

# Dell Crowbar Software Framework

A Dell User's Guide

Version 1.5

Revised October 2012



## Table of Contents

---

<b>Notes, Cautions, and Warnings</b>	<b>3</b>
<b>Introduction</b>	<b>5</b>
Concepts	5
Opscode Chef Server	5
Dell Specific Options	5
New for Version 1.5	5
<b>The Crowbar Framework</b>	<b>5</b>
Architecture	5
System End State	6
Node Provisioning	6
NTP	7
DNS	7
Nagios®	7
Ganglia	7
Logging	7
Network Setup	7
Default Networks	8
Crowbar Operation	9
User Interface / Using Crowbar	12
General Layout	13
Nodes (System Dashboard)	13
Network	15
Barclamps	17
Utilities	18
<b>Core Crowbar Barclamps</b>	<b>19</b>
Crowbar Barclamp	20
Deployer Barclamp	20
Provisioner Barclamp	21
Network Barclamp	21
RAID Barclamp	23
BIOS Barclamp	24
IPMI Barclamp	24
NTP Barclamp	24
Logging Barclamp	25
Nagios® Barclamp	25
Ganglia Barclamp	25
Test Barclamp	25
<b>Supplemental Material</b>	<b>26</b>
System Verification	26
Managing Barclamps	26
<b>Support</b>	<b>27</b>
Crowbar Support	27
<b>To Learn More</b>	<b>27</b>

## Tables

---

Table 1: Default Networks	8
Table 2: Network Parameters	8
Table 3: Default Network Addresses	9
Table 4: Changes to proposals based on their current state	10
Table 5: Service URLs	12
Table 6: Deployment States	14
Table 7: Details Panel Buttons	14
Table 8: Buttons on Bottom of Details Panel	15
Table 9: Proposal Status	17
Table 10: Barclamps	19
Table 11: Crowbar Barclamp Parameters	20
Table 12: User Name Key	20
Table 13: Deployer Barclamp Parameters	21
Table 14: BIOS Map Entry Keys	21
Table 15: Provisioner Barclamp Parameters	21
Table 16: Network Configuration Options	22
Table 17: Teaming Sub-Parameters	22
Table 18: Default Networks	22
Table 19: Network Parameters	23
Table 20: Range Map String Key	23
Table 21: IPMI Barclamp Parameters	24
Table 22: NTP Barclamp Parameters	24
Table 23: Logging Barclamp Configuration Parameters	25
Table 24: Nagios <sup>®</sup> Barclamp Parameters	25
Table 25: Ganglia Barclamp Parameters.	25
Table 26: Test Barclamp Parameters.	25

## Figures

---

Figure 1: Life Cycle of a Barclamp from Concept to Proposal and Deployment	9
Figure 2: The States a System Progresses through to Fully Provision a System	11

## Notes, Cautions, and Warnings

---



**NOTE:** A NOTE indicates important information that helps you make better use of your computer



**CAUTION:** A CAUTION indicates potential damage to hardware or loss of data if instructions are not followed.



**WARNING:** A WARNING indicates a potential for property damage, personal injury, or death.



This guide provides this additional Chef Server information as notes flagged with the Opscode logo.

Information in this document is subject to change without notice.

© 2012 Dell Inc. All rights reserved.

Reproduction of these materials is allowed under the Apache 2 license.

Trademarks used in this text: Dell™, the DELL logo, Nagios®, Ganglia, Opscode Chef™, OpenStack™, Canonical Ubuntu™, VMware™, Dell Precision™, OptiPlex™, Latitude™, PowerEdge™, PowerVault™, PowerConnect™, OpenManage™, EqualLogic™, KACE™, FlexAddress™ and Vostro™ are trademarks of Dell Inc. Intel®, Pentium®, Xeon®, Core™ and Celeron® are registered trademarks of Intel Corporation in the U.S. and other countries. AMD® is a registered trademark and AMD Opteron™, AMD Phenom™, and AMD Sempron™ are trademarks of Advanced Micro Devices, Inc. Microsoft®, Windows®, Windows Server®, MS-DOS® and Windows Vista® are either trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries. Red Hat Enterprise Linux® and Enterprise Linux® are registered trademarks of Red Hat, Inc. in the United States and/or other countries. Novell® is a registered trademark and SUSE™ is a trademark of Novell Inc. in the United States and other countries. Oracle® is a registered trademark of Oracle Corporation and/or its affiliates. Citrix®, Xen®, XenServer® and XenMotion® are either registered trademarks or trademarks of Citrix Systems, Inc. in the United States and/or other countries. VMware®, Virtual SMP®, vMotion®, vCenter®, and vSphere® are registered trademarks or trademarks of VMware, Inc. in the United States or other countries.

Other trademarks and trade names may be used in this publication to refer to either the entities claiming the marks and names or their products. Dell Inc. disclaims any proprietary interest in trademarks and trade names other than its own.

October 2012

## Introduction

---

This document provides instructions for operating the Dell™ Crowbar software framework. Please refer to additional user's guides for specific products that are deployed by Crowbar, such as OpenStack™ or Apache™ Hadoop™.

### Concepts

The purpose of this guide is to explain the user interface of Crowbar. Use the *Crowbar Deployment Guide* for assistance with installing Crowbar and configuring the target systems.



**Note:** Concepts beyond the scope of this guide are introduced as needed in notes and references to other documentation.

### Opscode Chef Server

Crowbar makes extensive use of Opscode Chef Server, <http://opscode.com>. To explain Crowbar actions, you should understand the underlying Chef implementation.



To use Crowbar, it is not necessary to log into the Chef Server; consequently, use of the Chef UI is not covered in this guide. Supplemental information about Chef is included.

This guide provides this additional Chef information as notes flagged with the Opscode logo.

### Dell Specific Options

The Dell End User License Agreement (EULA) version of Crowbar provides additional functionality beyond that in the open source version. It also has a skin that is different from the open source version, resulting in a different color palette. When divergences are relevant, they are identified.



To perform some configuration options and provide some integration, we use libraries that cannot be distributed under open source licenses.

Crowbar is not limited to managing Dell servers and components. Due to driver requirements, some barclamps, for example BIOS and RAID, must be targeted to specific hardware; however, those barclamps are not required for system configuration.

### New for Version 1.5

This section is provided as a shortcut for previous Crowbar users.

- Node Alias and Grouping feature in Dashboard
- Node Alias and Description population in Bulk Edit
- Networking Menu: Switch and VLAN views
- Utility Menu
  - Import Barclamps
  - Export Logs and Chef Objects

## The Crowbar Framework

---

### Architecture

Crowbar provides a modular platform containing the building blocks to provision, monitor, and operate a large-scale cloud deployment. Starting with bare metal installation, Crowbar automates all the required installation and configuration tasks. The core capabilities provided are:

- Hardware configuration—updating and configuring BIOS and BMC boards
- Deployment of base operating system
- Deployment of cloud components
- Providing core network infrastructure services (NTP, DNS, DHCP)
- Monitoring availability and performance of all deployed components

To accomplish this, Crowbar is installed on a dedicated administration node. From this admin node, Crowbar manages the initial discovery and configuration of the other nodes in the system. Each function in the cloud is controlled by a Crowbar component called a barclamp. There are barclamps for Nagios®, Ganglia, NTP, and a variety of other basic infrastructure services. Each barclamp is responsible for all the aspects of the underlying technology required to make it usable. To control the operation of a barclamp, you create a proposal for the barclamp (or may edit one already in place). A proposal comprises several parts:

- Parameters to customize the operation of the barclamp; for example: upstream DNS resolvers.
- List of machines in the deployment that fulfill the different roles in the barclamp.
- Internal system information.

When provisioning a function, you start with a proposal generated by Crowbar. Each core service running on the admin server has a default proposal included as part of the Crowbar installation. You can edit these proposals before installing these services on the admin node.

