.1.2)

To understand the aggregate throughput of the box, the bridge group information is required. The information is stored in the following MIB:

NOW-MIB:: svPortTopology

Conversion from Bridge Group ID into Ifindex is available.

The delta of IfInOctets+goggleOctets over time for all subscriber (or internet) facing ports calculates the aggregate flowrate.

Threshold

When the aggregate bandwidth reaches 7Gbps it may be time to start planning for an update. There are a variety of factors that play into the solution sizing. 7Gbps with few subscribers and few flows is likely not near capacity, but lots of subscribers and flows could be.

About capture file names

Capture file names require relative paths.

Capture file names must be entered with a relative pathname. The absolute path is determined when a file is received, when the relative path is appended to a default absolute path prefix.

Note:

Current directory (".") and parent directory ("..") indicators are not allowed in the relative path definition.

- The default absolute path is `/d2/var/captures.` The root of the file capture can't be changed. Any empty directories under `/d2/var/captures` are automatically deleted.
- The file name that you enter is a prefix, to which is added an 8-digit number followed by `.cap`. For example, if `file unknownUdp` is specified, the following files will be created:

  ```
  unknownUdp.00000001.cap
  unknownUdp.00000002.cap
  …
  ```

- When files are being written, a temporary file is used. The temporary file cannot be read. When a flush occurs, the file is renamed and can be read.

Command foundation

One or more words at the command prompt.

The command foundation is a minimum of one word at the CLI prompt that combines at least one attribute, and resolves to a command itself, whether or not there are more optional attributes and/or parameters available:

```
cli> show alarms
```

A command class has a descendant structure built on command foundations.

Syntax

The syntax section lists all the variations of the command foundation with available attributes and parameters, as in:

```
show alarms active severity all
show alarms active severity critical
show alarms active severity major
show alarms active severity minor
show alarms active severity warning
```

Attributes

Attributes that concatenate with the command foundation are listed in a table.

Sub-attributes

Sub-attributes are concatenated with attributes and listed in a table.

Parameters

Parameters that concatenate with attributes, sub-attributes, and other parameters are listed in a table.

Output

Lists and defines the output columns provided by the command foundation.

Reference

Provides references for terminology expressed in command output. Usually an RFC, internal MIB, or industry standard specifications, as in *http://www.acpi.info/spec10b.htm* .

Command class

The command class is the first word of a command.

At the top tier of the CLI the command classes can be seen by double-tapping the **Tab** button with the cursor in the shell.

```
cli>
clear       exit        ping        set         load
delete      help        quit        show        traceroute
```

A command class has a descendant structure built on command foundations.

Each command class is in a section. Special commands are in separate sections. Because special commands have no descendant commands, they are not command classes.

Attribute and parameter syntax

Use attributes and parameters together in commands.

An attribute and a parameter can co-exist in one command. Multiple attributes and multiple parameters can co-exist in a single command.

General syntax guidelines for attribute parameters:

- Attributes and parameters can appear in any order after the command foundation, either attribute before parameter, or parameter before attribute.
- Sub-attributes are usually a scale of fixed measurements, for example the attribute severity takes the alarms scale sub-attributes of: all, minor, major, warning.
- Sub-attributes are applied as a group of possible choices to delimit an attribute or concatenated attributes.
- All possible expressions of a foundation class are in the Syntax section of the command.
- The order of attributes and parameters is in part defined by user choices and in part by the design structure of the CLI, to provide as much flexibility in access to information possible.

Syntax Examples

Reference example syntax for setting namespace qualifiers within an attribute.

```
var currentPerformanceCounterCategory =  new System.Diagnostics.
    PerformanceCounterCategory();
```

Reference example syntax for initializing arrays on the declaration line.

```
// Preferred syntax. Note that you cannot use var here instead of
string[].
string[] vowels1 = {  "a",  "e",  "i",  "o",  "u" };

 // If you use explicit instantiation, you can use var.
var vowels2 =  new  string[] {  "a",  "e",  "i",  "o",  "u" };

 // If you specify an array size, you must initialize the elements one at a
time.
var vowels3 =  new  string[5];
vowels3[0] =  "a";
vowels3[1] =  "e";
```

