



# **Red Hat Enterprise MRG 2 Grid Installation Guide**

---

Installation information for the Grid component of Red Hat Enterprise MRG

Lana Brindley

Alison Young

David Ryan





# Red Hat Enterprise MRG 2 Grid Installation Guide

---

## Installation information for the Grid component of Red Hat Enterprise MRG

Lana Brindley  
Red Hat Engineering Content Services  
lbrindle@redhat.com

Alison Young  
Red Hat Engineering Content Services  
alyoung@redhat.com

David Ryan  
Red Hat Engineering Content Services  
dryan@redhat.com

**Legal Notice**

Copyright 2012 Red Hat, Inc. The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at [http://creativecommons.org/licenses/by-sa/3.0/](#). In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version. Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law. Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, MetaMatrix, Fedora, the Infinity Logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries. Linux is the registered trademark of Linus Torvalds in the United States and other countries. Java is a registered trademark of Oracle and/or its affiliates. XFS is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries. MySQL is a registered trademark of MySQL AB in the United States, the European Union and other countries. All other trademarks are the property of their respective owners. 1801 Varsity Drive Raleigh, NC 27606-2072 USA Phone: +1 919 754 3700 Phone: 888 733 4281 Fax: +1 919 754 3701

**Keywords****Abstract**

This book explains downloading and installation of the MRG Grid component of the Red Hat Enterprise MRG distributed computing platform. For detailed information on using MRG Grid, see the MRG Grid User Guide.

## Table of Contents

<b>Preface</b> .....	<b>4</b>
1. Red Hat Enterprise MRG	4
2. MRG Grid	4
3. Document Conventions	4
3.1. Typographic Conventions	4
3.2. Pull-quote Conventions	6
3.3. Notes and Warnings	6
4. Getting Help and Giving Feedback	7
4.1. Do You Need Help?	7
4.2. We Need Feedback!	7
<b>Chapter 1. Overview</b> .....	<b>8</b>
<b>Chapter 2. MRG Grid Benefits and Features</b> .....	<b>9</b>
<b>Chapter 3. MRG Grid Scale Requirements</b> .....	<b>12</b>
3.1. Small	12
3.2. Medium	12
3.3. Large	12
<b>Chapter 4. Installing MRG Grid</b> .....	<b>13</b>
4.1. Installing MRG Grid on Red Hat Enterprise Linux	13
4.2. Installing MRG Grid on Microsoft Windows	13
4.3. Upgrading MRG Grid Peacefully	18
4.3.1. Upgrading Execute Nodes	18
4.3.2. Upgrading Scheduler Nodes	18
<b>Chapter 5. Getting Started with MRG Grid</b> .....	<b>19</b>
<b>Chapter 6. More Information</b> .....	<b>20</b>
6.1. Reporting Bugs	20
6.2. Further Reading	20
<b>Revision History</b> .....	<b>21</b>



## Preface

### 1. Red Hat Enterprise MRG

This book contains basic overview and installation information for the MRG Grid component of Red Hat Enterprise MRG. Red Hat Enterprise MRG is a high performance distributed computing platform consisting of three components:

1. Messaging — Cross platform, high performance, reliable messaging using the Advanced Message Queuing Protocol (AMQP) standard.
2. Realtime — Consistent low-latency and predictable response times for applications that require microsecond latency.
3. Grid — Distributed High Throughput Computing (HTC) and High Performance Computing (HPC).

All three components of Red Hat Enterprise MRG are designed to be used as part of the platform, but can also be used separately.

### 2. MRG Grid

Grid computing allows organizations to fully utilize their computing resources to complete high performance tasks. By monitoring all resources - rack-mounted clusters and general workstations - for availability, any spare computing power can be redirected towards other, more intensive tasks until it is explicitly required elsewhere. This allows a standard networked systems to operate in ways similar to supercomputers.

MRG Grid provides high throughput and high performance computing and enables enterprises to achieve higher peak computing capacity as well as improved infrastructure utilization by leveraging existing technology to build high performance grids. MRG Grid provides a job-queuing mechanism, a scheduling policy, a priority scheme, resource monitoring, and resource management. Users submit their jobs to MRG Grid, where the jobs are placed into a queue. MRG Grid then chooses when and where to run the jobs based upon a policy, carefully monitors their progress, and ultimately informs the user upon completion.

MRG Grid is based on the [HTCondor Project](#) developed at the [University of Wisconsin-Madison](#). HTCondor offers a comprehensive library of freely available documentation in its [Manual](#).

### 3. Document Conventions

This manual uses several conventions to highlight certain words and phrases and draw attention to specific pieces of information.

In PDF and paper editions, this manual uses typefaces drawn from the [Liberation Fonts](#) set. The Liberation Fonts set is also used in HTML editions if the set is installed on your system. If not, alternative but equivalent typefaces are displayed. Note: Red Hat Enterprise Linux 5 and later includes the Liberation Fonts set by default.

#### 3.1. Typographic Conventions

Four typographic conventions are used to call attention to specific words and phrases. These conventions, and the circumstances they apply to, are as follows.

