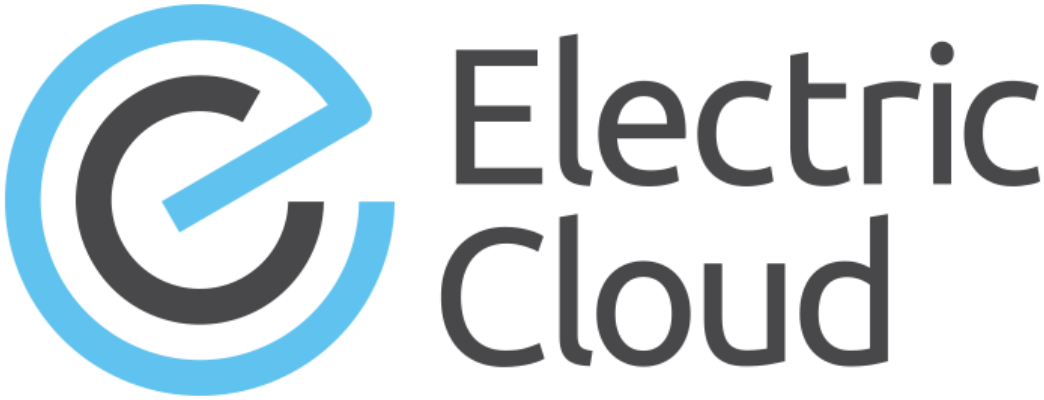


ElectricAccelerator

Installation and Configuration Guide

for version 7.1

Electric Cloud, Inc.
35 South Market Street, Suite 100
San Jose, CA 95113
www.electric-cloud.com

Copyright © 2002 - 2014 Electric Cloud, Inc. All rights reserved.

Published 9/15/2014

Electric Cloud® believes the information in this publication is accurate as of its publication date. The information is subject to change without notice and does not represent a commitment from the vendor.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED “AS IS.” ELECTRIC CLOUD, INCORPORATED MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Use, copying, and distribution of any ELECTRIC CLOUD software described in this publication requires an applicable software license.

Copyright protection includes all forms and matters of copyrightable material and information now allowed by statutory or judicial law or hereinafter granted, including without limitation, material generated from software programs displayed on the screen such as icons, screen display appearance, and so on.

The software and/or databases described in this document are furnished under a license agreement or nondisclosure agreement. The software and/or databases may be used or copied only in accordance with terms of the agreement. It is against the law to copy the software on any medium except as specifically allowed in the license or nondisclosure agreement.

Trademarks

Electric Cloud, ElectricAccelerator, ElectricCommander, ElectricDeploy, ElectricInsight, and Electric Make are registered trademarks or trademarks of Electric Cloud, Incorporated.

Electric Cloud products—ElectricAccelerator, ElectricCommander, ElectricDeploy, ElectricInsight, and Electric Make—are commonly referred to by their “short names”—Accelerator, Commander, Deploy, Insight, and eMake—throughout various types of Electric Cloud product-specific documentation.

Other product names mentioned in this guide may be trademarks or registered trademarks of their respective owners and are hereby acknowledged.

Contents

Chapter 1: Overview **1-1**
About ElectricAccelerator 1-2
Cluster Manager 1-2
Electric File System 1-2
ElectricAccelerator Agent 1-2
Electric Make 1-2
Electrify 1-3
Understanding Component Interactions 1-3
Electric Make and EFS 1-3
Electric Make and Cluster Manager 1-4
Cluster Manager and Other Components 1-4
When to Use the ElectricAccelerator Cluster 1-4
Grid Integration 1-5
Chapter 2: Supported Platforms and System Requirements **2-1**
Supported Linux Platforms 2-2
Known Linux Kernel Issue and ElectricAccelerator Performance 2-4
Supported Windows Platforms 2-5
Supported Solaris Platforms 2-7
Supported Third-Party Build Tools 2-8
System Requirements 2-9
Hardware Requirements 2-9
Web Browsers 2-10
Port Usage 2-10
Toolchain 2-11
Databases 2-11
Requirements for MySQL 5.5.16 on Linux 2-11
Location Limitations 2-11
Cygwin 2-12
Component Compatibility Matrix 2-12
Ensuring a Consistent Build Environment 2-13
Planning for Grid Use 2-13
Checksum Utility 2-14
Chapter 3: Installation **3-1**
Before You Install ElectricAccelerator 3-1
Cluster Manager and Agent Install Location Limitations 3-1
32-bit vs. 64-bit Information 3-1
Installing ElectricAccelerator on Linux/Solaris 3-2
Additional Linux/Solaris Install Information 3-2

Installing Cluster Manager 3-2
Installing Agent/EFS 3-5
Installing Electric Make 3-9
Path Settings 3-10
Installing ElectricAccelerator on Windows 3-10
Additional Windows Install Information 3-10
Installing Cluster Manager 3-11
Installing Electric Agent/EFS 3-13
Installing Electric Make 3-17
Silent Install 3-18
Installer Command-Line Options 3-18
Creating an Installer Properties File 3-24
Additional Install Options 3-25
Migrating an eMake-only Installation to a Different Machine 3-25
Installing the Backward-Compatible eMake Package on Agents 3-25
Installing on a Single Agent Machine 3-26
Installing on Multiple Agent Machines 3-26
Protocol Changes 3-26
Chapter 4: Configuration 4-1
Configuring Linux/Solaris 4-2
Configuring Windows 4-2
Windows Notes 4-5
Registry Information 4-6
Using the accelerator.properties File 4-8
Changing Log Locations 4-8
Disabling Agents by Default 4-8
User Authentication 4-8
Setting Up Email Notification 4-13
Adding Custom Protocol Mismatch Error Text 4-15
Using the Database Connection Monitor 4-15
Agent Log Rotation 4-15
Configuring Log Rotation 4-15
Alternate Methods for Configuring Log Rotation 4-16
Changing the Disk Cache Directory and Agent Temporary Storage Location 4-16
Default Directories 4-17
Installation Location 4-17
Log Files 4-17
Disk Cache Directory and Agent Temporary Storage Location 4-18
Logging In and Licensing 4-18
Enabling Resource Management 4-19
Installing an Apache Server Certificate 4-20
Generate a New Certificate and Private Key 4-21
Send the Request to the CA 4-22
Install the Key and Signed Certificate 4-22
Configuring LSF 4-22
Creating a Queue 4-23

Additional Information 4-23
Enabling Grid Host Manager 4-24
Grid Host Manager Examples 4-24
Documentation Roadmap 4-28
Chapter 5: Upgrade 5-1
Upgrading ElectricAccelerator on Linux/Solaris 5-2
Cluster Manager 5-2
Electric Agent/EFS 5-4
Electric Make 5-6
Upgrading ElectricAccelerator on Windows 5-6
Cluster Manager 5-6
Electric Agent/EFS 5-7
Electric Make 5-9
Using the ecconfig Command-Line Option 5-9
ecclusterupgrade Options 5-10
Chapter 6: Uninstall 6-1
Uninstalling Accelerator on Linux/Solaris 6-2
Uninstalling Accelerator on Windows 6-2
Index 7-1

Chapter 1: Overview

This guide describes the system requirements necessary to install and run ElectricAccelerator®. Installation walkthroughs are available for each component on every supported platform. Also described are upgrade instructions and initial configuration tasks essential to using the software successfully.

Topics:

- About ElectricAccelerator
- Understanding Component Interactions
- When to Use the ElectricAccelerator Cluster
- Grid Integration

About ElectricAccelerator

ElectricAccelerator (Accelerator) is a software build accelerator that dramatically reduces software build times by distributing the build over a large cluster of inexpensive servers. Using a patented dependency management system, Accelerator identifies and fixes problems in real time that would break traditional parallel builds. Accelerator plugs into existing software development environments seamlessly and includes web-based management and reporting tools.

During Accelerator installation, the following components will be installed:

- Cluster Manager
- Electric File System
- ElectricAccelerator Agent
- Electric Make
- Electrify

Cluster Manager

The Cluster Manager machine is a server that contains two layers: 1) a service layer to allocate Agents to builds, collect build results, and monitor system activity, and 2) a web server layer for the web interface.

The Cluster Manager also allows users to view upcoming and past builds, view the status of Agents in the cluster, and run reports.

Electric File System

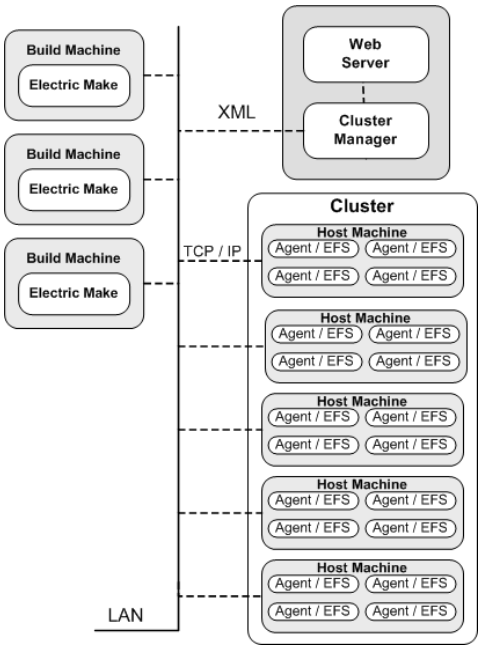
Electric File System (EFS) is a special-purpose file system driver, monitoring every file access and providing Electric Make with complete usage information. This driver collects dependency information, which allows Electric Make to automatically detect and correct out-of-order build steps. Each EFS driver instance is paired with an ElectricAccelerator Agent. During the installation process, the Agent and EFS are installed at the same time.

ElectricAccelerator Agent

As the user-level component running on the hosts, the Agent and EFS are inseparable—the Agent is an intermediary between Electric Make and EFS. Depending on your system configuration requirements, you may have one EFS/Agent installed per virtual CPU.

Electric Make

Electric Make® (eMake), the main build application, is a new Make version invoked interactively or through build scripts. It reads makefiles in several different formats, including GNU Make and Microsoft NMAKE. Electric Make distributes commands to the cluster for remote execution and services file requests.