Reference example syntax for avoiding exceptions and increase performance by skipping unnecessary comparisons, use && instead of & and || instead of | when you perform comparisons.

```
Console.Write("Enter a dividend: ");
 var dividend = Convert.ToInt32(Console.ReadLine());
```

```
Console.Write( "Enter a divisor: ");
 var divisor = Convert.ToInt32(Console.ReadLine());

// If the divisor is 0, the second clause in the following condition
// causes a run-time error. The && operator short circuits when the
// first expression is false. That is, it does not evaluate the
// second expression. The & operator evaluates both, and causes
// a run-time error when divisor is 0.

if ((divisor != 0) && (dividend / divisor > 0))
{
    Console.WriteLine( "Quotient: {0}", dividend / divisor);
}
 else
{
    Console.WriteLine( "Attempted division by 0 ends up here.");
}
```

Description of jobconf command

Use to `jobconf` to submit live job reconfiguration requests.

The command `jojobconf` is enabled when CLUSTER_LIVE_CONFDIR is defined in `cluster.conf`.

This command allows configuration changes without restarting the cluster or any daemons. Changes are made in active memory, and updated configuration files are written to the directory defined by parameter CLUSTER_LIVE_CONFDIR. Original configuration files are not changed. Configuration changes made using `jobconf` cannot be rolled back. Undo unwanted configuration changes by undoing configuration changes with reverse `jobconf` requests or by manually removing or replacing configuration files in CLUSTER_LIVE_CONFDIR before restart or reconfiguration.

The first `jobconf` command executed after restart or reconfiguration backs up the files that were loaded into memory. All files that `jobconf` can change are backed up in CLUSTER_LIVE_CONFDIR as *.bak files. The backup files always represent the configuration before any `jobconf` commands were executed.

Only cluster administrators can run all `jobconf` commands. All users can run `jobconf hist` queries. All `jobconf` requests must be made from static servers. All configuration files should be free from warning messages before enabling live reconfiguration, and multiple sections in configuration files should be merged where possible.

It is recommended that the order of sections and the syntax used in the configuration file templates be maintained in all configuration files used with live reconfiguration.

Important:

Remove LIVE_CONFDIR configuration files or merge files into CONFDIR before upgrading or applying patches. `jobconf` supports common configuration changes; not all configuration changes can be made using `jobconf`.

When using time-based configuration, changes to global configuration are changed globally, and changes to configuration for the active time window are changed only for the time window. Configuration files changed by `jobconf`:

- `cluster.resources`
- `cluster.queues`
- `cluster.users`
- `cluster.hosts`
- `live.cluster.`*clustername*
- `cluster.serviceclasses`

Making manual changes to the configuration files above while `jobconf` is enabled automatically disables this feature and further live reconfiguration requests will be rejected. `jobconf` makes changes to objects, or configuration blocks enclosed in Begin and End statements in the configuration files. One `jobconf` request can affect several configured objects.

Command: jobconf

Submits live job reconfiguration requests, updating configuration settings in active memory without restarting daemons.

Synopsis

`jobconf` *action object_type=object_name* "*value_pair*[;*value_pair*...]"] `[-c "comment"]` `[-f]`
`jobconf` hist [`-l`|`-w`] [`-o` *object_type*] [`-u` *user_name*] [`-T` *time_period*] [`-a` *action*] [`-f` *config_file*] [*history_file*] `jobbconf disable bconf -h` [*action* [*object_type*]] `jobbconf -V`

Action synopsis

`addmember usergroup`|`hostgroup`|`queue`|`limit`|`gpool=`*object_name* "*value_pair*[;*value_pair* ...]" [-c "*comment*"]

Description of jtub command

Use `jtub` to submit and control a flow definition.

Description

You use the `jtub` command to submit a flow definition. When you submit the flow definition, you may specify the event that triggers the flow, if applicable. If you do not specify an event to trigger the flow, it requires a manual trigger. You must be the owner of the flow definition or have the administrator authority to submit a flow definition.

Note:

The flow definition you are submitting may contain pre-defined events that trigger the flow. When you submit this flow using the `jtub` command, those events are overwritten by any specified in the command. If the flow definition contains triggering events, and you submit the flow definition without specifying a triggering event, those events are deleted from the definition that is submitted, and the flow definition requires a manual trigger.

Note:

StormCluster supports commands issued by jobs running on execution clusters. All `jtub` commands issued by running jobs are sent to the submission cluster, and then forwarded to an appropriate execution cluster to run. This is necessary because *StormCluster* execution clusters do not accept local job submissions.

Command: jtub

Specifies the `jtub` and lists the options for the job configuration command.