##### **Mono-spaced Bold**

Used to highlight system input, including shell commands, file names and paths. Also used to highlight

keys and key combinations. For example:

To see the contents of the file **my\_next\_bestselling\_novel** in your current working directory, enter the **cat my\_next\_bestselling\_novel** command at the shell prompt and press **Enter** to execute the command.

The above includes a file name, a shell command and a key, all presented in mono-spaced bold and all distinguishable thanks to context.

Key combinations can be distinguished from an individual key by the plus sign that connects each part of a key combination. For example:

Press **Enter** to execute the command.

Press **Ctrl+Alt+F2** to switch to a virtual terminal.

The first example highlights a particular key to press. The second example highlights a key combination: a set of three keys pressed simultaneously.

If source code is discussed, class names, methods, functions, variable names and returned values mentioned within a paragraph will be presented as above, in **mono-spaced bold**. For example:

File-related classes include **filesystem** for file systems, **file** for files, and **dir** for directories. Each class has its own associated set of permissions.

### Proportional Bold

This denotes words or phrases encountered on a system, including application names; dialog box text; labeled buttons; check-box and radio button labels; menu titles and sub-menu titles. For example:

Choose **System** → **Preferences** → **Mouse** from the main menu bar to launch **Mouse Preferences**. In the **Buttons** tab, click the **Left-handed mouse** check box and click **Close** to switch the primary mouse button from the left to the right (making the mouse suitable for use in the left hand).

To insert a special character into a **gedit** file, choose **Applications** → **Accessories** → **Character Map** from the main menu bar. Next, choose **Search** → **Find...** from the **Character Map** menu bar, type the name of the character in the **Search** field and click **Next**. The character you sought will be highlighted in the **Character Table**. Double-click this highlighted character to place it in the **Text to copy** field and then click the **Copy** button. Now switch back to your document and choose **Edit** → **Paste** from the **gedit** menu bar.

The above text includes application names; system-wide menu names and items; application-specific menu names; and buttons and text found within a GUI interface, all presented in proportional bold and all distinguishable by context.

### *Mono-spaced Bold Italic* or *Proportional Bold Italic*

Whether mono-spaced bold or proportional bold, the addition of italics indicates replaceable or variable text. Italics denotes text you do not input literally or displayed text that changes depending on circumstance. For example:

To connect to a remote machine using ssh, type **ssh username@domain.name** at a shell prompt. If the remote machine is **example.com** and your username on that machine is john, type **ssh john@example.com**.

The `mount -o remount file-system` command remounts the named file system. For example, to remount the `/home` file system, the command is `mount -o remount /home`.

To see the version of a currently installed package, use the `rpm -q package` command. It will return a result as follows: *package-version-release*.

Note the words in bold italics above — *username*, *domain.name*, *file-system*, *package*, *version* and *release*. Each word is a placeholder, either for text you enter when issuing a command or for text displayed by the system.

Aside from standard usage for presenting the title of a work, italics denotes the first use of a new and important term. For example:

Publican is a *DocBook* publishing system.

### 3.2. Pull-quote Conventions

Terminal output and source code listings are set off visually from the surrounding text.

Output sent to a terminal is set in **mono-spaced roman** and presented thus:

```
books      Desktop  documentation  drafts  mss    photos  stuff  svn
books_tests Desktop1  downloads      images  notes  scripts svgs
```

Source-code listings are also set in **mono-spaced roman** but add syntax highlighting as follows:

```
package org.jboss.book.jca.ex1;

import javax.naming.InitialContext;

public class ExClient
{
    public static void main(String args[])
        throws Exception
    {
        InitialContext iniCtx = new InitialContext();
        Object          ref    = iniCtx.lookup("EchoBean");
        EchoHome        home   = (EchoHome) ref;
        Echo             echo   = home.create();

        System.out.println("Created Echo");

        System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
    }
}
```

### 3.3. Notes and Warnings

Finally, we use three visual styles to draw attention to information that might otherwise be overlooked.



#### Note

Notes are tips, shortcuts or alternative approaches to the task at hand. Ignoring a note should have no negative consequences, but you might miss out on a trick that makes your life easier.



### Important

Important boxes detail things that are easily missed: configuration changes that only apply to the current session, or services that need restarting before an update will apply. Ignoring a box labeled 'Important' will not cause data loss but may cause irritation and frustration.



### Warning

Warnings should not be ignored. Ignoring warnings will most likely cause data loss.

## 4. Getting Help and Giving Feedback

### 4.1. Do You Need Help?

If you experience difficulty with a procedure described in this documentation, visit the Red Hat Customer Portal at <http://access.redhat.com>. Through the customer portal, you can:

- ▶ search or browse through a knowledgebase of technical support articles about Red Hat products.
- ▶ submit a support case to Red Hat Global Support Services (GSS).
- ▶ access other product documentation.