When a proposal is committed, Crowbar configures the Chef server and other components in the systems (TFTP, DHCP, and so on) to build the setup described in the proposal. Machines in the deployment affected by the proposals have their configuration updated using Chef client-side components. At the end of the process, the function described by the proposal is ready for use.

To allow easy integration into your existing environments, Crowbar allows customization of its barclamps, and additional barclamps can be added. You can disable the default monitoring tools (Nagios® and Ganglia) if you prefer to use your own existing monitoring tools, and internal cloud services can be connected to extant services; for example the cloud's NTP service can be configured to synchronize with existing servers.

Finally, a cloud deployment is dynamic. Machines come and go, break down, or get repurposed. Crowbar's operational model makes sure that machines are hooked into the key infrastructure services. Critical services (for example Nagios® monitoring) are installed automatically on newly provisioned machines by default, and those machines may be easily allocated for use with any additional Crowbar services desired.

## System End State

The sections that follow describe the services and capabilities available, assuming the system is installed with defaults. As mentioned above, the Crowbar framework allows for many customizations. This section focuses on the primary use cases for Crowbar, namely integrating all the functions into an existing network environment. Later sections describe more advanced customization options.

## Node Provisioning

When a new node is added to the system, the node should be set up to allow PXE booting. Once a machine is powered on, Crowbar uses the PXE boot protocol to manage the provisioning process.

After a system is fully installed by Crowbar, it has the following characteristics:

- BIOS is updated and configured based on the system's usage.
- BMC (baseboard management controller) is configured to allow management and IPMI support.
- Base operating system (OS) is installed.
- Administration access to the OS is configured—IP addressing and SSH keys are installed.
- Nagios® and Ganglia monitoring scripts are installed for the functions deployed on the system.
- Chef-client daemon is configured to maintain the system's state in sync.
- NTP sync client is configured.

Additionally, the system is configured to fulfill the functions that are deployed on it and is added to the appropriate cluster.

Crowbar's network barclamp carries on responsibilities related to L2/L3 management, namely:

- Networking administration.
- Physical NIC configuration—BMC port allocation (teamed or not).
- VLAN configuration on nodes.
- IP address location service, used by the rest of Crowbar. Addresses can be allocated from different pools, meant for different usages; for example: admin network, BMC, storage, and public.

The above functions involve managing information on the admin node and remotely executing operations on the compute nodes as they are provisioned. On each compute node, NICs are defined to match VLANs and appropriate addressing information is configured.

## NTP

The admin node runs an NTP server to synchronize time on all the machines in the cluster. Optionally, the NTP server can synchronize with upstream servers, in which case nodes are configured to sync their local time to those servers instead.



**Note:** Since all nodes in the cluster rely on the admin node, it is important that the time, date, and time zone are correctly set on this node. Crowbar expects the hardware clock on the admin node to be set to the UTC time zone.

## DNS

The admin node runs a DNS server to allow resolution of internal and (optionally) external names. The DNS server can be configured with the following:

- A list of upstream DNS servers to contact.
- A set of static mappings.
- The default domain name suffix.
- Crowbar makes sure that when a new machine is added to the deployment, it has a default entry added to the DNS zone. The default host name is the machine's MAC address prefixed by the letter "d" (for example: d00-a4-b3-c2-d1-e0.yourdomain.com).

## Nagios®

Nagios® monitors provisioned services for availability. Each cluster instance is represented as a host group, letting you quickly identify the health of a given instance.

Crowbar installs the Nagios® server on the admin node and configures it to monitor all the nodes in the system. As new nodes are brought online, Crowbar dynamically updates Nagios® to include them.

## Ganglia

Ganglia monitors the installed cluster for capacity and performance information, letting you easily gauge the cluster's capacity and check recent activity.

## Logging

The admin node serves as a central log repository. Its syslog daemon is configured to accept remote messages and each node is configured to forward all messages there.

## Network Setup

Due to the nature of the different software used, the network is set up as flat as possible using a dedicated BMC port and bonded LOMs. Crowbar manages all networks, and comes out of the box preconfigured to allow the initial configuration to come up quickly by predefining the storage, admin, public, and BMC networks.

The Crowbar network configuration can be customized to better map to site-specific networking needs and conventions. These changes include adding additional VLANs, changing VLAN mappings, and teaming NICs.

Please refer to the *Network Barclamp* section for specific details.



**NOTE:** Networks for the environment are configured when the Crowbar installation is performed. They cannot be changed without re-installing Crowbar. For specific information on how to configure the network JSON file, please see the Crowbar Deployment Guide.

## Default Networks

The default networks are presented in the following table. These defaults can be modified prior to installing Crowbar for your specific environment.

**Table 1: Default Networks**

Usage	Description	Default reserved VLAN tag	Tagged
<b>Admin/Internal VLAN</b>	Used for administrative functions such as Crowbar node installation, TFTP booting, DHCP assignments, KVM, system logs, backups, and other monitoring. There is only one VLAN set up for this function and it is spanned across the entire network.	100	Not tagged
<b>BMC VLAN</b>	Used for connecting to the BMC of each node.	100	Not tagged
<b>Storage VLAN</b>	Used by the Swift storage system for replication of data between machines, monitoring of data integrity, and other storage-specific functions.	200	802.1q Tagged
<b>Edge/External VLANs</b>	Used for connections to devices external to the Cloud infrastructure; these include externally visible services, such as load balancers and web servers. Use one or many of these networks, dependent on the need to segregate traffic among groups of servers.	300	802.1q Tagged



**Note:** The admin and BMC networks are expected to be in the same L2 network.

Each network defined in the system has the following parameters:

**Table 2: Network Parameters**

Name	Default	Description
<b>vlan</b>	Integer	The VLAN to use on the switch and interfaces for this network
<b>use_vlan</b>	true	A value of true indicates that the VLAN should be applied to the interface. A value of false assumes that the node will receive untagged traffic for this network.
<b>add_bridge</b>	false	Indicates if the network should have a bridge built on top of it. If a bridge is created, it will be named "br0".
<b>subnet</b>	IP Address	The subnet for this network
<b>netmask</b>	Netmask	The netmask for this network
<b>router</b>	IP Address	The default router for this network
<b>broadcast</b>	IP Address	The default broadcast address for this network
<b>ranges</b>	map	This contains a map of strings to start and stop values for the network. This allows for sub-ranges with the network for specific uses, e.g. dhcp, admin, bmc, hosts.



## IP Addressing

By default, IP addresses are assigned in the following fashion, using large subnets to support many machines on the production network. The table below shows the networks that are installed when the default network configuration is used. In each network, the first 10 IP addresses are reserved for switches, routers, and firewalls.

**Table 3: Default Network Addresses**

LAN	VLAN	Network	Subnet	Gateway	802.1q	Bridged
Storage	200	192.168.125.0	255.255.255.0	none	Yes	No
Public	300	192.168.122.0	255.255.255.0	192.168.122.1	Yes	No
Admin	100	192.168.124.0	255.255.255.0	192.168.124.1	No	No
BMC	100	192.168.124.0	255.255.255.0	192.168.124.1	No	No
BMC_Vlan	100	192.168.124.0	255.255.255.0	192.168.124.1	No	No