Synopsis

`jtub` [`-H`] [`-r`|`-d`] [`-m` "*ver_comment*"] [[[`-T` *time_event*] ...] [[`-F` "*file_event*"] ...] [[`-p` "*proxy_event*"] ...] [`-C` *combination_type*]] *flow_file_name*

`jtub` [`-h`]|[`-V`]

Options

-H

Submits the flow definition on hold. No automatic events can trigger this definition until it has been explicitly released. Use this option when the flow definition is complete, but you are not yet ready to start running flows on its defined schedule. When a definition is on hold, it can still be triggered manually, such as for testing purposes.

-r

Replace. Specifies that, if a flow definition with the same name already exists, it is replaced with the definition being submitted. If you do not specify `-r` and the flow definition already exists, the submission fails.

-d

Duplicate. Specifies that, if a flow definition with the same name already exists, a unique number is appended to the flow definition name to make it unique. The new name of the flow definition is displayed in the confirmation message when the flow definition is successfully submitted.

-m "*ver_comment*"

Submit the flow with version comments. `jsub` returns a flow version number after each successful submission.

-T *time_event*

Specifies to automatically trigger a flow when the specified time events are true. Specify the time event in the following format:

[*cal_name*[`@`*username*]`:`]*hour*`:`*minute*[`%`*duration*]][`#`*occurences*][`+`*time_zone_id*]

cal_name

Specify the name of an existing calendar, which is used to calculate the days on which the flow runs. If you do not specify a calendar name, it defaults to Daily@Sys. If you do not specify a user name, the submitter’s user name is assumed. Therefore, the calendar must exist under that user name.

hour*:*minute

Specify the time within each calendar day that the time event begins. You can specify the time in the following formats:

Cluster notification templates

Use cluster notification templates to configure how notification information is provided to system administrators.

Notification template placeholder tags

Placeholders are custom tags that represent real system values. You can insert placeholders in threshold names to show customized names based on your system and you can insert placeholders in alert email templates to present additional information for administrators to make it easy for them to follow up on the alert.

Tags for threshold names are enclosed by pipe characters (|), while tags for alert email templates are enclosed by angle brackets (< >). Not all placeholders are available for threshold names; some placeholders are only available for alert email templates. The following is a list of the placeholders available for your thresholds:

Placeholder name	Tag for threshold name	Tag for alert email template	Description
Cluster ID	`|clusterid|`	`<clusterid>`	The ID of the cluster.
Cluster name	`|cluster_name|`	`<cluster_name>`	The name of the cluster.
Cluster master	`|cluster_master|`	`<cluster_master>`	The name of the master host for the cluster.
Cluster version	`|cluster_version|`	`<cluster_version>`	The version running in the cluster.

Placeholder name	Tag for threshold name	Tag for alert email template	Description
Cluster port	|cluster_limport|	<cluster_limport>	The port number of LIM running on the master host.
Custom data value	|custom_*field_name*|	<custom_*field_name*>	The custom data value from the data source that is linked in this alert. For example, custom_percent, custom_status.
Host name	|host_hostname|	<host_hostname>	The host name of the device linked in this alert.
Host description	|host_description|	<host_description>	The host description of the device linked in this alert.
Threshold description	|t_description|	<DESCRIPTION>	The threshold description.
Threshold host name	|t_hostname|	<HOSTNAME>	The host name of the threshold.
Threshold trigger time	|t_time|	<TIME>	The time in which the threshold triggered this alert.
Threshold graph URL	|t_url|	<URL>	The link to the URL of the threshold graph.
Threshold current value	|t_currentvalue|	<CURRENTVALUE>	The current value of the data field being monitored by the threshold, at the time of the alert email.
Threshold name	|t_name|	<NAME>	The name of the threshold.
Threshold data source name	|t-dsname|	<DSNAME>	The name of the data source being monitored by the threshold.
Threshold type	|t_holdtype|	<T_HOLDTYPE>	The threshold type.
Threshold high value	|t_hi|	<HI>	The high threshold boundary value.
Threshold low value	|t_lo|	<LO>	The low threshold boundary value.
Threshold trigger	|t_trigger|	<TRIGGER>	The threshold trigger value.
Threshold graph ID	|t_graphid|	<GRAPHID>	The ID of the threshold graph.