Red Hat also hosts a large number of electronic mailing lists for discussion of Red Hat software and technology. You can find a list of publicly available mailing lists at <https://www.redhat.com/mailman/listinfo>. Click on the name of any mailing list to subscribe to that list or to access the list archives.

### 4.2. We Need Feedback!

If you find a typographical error in this manual, or if you have thought of a way to make this manual better, we would love to hear from you! Please submit a report in Bugzilla: <http://bugzilla.redhat.com/> against the product **Red Hat Enterprise MRG**.

When submitting a bug report, be sure to mention the manual's identifier: *Grid\_Installation\_Guide*

If you have a suggestion for improving the documentation, try to be as specific as possible when describing it. If you have found an error, please include the section number and some of the surrounding text so we can find it easily.

## Chapter 1. Overview

MRG Grid provides high throughput and high performance computing and enables enterprises to achieve higher peak-computing capacity as well as higher IT-utilization by leveraging existing infrastructure to build high performance grids. High throughput computing (HTC) delivers large amounts of computing power over a sustained period of time (months or years), whereas high-performance computing (HPC) delivers significant computing power over a short period of time (hours or days).

MRG Grid enables enterprises and research organizations to apply distributed computing across their infrastructure to tackle large computational problems in a highly-efficient and effective manner.

## Chapter 2. MRG Grid Benefits and Features

### Benefits

MRG Grid provides significant benefits and value for enterprises, including:

#### Power

MRG Grid can process the largest computational workloads, from massively parallel High-Performance Computing jobs to long-running High-Throughput Computing jobs.

#### Peak Workload Handling

MRG Grid adds on-demand computational power for handling peak loads. This is achieved by means of cycle-stealing, scheduling, and the use of remote grids.

#### Flexibility

MRG Grid provides flexibility and can run high-burst or lengthy computations in both centralized and distributed grids. Jobs can be run on platforms including Linux and Windows. MRG Grid can schedule virtualized environments and workloads for the utmost flexibility in utilizing infrastructure.

#### Powerful Management Tools

MRG Grid offers a browser-based management console. The Red Hat Enterprise MRG integrated management tools enable administrators to manage, configure, provision, deploy, and monitor their grid deployments using the same tools they use for MRG Messaging and MRG Realtime.

### Features

MRG Grid provides a broad set of features involving both High-Throughput Computing and High-Performance Computing, including:

#### Virtualization

Allows for submission of a virtual machine (VM) as a user job, supporting migration of the virtual machine.

#### Dedicated and Undedicated Node Management (Cycle-Stealing)

Allows for dedicated resources (clusters) to be augmented with otherwise undedicated (desktops) resources. Cycle-stealing is managed by means of policies.

#### Multiple Standards-Based APIs

The web service interface allows you to submit jobs and manage the environment. The command line interface allows you to make use of scripts.

#### Security

Authentication using multiple mechanisms.

Privacy provided by network encryption.

Integrity of network traffic.

Authorization through flexible configuration policies.

### **Management Tools**

Powerful browser-based management tools for managing daemons and machines, security, compute jobs, scalability settings, priorities, and more. Also provides sophisticated monitoring capabilities.

### **Workflow Management**

The ability to specify job dependencies, via [DAGMan](#), allows for construction and execution of complex workflows.

### **Accounting**

User and group resource utilization is tracked and accessible to administrators.

### **ClassAds**

A flexible language for policy and meta-data description.

### **Policies**

Flexible, customizable policies specified by jobs and resources via ClassAds.

### **High Availability**

The Negotiator and Collector (via the High Availability Daemon) and the Schedd (via Schedd Fail-over) can have their respective states replicated to allow for graceful fail-over upon service disruption.

### **Compute On-Demand (COD)**

The ability for a node or set of nodes to be claimed by a user in a way that others may use the claimed nodes until the user needs them.

### **Priority Based Scheduling**

Priority scheduling is performed at the granularity of a user.

Fair-share scheduling can be performed on groups of users.

Priority management is controllable by administrators.

### **Account Remapping**

Allows for execution across administrative domains.

Enhance security by using a restricted pool of users to run jobs on execute machines.

### **Parallel Universe**

Provides an extensible framework for running parallel - including Message Passing Interface (MPI) - jobs.

Co-allocation of compute nodes is done automatically.

Framework implementation for MPICH1, [MPICH2](#), and [LAM](#) provided.

**Java Universe**

Explicit support of jobs written in Java.

**Time Scheduling for Job Execution (Cron)**

Allows a job or multiple jobs to be started at specific times, with customizable policy for failures such as missed deadlines.

**File Staging**

Support for automatic file staging in the absence of a shared file system. Job input and online file streaming from submit to execute nodes using [Chirp](#).

## Chapter 3. MRG Grid Scale Requirements

MRG Grid is designed for deployments of any scale, from deployments of fewer than a dozen nodes to deployments of tens of thousands of nodes.

The deployments described below are examples of scale sizes.