Electrify

The Electrify component accelerates builds by parallelizing the build process and distributing build steps across clustered resources. You can use one of two methods to monitor your tools: a wrapper application or automatic process interception. The *ElectricAccelerator Electric Make Users Guide* contains information about using Electrify.

Understanding Component Interactions

To a user, Accelerator may appear identical to other Make versions—reading makefiles in several different formats and producing identical results. Using a cluster for builds is transparent to the Accelerator user.

Important differences in Accelerator build processing versus other distributed systems:

- Accelerator components work together to achieve faster, more efficient builds. Instead of running a sequential build on a single processor, Accelerator executes build steps in parallel on a cluster of hosts.
- For fault tolerance, job results are isolated until the job completes. If an Agent fails during a job, Accelerator discards any partial results it might have produced and reruns the job on a different Agent.
- Missing dependencies discovered at runtime are collected in a history file that updates each time a build is invoked. Accelerator uses this collected data to improve performance of subsequent builds.

Electric Make and EFS

High concurrency levels in Accelerator are enabled by the Electric File System (EFS). When a job such as a compilation runs on a host, it accesses files such as source files and headers through EFS. EFS records detailed file access data for the build and returns that data to Electric Make.

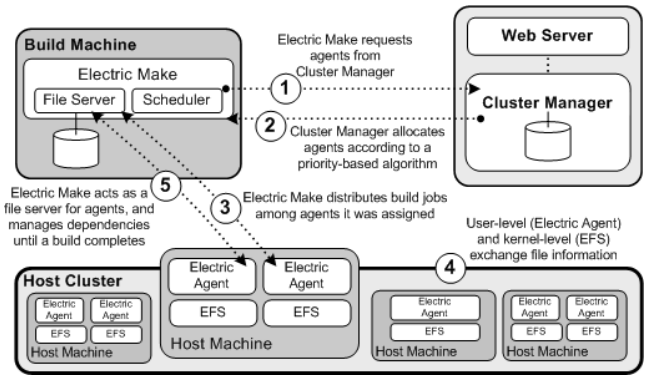
Electric Make acts as a file server for Agents, reading the correct file version from file systems on its machine and passing that information back to the Agents. Agents retain different file version information and do not rely on Electric Make's file sequencing ability to provide the correct version for a job. The Agent receives file data, downloads it into the kernel, notifying EFS, which then completes the original request. At the end of a job, Electric Agent returns any file modifications to Electric Make so it can apply changes to its local file systems.

Electric Make and Cluster Manager

When Electric Make is invoked on the build machine, it communicates with Cluster Manager to acquire a set of Agents it can use for the build. When Electric Make finishes, it sends Cluster Manager the build results, and tells Cluster Manager that Agents are free now to work on other builds. If more than one build is invoked, Cluster Manager allocates agents using a priority-based algorithm. Builds with the same priority share Agents evenly, while higher priority builds are allocated more Agents than lower priority builds. By default, agents running on the same host machine are allocated to the same build. In real time, Cluster Manager dynamically adjusts the number of agents assigned to running builds as each build's needs change, which allows Accelerator to make the best use of cluster resources.

Cluster Manager and Other Components

Cluster Manager's primary responsibility is Agent allocation. However, through Cluster Manager, users also create and maintain the build infrastructure: build classes, user access, build history details, and manage Agent activity.



When to Use the ElectricAccelerator Cluster

Accelerator is designed to be a complete replacement for GNU Make or Microsoft NMAKE. It has the same command-line switches, produces identical output, and (with a few exceptions) reads your existing makefiles without change. Wherever you are currently invoking *gmake* or *nmake*—in a script, from a batch process, or interactively—you can run Electric Make to distribute the work to the Cluster Manager and complete your build faster.

In some cases, however, distributed builds are not always faster. In particular, if a build has very little work to do and/or has much higher I/O activity compared to computation, it can be faster to allow the build to execute locally. The best examples of these build types are:

- "Clean" targets – often do little more than large recursive deletes.
- "One touch" build – a build consisting of nothing more than zero or one compilations followed immediately by a large link.

But then, depending on the build specifics, the cluster may outperform local builds for a large class of "clean" target or "one touch" builds—it is best to experiment with a few common configurations to determine which mode is more efficient. Typically, if a build has two or more jobs it can execute concurrently, running the build on the cluster is more efficient.

If you find that some common "clean" or "one touch" builds are faster when run locally, you can run Electric Make in local mode. In local mode, Electric Make behaves exactly like your existing Make, running all jobs in series on the local system.

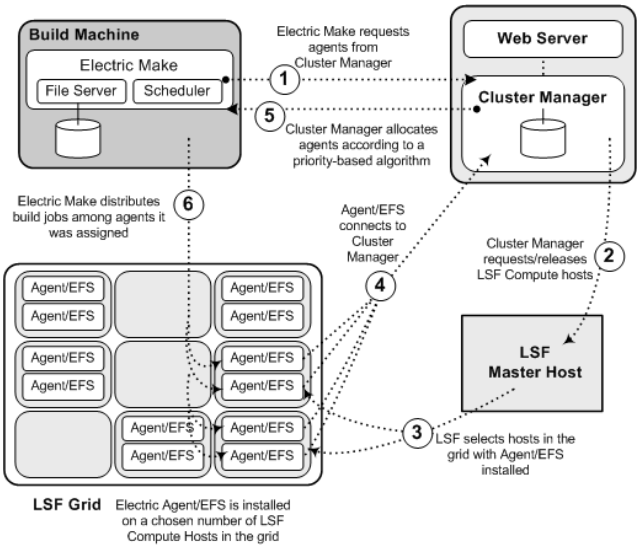
Grid Integration

If you plan to use a host grid with Accelerator, the following list shows the relationship between Accelerator components and the grid management software, such as LSF or Oracle Grid Engine.

- Electric Make does not interact directly with the grid management software—Electric Make can run inside or outside of a grid.
- Electric Agent/EFS software is designed to run on grid execution hosts. While the Agent/EFS does not "talk" to the grid management software, the Agent/EFS can be activated by it, effectively creating a dynamically sized Accelerator cluster.
- Cluster Manager interfaces with the grid management software to add or subtract machines to/from the Accelerator cluster as build demands increase or decrease.

Note: Grid management client software must be installed on the designated Cluster Manager server *before* Cluster Manager is used in grid mode.

The following diagram illustrates how Accelerator interacts with grid management software, using LSF as an example.



For additional information about planning for and using a grid, see these topics:

- Planning for Grid Use
- Enabling Grid Host Manager
- Configuring LSF

Chapter 2: Supported Platforms and System Requirements

This section discusses ElectricAccelerator's hardware and system requirements. Make sure you are familiar with all of this section's information ***before*** installing Accelerator.

Topics:

- Supported Linux Platforms
- Supported Windows Platforms
- Supported Solaris Platforms
- Supported Third-Party Build Tools
- System Requirements
- Component Compatibility Matrix
- Ensuring a Consistent Build Environment
- Planning for Grid Use
- Checksum Utility

Supported Linux Platforms

Refer to the Release Notes for the latest updates to this information.

ElectricAccelerator Cluster Manager, Electric Make, and Electric Agent/EFS currently run on:

- Red Hat Enterprise Linux Platforms
- SUSE Linux Enterprise Server Platforms
- Ubuntu Platforms

Red Hat Enterprise Linux Platforms

Platform	Notes
RHEL 6.4 (kernel 2.6.32-358) RHEL 6.3 (kernel 2.6.32-279) RHEL 6.2 (kernel 2.6.32-220) RHEL 6.1 (kernel 2.6.32-131) RHEL 6.0 (kernel 2.6.32-71) (32 and 64-bit)	• For 64-bit systems, you must install 32-bit libraries before invoking the installer. • Run `yum install glibc.i686`. If yum server is not present, the required RPMs are `glibc-2.11.1.1.10.el6.i686` and `nss-softokn-freebl-3.12.4-11.el6.i686`. • Be advised of a known Linux kernel issue. • SELinux must be disabled. To disable SELinux, modify `/etc/selinux/config` by changing SELINUX=enforcing to SELINUX=disabled. • Extended attributes (xattr) are not supported. Attempting to query or set extended attributes for a file returns an ENOTSUPP ("Operation not supported") error. • For 32-bit, only x86 is supported. For 64-bit, only x86-64 is supported. IA-64 (Itanium) is not supported.

Platform	Notes
RHEL 5.9 (kernel 2.6.18-348) RHEL 5.8 (kernel 2.6.18-308) RHEL 5.7 (kernel 2.6.18-274) RHEL 5.6 (kernel 2.6.18-238) RHEL 5.5 (kernel 2.6.18-194) RHEL 5.4 (kernel 2.6.18-164) (32 and 64-bit)	• For agent hosts, you must install the corresponding `kernel-devel` version package before invoking the installer. • Be advised of a known Linux kernel issue. • SELinux must be disabled. To disable SELinux, modify `/etc/selinux/config` by changing SELINUX=enforcing to SELINUX=disabled. • Extended attributes (xattr) are not supported. Attempting to query or set extended attributes for a file returns an ENOTSUPP ("Operation not supported") error. • For 32-bit, only x86 is supported. For 64-bit, only x86-64 is supported. IA-64 (Itanium) is not supported.
RHEL 4.9 (kernel 2.6.9-100) RHEL 4.8 (kernel 2.6.9-89) (32 and 64-bit)	• Agent hosts require these three packages: `kernel-devel`, `gcc`, and `gcc-c++`, before invoking the installer. • SELinux must be disabled. To disable SELinux, modify `/etc/selinux/config` by changing SELINUX=enforcing to SELINUX=disabled. • Extended attributes (xattr) are not supported. Attempting to query or set extended attributes for a file returns an ENOTSUPP ("Operation not supported") error. • For 32-bit, only x86 is supported. For 64-bit, only x86-64 is supported. IA-64 (Itanium) is not supported.