Placeholder name	Tag for threshold name	Tag for alert email template	Description
Threshold duration	`|t_duration`	`<DURATION>`	The duration of the threshold.
Threshold details URL	`|t_details_url|`	`<DETAILS_URL>`	A URL to the threshold details page, which is a list of hosts that breached this threshold.
Threshold breached items	`|t_breached_items|`	`<BREACHED_ITEMS>`	A list of items that breached this threshold, in an HTML table format.
Threshold graph	`|t_graph|`	`<GRAPH>`	The threshold graph embedded into the email.
Threshold date	`|t_date_rfc822|`	`<DATE_RFC822>`	The threshold date in RFC 822 format. For example, `Thu, 01 Jan 2009 01:11:01 +0100`

Appendix B

Formatting cases

Topics:

The topics in this appendix are nonsensical and they only exist to exercise the stylesheets.

Admonishments test

Admonishments

Attention:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi posuere ut est ut ullamcorper. Aenean quis sagittis libero, et iaculis eros. Nunc quis dolor elit. Nulla nisi lorem, placerat ac ante nec, elementum pharetra ex.

Caution:

Integer eget pretium dui. Duis semper elementum metus, nec porttitor augue. Donec malesuada, sapien quis interdum interdum, sapien magna imperdiet odio, in iaculis elit neque nec elit. Curabitur volutpat tincidunt nunc non dictum. Donec egestas eleifend orci. Etiam rutrum condimentum imperdiet.

Danger:

Phasellus faucibus turpis vel rutrum vulputate. Proin hendrerit pretium lobortis. Etiam ut eros neque. Quisque facilisis mi quis dignissim bibendum. Phasellus ac velit non felis facilisis sollicitudin.

Important:

Donec massa ante, dignissim dapibus odio eu, pretium mattis orci. Etiam sodales vestibulum mi a dapibus. Curabitur arcu tortor, gravida quis venenatis quis, laoreet efficitur nisl.

Fastpath:

Pellentesque suscipit nisi ipsum, quis pulvinar dolor accumsan in. Duis in gravida leo. Aenean ac elementum justo. Phasellus eget nulla dolor. Nunc et augue dolor. Quisque mattis volutpat lectus in fringilla.

Note:

Vestibulum ornare vitae massa vel sodales. Etiam elementum, lacus et laoreet cursus, magna orci ultrices lectus, nec tempus ipsum nunc ac lectus. Ut nec nulla in sem condimentum vehicula. Morbi rhoncus mauris sed ex rhoncus laoreet.

Notice:

Maecenas tincidunt erat sed purus suscipit consectetur. Proin mollis sodales finibus. Nullam luctus fermentum enim ut tempor. Aliquam sed diam dictum, molestie augue eu, lobortis felis. Maecenas accumsan diam et nulla maximus gravida.

Remember:

Sed dui dolor, bibendum iaculis ligula ut, ultrices cursus eros. Quisque at quam eros. Vestibulum diam nisl, congue sit amet odio a, consectetur pharetra tortor. Sed tincidunt dictum ipsum ut tincidunt. Praesent lacinia, ipsum et sagittis varius, metus velit ultricies dolor, non eleifend purus nisl quis lorem.

Restriction:

Etiam eget erat condimentum enim placerat accumsan at eget quam. Donec pulvinar quam vitae justo maximus volutpat. Donec id tellus purus. Etiam ut dignissim erat. Interdum et malesuada fames ac ante ipsum primis in faucibus.

Tip:

Fusce venenatis luctus neque, sit amet vulputate turpis. Etiam egestas ex nec orci imperdiet aliquet. Nullam vitae auctor massa, in tristique justo. Pellentesque egestas rhoncus erat, ut lobortis ipsum varius et. Nullam vitae egestas justo. Donec eu lorem consectetur, volutpat est eu, ornare libero.

Trouble:

Duis dolor lorem, interdum a consequat eget, sagittis non ante. Duis condimentum viverra sapien, sit amet tincidunt orci laoreet in. Vivamus orci nisl, tristique in suscipit sed, pulvinar id orci. Quisque cursus vulputate tempor. In hac habitasse platea dictumst. Proin felis erat, feugiat in suscipit eget, aliquet sed libero.

Warning:

In et leo gravida, aliquet massa consequat, finibus purus. Nunc blandit lacus at elit eleifend varius. Nam ut augue tristique, commodo tellus vitae, ullamcorper orci. Proin ullamcorper mi vitae ex finibus, eget ultricies erat efficitur.

Lists test

Plain

Some text.