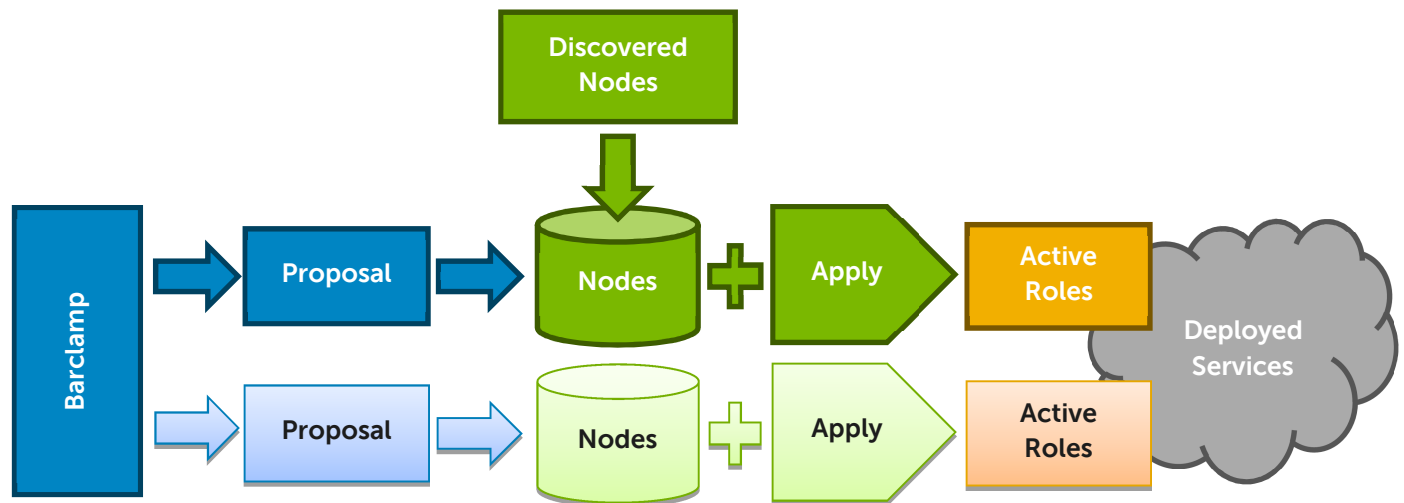


**Note:** Each network's ".1" address is reserved for the network gateway.

## Crowbar Operation

### Barclamp Life Cycle

Understanding the barclamp life cycle is essential to understanding the Crowbar user interface layout.



**Figure 1: Life Cycle of a Barclamp from Concept to Proposal and Deployment**

Crowbar progresses all deployments through a fixed lifecycle. It is important to understand this life cycle to use Crowbar.

**Figure 1** shows the life cycle of a barclamp within Crowbar. A barclamp defines the capability for a service but cannot be deployed. To deploy a barclamp, you must create a proposal. Once the proposal is created, you must select one or more nodes that will fulfill the roles provided by the barclamp. As discussed in the next sections, you may also edit the proposal's attributes as needed.

Applying the proposal tells Crowbar to deploy the proposal onto the nodes. Nodes return to the Ready state when deployment is completed. At this point, the proposal is considered active.

Once a proposal has become active, you may still edit it and reapply the changes. Crowbar will adjust the deployed proposal. Removing nodes will change the configuration for nodes remaining in the proposal;

however, it does *not* perform cleanup actions on the removed node. You may also “deactivate” an applied proposal. This will remove the nodes from the proposal but keeps the configuration data.

If you delete an inactive proposal then your configuration changes will be lost.



**Note:** Many barclamps require multiple nodes before they can be deployed. A proposal is not deployed until it has a sufficient number of nodes.

At the time a proposal is applied, Crowbar updates the “Run List” of the Crowbar managed node role in Chef.

The following table shows options for changes to proposals based on their current state.

**Table 4: Changes to proposals based on their current state**

State	Icon	Forwards	Backwards	Comment
No Proposal		Create	None	You must create proposal
User Input		Apply	Delete	
Active		Apply	Deactivate	You may reapply without deactivating a proposal.
Pending		None: Proposal Queued	Dequeue	Crowbar is building nodes during this phase.
In Progress		None: System Working	None: System Working	
Failed		Apply (after fixing issue)	Delete	In some cases, reapplying the proposal is sufficient to fix.

### Crowbar State Machine

When Crowbar brings systems up, the systems run through a series of states, each of which performs different actions on the systems. This is controlled by the Provisioner barclamp, and the status of any of the systems is indicated by the icon next to the system on the Dashboard page. You can see a description of the state a system is in by hovering over the status icon.

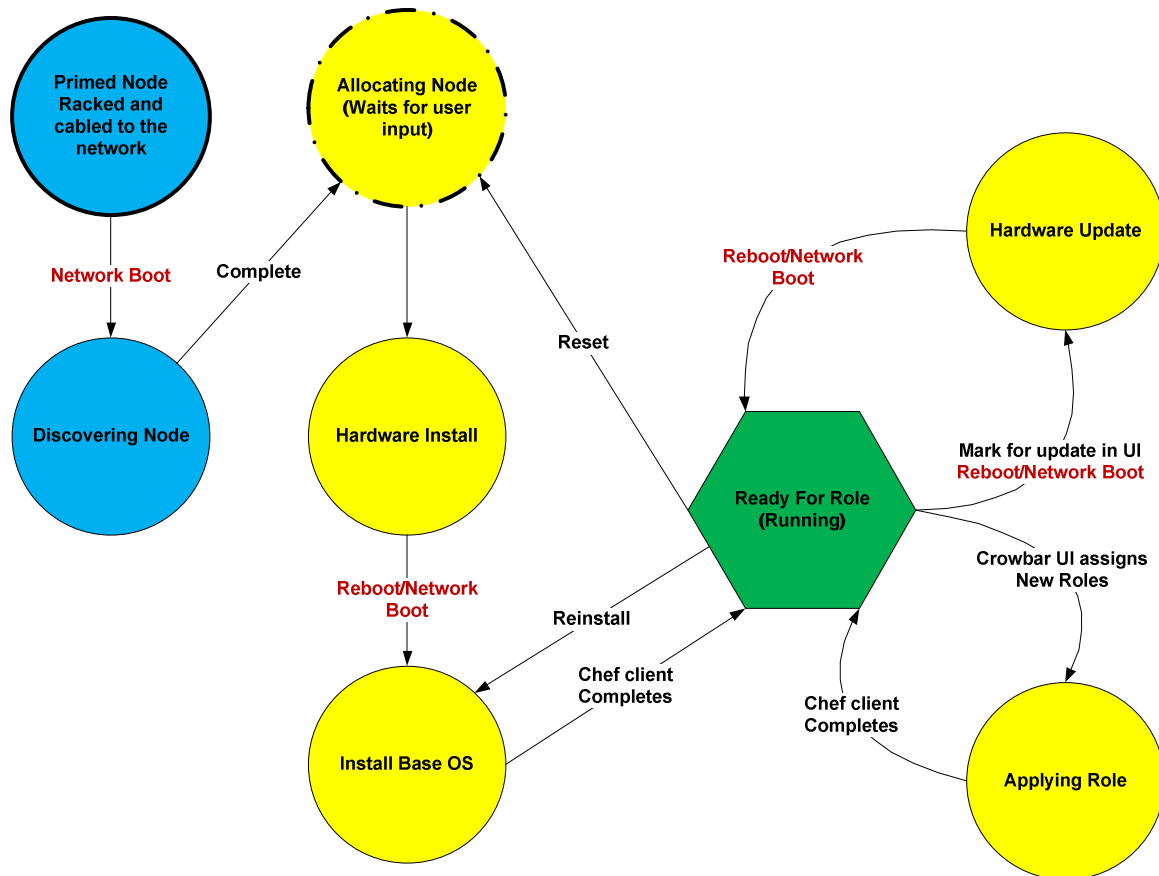


Figure 2: The States a System Progresses through to Fully Provision a System

### Discovering Node

During the Discovering Node state, the node network boots a CentOS LiveCD image to make sure that the node can run some basic components. Once this is complete, the LiveCD image registers with the Admin node to declare its state. The admin node adds the machine to the Chef database, and allocates a BMC and Admin network address for the node. After discovery, no persistent changes have been made to the node or its configuration. The node must be allocated before any permanent changes are made.

### Allocating Node

By default the nodes pause after the discovered state. The node awaits allocation either manually through the API (CLI or UI) or by being included in a proposal that is committed to the system. Since the node is not in the ready state, the proposal is queued until the newly allocated node is ready. During the allocation process, Crowbar examines what roles will be applied to the node and uses this information to influence the configuration of the BIOS and RAID systems.