### 3.1. Small

#### Small

An example small scale deployment using a management node with a scheduler and central manager is:

- ▶ 512 cores
- ▶ 5 concurrent grid users, accessing the grid at 1 page view per second (peak)
- ▶ 10 job submitters, submitting 1 job per second concurrently (peak)
- ▶ 10 job completions per minute (sustained), 3 years of job history (1 million jobs)
- ▶ Ability to sustain peak rates for at least 5 minutes

### 3.2. Medium

#### Medium

An example medium scale deployment using a management node with a scheduler and central manager is:

- ▶ 4000 cores
- ▶ 20 concurrent grid users, accessing the grid at 1 page view per second (peak)
- ▶ 20 job submitters, submitting 2 jobs per second concurrently (peak)
- ▶ 100 job completions per minute (sustained), 3 years of job history (10 million jobs)
- ▶ Ability to sustain peak rates for at least 5 minutes

### 3.3. Large

Large-scale grids support 40000 cores, or deployments of a similar magnitude. Red Hat, Inc recommends that customers configure large-scale MRG Grid installations in cooperation with a Solution Architect through Red Hat, Inc Consulting.

## Chapter 4. Installing MRG Grid

To install MRG Grid you will need to have registered your system with [Red Hat Network](#). This table lists the Red Hat Enterprise MRG channels available on Red Hat Network for MRG Grid.

**Table 4.1. Red Hat Enterprise MRG Channels Available on Red Hat Network**

Channel Name	Operating System	Architecture
Red Hat MRG Grid	RHEL-5 Server	32-bit, 64-bit
Red Hat MRG Grid	RHEL-6 Server	32-bit, 64-bit
Red Hat MRG Grid	non-Linux	32-bit



### Important

Before you install Red Hat Enterprise MRG check that your hardware and operating system are supported. A complete list is available on the [Red Hat Enterprise MRG Supported Hardware Page](#).

### 4.1. Installing MRG Grid on Red Hat Enterprise Linux

1. Install the MRG Grid group using the **yum** command.

```
# yum groupinstall "MRG Grid"
```

2. You can check the installation location and that the components have been installed successfully by using the **rpm -q1** command with the name of the package you installed. For example:

```
# rpm -q1 condor
/etc/condor
/etc/condor/condor_config
/usr/bin/condor
...
[output truncated]
```

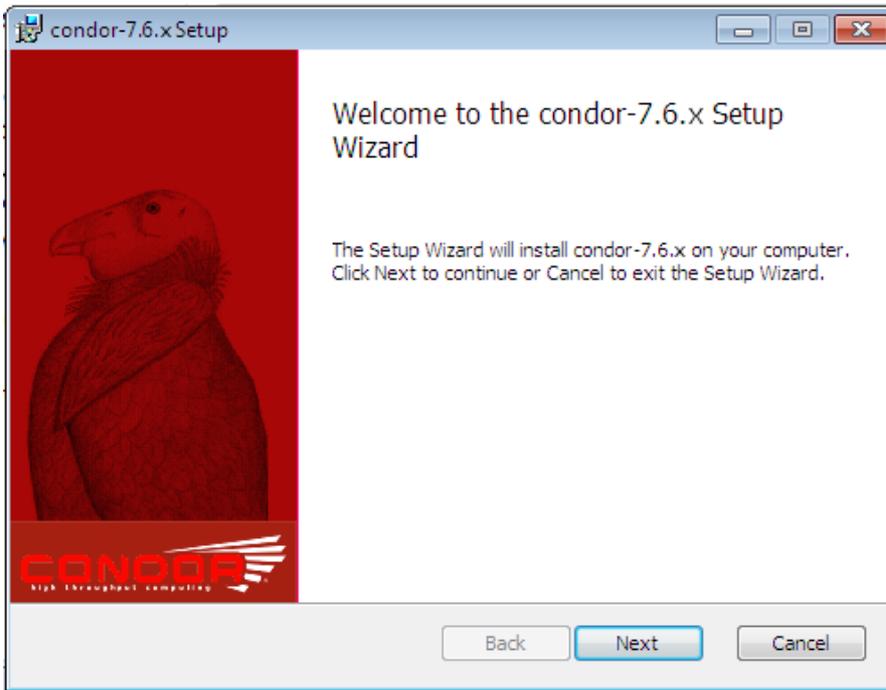


### Note

If you find that yum is not installing all the dependencies you require, make sure that you have registered your system with [Red Hat Network](#).