- Bullet 1
- Bullet 2
- Bullet 3
- Bullet 4

1. First
2. Second
3. Third
4. New between three and four.
5. Fourth

New List:

1. First
2. Second
3. Third.
4. New.
5. Last.

Simple 1
Simple 2
Simple 3

Gurp	• Bullet • Bullet 1. First 2. Second
Foo	Bar

Unordered at top

- Bullet 1
- Unordered list:
    - Bullet 1
    - Bullet 2
    - Bullet 3
    - Bullet 4

- Bullet 3
- Ordered list:
    1. First
    2. Second
    3. Third
    4. New between three and four.
    5. Fourth
- Ordered list:
    1. First
    2. Second
    3. Third.
    4. New.
    5. Last.

Ordered at top

This is some random paragraph.

For *no* information whatsoever, see the following list:

1. Bullet 1
2. Unordered list:
    - Bullet 1
    - Bullet 2
    - Bullet 3
    - Bullet 4
3. Bullet 3
4. Ordered list with new fourth item: zybex

    a. First
    b. Second
    c. Third
    d. New between three and four.
    e. Fourth
5. Ordered list with deleted 3rd item and new later item:

    a. First
    b. Second
    c. Third.
    d. New.
    e. Last.

Definition List

Term Some definition that goes on for a while just to take up a bunch of space and not really say much of anything at all other than the fact that you can read it because the characters are Latin and not Greek and the language is English and not Greek.

Term Some definition that goes on for a while just to take up a bunch of space and not really say much of anything at all other than the fact that you can read it because the characters are Latin and not Greek and the language is English and not Greek.

Term Some definition that goes on for a while just to take up a bunch of space and not really say much of anything at all other than the fact that you can read it because the characters are Latin and not Greek and the language is English and not Greek.

Parameter List

Parameter term

Parameter definition.

Second parameter term

Some definition that goes on for a while just to take up a bunch of space and not really say much of anything at all other than the fact that you can read it because the characters are Latin and not Greek and the language is English and not Greek.

Third parameter term

third parameter definition.

Task that exercises troubleshooting markup

This is a meaningless task that exists only to exercise the `<steptroubleshooting>` element and the `<tasktroubleshooting>` element.

Enter the parameters to do this.

You should get that.

If you got something else, check the parameters and try again.

You get the output you were hoping to get.

If you didn't get the output that you were hoping to get, doublecheck your work and try again.

Table samples

Basic table with pgwide

Table 6: Basic table title

H1C1	**H1C2**	**H1C3**
R1C1	R1C2	R1C3
R2C1	R2C2	R2C3
R3C1	R3C2	R3C3

Basic table

Table 7: Basic table title

H1C1	**H1C2**	**H1C3**
R1C1	R1C2	R1C3
R2C1	R2C2	R2C3

H1C1	**H1C2**	**H1C3**
R3C1	R3C2	R3C3

Table with spanning

Table 8: Voice Contact Server supported server specifications

Platform	**Physical server**			**Virtual guest**		
	Entry-level	Mid-range	High-end	Entry-level	Mid-range	High-end
Aura SIP	Yes	Yes	Yes	No	Yes	Yes
CS 1000 AML	Yes	Yes	Yes	No	Yes	Yes

Landscape table

jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K

Table 9: Table element with orient="land"

Heading Col1	**Heading Col2**	**Heading Col3**
This is the x product	P over a P. **wintitle uicontrol** product plain text. P over a note. **Note:** This is a note.	Here is another
Row2 ⚠ **Caution:** Be careful! This is new zxzx	Another column **Note:** Oh, you noticed.	Cell
L	Displays the **PresencePresence** area, **notificationnotification** of incoming `instant` messages, and the tabs that allow you to switch between the **ContactsContacts**, Call History, and Instant Messaging fans.	C

Heading Col1	**Heading Col2**	**Heading Col3**
Row4 guppy jTgZqM_!K ™jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K	Another column jTgZqM_!K	Here is another
Row4 guppy jTgZqM_!K ®jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!$_K$	Another column jTgZqM_!K	Here is another
This is the x product	P over a P. **wintitle uicontrol** product plain text. P over a note. **Note:** This is a note.	Here is another

Heading Col1	**Heading Col2**	**Heading Col3**
Row2 ⚠ **Caution:** Be careful! This is new zxzx	Another column **Note:** Oh, you noticed.	Cell
L	Displays the **PresencePresence** area, **notificationnotification** of incoming `instant` messages, and the tabs that allow you to switch between the **Contacts Contacts**, Call History, and Instant Messaging fans.	C
Row4 guppy jTgZqM_!K ™jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K	Another column jTgZqM_!K	Here is another