SUSE Linux Enterprise Server Platforms

Platform	Notes
SLES 11 SP2 (kernel 3.0.10) SLES 11 SP1 (kernel 2.6.32) SLES 11 (kernel 2.6.27) (32 and 64-bit)	• For agent hosts, you must install the `gcc` and `kernel-source` packages before invoking the installer. • To install `gcc`, run `sudo zypper install gcc` This installs the following: `gcc`, `gcc43`, `glibc-devel`, `linux-kernel-headers`. • To install kernel-source, run `sudo zypper install kernel-source` • SELinux must be disabled. To disable SELinux, modify `/etc/selinux/config` by changing SELINUX=enforcing to SELINUX=disabled. • Extended attributes (xattr) are not supported. Attempting to query or set extended attributes for a file returns an ENOTSUPP ("Operation not supported") error. • For 32-bit, only x86 is supported. For 64-bit, only x86-64 is supported. IA-64 (Itanium) is not supported.

Platform	Notes
SLES 10 SP4 (kernel 2.6.16) (32 and 64-bit)	• SELinux must be disabled. To disable SELinux, modify `/etc/selinux/config` by changing SELINUX=enforcing to SELINUX=disabled. • Extended attributes (xattr) are not supported. Attempting to query or set extended attributes for a file returns an ENOTSUPP ("Operation not supported") error. • For 32-bit, only x86 is supported. For 64-bit, only x86-64 is supported. IA-64 (Itanium) is not supported.

Ubuntu Platforms

Platform	Notes
Ubuntu 12.10 (kernel 3.5) Ubuntu 12.04 (kernel 3.2) (32 and 64-bit)	• For 64-bit systems, you must install 32-bit libraries before invoking the installer. • Run sudo apt-get install ia32-libs to install the 32-bit libraries. • SELinux must be disabled. To disable SELinux, modify `/etc/selinux/config` by changing SELINUX=enforcing to SELINUX=disabled. • Extended attributes (xattr) are not supported. Attempting to query or set extended attributes for a file returns an ENOTSUPP ("Operation not supported") error. • For 32-bit, only x86 is supported. For 64-bit, only x86-64 is supported. IA-64 (Itanium) is not supported. • Be advised of a known Linux kernel issue.
Ubuntu 11.10 (kernel 3.0) Ubuntu 11.04 (kernel 2.6.38) (32 and 64-bit)	
Ubuntu 10.10 (kernel 2.6.35) Ubuntu 10.04.1 (kernel 2.6.32) (32 and 64-bit)	

Known Linux Kernel Issue and ElectricAccelerator Performance

Affected Kernel Versions

- RHEL versions later than 2.6.18-194.32 and earlier than 2.6.32-131.
- Ubuntu versions 2.6.31, 32, 33, and 34

Symptoms

Affected systems may encounter reduced performance on both ext3 and ext4 filesystems. Symptoms may include:

- hung_task_timeout_secs messages in system dmesg logs
- widely variable agent availability (entering and exiting agent "penalty" status frequently)
- contention over the ecagent.state file
- slower builds (with unexplained variances)

To help determine if your environment has this kernel issue, run

```
dmesg | grep hung_task_timeout
```

If hung_task_timeout errors are present, this indicates that a known Linux kernel issue is present. Contact your kernel provider to obtain a different version of the precompiled kernel.

Corrective Actions

For systems running RHEL 5.6, 5.7, 5.8, and 6.0

Consider upgrading to 2.6.32-131 (RHEL 6.1), or downgrading to 2.6.18-194.32 (RHEL 5.5).

For systems running Ubuntu 10.04

Consider upgrading to kernel version 2.6.35 or later.

To install the upstream kernel, do the following, for example (replace * with the version you want to install):

```
sudo add-apt-repository ppa:kernel-ppa/ppa
sudo aptitude update
sudo apt-get install linux-image-2.6.35-*-generic
sudo apt-get install linux-headers-2.6.35-*-generic
sudo apt-get linux-maverick-source-2.6.35 linux-maverick-headers-2.6 (optional)
```

Upgrading the kernel may require you to reinstall various kernel modules such as video drivers, efs, and vm-tools. You can do so by running: apt-get dist-upgrade and then rebooting.

Supported Windows Platforms

Refer to the Release Notes for the latest updates to this information.

ElectricAccelerator Cluster Manager, Electric Make, and Electric Agent/EFS currently run on:

Platform	Notes

Windows 8
(32 or 64-bit)
	• NTFS is required for all Windows machines.
	• For 32-bit, only x86 is supported. For 64-bit, only x86-64 is supported. IA-64 (Itanium) is not supported.
	• To run Electric Make with Visual Studio on Windows, one of the following is required:
	  • Visual Studio 2005 SP1
	  • Visual Studio 2005 SP1 Redistributable Package (if you use Visual Studio .NET 2002 or 2003). Links are provided for your convenience:
	  32-bit
	  64-bit

Platform	Notes

Windows 7
(32 or 64-bit)
	• NTFS is required.
	• For 32-bit, only x86 is supported. For 64-bit, only x86-64 is supported. IA-64 (Itanium) is not supported.
	• To run Electric Make with Visual Studio on Windows, one of the following is required:
	  • Visual Studio 2005 SP1
	  • Visual Studio 2005 SP1 Redistributable Package (if you use Visual Studio .NET 2002 or 2003). Links are provided for your convenience:
	  32-bit
	  64-bit

Platform	Notes

Windows Server 2008 R2
(64-bit only)
	• NTFS is required.
	• For 64-bit, only x86-64 is supported. IA-64 (Itanium) is not supported.
	• 64-bit registry mirroring is supported only if you use a 64-bit Agent/EFS (running on 64-bit Server 2008 R2) with 64-bit eMake (running on any 64-bit Windows platform).
	• To run Electric Make with Visual Studio on Windows, one of the following is required:
	  • Visual Studio 2005 SP1
	  • Visual Studio 2005 SP1 Redistributable Package (if you use Visual Studio .NET 2002 or 2003). Links are provided for your convenience:
	  32-bit
	  64-bit

Windows Server 2003 R2
(32-bit only)
	• SP1 and SP2 required
	• NTFS is required.
	• For 32-bit, only x86 is supported.
	• To run Electric Make with Visual Studio on Windows, one of the following is required:
	  • Visual Studio 2005 SP1
	  • Visual Studio 2005 SP1 Redistributable Package (if you use Visual Studio .NET 2002 or 2003). Links are provided for your convenience:
	  32-bit
	  64-bit

Windows XP SP3
(32-bit only)
	• NTFS is required.
	• For 32-bit, only x86 is supported.
	• To run Electric Make with Visual Studio on Windows, one of the following is required:
	  • Visual Studio 2005 SP1
	  • Visual Studio 2005 SP1 Redistributable Package (if you use Visual Studio .NET 2002 or 2003). Links are provided for your convenience:
	  32-bit
	  64-bit

Supported Solaris Platforms

Refer to the Release Notes for the latest updates to this information.

ElectricAccelerator Electric Make and Electric Agent/EFS currently run on:

Platform	Notes
Solaris 10 (SPARC and x86)	• Agent and eMake only, no Cluster Manager support as of Accelerator v7.1 • Application is 32-bit only for Cluster Manager and eMake, and 64-bit only for agents. • For Solaris x86, eMake relies on several third-party open source libraries. The version of libxml2 that Electric Cloud builds in-house is different from the version that is available on SunFreeware.com. If you run eMake and get an error such as: `ld.so.1: emake: fatal: libxml2.so.2: version `LIBXML2_2.7.6' not found (required by file /opt/ecloud/i686_SunOS.5.10/bin/emake)` `ld.so.1: emake: fatal: libxml2.so.2: open failed: No such file or directory` then it is likely that your LD_LIBRARY_PATH environment variable contains a directory that includes a version of libxml2.so.2 that is incompatible with eMake. Unset LD_LIBRARY_PATH entirely to make eMake's rpath find all of its necessary libraries.

Supported Third-Party Build Tools

See System Requirements as well.

- GNU Make 3.80 and 3.81
- Microsoft NMAKE 7.x and 8.x
- Symbian Make (Windows)
- Visual Studio .NET 2002, 2003, 2005, 2008, 2010, 2012

  Note: Accelerator can support Visual Studio 6 workspaces if they are exported to NMAKE files. However, every time something changes in the workspace, an export must be done again.

- Rational ClearCase 7.1.1, 7.1.2, and 8.0 (exceptions detailed below)

  Note: ClearCase versions 7.1.2.9 and 8.0.0.5 contained a known missing lib issue. IBM fixed this issue in ClearCase versions 7.1.2.10 and 8.0.0.6 or 8.0.0.7. Electric Cloud recommends using one of those ClearCase versions with Accelerator.

  Accelerator supports building within ClearCase dynamic views and provides the Ledger feature, which can track files that change as a result of a change to the view's configuration specification. (See "ElectricAccelerator Ledger File" in the *ElectricAccelerator Electric Make Users Guide*.)

  Accelerator 7.1 in GNU Make 3.81 emulation mode was tested against the following ClearCase environment:

ClearCase server
RHEL 4.6
ClearCase version 7.0.1 (Wed May 30 17:04:58 EDT 2007) @(#) MVFS version 7.0.1.0 (Wed Apr 11 21:19:21 2007) built at $Date: 2008-10-24.19:30:48 (UTC) $

ClearCase client
RHEL 5.6
ClearCase version 8.0.0.06 (Fri Mar 08 10:35:58 EST 2013) (8.0.0.06.00_2013A.D130307) @(#) MVFS version 8.0.0.6 (Thu Feb 21 05:02:58 2013) built at $Date: 2013-02-21.10:13:08 (UTC) $

ClearCase client
RHEL 5.6
ClearCase version 7.1.2.10 (Fri Mar 08 11:01:39 EST 2013) (7.1.2.10.00_2013A.D130307) @(#) MVFS version 7.1.2.10 (Thu Feb 21 00:53:03 2013) built at $Date: 2013-05-16.22:01:13 (UTC) $

Note: Accelerator does not currently support ClearCase integration on Solaris x86. If you have need of this support, please contact your Electric Cloud sales representative. You can use ecclearcase_fake.so to provide information about your ClearCase setup through an ini file. See the “eMake’s “fake” interface for ClearCase” section in the *ElectricAccelerator Electric Make Users Guide*.

System Requirements

See Supported Third-Party Build Tools as well.