### Hardware Install

The LiveCD image reboots forcing another network boot. During the Hardware Install state, the node network boots a CentOS LiveCD image to make sure that the BIOS/BMC/RAID controller are up-to-date. The CentOS LiveCD also updates the BIOS and RAID configurations. Once this is complete, the LiveCD image registers with the Admin node to declare its state. The Admin node updates the Chef database and resets the DHCP entry to installation state.

### Base OS Install

The LiveCD image reboots forcing another network boot. The node boots into a network installation of Ubuntu or Red Hat. The installation process reboots the node. Upon reboot and completing the one-time setup script, the system transitions to the Ready for Role state. The first time the Chef client is run, the node gets a base set of configuration (Nagios® client, Ganglia client, NTP client, DNS resolver). Reinstall will reinstall

the base OS on the node, as well as redeploying the roles applied to the node. This effectively uninstalls any custom software that was installed on it and removes all custom configuration that was done to it.

### Ready for Role

This is the state that a node spends most of its time in. The node runs the Chef client periodically to ensure that it is up-to-date. It waits for changes to its configuration. If the node reboots, then the node network boots the local boot image again and returns to this state.

### Hardware Update

To transition to this state, mark the node as needing hardware updates in the Crowbar UI and then reboot the node. The node network boots and does the same actions as the Hardware Install state, except that it makes no changes to the RAID configuration. Once this is complete, the LiveCD image registers with the Admin node to declare its state. The Admin node updates the Chef database and resets the DHCP entry to installation state. The return state is back to the Ready for Role state. This state allows for the update of the BIOS on a provisioned node without making any other changes to the server.

### Applying Role

This is a transient state that represents the running Chef client. It is the time that the node is applying a new configuration before returning to Ready for Role. The transition to this state happens periodically or can be forced by the Admin node.

## User Interface / Using Crowbar

Crowbar is delivered as a web application available on the Admin node using HTTP on port 3000. By default, you can access it using <http://192.168.124.10:3000> (see table below). The Crowbar dashboard includes links to the other tools installed, including Ganglia, Nagios®, and Chef.

Table 5: Service URLs

Service	URL	Credentials
SSH	<a href="ssh://root@192.168.124.10">root@192.168.124.10</a>	crowbar
Crowbar UI	<a href="http://192.168.124.10:3000/">http://192.168.124.10:3000/</a>	crowbar / crowbar
Nagios® UI	<a href="http://192.168.124.10/nagios3">http://192.168.124.10/nagios3</a>	nagiosadmin / password
Ganglia UI	<a href="http://192.168.124.10/ganglia">http://192.168.124.10/ganglia</a>	nagiosadmin / password
Chef UI	<a href="http://192.168.124.10:4040/">http://192.168.124.10:4040/</a>	admin / password



**Note:** Crowbar is supported on the following browsers: FireFox 3.6 and later, Internet Explorer 8, Internet Explorer 9, Google Chrome, and Safari 5. HTML5 compatibility and a minimum screen resolution of 1024x768 are recommended.

The IP address 192.168.124.10 is the default address. Replace it with the address assigned to the Admin node.

The Crowbar interface has two primary concepts: nodes and barclamps. All actions are focused on management of these two elements. Before discussing the UI, it's important to understand how they are used by Crowbar.

### Nodes

Nodes represent distinct servers in your environment. A server is a single operating system with multiple NICs and HDDs. Each server is identified uniquely by the MAC address of the NIC on the administrative network.



Crowbar nodes map directly to Chef nodes. In fact, all data used in Crowbar is stored entirely in Chef. Chef is the database for Crowbar. Changing a node's data in Chef changes it in Crowbar.

### Barclamps

- Barclamps represent modular capabilities that you can install onto none, some, or all of the nodes in your environment.
- Barclamps are activated by generating a proposal for that Barclamp. It is possible to generate multiple proposals for some barclamps, while others may only have a single instance.
- Once a proposal is reviewed, you must activate it before it becomes active in the system.



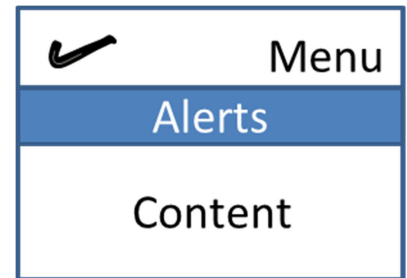
Barclamps are decomposed in Chef as multiple components: Crowbar data bag entries, cookbooks, recipes, and roles. Our objective is to allow the Chef components used by barclamps to operate in Chef even without Crowbar.

Barclamps have a specific life cycle that is discussed below in more detail as we explore the user interface. Information about using, creating, and extending barclamps is included in the Supplemental Material section.

## General Layout

The menu for Crowbar is displayed on the upper right side of the page. Clicking on one of the menu sections causes related content to display on the lower section of the screen.

Alerts or confirmation messages may be displayed between the menu and the page content. Most Crowbar screens automatically update state information so you should not have to refresh the page to get the most current information.



## Nodes (System Dashboard)

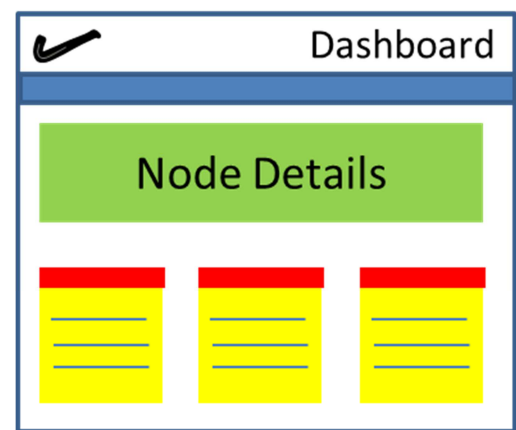
The Dashboard shows all the nodes in the system and lets you manipulate their power and configuration states.

### Node Alias and Description

Users can override the default MAC addressed-based names in the UI from the Node edit page or Bulk Edit pages. When set, the entire UI will display the Node's alias instead of its name. The system name and description are displayed when the mouse hovers over the node alias.

### Node Groups

The top of the group box (red in the illustration above) shows the Group Name and a pie chart of the nodes' status within each group. Nodes (yellow in the illustration) are shown based on the group order with their current deployment state shown by the status light.



Nodes are automatically arranged into a series of default groups where each default group represents a switch. The default groups are named using the MAC address of the switch associated with them. Nodes are shown within each group ordered by port order (lowest on top). Switches and ports are discovered automatically during the configuration process. An italic group name indicates that the group is a default group, and a non-italicized group name indicates that the group is a user created group.









**Note:** If you use a consistent pattern for connecting nodes to switches then the Crowbar display matches your nodes' physical location.

You can override the default group behavior from the Dashboard. To create a new group, enter the desired group name and click "Add Group," then drag the desired node into the empty group that appears on the top left of the Dashboard. To rearrange nodes between groups, drag the node into the name of the desired group. To remove a node from a user-defined group setting, you must drag the node into the "[here]" portion of the hint text area.



**Note:** Nodes may be moved between user-defined groups, and between a user-defined group and a default group, but cannot be moved between default groups.

Table 6: Deployment States

Status	Icon	Comment	User Action
<b>Ready</b>		Requested services provisioned.	Configure as needed.
<b>Waiting</b>	 blinking	Waiting for user input.	Node waiting to be allocated. See "Bulk Edit" or include node in a proposal.
<b>Pending</b>	 solid	Hardware and operating system installation.	<i>None: Crowbar is provisioning the node.</i>
<b>In Process</b>	 spinning	Crowbar and Chef actively deploying.	<i>None: Crowbar is provisioning the node.</i>
<b>Failed</b>	 blinking	Failure detected operating on node.	Correct issue.
<b>Unknown</b>		In between states or not reporting for 20 minutes (likely powered off).	Restart server if desired.