### 4.2. Installing MRG Grid on Microsoft Windows

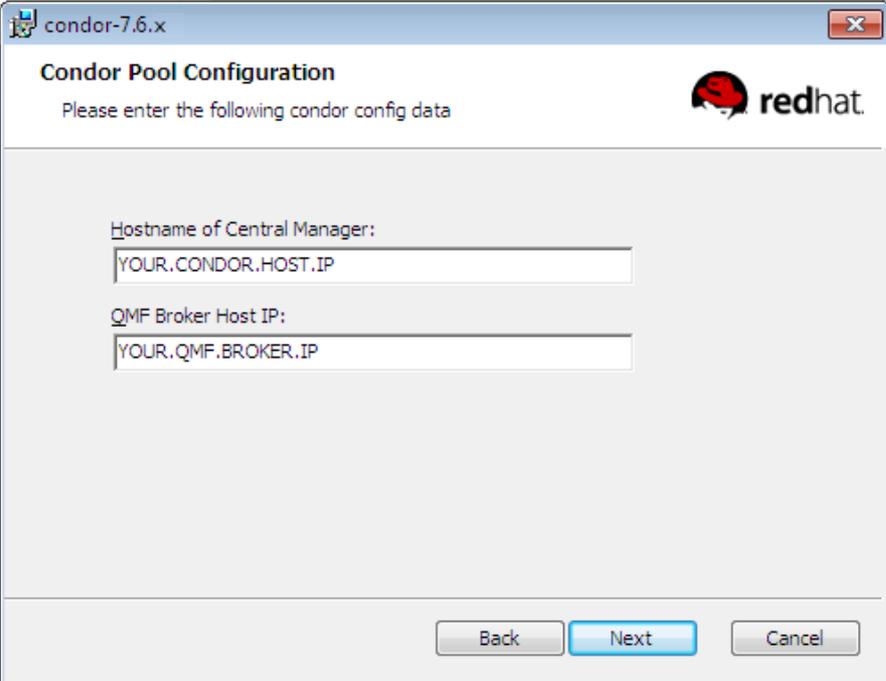
1. Download the **MRG Grid 2.2 for Windows installer** file from the **MRG Grid v.2 (for non-Linux platforms)** channel on Red Hat Network. It is located under the **Downloads** tab.
2. Save the **condor-version.number.msi** installation file in a location of your choice.
3. Navigate to the directory where the installation file has been saved. Double-click on the **.msi** file.
4. Click **Next** to begin the installation.



5. Read the End-User License Agreement. Check **I accept the terms in the License Agreement** and click **Next** to continue.



6. Enter the IP configuration information for the installation. Click **Next** to continue.



condor-7.6.x

### Condor Pool Configuration

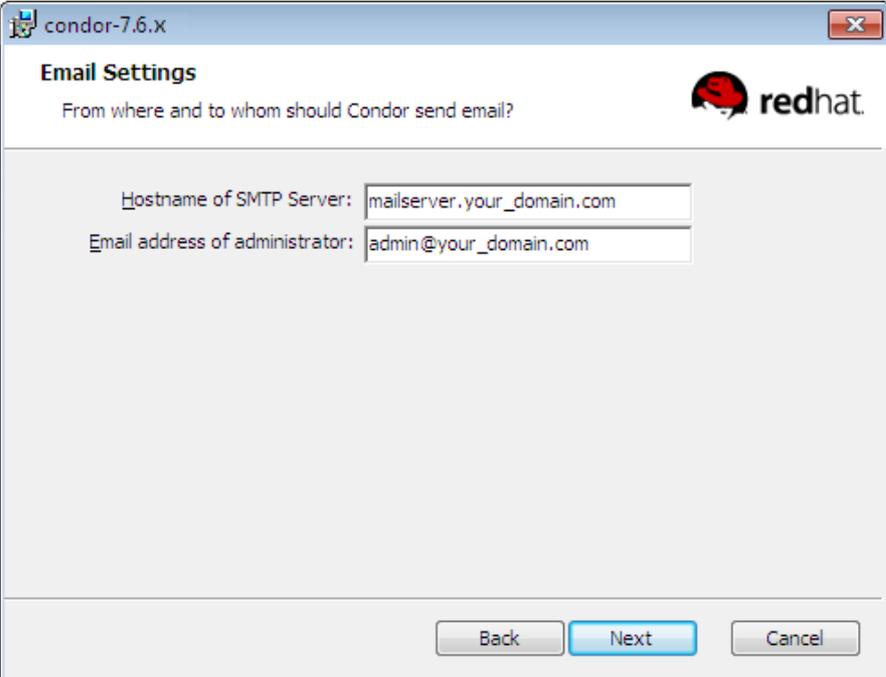
Please enter the following condor config data