Topics:

- Hardware Requirements
- Web Browsers
- Port Usage
- Toolchain
- Databases
- Cygwin

Hardware Requirements

Electric Cloud recommends installing Cluster Manager, Electric Make, and Electric Agent components on independent machines so each component can run efficiently without competing for system resources.

The following are ***minimum*** hardware requirements for each machine where ElectricAccelerator components are installed (Agent, Cluster Manager, Electric Make):

- **Processor** - Pentium 4 (Linux and Windows) or SPARC or x86 (Solaris)
- **Installer disk space** - 300 MB, additional component disk space usage varies and depends on the size of your builds.

Agent Machines

Agents use system memory to cache small or unmodified files. For optimal performance, each host in the cluster must have enough memory for your link step, which is typically the largest single build step, plus another 200 MB.

If your builds have increased in size substantially since the system was originally delivered, and you have noticed a slowdown in build times, consider upgrading the host hardware.

	Memory	Disk Space
Recommended	2 - 3 GB per agent (if your build contains very large build steps)	Free disk space should be at least 3 - 4 times the size of a complete build (input and output).
Minimum	1 GB per agent (machine minimum of 2 GB)	

Cluster Manager Machine

	Memory	Disk Space
Minimum	2 GB	5 GB (to allow space for build log storage)

Electric Make Machine

	Memory	Disk Space
Minimum	2 GB	Free disk space should be at least 200 MB or equal to the size of your largest build.

Web Browsers

To connect to the Cluster Manager, you must use one of these web browsers:

- Microsoft Internet Explorer 7.0 or later
- Mozilla Firefox 3.5 or later

Port Usage

If you have a web server (for example, Apache, IIS) or other application currently using port 80 on the Cluster Manager host, you have the following options:

- Uninstall the web server.
- Disable the web server.
- Reconfigure the web server to use another port.

- Choose a non-default port (not 80) for Apache during Cluster Manager installation. The default Apache SSL port is 443.

By default, the Cluster Manager server uses port 8030 and secure port 8031.

Toolchain

Your toolchain should be installed on each agent host. You can virtualize the toolchain in some cases, but this can be tricky if the Windows registry is involved. Also, for performance reasons it is better to install the toolchain on each agent host. Many companies use Norton Ghost software to mirror all agent hosts.

Databases

Accelerator supports the following databases:

- MySQL 5.5.16
- MariaDB 5.5.31
- Oracle 11g Release 2
- MS SQL Server 2008

If you plan to use an Oracle or MS SQL database, you must create it before installing Accelerator.

Requirements for MySQL 5.5.16 on Linux

If you want to run MySQL 5.5.16 on a Linux Cluster Manager, ***it must have the 32-bit version of the following libraries***:

- libaio.so.1
- libncurses.so.5

You may need to add those libraries to the Cluster Manager machine.

RHEL example using a package manager:

```
yum install libaio.i686
yum install ncurses-libs.i686
```

RHEL example without a package manager:

```
up2date --nox --arch=i386 libaio
up2date --nox --arch=i386 ncurses
```

Ubuntu example using a package manager:

```
apt-get install libation
```

Location Limitations

Electric Cloud does ***not*** support storing your database on Network Appliance systems or NFS.

Issues that you may encounter include the following:

- Accelerator is unable to start the database because the database attempts to lock files that are already locked.
- If the database starts, you may encounter issues related to the database's UTF-8 support.

Cygwin

If you run builds on Windows in Cygwin environments, ensure you have a supported cygwin1.dll version installed:

- 1.7.9
- 1.7.7
- 1.5.25

Install the same version of Cygwin on all agent hosts and eMake machines . Mixing different Cygwin versions (for example, running v1.5 on an eMake machine and v1.7 on agents) is not supported. (In particular, Cygwin versions 1.5 and 1.7 default to different incompatible representations for symbolic links.)

Note: Cygwin version 1.7.x is supported for x = 7 or 9 only. There are known problems for other versions of 1.7.x.

By default, Cygwin 1.7.7 applies overly restrictive permissions to most directories. The permissions prevent the Administrators group from creating new subdirectories and may prevent the agent software from creating new directories to serve as mount points in order to reflect eMake client mount points.

On all agent hosts modify the permissions for the Cygwin installation directory and any other directories under which you want the agent software to dynamically create Cygwin mount points. For agent installations that use standard ECloudInternalUser* accounts, grant the “Administrators” group permission to “create folders / append data.” For custom agent users, grant permission for subdirectory creation to those agent users.

Component Compatibility Matrix

In general, Electric Cloud tries to maintain compatibility between older versions of Electric Make and newer versions of the Cluster Manager and the Agent. However, cross-version compatibility between other components is not ensured. Newer Electric Make version compatibility with older clusters is not ensured. The following matrix illustrates Electric Cloud’s compatibility philosophy:

	Newer Agent	Older Agent	Newer Cluster Manager	Older Cluster Manager	Newer Electric Make	Older Electric Make
Electric Make	Yes	No	Yes	No	No	No
Electric Agent	N/A	N/A	No	No	No	Yes
Cluster Manager	No	No	N/A	N/A	No	Yes

Electric Make and the Agent use a custom protocol to communicate with each other. Electric Cloud evolves this protocol in a backwards compatible manner so older Electric Make versions are compatible with newer Agent versions. This means you can upgrade the cluster anytime you want, then upgrade Electric Make installations at your leisure.

Electric Make also uses a custom protocol to communicate with remote instances of itself, for example, during remote parse jobs. This protocol is less tolerant than the Electric Make/Agent protocol. In general, it is best to have the same version of Electric Make on the cluster as you have on your build machine—Electric Cloud simplifies this because old versions of Electric Make are preserved on the cluster whenever the cluster is upgraded. For example, if your cluster originally had version 6.x installed and was upgraded to 7.x, the 6.x Electric Make version remains on the cluster—in a special directory in the install tree. You do not have to do anything to ensure this version is used when you run a build using a 6.x Electric Make version (the system handles that for you).

Note: Old versions are ***not*** installed! What is already there during upgrades is preserved. So, if you install 7.1 on a fresh agent machine, you will ***not*** be able to use a 7.0 Electric Make with that agent unless you obtain archives of that earlier eMake version from the Electric Cloud FTP site. See Installing the Backward-Compatible eMake Package on Agents.

Electric Make and the Cluster Manager use yet another protocol to communicate with each other. Again, Electric Cloud evolves this protocol so old Electric Make versions can work with newer Cluster Managers. For example, you should have no problem using a 6.x Electric Make version with a 7.x Cluster Manager.

Finally, the Agent uses a custom protocol to communicate with the Cluster Manager. Electric Cloud does not try to maintain compatibility between these two components—generally, if the agent version is "close" to the Cluster Manager version, it is compatible. However, you cannot use 7.x Agents with a 6.x Cluster Manager, or vice-versa.

Ensuring a Consistent Build Environment

Ideally, all Agent host machines should be the same, using the same operating system and hardware configuration. You may experience difficulties if you do not follow this recommendation. In addition, if any of your tool applications (compilers and so on) are installed locally, ensure they are all the same version and kept in sync.

Note: Ghosting with the Electric Agent already installed is not recommended.

Electric Cloud also recommends all hosts in a cluster have synchronized clocks, which is important particularly if you are running in the "shared" agent allocation policy.

- For UNIX machines, you can use ntpdate to sync clocks.
- For Windows machines, participation in a domain generally ensures synchronized clocks.

Planning for Grid Use

If you plan to use a host grid with Accelerator, here is a planning checklist:

- Choose a machine for the Cluster Manager.

  This machine will use the grid as any other client-only host. If you are upgrading an existing Cluster Manager, you must install the grid client-only product—and ensure it is 32-bit only.

  IMPORTANT: A Solaris x86 Cluster Manager cannot integrate with LSF.

- Choose hosts within the grid to run your builds.

  In general, these are machines you already use for software builds. Accelerator will use these machines in parallel to run builds, but machine availability is controlled by the grid management software. Electric Make allows you to select a subset of these machines using grid resource query strings. See the *ElectricAccelerator Electric Make Users Guide*, Electric Make Command-Line Options and Environment Variables section for information on the `emake --emake-resource` option.

- Choose hosts for Electric Make.

  If you will be running Electric Make within the grid, you must install Electric Make on those hosts.

- Select or create a user ID to run the Cluster Manager.

  IMPORTANT: This user ID must be set up with appropriate permissions and able to execute grid commands.

Note: If this is a UNIX user ID, you may need to run, for example, "`profile.lsf`" in your shell startup script.
If this is a Windows user ID, you may need to do the following:

1. Go to Services. (The process may vary depending on your Windows version.)
2. Right-click the ElectricAccelerator Cluster Manager service and go to Properties.
3. Click the Log On tab.
4. Select This account (Local System account would normally be selected).
5. Type-in the user ID and password for the user account.

- Select a queue for Accelerator.

  This queue must be allowed to submit jobs that request exclusive host use (`bsub -x` option) or a queue that specifies `HJOB_LIMIT=1`. Set up the queue so jobs can be submitted on any platform where you intend to use Electric Make. For example, if your software builds are on Linux and Windows, the queue must allow jobs on either platform.

For additional information, see:

- Enabling Grid Host Manager
- Configuring LSF

Checksum Utility

An MD5 checksum file is available on the Electric Cloud FTP site. If you choose to verify that the install files are intact and unaltered from their original form and content after you download them, download the corresponding MD5 checksum file also.

MD5 utilities are available for supported operating systems.

- On Linux, verify with `md5sum --check md5.txt`
- Most Linux installations provide an md5sum command for calculating MD5 message digests.
- An MD5 utility for Windows can be downloaded at http://www.fourmilab.ch/md5/.

Chapter 3: Installation

You can install the software using a GUI or an interactive command-line interface (for Linux and Solaris), or by using a "silent" installation. Not all instructions are the same for each platform. Follow the instructions carefully for your particular platform. Read the procedures thoroughly ***before*** attempting to install, configure, or uninstall any component.