The Admin node is the node that runs Crowbar, Chef, and other core services. It exists in the system before the user interface is available.

### Node Details

Clicking on a node's name displays details about the selected node in the details panel (green in the illustration). The details panel displays important information about the node, including its Full Qualified Domain Name (FQDN), uptime, switch connectivity hardware profile, and a detailed list of all active network connections.



Node detail shows only a subset of the total details that Chef tracks for each node. To see the complete list, examine the Run List and Attributes for each node in Chef.

The Links list is barclamp-specific and expands depending on which barclamps are applied to the selected node. Links open a new window to view additional information about the node. The Barclamps and Roles lists indicate what capabilities have been assigned to the node.



**Note:** When a role or barclamp is selected in the details panel, the nodes that share the same barclamp or role are highlighted in the group panel. This helps quickly identify groups of nodes that are similarly configured.

The buttons on the top of the details panel (Identify, Power On, Shutdown, and Reboot) use the node's IPMI interface to change the node's physical state. These states cause the node status to be unknown. These buttons are available only if the system is able to successfully configure the BMC on the target system.

Table 7: Details Panel Buttons

Button	Action	Useful When
<b>Identify</b>	Causes the identify light to blink for 15 seconds.	Trying to identify a node within a rack.
<b>Power On</b>	Sends a power-on signal to the BMC of the selected system.	Remotely powering on a system.
<b>Shutdown</b>	Sends a power-off signal to the BMC of the selected system.	Remotely powering off a system.
<b>Reboot</b>	Sends a power-cycle signal to the BMC of the selected system.	Remotely power cycling a system, which has stopped responding.

The buttons on the bottom of the details panel (Delete, Reset, Reinstall, and Hardware Update) reset the node's deployment state. These functions are very useful during lab configurations when the system is being continuously reconfigured. The buttons take the following actions:

**Table 8: Buttons on Bottom of Details Panel**

Button	Action	Config Lost?	Reboot?	Useful When
<b>Delete</b>	Completely removes all records of the node from the Crowbar/Chef database. If a node is deleted, it is rediscovered when it reboots.	Yes	No	Removing nodes.
<b>Reset</b>	Removes all the roles assigned to the node and reimage it back to an unallocated node.	Yes	Yes	Reallocate the node for a new purpose.
<b>Reinstall</b>	Reimages the node and then reapplies the current deployment profile to the node. This rebuilds the server, returning it to a freshly deployed state.	No	Yes	Tuning the Chef recipes or configuration details.
<b>Hardware Update</b>	Keeps the current configuration, forces the node to reboot and apply BIOS and RAID updates.	No	Yes	To apply BIOS or RAID updates.

Using the Edit link (after the node name in the top left) lets you make per-node decisions about how the node is deployed.

To allocate a node, manually allocate the node from the edit page or include that node in an applied barclamp proposal.



**Note:** No changes are made to a node when it is discovered. This provides a safeguard against data loss if a system is accidentally PXE booted on the admin network. If this happens, delete the system from Crowbar, reboot it, and set the boot order in the BIOS to boot from the local disk first. Proceeding beyond the discovered state for a system by allocating it will immediately wipe all data on the system.

### Bulk Edit

The Bulk Edit screen allows you to quickly update the alias, description, RAID, BIOS, and allocation state for all the nodes in the system.



**Note:** Allocation is a destructive process. Allocating a node will immediately wipe all data on the node. You can return a node to the unallocated state using the Reset button as described above, though this will not restore any data.

The RAID and BIOS selections highlight the nodes' current value using [brackets]. The choices offered for BIOS and RAID selectors are determined by the BIOS and RAID barclamps. They can be expanded after installation.


If no change is made then the node will not be updated.

### Network

The Network menu exposes Crowbar network data and configuration options. Unlike the Nodes menu, the purpose of this menu is to visualize and manage the system network topology. Any barclamp is able to extend the Network menu.

Switches


The Switches menu item attempts to recreate the switch/port layout of your data center using information collected from the nodes. While the node view shows only the information about the node's administrative network, the Switch view shows what node interfaces are plugged in to which switch ports.

 **Note:** If the 29-48 unit ports are not used or are inactive, then the switch view will show only a 24-port switch instead of a 48-port switch.

The Switch view models either 24- or 48-port switches with the Switch MAC address and switch unit number at the top of the view. Ports are shown in two rows numbered from top-left to bottom-right with port numbers and include a status light that matches the nodes' current status. Inactive ports use the grey (●) status indicator and darker background color.

Hovering over any port causes two actions:

- 1. A tag showing the node's interface identifier (e.g.: eth1), alias, and description will be shown.
- 2. All other ports also connected to that node will be highlighted using a red cell border. This highlighting helps quickly identify a node's connection.

 **Note:** If you want to just see the connections for a single node, you can add "?node=[node name]" to the URL. This call is linked on the switch name from the node view.

The Switch view uses a compact layout to render switches. To expand the view to show additional details without hovering over the node, click on the switch name. This will reveal the expanded switch view. If the switch is 48 ports, then the expanded view will be split into 1..24 and 25..48 port blocks.

The expanded view shows the port number, status, network interface, and node alias inside the port cell. The same node highlight hovering effect is applied on the expanded view. Screen rendering may not be regular depending on the length of the node alias.

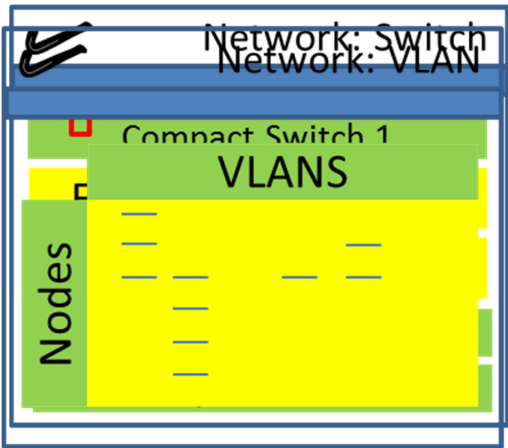
You cannot edit or change the switch configuration from this view.

VLANs

The VLANs menu item builds a matrix mapping between the VLANs tracked by the network barclamp and the nodes. This matrix represents both active and inactive connections between the nodes and the VLANs.

The VLAN view is a simple table with VLANs represented by columns and nodes as rows. If a node is active on a VLAN then "Active VLAN" is shown at the intersection. If a node is connected but inactive then "not active" or "N/A" is shown. If a node and VLAN are not mapped then the cell is empty. The IP address that the node has on a given VLAN can be seen by hovering the mouse over the appropriate cell. The VLAN ID can be seen by hovering the mouse over the network name.

You cannot edit or change the switch configuration from this view.






## Barclamps

The Barclamps page lets you create, edit, review, and deploy proposals for barclamps. These activities are the way that Crowbar decides which nodes to deploy and how to configure them.

### Barclamp List

The All Barclamps page shows a list of all available barclamps (see “**Error! Reference source not found.**” table). The barclamps are represented as the blue lines in the figure to the right. Expanding a barclamp (by clicking ) displays the associated proposals for the selected barclamp (red box in the figure). You jump directly to the relevant proposal by clicking on its name under its barclamp.

A barclamp will show the status of the proposals that are attached to it using a status light (see table below). If multiple proposals are assigned, then multiple lights are displayed. If there are no proposals, a diamond is displayed. Hovering over the light will show you the name and status of the matching proposal. The proposals status updates automatically without a refresh.