Hostname of Central Manager:  
YOUR.CONDOR.HOST.IP

QMF Broker Host IP:  
YOUR.QMF.BROKER.IP

Back Next Cancel

7. Enter your email configuration information for the installation. Click **Next** to continue.



condor-7.6.x

### Email Settings

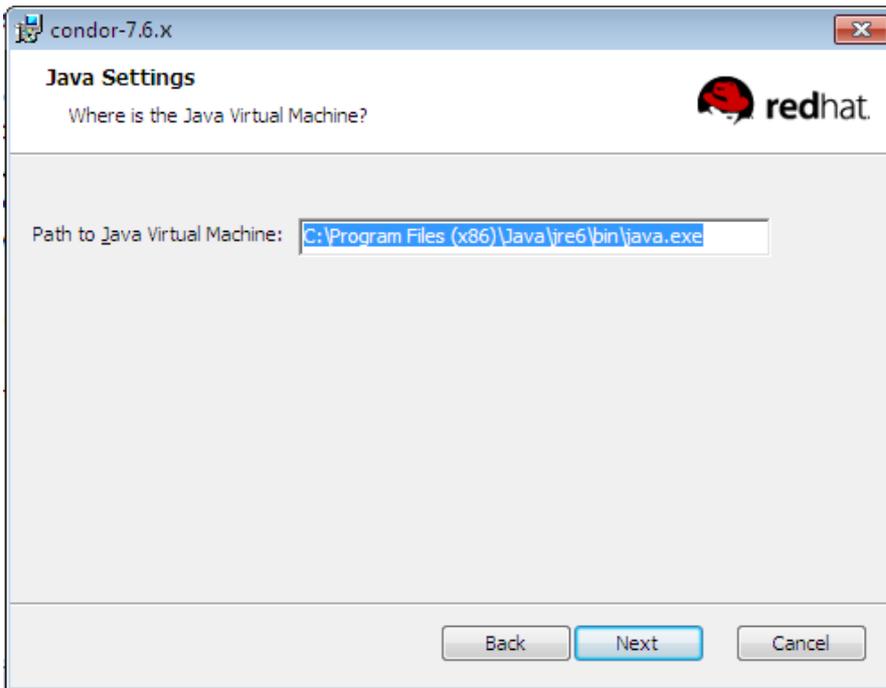
From where and to whom should Condor send email?

Hostname of SMTP Server: mailserver.your\_domain.com

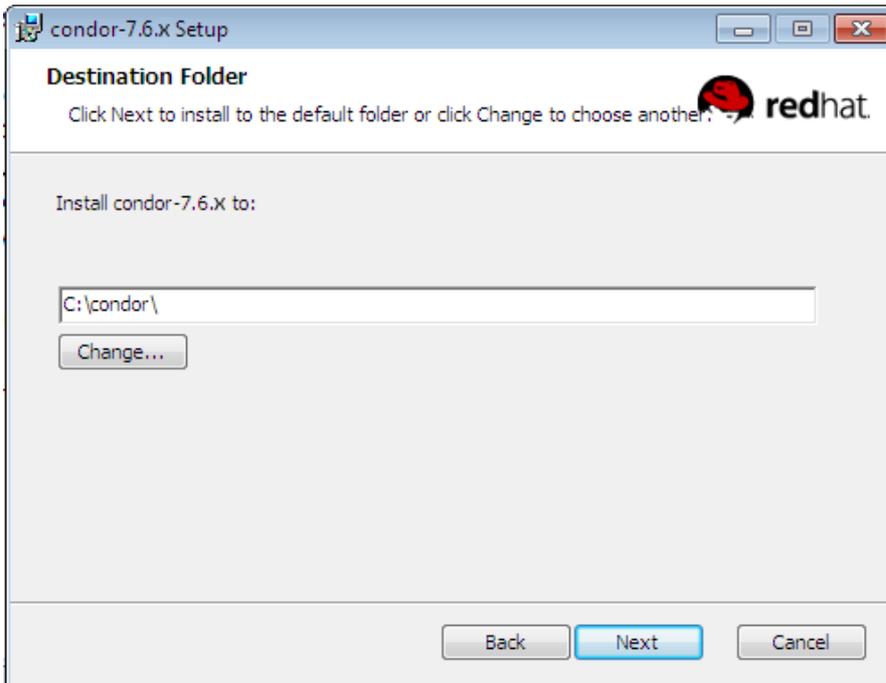
Email address of administrator: admin@your\_domain.com

Back Next Cancel

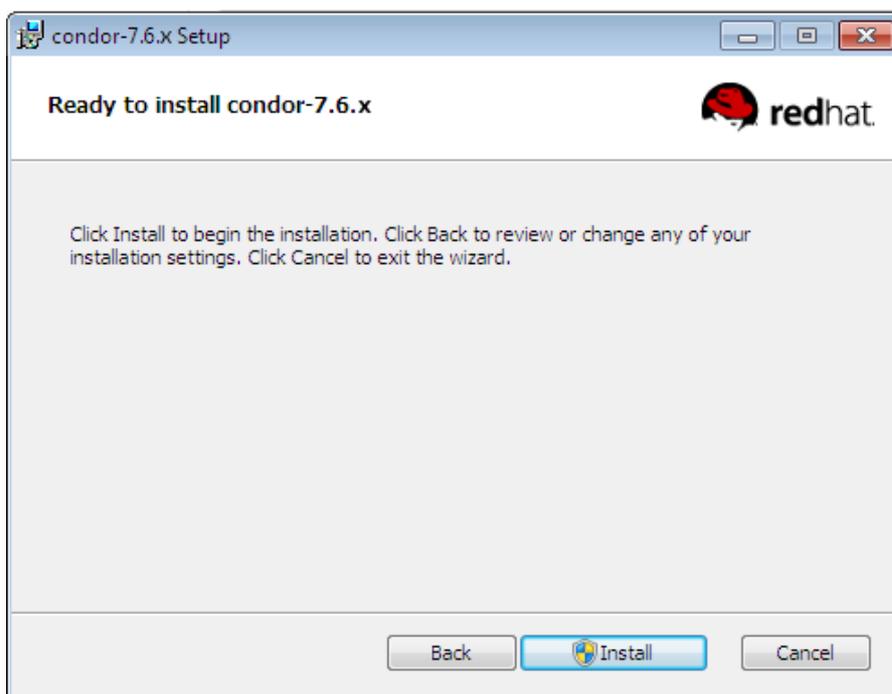
8. Enter the location of your Java installation. Click **Next** to continue.