Topics:

- Before You Install ElectricAccelerator
- Installing ElectricAccelerator on Linux/Solaris
- Installing ElectricAccelerator on Windows
- Silent Install
- Migrating an eMake-only Installation to a Different Machine
- Installing the Backward-Compatible eMake Package on Agents

Before You Install ElectricAccelerator

Cluster Manager and Agent Install Location Limitations

Electric Cloud does not support installation of Cluster Manager or Agents on the following:

- NFS
- CIFS
- Samba shares

32-bit vs. 64-bit Information

- The installer for Windows and Linux provides both 32-bit and 64-bit versions of Electric Make. (For Solaris, only 32-bit is available.)
- The agent installer automatically determines whether to install the 32-bit or 64-bit agent based on the machine architecture. No user action is required.
- The Accelerator installer automatically sets the path to 32-bit. To use 64-bit, you must edit the environment variable path to include the 64-bit bin location *before* the 32-bit location.
- 64-bit executables are installed in a subdirectory of the install location, `<installDir>/64/bin`.

For example:
If you install into the Windows `C:\ECLoud\i686_win32` directory, the 64-bit executables are in the `C:\ECloud\i686_win32\64\bin` directory.
For Linux, the directory is `/opt/ecloud/i686_Linux/64/bin`.

Installing ElectricAccelerator on Linux/Solaris

Installing any component completely replaces the current installation on that machine. Only one ElectricAccelerator installation can exist on a machine and it is whatever combination of components were chosen from the most recent install. For example, installing the Cluster Manager on a machine with the Electric Agent, results in a machine with Cluster Manager only, not a machine with the Electric Agent and Cluster Manager.

If you plan to install Electric Agent on a machine that has ElectricAccelerator Developer Edition installed, Electric Cloud recommends doing a full uninstall of ElectricAccelerator Developer Edition first.

Topics:

- Additional Linux Install Information
- Installing Cluster Manager
- Installing Agent/EFS
- Installing Electric Make
- Path Settings
- Silent Install
- Additional Install Options

Additional Linux/Solaris Install Information

Antivirus Software

Some antivirus software may affect the installer. Turn antivirus software off during installation. If antivirus software is running when you start the installer, you may receive an error dialog. The antivirus software may have reported Accelerator files as a virus and removed them from the temp location. As a workaround, turn off the antivirus software and rerun the installer.

If Symantec AntiVirus software is installed, disable it before installation to avoid serious filesystem conflicts. If Symantec AntiVirus cannot be disabled, put the ECloud directory in an exclusion list or disable the AutoProtect feature. If you need more information, contact Electric Cloud technical support.

umask

Electric Cloud recommends umask 0022. Do not set a different umask during installation.

Absolute Path for Solaris

On Solaris systems, you must use an absolute path for the installation directory.

mysqlcheck

By default, mysqlcheck looks for my.cnf in `/etc` and `~/.my.cnf` for configuration information. The installer does not currently copy my.cnf to `/etc`, causing mysqlcheck to fail. You must add the following to the mysqlcheck command line parameter: `--defaults-file=<path to my.cnf>`
For example: `mysqlcheck --defaults-file=/opt/ecloud/i686_Linux/mysql/my.cnf -o ecloud`

Installing Cluster Manager

Note: Starting with v7.1, Accelerator does not support installing the Cluster Manager on Solaris.

The Cluster Manager is a web server and can be installed on any networked machine with access to Electric Make and the cluster hosts, or on the main build machine where Electric Make is installed.

Install only one copy of the Cluster Manager, regardless of the number of hosts in the cluster or the number of machines with Electric Make installed. Electric Make is installed with the Cluster Manager by default.

If you are upgrading a previously installed ElectricAccelerator version, follow the upgrade instructions *before* attempting to install the new version.

Note: Electric Cloud no longer bundles MySQL as the default local database for ElectricAccelerator. Starting with ElectricAccelerator v7.1, Electric Cloud bundles MariaDB and installs it as the default local database on the Cluster Manager. When you perform a Cluster Manager upgrade, MariaDB replaces the local MySQL database (if applicable) and migrates all information from it.

MariaDB is an open-source database that is fully compatible with MySQL. As open-source software, there is no enterprise license support model available with MariaDB. If using an open-source database is of concern to your organization, you continue to have the option to use any of the following databases as a replacement: MySQL, Oracle, or SQL Server. See the online help topic “Database Configuration” for information about configuring Accelerator to use an external database.

Using the GUI to Install the Cluster Manager

1. Log in as root.
2. Double-click the `ElectricAccelerator-<version>` installer file to start installation.

   Note: It may take a few minutes to extract the installation packages to your machine before you see the installation wizard.

3. When the Welcome screen appears, click **Next**.
4. On the Setup Type screen, select Cluster Manager (Electric Make also installs automatically with the Cluster Manager) and click **Next**.
5. On the Choose Destination Location screen, accept the default installation directory or browse to select an alternative directory.

   Notes:

   - Avoid selecting an alternative directory that includes spaces in the name. Spaces can create problems when configuring connections with other command-line-based components.
   - If you plan to install Electric Make in a non-default location, use the same location on all agent hosts and on the Cluster Manager host.

   Click **Next**.

6. If the installer cannot locate a PDF viewer, it will prompt you to browse for one. If you wish, browse for a PDF viewer to use. Click **Next**.
7. On the Base Setup screen, choose whether you want to install Electric Runner client applications (which enable you to run Accelerator administration tools) and click **Next**.
8. On the Cluster Manager screen, enter Cluster Manager configuration options:

   - Accept the default ports or type-in alternative port numbers if the defaults are already in use. For example, IIS and Apache also use port 80 by default.
   - Accept the default to keep old log files or select the checkbox to remove old log files.

- Accept the default [checked checkbox] to leave web server log rotation turned on or clear the checkbox to turn off log rotation. **IMPORTANT -** Log rotation may affect Cluster Manager performance.
- Select the type of database to use.

Note: If you do not use MariaDB, the database must already exist and you must be able to connect to it. The installer does ***not*** validate database connections.

Click **Next**.

9. On the next Cluster Manager screen, continue to enter Cluster Manager configuration options:
   - You can change the Cluster Manager admin password or accept the default password (*changeme*).
   - You do not need to migrate the existing database or back up the local database if this is your first Cluster Manager installation.
   - Fill-in the user name or choose a user that the Cluster Manager service should run as or leave blank. Choose a password for the Cluster Manager service user or leave blank.

   Note: If you plan to use a host grid, the user selected for the Cluster Manager Service User must be fully enabled to start and stop grid jobs.

   Click **Next**.
10. On the Cluster Manager Database screen, type-in the Cluster Manager database port number. For MS SQL, use port 1433.

    If you did not select MariaDB as your database type, the following additional fields are available:
    - Database Host/IP Address
    - Database User
    - Database Password
    - Database Name (**IMPORTANT:** Do ***not*** use a . (period) in the database name. A period in the name will prevent the Cluster Manager from starting.)

    Click **Next**.
11. When the Start Copying Files screen appears, click **Next**.
12. The Installing screen displays while the installation proceeds. When installation is finished, the Complete screen displays. Click **Finish**.

    Installation is complete. The installation log file is in the install directory's root by default.

Note: If Apache fails to start properly after the Cluster Manager installs, reboot the system.

Using an Interactive Command-Line to Install the Cluster Manager

1. Log in as root.
2. Run `chmod +x` on the installer (`ElectricAccelerator-<version>`) to ensure it is executable.
3. Run `./<installer filename> --mode console` to start the installation.
4. When the welcome message displays, enter **3** to select the **Cluster Manager** package. Press Enter.

5. After choosing option 3, provide the following Cluster Manager configuration information. Accept the defaults or type-in alternatives.

   Notes:

   - Turning on log rotation may affect Cluster Manager performance.
   - If the installer cannot locate a PDF viewer, it will prompt you to provide a path to one.
   - If you plan to use a host grid, the user selected for the Cluster Manager Service User must be fully enabled to start and stop grid jobs.

```
Where do you want to install ElectricAccelerator? [opt/ecloud]
Install Electric Runner client apps? [y/n]Cluster Manager HTTP port: [80]
Cluster Manager HTTPS port: [443]Cluster Manager Accelerator Server port:
[8030]
Cluster Manager Secure Accelerator Server port: [8031]
Remove old logs [y/N]
Rotate Cluster Manager logs [y/N]
Database Type: [MariaDB]
Cluster Manager Service User: [eacmuser]
Cluster Manager admin password: [changeme]
Migrate existing database [y/N]
Backup local MariaDB database [y/N]
Database port: [0]
Database Host/IP address: [localhost]
Database User: [ecloud]
Database Password: [ecloud]
Database Name: [ecloud]
```

   IMPORTANT: Do not use a . (period) in the database name. A period in the name will prevent the Cluster Manager from starting.

   The installer installs Cluster Manager using the configuration details you entered, followed by "Installation complete" when the install completes.

The installation log file is in the install directory's root, `/opt/ecloud` by default.

Note: Linux operating systems contain a 1024 default file descriptor limit for each process. However, this does not mean 1024 builds can be run concurrently. Depending on the size of your builds, you may be able to run only 300-400 builds concurrently because of other file descriptor activity and the collection activity between eMake and the Cluster Manager. If you must run more than 300-400 builds simultaneously, increase the default file descriptor limit.

If Apache fails to start properly after a fresh Cluster Manager install, reboot the system.

Installing Agent/EFS

Electric Agent/EFS software must be installed on each machine in the cluster. Electric File System (EFS) and Electric Agent (collectively, the Agent) are installed simultaneously on each host machine during Electric Agent/EFS installation. Multiple instances of Electric Agent/EFS pairings can be set up to run in parallel on a host. For example, if you have a multiprocessor machine, you may want to set up an Agent for each processor. More information on running multiple Agents is available in the ElectricAccelerator online help.

If you are upgrading a previously installed ElectricAccelerator version, follow the upgrade instructions *before* attempting to install the new version.

Your tool chain should be installed on each agent host. You can virtualize the tool chain in some cases, but this can be tricky especially when the Windows registry is involved. For performance reasons, it is better to install

the tool chain on each agent host. Many companies use a “ghosting” software for this purpose. **IMPORTANT:** “Ghosting” with the Electric Agent already installed is **not** recommended.