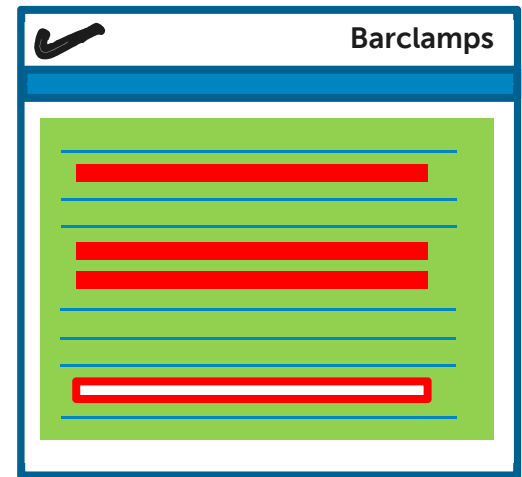








Table 9: Proposal Status

Status	Icon	Next Step	Comment	User Interaction
No Proposal		Create	No proposal has assigned to the barclamp.	Create a proposal for the barclamp if desired.
User Input		Delete or Apply	Proposal waiting for user input and activation.	Edit the proposal. Apply proposal after review.
Active		Deactivate or Apply	Proposal has been deployed.	Ready for use.
Pending	 blinking	Wait or Dequeue	Queued for deployment.	Crowbar is building nodes during this phase.
In Progress	 spinning	None	Proposal is being configured.	None. Crowbar is working.
Failed	 blinking	Apply	Proposal failed during Apply	Correct error and reapply proposal.

From the Barclamp list, you may take actions on the proposals based on their state, as shown in the table above. Please review the **Error! Reference source not found.** section for more information about the different proposal states.

All core barclamps automatically create proposals and do not allow users to create additional proposals. Some barclamps allow the creation of multiple proposals. These additional proposals can be used to manage deployment configurations or control which parts of the system are active in which barclamps.

To create a new proposal, expand the barclamp row to expose the create form for the new proposal. You must supply a name for the proposal but descriptions are optional. Clicking “create” will take you to the proposal editor (*details below*). The create form will not be shown for barclamps that only allow a single proposal.



**Note:** Naming for proposals is limited to lowercase letters and numbers only (spaces are not allowed). Capitalization is automatic for formatting only.



This limitation is necessary because activated proposals are created as roles in Chef and follow a prescribed naming convention.

Crowbar stores barclamps in Chef under the Crowbar data bag using `bc-template-[barclamp]` as the naming pattern. When a proposal is created, the instance copy is also stored in the Crowbar data bag. Only active proposals have roles created for them.

### Proposal View/Edit

Selecting a proposal from on the list navigates to the proposal details page.

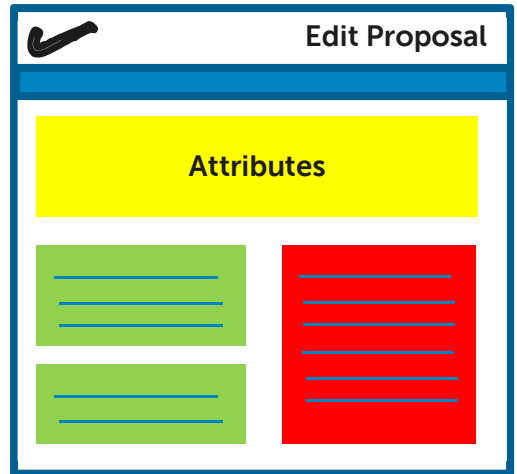


**Note:** If a proposal is active, you will be initially taken to a read-only view of the proposal. It is acceptable to edit an active proposal and re-apply. To access the proposal editor from the read-only view, click the edit button.

Clicking the Edit button opens the Edit Proposal page. All proposals have two primary edit areas: Attributes (yellow in figure) and Node Deployment. Attributes are configurable data that is used by the Chef recipes. Node Deployment shows the Chef roles and nodes assigned to those roles.

Since each barclamp has unique attributes and roles you should consult the documentation for each barclamp if you plan to change its defaults.

Each barclamp may provide a custom editor for its attributes and node deployment information. The typical custom editor lets you set attribute values using a form, and drag and drop nodes from the available list (left column) into the roles associated with the barclamp (right, red on figure). Each barclamp may have specific logic that requires minimums or maximums for node assignments.



**Note:** While most barclamps coordinate with Chef to perform node deployments, Crowbar includes some special-function barclamps that can be used to change how Crowbar operates.

If the barclamp does not have a custom editor or your browser does not support the editor, Crowbar automatically uses a raw JSON editor. You can also use this view if you want to see the entire configuration details. Selecting the Raw view option on the right side of the Attributes or Deployment panel opens the JSON editor for that section. This option lets you directly edit the JSON configuration details for the proposal. This option is typically used when developing new barclamps or for advanced users only.

When you have finished editing the proposal, you may save or apply it. "Save" retains your configuration settings. "Apply" saves and then applies your proposal so that Crowbar begins deploying the barclamp on the selected nodes. Deleting a proposal removes it from the system and you lose your configuration.



**Note:** If you attempt to apply a proposal to nodes that have not yet been allocated, then Crowbar will queue the proposal while it automatically allocates the nodes. Following allocation, the proposal will be dequeued and applied to the nodes.



When you apply a proposal, Crowbar creates Chef roles, and then puts them into the run list of the selected nodes.

Crowbar uses a naming pattern for Roles that lets you quickly figure out which barclamp and proposal is being applied to a node's run list in Chef. The instantiated barclamp naming pattern is [barclamp]-config-[proposal]. Barclamps then use additional roles to control node proposal membership (aka the Run List).

### Utilities

The Utilities menu is used by various barclamps to provide general-purpose tools that do not map within the normal node, network, or barclamp context. Any barclamp is able to extend the Utilities menu.



**Note:** Utilities is a native part of the Crowbar barclamp; however, several menu items are populated by barclamps.

### Exported Items

The Exported Items menu shows a list of files that have been prepared for export. There will be no items available until an export has been requested (see below).

The exported items are shown grouped by category. Clicking on an item will download it through the browser download function. Clicking the delete icon (🗑️) will delete the file.

The export items page refreshes automatically.

### Chef Export

The Chef Export menu item creates a time-stamped archive snapshot of the underlying Crowbar database. This snapshot can be used to replicate the installation for troubleshooting or backup.

This action generally takes only a few seconds to complete. Crowbar does this work in the background so that you can continue to use the UI while the export is operating.

### Log Export

The Log Export menu item compiles all the system logs into a time-stamped single compressed archive. Typically, the first step in any troubleshooting activity will be to capture and send the logging information.

This action may take several minutes to complete and runs in the background. The Exported Items screen will automatically update when the export is complete.

## Core Crowbar Barclamps

There are a set of core barclamps that are a part of every Crowbar installation. They are listed in the table below.

**Table 10: Barclamps**

Barclamp	Function / Comments
<b>Crowbar</b>	The roles and recipes to set up the barclamp framework. References other barclamps. Modify the default proposal to change the Usernames and passwords for access to the Crowbar UI.
<b>Deployer</b>	Initial classification system for the Crowbar environment (aka the state machine).
<b>Provisioner</b>	The roles and recipes to set up the provisioning server and a base environment for all nodes.
<b>Network</b>	Instantiates network interfaces on the Crowbar managed systems. Also manages the address pool.
<b>RAID</b>	Sets up LSI RAID controllers in a variety of configurations. If missing, the RAID controllers can be set up manually.
<b>BIOS</b>	Configures BIOS options for Dell™ PowerEdge™ servers. If missing, the BIOS options can be configured manually.
<b>IPMI</b>	Allows management of the IP Management Interface (IPMI) on servers when the BMC network is enabled.
<b>NTP</b>	Common NTP service for the cluster (required for secure access). An NTP server can be specified.
<b>DNS</b>	Manages the DNS subsystem for the cluster.
<b>Logging</b>	Centralized logging system based on syslog.
<b>Nagios</b>	System monitoring service for the cluster that can be used by other barclamps.
<b>Ganglia</b>	Performance monitoring service for the cluster that can be used by other barclamps.
<b>Test</b>	Provides a shell for writing tests against.

Details about these barclamps are provided below.

## Crowbar Barclamp

The Crowbar barclamp provides the roles and recipes to set up the barclamp framework.