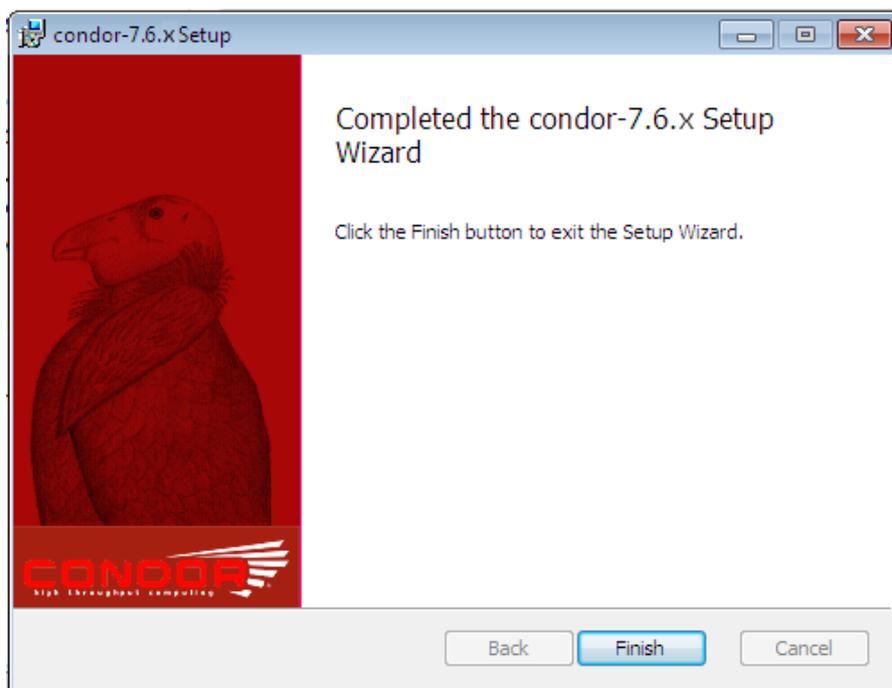
9. Enter the installation location. Click **Next** to continue.



10. Click **Install** to complete the installation.



11. Click **Finish** to exit the Setup Wizard.



12. The Setup Wizard will prompt for the machine to be restarted.  
Remote configuration is enabled by default. After the restart, MRG Grid will attempt to connect to the remote configuration store to download the new configuration.

 Note

The installer can also be run in unattended mode. Open the shell by selecting **Start** → **Run**, then typing in **cmd**. At the shell prompt, use a command like this:

```
> msiexec /passive /Lv install.log /package <condor-version.number.msi>
CONDOR_HOST="YOUR.CONDOR.HOST.IP" QMF_BROKER_HOST="YOUR.QMF.BROKER.IP"
CONDOR_ADMIN="admin@your_domain.com"
SMTP_SERVER="mailserver.your_domain.com"
INSTALLDIR="C:\condor"
```

## 4.3. Upgrading MRG Grid Peacefully

Shutting down a node peacefully allows jobs to complete prior to HTCondor exiting. This is the recommended way to upgrade machines which may be executing jobs. As it is not necessary to peacefully shutdown a central manager, the **service condor stop** command is sufficient.

### 4.3.1. Upgrading Execute Nodes

Issue the following command from the central manager, or alternatively from wherever and with whatever user you require to issue administrative commands.

For a single node:

```
# condor_off -startd -peaceful hostname
```

For all execute nodes:

```
# condor_off -all -startd -peaceful
```

This will allow jobs to exit gracefully. The administrator can then update specific sub-packages with **yum update**.

### 4.3.2. Upgrading Scheduler Nodes

It is recommended to disable new submissions to the schedd prior to upgrading by adding the following to the configuration file:

```
'MAX_JOBS_SUBMITTED=0'
```

Once all jobs have completed, turn off HTCondor by issuing the following command from the central manager, or alternatively from wherever and with whatever user you require to issue administrative commands.

```
# condor_off -schedd -peaceful hostname
```

Once the scheduler has been shutdown, the administrator can then update specific subpackages with **yum update**.