Installing Electric Agent/EFS in a Grid Environment

You must install Electric Agent/EFS on each machine in the grid that ElectricAccelerator will use. The actual installation process for Electric Agent/EFS is the same whether or not you use a grid—*except*, you must answer “yes” **(y)** to any prompt that asks whether or not you plan to use a grid environment, such as LSF.

Note: After Agent/EFS is installed, it remains dormant until the grid management software schedules an agent connect script to run on that host.

Note: Agent/EFS builds a kernel module during installation, so you may need to take this into consideration.

Using the GUI to install Electric Agent/EFS

1. Log in as root.
2. Double-click the `ElectricAccelerator-<version>` installer file to start installation.

   Note: It may take a few minutes to extract the installation packages to your machine before you see the installation wizard.

3. When the Welcome screen appears, click **Next** to continue to the Setup Type screen.
4. On the Setup Type screen, select **Electric Agent**, and click **Next**.
5. On the Choose Destination Location screen, accept the default installation directory or browse to select an alternative directory.

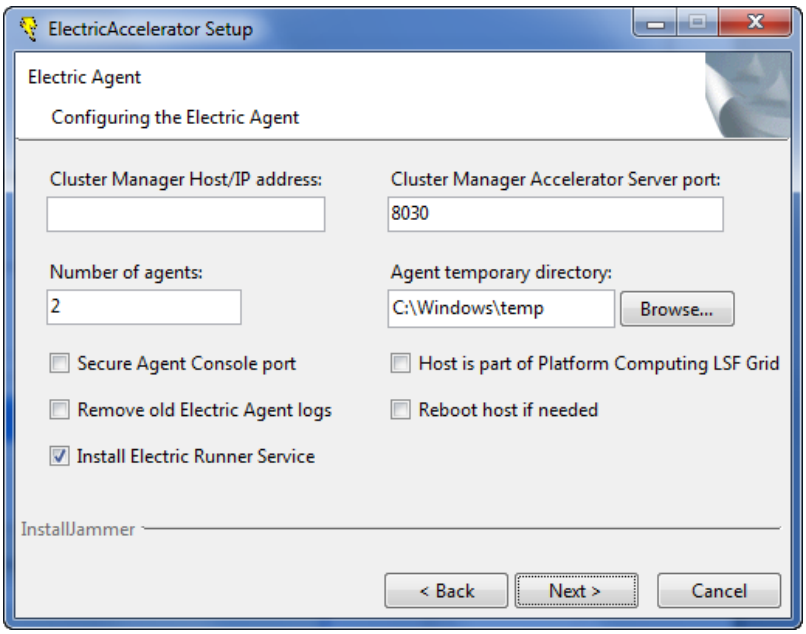
   Note: Avoid selecting an alternative directory that includes spaces in the name. Spaces can create problems when configuring connections with other command-line-based components.

   Click **Next**.

6. If the installer cannot locate a PDF viewer, it will prompt you to browse for one. If you wish, browse for a PDF viewer to use, and click **Next**.
7. On the Base Setup screen, choose whether you want to install Electric Runner client applications, and click **Next**.
8. On the Electric Agent screen, enter options for the locally installed Electric Agent/EFS.

- Type-in the Cluster Manager host name or IP address.
- Accept the default Cluster Manager port 8030 or type-in the alternative port you supplied during the Cluster Manager server installation.
- Enter the number of Agents to run—the installer calculates the default number of Agents based on the effective number of CPUs on the machine. Default=1, if one CPU is present. If more than one CPU is present, the default value is the number of effective CPUs. The maximum number of Agents allowed on one 32-bit machine is 32; the maximum for one 64-bit machine is 64.
- Accept the default agent temporary directory, or choose an alternative.
- Accept the default for the Secure Agent Console Port checkbox if you do not plan to use the secure port, or select the checkbox if you want to use the secure port.
- If this agent machine is part of the grid, select the Host is part of Platform Computing LSF Grid checkbox. (For this field, the term “LSF” refers to all grid engines.
- Accept the default to keep all existing log files or select the checkbox to Remove existing log files.
- Select the checkbox to Reboot host if needed.
- Accept the default to install the Electric Runner service, or clear the checkbox to not install this service.

  Note: eRunner is a “helper” service normally installed on all agent machines. This service listens for commands to execute on agents, facilitating administration tasks. For example, with eRunner you can upgrade all agents at the same time, remotely.



Click **Next**.

9. (Windows only) Select the user accounts you will use to run agents. Accept the default ECloud Internal User or select Existing User. If you select Existing User, click **Add User** to add a user

account. Click **Next**.

10. On the Electric Runner Server screen, enter options for the Electric Runner server.
    - Accept the default Electric Runner Server port, or type-in an alternative.
    - Accept the default to leave old Electric Runner logs, or select the checkbox to remove old logs.

    Click **Next**.
11. When the Start Copying Files screen appears, click **Next**.
12. The Installing screen displays while the installation proceeds. When installation is finished, the Complete screen displays. Click **Finish**.

    Installation is complete. The installation log file is in the install directory's root by default.

Using an Interactive Command-Line Install Electric Agent/EFS

1. Log in as root.
2. Run `chmod +x` on the installer (`ElectricAccelerator-<version>`) to ensure it is executable.
3. Run `./<installer filename> --mode console` to start the installation.
4. When the welcome message displays, enter **1** to select the **Electric Agent/EFS** package. Press Enter.
5. After choosing option 1, provide Electric Agent/EFS configuration information. Accept the defaults or type-in alternatives.

Note: If the installer cannot locate a PDF viewer, it will prompt you to provide a path to one.

- Use the Cluster Manager default server listening port 8030, or type-in the alternative port if you did not accept the default when you installed Cluster Manager.
- Select the number of Agents to run.
  The installer calculates the default number of Agents based on the number of effective CPUs on the machine. For example, you may have one Agent per virtual CPU, a single dual-core CPU supports two Agents, three dual-cores support six Agents. The maximum number of Agents allowed on one 32-bit machine is 32; the maximum for one 64-bit machine is 64.
- Specify the Agent temporary directory. The default is `/tmp`. If you specify a different directory, it must already exist, otherwise the temporary directory defaults to `/tmp`.
- Regarding host grid usage
  In a grid environment, such as LSF, Agents are not bound to a Cluster Manager during installation. Instead, the grid management software tells the agents which Cluster Manager to talk to—which means the Cluster Manager host name or port number you may have specified will not be used.
- Decide whether to install Electric Runner.
  eRunner is a "helper" daemon normally installed on all agent machines. This daemon listens for commands to execute on agents, facilitating administration tasks. For example, with eRunner you can upgrade all agents at the same time, remotely.

The installer installs Electric Agent/EFS using the configuration details you provided, followed by "Installation complete" when the install completes.

The installation log file is in the install directory's root, `/opt/ecloud` by default.

Installing Electric Make

Electric Make can be installed on a networked machine or on a shared file server accessible to other build machines. Wherever Electric Make is installed, it must have access to the Cluster Manager machine and Agents on the cluster host machines. After Electric Make is installed, configure Electric Make for your build environment and enable communication with Cluster Manager. For information on enabling communication with Cluster Manager, see "Configuring ElectricAccelerator" in the *ElectricAccelerator Electric Make Users Guide*.

If you are upgrading a previously installed ElectricAccelerator version, follow the upgrade instructions *before* attempting to install the new version.

Using the GUI to Install Electric Make

1. Log in as root.
2. Double-click the `ElectricAccelerator-<version>` installer file to start installation.

   Note: It may take a few minutes to extract the installation packages to your machine before you see the installation wizard.

3. When the Welcome screen appears, click **Next** to continue to the Setup Type screen.
4. On the Setup Type screen, select **Electric Make**, and click **Next**.
5. On the Choose Destination Location screen, accept the default installation directory or browse to select an alternative directory. Ensure the Electric Make path matches the path used for the installed Agent.

   Note: Avoid selecting an alternative directory that includes spaces in the name. Spaces can create problems when configuring connections with other command-line-based components.

   Click **Next**.

6. If the installer cannot locate a PDF viewer, it will prompt you to browse for one. If you wish, browse for a PDF viewer to use, and click **Next**.
7. On the Base Setup screen, choose whether you want to install Electric Runner client applications, and click **Next**.
8. When the Start Copying Files screen appears, click **Next**.
9. The Installing screen displays while the installation proceeds. When installation is finished, the Complete screen displays. Click **Finish**.

   Installation is complete. The installation log file is in the install directory's root, by default.

Using an Interactive Command-Line to Install Electric Make

1. Log in as root.
2. Run `chmod +x` on the installer (`ElectricAccelerator-<version>`) to ensure it is executable.
3. Run `./<installer filename> --mode console` to start the installation.
4. When the welcome message displays, enter **2** to select the **Electric Make** package. Press Enter.
5. After choosing option 2, provide Electric Agent/EFS configuration information. Accept the defaults or type-in alternatives.

Notes:

- If you install Electric Make in a non-default location, ensure the same location is used for all agent hosts and the Cluster Manager host.
- If the installer cannot locate a PDF viewer, it will prompt you to provide a path to one.

The installer installs Electric Make using the configuration details you provided, followed by "Installation complete" when the install completes.

The installation log file is in the install directory's root, `/opt/ecloud` by default.

Path Settings

All ElectricAccelerator components are installed in the `/opt/ecloud` directory.

Scripts to add the necessary environment variables are installed in `/opt/ecloud/<arch>/conf`. arch is:

- `i686_Linux` for Linux, or
- `sun4u_SunOS` for Solaris SPARC or `i686_SunOS.5.10` for Solaris x86

The scripts are called `ecloud.bash.profile` (for bash shells) or `ecloud.csh.profile` (for csh). You can source the appropriate file in your shell to ensure your PATH includes ElectricAccelerator libraries.

Installing ElectricAccelerator on Windows

Installing any component completely replaces the current installation on that machine. Only one ElectricAccelerator installation can exist on a machine and it is whatever combination of components were chosen from the most recent install. For example, installing the Cluster Manager on a machine with the Electric Agent, results in a machine with Cluster Manager only, not a machine with the Electric Agent and Cluster Manager.