The Crowbar barclamp initializes the system, creates initial proposals of other barclamps defined in its configuration, and creates the users to access the Crowbar API and UI. By default, the system creates Network, Ganglia, Nagios®, NTP, DNS, Provisioner, Deployer, IPMI, RAID, and BIOS proposals based upon the default configuration of their barclamps. The initialization function of the Crowbar barclamp works exactly like other barclamps. A proposal is created and can be committed during installation.

All barclamps' transition functions can be called directly, but the Crowbar barclamp calls these in an order specified in its configuration, which is determined by their priority. The default unspecified priority is 100. The special cases are the Provisioner, which is last, and the Deployer and Network, which are first and second, respectively.

**Table 11: Crowbar Barclamp Parameters**

Name	Default	Description
<b>instances</b>	The starting barclamps using their default configurations.	A map of barclamp names that reference a list of JSON files (default is special to mean to take the defaults) that represent barclamp instances to create.
<b>users</b>	A map of users—containing Crowbar.	This map defines the users allowed to access Crowbar's UI and REST API.

The users map contains a map. The key is the user name and the rest of the required fields are as follows:

**Table 12: User Name Key**

Name	Description
<b>password</b>	Clear text password of the user.
<b>description</b>	A description of the user.

## Deployer Barclamp

The Deployer provides an initial classification system for the Crowbar environment. As nodes are discovered, the Deployer makes sure that discovery tools are run on the node by making sure that the Deployer-client role is assigned to the node. The results of that discovery are classified, and the node's attributes are updated to reflect its potential usage. The Deployer also builds a map of valid and usable disks.

The Deployer gives the primary name to the node at the discovered state. The names default to the letter "d" and the MAC address (with dashes instead of colons). The Deployer also allocates the admin and BMC addresses from the network barclamp.

In addition, the Deployer defines and provides the node's configuration for RAID and BIOS. These values are assigned part of the hardware-installing state transition. The Deployer uses a list of role name patterns that define what the RAID and BIOS configurations should be. These are applied as values in the node attributes under crowbar -> hardware. bios\_set can be either Virtualization or Storage. RAID set can be either JBODOnly or SingleRaid10.

The Deployer is also responsible for manipulating the run list during the hardware-installing and update (or hardware-updating) states. The run list should only include BIOS, RAID, and IPMI operations.

The Deployer also controls the allocate flag on the node. The allocate flag is used to pause the node after discovery. The node waits for it to be allocated to continue. The Deployer has a configuration option to indicate if the allocate flag should be set to false (and cause a pause) or just allocate all nodes.

Table 13: Deployer Barclamp Parameters

Name	Default	Description
<b>bios_map</b>	A list of default settings for bios and raid for swift and nova.	The map defines a list of patterns that would apply a configuration setting for BIOS and RAID.
<b>use_allocate</b>	true	A Boolean value true indicates that a pause should be injected after the discovered state to allow the admin to accept and allocate the node.

Table 14: BIOS Map Entry Keys

Name	Description
<b>pattern</b>	Regular expression applied to the role names on the node.
<b>bios_set</b>	The BIOS set of parameters to apply. Values are: Virtualization or Storage.
<b>raid_set</b>	The RAID set of parameters to apply. Values are: JBODOnly or SingeRaid10.

## Provisioner Barclamp

The Provisioner provides the roles and recipes to set up the provisioning server and a base environment for all provisioned nodes. The Provisioner also provides the transition entry point for nodes that need to have DHCP transitions done. The Provisioner assumes that IP addressing is handled outside of this barclamp.

Table 15: Provisioner Barclamp Parameters

Name	Default	Description
<b>default_user</b>	crowbar	User to create for external login.
<b>default_password</b>	unset	Clear text password to use for external login.
<b>default_password_hash</b>	Hash of crowbar	MD5 hash of password to use for external login. printf 'password'   mkpasswd -s -m md5 will generate the hash.
<b>web_port</b>	8091	The default Web port that the repository web server uses.
<b>use_local_security</b>	true	This defaults the security updates path in the install to use the admin node instead of the Internet.
<b>dhcp</b>	map	This is a map that contains the DHCP parameters (lease-time and state machine).
<b>lease-time</b>	60	The number of seconds a DHCP lease is valid for the system.
<b>state_machine</b>	map	This is the state machine that the DHCP server uses in this instance of the barclamp.



Note: While neither is required, one of default\_password or default\_password\_hash is required.

## Network Barclamp

The Network barclamp provides two functions for the system. The first is a common role to instantiate network interfaces on the Crowbar managed systems. The other function is address pool management.

The network interfaces are controlled by the network role that is applied by the barclamp as a node transition to "installed". Based upon assigned addresses, the network recipe creates the appropriate single, dual, or team mode interface sets.

The network assignment function is handled by the creation of an API extension of the base barclamp. The barclamp adds the `allocate_ip` REST API call. This function allocates an IP address from a requested network and updates the node's attributes and the network's data bag. The available networks (and their parameters) are defined in the configuration for the barclamp.

Modification of the following parameters should only be done when installing Crowbar, prior to running the `./install systemname.yourdomain.com` command. See the *Crowbar OpenStack Deployment Guide* for more information.

**Table 16: Network Configuration Options**

Name	Default	Description
<b>mode</b>	single	A string value of single, dual, or team. This specifies the default network interface construction model.
<b>teaming</b>	map	A map of values specific to teaming.
<b>networks</b>	map	A map of networks that this barclamp should manage.

**Table 17: Teaming Sub-Parameters**

Name	Default	Description
<b>mode</b>	6	The default teaming algorithm to use for the bonding driver in Linux.

**Table 18: Default Networks**

Name	Usage	Notes
<b>admin</b>	Private network for node-to-node communication	A router, if wanted, is external to the system. This network must be owned by the Crowbar system to run DHCP.
<b>bmc</b>	Private network for BMC communication	This can be the same as the admin network by using the ranges to limit what IP goes where. A router, if wanted, is external to the system.
<b>bmc_vlan</b>	Private network for admin nodes on the BMC network	This must be the same as the BMC network and have the same VLAN. This is used to generate a VLAN-tagged interface on the admin nodes that can access the BMC LAN.
<b>storage</b>	Private network for storage traffic	A router, if wanted, is external to the system.
<b>public</b>	Public network for Crowbar and other components	A router, if wanted, is external to the system.

Table 19: Network Parameters

Name	Default	Description
<b>vlan</b>	Integer	The VLAN to use on the switch and interfaces for this network.
<b>use_vlan</b>	true	A value of true indicates that the VLAN should apply to the interface. A value of false assumes that the node receives untagged traffic for this network.
<b>add_bridge</b>	false	Indicates if the network should have a bridge built on top of it. The bridge will be br. This is mostly for Nova compute.
<b>subnet</b>	IP Address	The subnet for this network.
<b>netmask</b>	Netmask	The netmask for this network.
<b>router</b>	IP Address	The default router for this network.
<b>broadcast</b>	IP Address	The default broadcast address for this network.
<b>ranges</b>	map	This contains a map of strings to start and stop values for the network. This allows allocating sub-ranges with the network for specific uses. For example: DHCP, admin, BMC, hosts.

Table 20: Range Map String Key

Name	Type	Description
<b>start</b>	IP Address	First address in the range, inclusive.
<b>end</b>	IP Address	Last address in the range, inclusive.



Settings in the Network barclamp should not be changed after the installation of the Admin Node.

## RAID Barclamp

RAID is an acronym for “Redundant Array of Independent Disks.” This means that a RAID controller makes (or can make) multiple disks look like one big/smart/safe disk.

Different RAID controllers have different capabilities. In general, a RAID controller can support one or more RAID volumes of various types. Any disks not included in a RAID volume are directly exposed to the operating system. Disks that are directly exposed like this are known as Just a Bunch Of Disks (JBOD).

Crowbar supports specific RAID controllers. See the *Dell OpenStack Reference Architecture Guide* for more information.

The Crowbar code makes sure that the configuration on the RAID controller matches that specified within the Crowbar configuration.