## Chapter 5. Getting Started with MRG Grid

1. Review the local configuration file before starting MRG Grid. By default, this file is located at `/etc/condor/config.d/00personal_condor.config`.
2. The default configuration sets up a *Personal HTCondor*. This is a specific configuration was built for individual users who do not have their own pool of machines. To find out more about Personal HTCondor, see the [HTCondor Manual](#).
3. Additional configuration is necessary in order for other machines to join your pool. Open the local configuration file in your preferred text editor and create or locate the **ALLOW\_WRITE** option. Change this option to allow machines to join your pool and submit jobs. Any machine given write access using the **ALLOW\_WRITE** option should also be given read access using the **ALLOW\_READ** option.

```
ALLOW_WRITE = *.your.domain.com
```



### Warning

The simplest option is to change the **ALLOW\_WRITE** option to **ALLOW\_WRITE = \***. However, this will allow anyone to submit jobs or add machines to your pool. This is a serious security risk and therefore not recommended.

4. MRG Grid is usually run as a service. To start MRG Grid run the following command:

```
# service condor start
```



### Note

See [Further Reading](#) if you have trouble installing MRG Grid. For detailed configuration information, see the *MRG Grid User Guide*.

## Chapter 6. More Information

### 6.1. Reporting Bugs

Follow these instructions to enter a bug report:

1. Create a [Bugzilla](#) account.
2. Log in and click on [Enter A New Bug Report](#).
3. You will need to identify the product (Red Hat Enterprise MRG), the version (2.2), and whether the bug occurs in the software (component=grid) or in the documentation (component=Grid\_Installation\_Guide).

### 6.2. Further Reading

#### Further Reading

**Red Hat Enterprise MRG and MRG Grid Product Information**

<http://www.redhat.com/mrg>

**MRG Grid User Guide and other Red Hat Enterprise MRG manuals**

<https://access.redhat.com/knowledge/docs/>

**HTCondor Manual**

<http://research.cs.wisc.edu/htcondor/manual/>

**Red Hat Knowledgebase**

<https://access.redhat.com/knowledge/search>

## Revision History

<b>Revision 3.0-5</b>	<b>Fri Feb 22 2013</b>	<b>David Ryan</b>
Various updates.		
<b>Revision 3.0-3</b>	<b>Sun Feb 17 2013</b>	<b>David Ryan</b>
Various updates.		
<b>Revision 3.0-2</b>	<b>Wed Jan 30 2013</b>	<b>David Ryan</b>
Various updates.		
<b>Revision 3.0-0</b>	<b>Fri Sep 14 2012</b>	<b>Cheryn Tan</b>
Prepared for publishing (MRG 2.2).		
<b>Revision 2-4</b>	<b>Fri Aug 10 2012</b>	<b>Cheryn Tan</b>
Rebuild for Publican 3.0.		
<b>Revision 2-2</b>	<b>Tue Apr 19 2012</b>	<b>Cheryn Tan</b>
BZ#798072 - Updated unattended installation command and screenshots for Windows. Included fixes from documentation QE review.		
<b>Revision 2-1</b>	<b>Tue Feb 28 2012</b>	<b>Tim Hildred</b>
Updated configuration file for new publication tool.		
<b>Revision 2-0</b>	<b>Tue Dec 6 2011</b>	<b>Alison Young</b>
Prepared for publishing		
<b>Revision 1-4</b>	<b>Fri Nov 18 2011</b>	<b>Alison Young</b>
BZ#733374 - Re-validate scale numbers		
<b>Revision 1-3</b>	<b>Tue Nov 15 2011</b>	<b>Alison Young</b>
BZ#752406 - change RHEL versions		
<b>Revision 1-2</b>	<b>Mon Oct 31 2011</b>	<b>Alison Young</b>
BZ#733374 - updated scale wording		
<b>Revision 1-1</b>	<b>Wed Sep 07 2011</b>	<b>Alison Young</b>
Prepared for publishing		
<b>Revision 1-0</b>	<b>Thu Jun 23 2011</b>	<b>Alison Young</b>
Prepared for publishing		
<b>Revision 0.1-7</b>	<b>Fri Jun 17 2011</b>	<b>Alison Young</b>
Rebuilt for docs stage		
<b>Revision 0.1-6</b>	<b>Fri Jun 17 2011</b>	<b>Alison Young</b>
Minor scale updates		
<b>Revision 0.1-5</b>	<b>Tue Jun 14 2011</b>	<b>Alison Young</b>

Minor updates

<b>Revision 0.1-4</b>	<b>Thu Jun 09 2011</b>	<b>Alison Young</b>
BZ#677495 - updated Windows Installation images		
<b>Revision 0.1-3</b>	<b>Fri Jun 03 2011</b>	<b>Alison Young</b>
rebuilt as stage didn't update		
<b>Revision 0.1-2</b>	<b>Thu Jun 02 2011</b>	<b>Alison Young</b>
BZ#677495 - Updates for 2.0		
BZ#708789 - typo fix		
BZ#708792 - typo fix		
<b>Revision 0.1-1</b>	<b>Fri May 20 2011</b>	<b>Alison Young</b>
Minor xml updates		
<b>Revision 0.1-0</b>	<b>Tue Feb 22 2011</b>	<b>Alison Young</b>
Fork from 1.3		