If you plan to install Electric Agent on a machine that has ElectricAccelerator Developer Edition installed, Electric Cloud recommends doing a full uninstall of ElectricAccelerator Developer Edition first.

Topics:

- Additional Windows Install Information
- Installing Cluster Manager
- Installing Electric Agent/EFS
- Installing Electric Make
- Silent Install
- Additional Install Options

Additional Windows Install Information

Installer Activity

- For Windows Server 2008 R2, the installer automatically does the following:
  - Disables the Windows error reporting service.
  - Sets HKLM\SYSTEM\CurrentControlSet\Control\FileSystem\NtfsDisableLastAccessUpdate to 0. The default value for Windows Server 2008 R2 is 1.
  - Disables User Account Control (UAC) for 64-bit versions. Disabling UAC avoids popup windows for applications that require administrator privileges. If UAC is enabled, application registry access is redirected to each user's virtual store, even if it runs under the Administrator account.

- If you invoke the installer from a network drive, you may receive an Unknown Publisher security warning. You can disregard this warning and proceed with installation.

Installing from the Cygwin Shell

If you choose to run the installer from the Cygwin shell, be advised of the following:

- Before running the installer, disable UAC, or start the Cygwin shell using the Run as Administrator menu item (right-click the Cygwin desktop icon). Running the installer with UAC enabled may result in a "permission denied" error. This is applicable for all Windows versions that use UAC (Windows 7 and Windows Server 2008 R2).
- You may encounter issues when running the installer from the Cygwin `/tmp` directory. Electric Cloud recommends running the installer from a different directory. This is applicable for all Windows versions.

Antivirus Software

Some antivirus software may affect the installer. Turn antivirus software off during installation. If antivirus software is running when you start the installer, you may receive an error dialog. The antivirus software may have reported Accelerator files as a virus and removed them from the temp location. As a workaround, turn off the antivirus software and rerun the installer.

If Symantec AntiVirus software is installed, disable it before installation to avoid serious filesystem conflicts. If Symantec AntiVirus cannot be disabled, put the ECloud directory in an exclusion list or disable the AutoProtect feature. If you need more information, contact Electric Cloud technical support.

umask

Electric Cloud recommends umask 0022. Do not set a different umask during installation.

Installing from a Network Drive

If you invoke the installer from a network drive, you may receive an Unknown Publisher security warning. You can disregard this warning and proceed with installation.

mysqlcheck

By default, mysqlcheck looks for my.cnf in `/etc` and `~/.my.cnf` for configuration information. The installer does not currently copy my.cnf to `/etc`, causing mysqlcheck to fail. You must add the following to the mysqlcheck command line parameter: `--defaults-file=<path to my.cnf>`
For example: `mysqlcheck --defaults-file=/opt/ecloud/i686_Linux/mysql/my.cnf -o ecloud`

Installing Cluster Manager

The Cluster Manager is a web server and can be installed on any networked machine with access to Electric Make and the cluster hosts, or on the main build machine where Electric Make is installed.

Install only one copy of the Cluster Manager, regardless of the number of hosts in the cluster or the number of machines with Electric Make installed. Electric Make is installed with the Cluster Manager by default.

If you are upgrading a previously installed ElectricAccelerator version, follow the upgrade instructions *before* attempting to install the new version.

Note: Electric Cloud no longer bundles MySQL as the default local database for ElectricAccelerator. Starting with ElectricAccelerator v7.1, Electric Cloud bundles MariaDB and installs it as the default local database on the Cluster Manager. When you perform a Cluster Manager upgrade, MariaDB replaces the local MySQL database (if applicable) and migrates all information from it.

MariaDB is an open-source database that is fully compatible with MySQL. As open-source software, there is no enterprise license support model available with MariaDB. If using an open-source database is of concern to your organization, you continue to have the option to use any of the following databases as a replacement: MySQL, Oracle, or SQL Server. See the online help topic “Database Configuration” for information about configuring Accelerator to use an external database.

Note: On Windows Server 2008 R2, you may encounter a “Not Responding” error during install. Electric Cloud recommends waiting for up to 10 minutes for the install to continue normally.

Installation Procedure

1. Log in as Administrator (You *must* be a member of the Administrator group—Administrator privileges are not sufficient.)
   If you are running *rdp* on the target Cluster Manager server, ensure *rdp* is in installation mode: `change user/install`.
2. Double-click the `ElectricAccelerator-<version>` installer file to start installation. (For systems running Windows Server 2008 or later, the administrator user must right-click the installer and select Run as administrator.)

   Notes:

   - It may take a few minutes to extract the installation packages to your machine before you see the installation wizard. Also, during installation, if you see a security alert pop-up, click **unblock** and continue.
   - ElectricAccelerator requires the Microsoft Visual C++ 2005 SP1 Redistributable. If it is already installed, select the checkbox.

   Click **Next** to continue.
3. When the Welcome screen appears, click **Next**.
4. On the Setup Type screen, select Cluster Manager (Electric Make also installs automatically with the Cluster Manager) and click **Next**.
5. On the Choose Destination Location screen, accept the default installation directory or browse to select an alternative directory.

   Notes:

   - Avoid selecting an alternative directory that includes spaces in the name. Spaces can create problems when configuring connections with other command-line-based components.
   - If you plan to install Electric Make in a non-default location, use the same location on all agent hosts and on the Cluster Manager host.

   Click **Next**.
6. If the installer cannot locate a PDF viewer, it will prompt you to browse for one. If you wish, browse for a PDF viewer to use. Click **Next**.
7. On the Base Setup screen, choose whether you want to install Electric Runner client applications (which enable you to run Accelerator administration tools) and click **Next**.
8. On the Cluster Manager screen, enter Cluster Manager configuration options:

- Accept the default ports or type-in alternative port numbers if the defaults are already in use. For example, IIS and Apache also use port 80 by default.
- Accept the default to keep old log files or select the checkbox to remove old log files.
- Accept the default [checked checkbox] to leave web server log rotation turned on or clear the checkbox to turn off log rotation. **IMPORTANT -** Log rotation may affect Cluster Manager performance.
- Select the type of database to use.

Note: If you do not use MariaDB, the database must already exist and you must be able to connect to it. The installer does ***not*** validate database connections.

Click **Next**.

9. On the next Cluster Manager screen, continue to enter Cluster Manager configuration options:
   - You can change the Cluster Manager admin password or accept the default password (*changeme*).
   - You do not need to migrate the existing database or back up the local database if this is your first Cluster Manager installation.
   - Fill-in the user name or choose a user that the Cluster Manager service should run as or leave blank. Choose a password for the Cluster Manager service user or leave blank.

   Note: If you plan to use a host grid, the user selected for the Cluster Manager Service User must be fully enabled to start and stop grid jobs.

   Click **Next**.

10. On the Cluster Manager Database screen, type-in the Cluster Manager database port number. For MS SQL, use port 1433.

    If you did not select MariaDB as your database type, the following additional fields are available:

    - Database Host/IP Address
    - Database User
    - Database Password
    - Database Name (**IMPORTANT:** Do ***not*** use a . (period) in the database name. A period in the name will prevent the Cluster Manager from starting.)

    Click **Next**.

11. When the Start Copying Files screen appears, click **Next**.
12. The Installing screen displays while the installation proceeds. When installation is finished, the Complete screen displays. Click **Finish**.

    Installation is complete. The installation log file is in the install directory's root by default.

Note: If Apache fails to start properly after the Cluster Manager installs, reboot the system.

Installing Electric Agent/EFS

Electric Agent/EFS software must be installed on each machine in the cluster. Electric File System (EFS) and Electric Agent (collectively, the Agent) are installed simultaneously on each host machine during Electric Agent/EFS installation. Multiple instances of Electric Agent/EFS pairings can be set up to run in parallel on a

host. For example, if you have a multiprocessor machine, you may want to set up an Agent for each processor. More information on running multiple Agents is available in the ElectricAccelerator online help.

If you are upgrading a previously installed ElectricAccelerator version, follow the upgrade instructions *before* attempting to install the new version.

Your tool chain should be installed on each agent host. You can virtualize the tool chain in some cases, but this can be tricky especially when the Windows registry is involved. For performance reasons, it is better to install the tool chain on each agent host. Many companies use a "ghosting" software for this purpose. **IMPORTANT:** "Ghosting" with the Electric Agent already installed is **not** recommended.

Installing Electric Agent/EFS in a Grid Environment

You must install Electric Agent/EFS on each machine in the grid that ElectricAccelerator will use. The actual installation process for Electric Agent/EFS is the same whether or not you use a grid—*except*, you must answer "yes" **(y)** to any prompt that asks whether or not you plan to use a grid environment, such as LSF.

Note: After Agent/EFS is installed, it remains dormant until the grid management software schedules an agent connect script to run on that host.

IMPORTANT: If Symantec AntiVirus software is installed on your intended agent machines, disable it before installing Electric Agent/EFS to avoid serious filesystem conflicts. If Symantec AntiVirus cannot be disabled, put `C:\ECloud` in an exclusion list or disable the AutoProtect feature. Electric Cloud generally considers agent machines safe because they are internal machines residing behind a firewall. If you need more information, contact Electric Cloud technical support.

Installation Procedure

1. Log in as Administrator (You *must* be a member of the Administrator group—Administrator privileges are not sufficient.)
   If you are running *rdp* on the target Cluster Manager server, ensure *rdp* is in installation mode: `change user/install`.
2. Double-click the `ElectricAccelerator-<version>` installer file to start installation. (For systems running Windows Server 2008 or later, the administrator user must right-click the installer and select Run as administrator.)

   Notes:

   - It may take a few minutes to extract the installation packages to your machine before you see the installation wizard. Also, during installation, if you see a security alert pop-up, click **unblock** and continue.
   - ElectricAccelerator requires the Microsoft Visual C++ 2005 SP1 Redistributable. If it is already installed, select the checkbox.