The parts that determine the configuration for a node are:

- A set of Chef data bags, which contain the RAID configuration (in data bags/crowbar-data). The defaults are SingleRaid10 and JBOD Only.
- An attribute on the Chef node of the machine that identifies which data bag (described above) should be applied to this node (the attribute is node[:crowbar][:hardware][:raid\_set], and it should include the name of a data bag).
- Crowbar (the Deployer barclamp) sets the above property when a node is allocated to a proposal.

When invoked, the recipe uses a Chef Lightweight Resources and Providers (LWRP) to inspect the current configuration on the system and compare it to the desired state. If the two diverge, the code will:

- Delete any RAID sets that are not required any more
- Allocate available disks among the desired RAID sets, according to the order attribute.
- Issue commands to apply the configuration.

## BIOS Barclamp

The BIOS barclamp provides the following specific control features for certain Dell servers:

- Uploading a known BIOS firmware into flash. Setting parameters to a defined set, based on the machine's role.
- IPMI barclamp.
- LAN parameters (IP address, netmask, gateway) and User credentials.
- The IPMI barclamp has a couple of list parameters.

## IPMI Barclamp

The IPMI barclamp configures IPMI access on platforms that support it. It selects an IP address and assigns it to the BMC. Prior to doing this, it checks the IP address that is currently assigned to the BMC. If the address falls within the range of addresses that are configured for the BMC network in the network JSON, then it will use that IP address instead of assigning a new one.

The IPMI barclamp allows the user to control the following parameters:

**Table 21: IPMI Barclamp Parameters**

Name	Default	Description
<b>bmc_enable</b>	true	Controls if the barclamp attempts to work on the BMC.
<b>bmc_password</b>	cr0wBar!	The password that will be configured on the BMC.
<b>bmc_user</b>	root	The username that will be configured on the BMC.
<b>debug</b>	true	Turns on more verbose output.

## NTP Barclamp

The NTP barclamp provides a common NTP service for the cluster. You can specify an NTP server or servers and all other nodes are clients of them. By default, the time on all nodes are synced to the hardware clock of the Crowbar admin node. The time zone of all Crowbar managed nodes in the cluster is set to UTC.

**Table 22: NTP Barclamp Parameters**

Name	Default	Description
<b>external_servers</b>	empty list	A list of IP addresses or hostnames that should be used as external NTP servers. Hostname can be used if the DNS barclamp is configured to have access to an external resolver.
<b>admin_ip_eval</b>	"Chef::Recipe::Barclamp::Inventory.get_network_by_type(node, \"admin\").address"	The Ruby eval expression that returns the admin IP address of a node.



Note: If you are setting up an external server it can take up to 5 minutes for the nodes to sync with the server. Systems should not be rebooted during this process. If they are rebooted, then they will pause during bootup for time synchronization.



## Logging Barclamp

The Logging barclamp provides a centralized logging system based on syslog. The barclamp enables a centralized log server that can then forward information to external syslog servers. The Crowbar installation process sends logs to the admin node by default, but the configuration from the logging barclamp can override this initial configuration.

**Table 23: Logging Barclamp Configuration Parameters**

Name	Default	Description
<b>external_servers</b>	Empty list	A list of IP addresses for the logging-server to which to forward logs.

## Nagios® Barclamp

The Nagios® barclamp provides a common Nagios® service for the cluster. A Nagios® server or servers can be specified and all other nodes are clients of them. The barclamp attempts to direct all traffic over the admin network.

**Table 24: Nagios® Barclamp Parameters**

Name	Default	Description
<b>admin_interface_eval</b>	<code>Chef::Recipe::Barclamp::Inventory.get_network_by_type(node, \"admin\").interface</code>	The Ruby eval expression that returns the admin interface of the node.
<b>admin_ip_eval</b>	<code>"Chef::Recipe::Barclamp::Inventory.get_network_by_type(node, \"admin\").address"</code>	The Ruby eval expression that returns the admin IP address of the node.

## Ganglia Barclamp

The Ganglia barclamp provides a common Ganglia service for the cluster. Ganglia server or servers can be specified and all other nodes are clients of them.

**Table 25: Ganglia Barclamp Parameters.**

Name	Default	Description
<b>interface_eval</b>	<code>Chef::Recipe::Barclamp::Inventory.get_network_by_type(node, \"admin\").interface</code>	The Ruby evaluation string that gets the interface of the admin interface.

## Test Barclamp

The Test barclamp provides a shell for writing tests against. It allows for failures to be injected and other barclamps can be validated against it.

**Table 26: Test Barclamp Parameters.**

Name	Description
<b>barclamps</b>	A list of supported barclamps that are used as the return value for the barclamp list API call
<b>instances</b>	A map of barclamp names that reference a list of JSON files (default is special to mean to take the defaults) that represent starting barclamp instances to create

This barclamp is currently used only to perform a quick smoke test of Crowbar following Crowbar installation, and as a result should never be deployed.

## Supplemental Material

---

### System Verification

As a final step, it is important to verify that your deployment has succeeded. Crowbar provides limited feedback to confirm that the Chef recipes were successfully deployed.

You should consult the *getting started guide* and barclamps specific to your system for details on verification of deployment.

### Managing Barclamps

This section briefly describes barclamps, and how to import barclamps.



**Note:** For information about creating barclamps, please visit <https://github.com/dellcloudedge/crowbar/wiki/Barclamp:-create-&-install-steps>

#### Introduction

A barclamp is a deployment module that is imported from its own code repository into the Crowbar framework. A barclamp cannot operate without Crowbar, but you do not have to create a unique build of Crowbar in order to create a barclamp.



**Note:** You must install Crowbar before importing barclamps.

#### Importing a Barclamp

Once you have created a barclamp, you can import the barclamp into Crowbar and Chef. Assuming that you already created the foo barclamp in /barclamps (see Creating a Barclamp), proceed as follows:

1. From the Crowbar server, become the super admin: `sudo -i`
2. Run the barclamp install script: `/opt/dell/bin/barclamp_install /barclamps/foo.tar.gz`
3. `"/barclamps/foo.tar.gz"` is the file name of your barclamp. It could be anything.
4. The core barclamps are in `/opt/dell/barclamps`.

Your barclamp should now appear in the Crowbar UI. You can also see it in Chef under the Crowbar data bag.

While barclamps are generally safe to install multiple times, you can uninstall a barclamp using `"barclamp_uninstall.rb /path/to/barclamp"`.

#### More Information

Information on how to develop your own barclamp is available in the *Developers Guide*. If you are interested in creating, extending or contributing barclamps, please use one of the following contact methods.

## Support

### Crowbar Support

To obtain support for Crowbar:

- See the Crowbar wiki on GitHub: <https://github.com/dellcloudedge/crowbar/wiki>
- Gather log information.
- Email the Crowbar listserv—join at: <https://lists.us.dell.com/mailman/listinfo/crowbar>

To help facilitate troubleshooting of the environment, a utility to gather logs has been provided. Use the Log Export function in the user interface as detailed above, or browse to [http://<Admin\\_ip>:3000/support/logs](http://<Admin_ip>:3000/support/logs). This creates a tar archive of the relevant logs and asks the user for a location to save the resulting archive.



**Note:** Depending on the size of the logs to be gathered, this utility may take a while to run.

### To Learn More

For more information on the Crowbar, visit:

[www.dell.com/crowbar](http://www.dell.com/crowbar)

©2012 Dell Inc. All rights reserved. Trademarks and trade names may be used in this document to refer to either the entities claiming the marks and names or their products. Specifications are correct at date of publication but are subject to availability or change without notice at any time. Dell and its affiliates cannot be responsible for errors or omissions in typography or photography. Dell's Terms and Conditions of Sales and Service apply and are available on request. Dell service offerings do not affect consumer's statutory rights.

Dell, the DELL logo, and the DELL badge, PowerConnect, and PowerVault are trademarks of Dell Inc.