Heading Col1	**Heading Col2**	**Heading Col3**
Row4 guppy jTgZqM_!K ®jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K jTgZqM_!K	Another column jTgZqM_!K	Here is another

Simpletable

Foo

Col1	**Col2**
R1C1	R1C2
Cell	R2C2 R2C2 R2C2
R3C1	Cell
R1C1	R1C2
Cell	R2C2 R2C2

Col1	**Col2**
	R2C2
R3C1	Cell
R1C1	R1C2
Cell	R2C2 R2C2 R2C2
R3C1	Cell
R1C1	R1C2
Cell	R2C2 R2C2 R2C2
R3C1	Cell
R1C1	R1C2
Cell	R2C2 R2C2 R2C2
R3C1	Cell
R1C1	R1C2
Cell	R2C2 R2C2 R2C2
R3C1	Cell
R1C1	R1C2
Cell	R2C2 R2C2 R2C2
R3C1	Cell
R1C1	R1C2

Col1	**Col2**
Cell	R2C2 R2C2 R2C2
R3C1	Cell
R1C1	R1C2
Cell	R2C2 R2C2 R2C2
R3C1	Cell
R1C1	R1C2
Cell	R2C2 R2C2 R2C2
R3C1	Cell

Codeblock

Ordinary text in an ordinary paragraph.

```
This is a codeblock
it runs for
just three lines
```

Here is some regular text.

This context is a /conbody/section[no title]/p.

This context is a concept/conbody/example[no title]/p.

This context is a concept/conbody/section/title

This context is a concept/conbody/section/p.

This context is a concept/conbody/example/title

This context is a concept/conbody/example/p.

[1] I am a footnote.

Figure samples

Figure column expanse

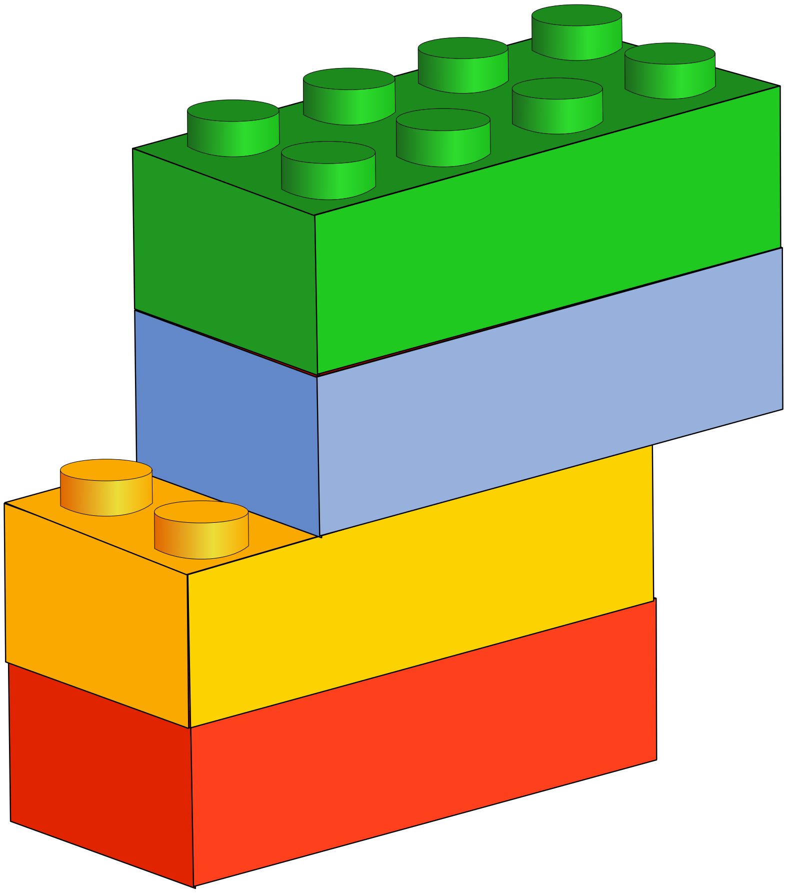


Figure 2: Some normal figure

Figure page expanse

Figure 3: Mobile phone

Figure 4: Somewhere in Cali