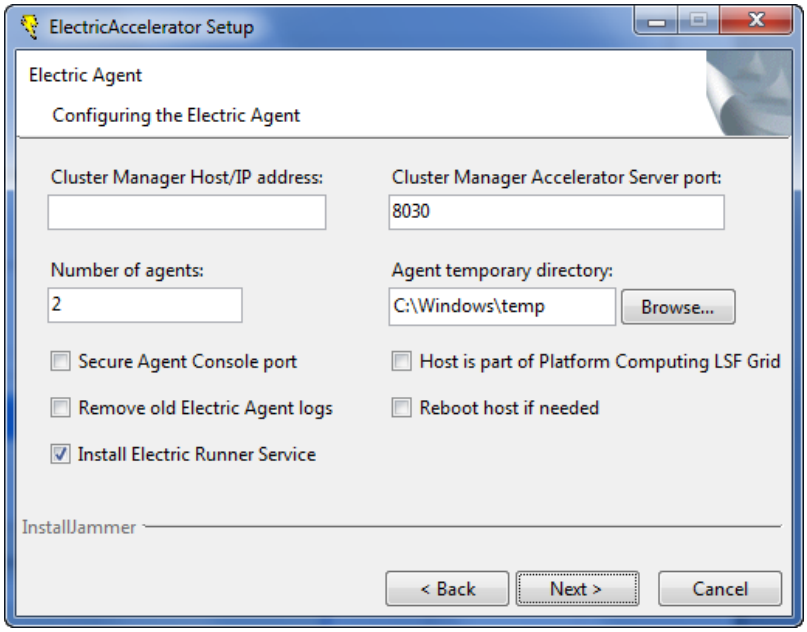
   Click **Next** to continue.
3. When the Welcome screen appears, click **Next** to continue to the Setup Type screen.
4. On the Setup Type screen, select **Electric Agent**, and click **Next**.
5. On the Choose Destination Location screen, accept the default installation directory or browse to select an alternative directory.

   Note: Avoid selecting an alternative directory that includes spaces in the name. Spaces can create problems when configuring connections with other command-line-based components.

   Click **Next**.

6. If the installer cannot locate a PDF viewer, it will prompt you to browse for one. If you wish, browse for a PDF viewer to use, and click **Next**.
7. On the Base Setup screen, choose whether you want to install Electric Runner client applications, and click **Next**.
8. On the Electric Agent screen, enter options for the locally installed Electric Agent/EFS.
    - Type-in the Cluster Manager host name or IP address.
    - Accept the default Cluster Manager port 8030 or type-in the alternative port you supplied during the Cluster Manager server installation.
    - Enter the number of Agents to run—the installer calculates the default number of Agents based on the effective number of CPUs on the machine. Default=1, if one CPU is present. If more than one CPU is present, the default value is the number of effective CPUs. The maximum number of Agents allowed on one 32-bit machine is 32; the maximum for one 64-bit machine is 64.
    - Accept the default agent temporary directory, or choose an alternative.
    - Accept the default for the Secure Agent Console Port checkbox if you do not plan to use the secure port, or select the checkbox if you want to use the secure port.
    - If this agent machine is part of the grid, select the Host is part of Platform Computing LSF Grid checkbox. (For this field, the term “LSF” refers to all grid engines.
    - Accept the default to keep all existing log files or select the checkbox to Remove existing log files.
    - Select the checkbox to Reboot host if needed.
    - Accept the default to install the Electric Runner service, or clear the checkbox to not install this service.

Note: eRunner is a “helper” service normally installed on all agent machines. This service listens for commands to execute on agents, facilitating administration tasks. For example, with eRunner you can upgrade all agents at the same time, remotely.



Click **Next**.

9. (Windows only) Select the user accounts you will use to run agents. Accept the default ECloud Internal User or select Existing User. If you select Existing User, click **Add User** to add a user account. Click **Next**.
10. On the Electric Runner Server screen, enter options for the Electric Runner server.
    - Accept the default Electric Runner Server port, or type-in an alternative.
    - Accept the default to leave old Electric Runner logs, or select the checkbox to remove old logs.

    Click **Next**.
11. When the Start Copying Files screen appears, click **Next**.
12. The Installing screen displays while the installation proceeds. When installation is finished, the Complete screen displays. Click **Finish**.

    Installation is complete. The installation log file is in the install directory’s root by default.

Agent Communication and Firewall Settings

Because the ports used to communicate with agents are dynamic, you may experience issues on Windows agent hosts depending on your firewall settings.

If you experience installation issues, exclude the following programs from the block list during agent host installation:

- `C:\ecinst\ecinstconf.exe`
- `C:\ecinst\ecloduninst.exe`

- `C:\ecinst\fsize.exe`
- `C:\ecinst\Setup.exe`
- `C:\ecinst\SetupErunnerTmp.exe`
- `C:\ERunnerTmp\i686_win32\bin\erunnerdtmp.exe`

These programs exist only during Accelerator installation. You might find it easier to disable communication restrictions with the agent host during agent installation. If you do not want to do this:

1. Create folders `C:\ecinst` and `C:\ErunnerTmp\i686_win32\bin`.
2. Move files with those specified names into these directories.
3. Exclude these programs from the “blocked programs” list.
4. Delete the directories `C:\ecinst` and `C:\ErunnerTmp`.
5. Install agents.

Installing Electric Make

Electric Make can be installed on a networked machine or on a shared file server accessible to other build machines. Wherever Electric Make is installed, it must have access to the Cluster Manager machine and Agents on the cluster host machines. After Electric Make is installed, configure Electric Make for your build environment and enable communication with Cluster Manager. For information on enabling communication with Cluster Manager, see “Configuring ElectricAccelerator” in the *ElectricAccelerator Electric Make Users Guide*.

If you are upgrading a previously installed ElectricAccelerator version, follow the upgrade instructions *before* attempting to install the new version.

Installation Procedure

1. Log in as Administrator (You *must* be a member of the Administrator group—Administrator privileges are not sufficient.)
   If you are running *rdp* on the target Cluster Manager server, ensure *rdp* is in installation mode: `change user/install`.
2. Double-click the `ElectricAccelerator-<version>` installer file to start installation. (For systems running Windows Server 2008 or later, the administrator user must right-click the installer and select Run as administrator.)

   Notes:

   - It may take a few minutes to extract the installation packages to your machine before you see the installation wizard. Also, during installation, if you see a security alert pop-up, click **unblock** and continue.
   - ElectricAccelerator requires the Microsoft Visual C++ 2005 SP1 Redistributable. If it is already installed, select the checkbox.

   Click **Next** to continue.
3. When the Welcome screen appears, click **Next** to continue to the Setup Type screen.
4. On the Setup Type screen, select **Electric Make**, and click **Next**.

5. On the Choose Destination Location screen, accept the default installation directory or browse to select an alternative directory. Ensure the Electric Make path matches the path used for the installed Agent.

   Note: Avoid selecting an alternative directory that includes spaces in the name. Spaces can create problems when configuring connections with other command-line-based components.

   Click **Next**.

6. If the installer cannot locate a PDF viewer, it will prompt you to browse for one. If you wish, browse for a PDF viewer to use, and click **Next**.

7. On the Base Setup screen, choose whether you want to install Electric Runner client applications, and click **Next**.

8. When the Start Copying Files screen appears, click **Next**.

9. The Installing screen displays while the installation proceeds. When installation is finished, the Complete screen displays. Click **Finish**.

   Installation is complete. The installation log file is in the install directory's root, by default.

Silent Install

If you are installing on a series of identical machines (such as a series of cluster hosts), you can use the "silent install" method, which installs components automatically, without user interaction.

Note: On Windows, if you invoke the installer from the command line, you may receive an "Unknown Publisher" security warning. You can disregard this warning and proceed with installation.

Installer Command-Line Options

Use the following command-line options when performing a silent install. The options are the same values a user would normally set through the installer interface. Use this format:

```
<installer filename> [options]
```

This table lists each command-line option's equivalent in the installer UI and the variable that is set in the installer properties file. You can use the resulting properties file for running silent installs.

Notes:

- You can use the values "yes", "y", "1" and "no", "n", "0" interchangeably within installer command-line options.
- For `--cmdbtype` and `--type`, differences exist between valid command-line option arguments and valid properties file parameters. Ensure you provide the correct input for each method.

Command-line option **Equivalent installer UI field**	**Variable set in the installer**	**Description**
`--adminuserpassword [ARG]` Cluster Manager Admin Password	`EC_CM_ADMINUSER_PASSWORD=`	Sets the Cluster Manager admin user password.

Command-line option Equivalent installer UI field	Variable set in the installer	Description
`--agentallowreboot <y or n>` Reboot host if needed	`EC_AGENT_REBOOT=y or n`	Indicates if you want to reboot after installing Agent/EFS. Default: n For Windows, if you use n, the installer does not restart the Agent service; reboot the host to ensure EFS works properly. Windows may prompt before the host is rebooted. For UNIX, the machine does not reboot unless required, even if you specify `EC_AGENT_REBOOT=y`.
`--agentcmhost [ARG]` Cluster Manager Host/IP address	`EC_AGENT_CMHOST=<`*`hostname or IP address`*`>`	Sets the Cluster Manager hostname or IP address for this host to connect to. Example: 192.205.2.19 or winlong-cm ***Not required*** if using a grid.
`--agentcmport [ARG]` Cluster Manager Accelerator Server port	`EC_AGENT_CMPORT=8030 (default)`	Sets the Cluster Manager server port number for this host to connect to. Not required if using a grid.
`--agentinstallerunnerd <y or n>` Install Electric Runner Service	`EC_AGENT_INSTALL_ERUNNERD=y or n`	y installs eRunnerd on agent machines. Default: y
`--agentislsfhost <y or n>` Host is part of LSF Computing LSF Grid	`EC_AGENT_IS_LSF_HOST=y or n`	Indicates if the Agent is on the grid. For this option/parameter, the term “LSF” refers to all grid engines. Default: n
`--agentnumber [ARG]` Number of Agents	`EC_AGENT_AGENT_NUMBER=1 to 32`	Sets the number of Agents to set up on the host. The maximum is 32 for 32-bit and 64 for 64-bit machines. For Developer Edition, the maximum is the number allowed by your license.
`--agentpassword [ARG]` ECloud Internal User Shared Password	`EC_AGENT_WINUSER_PASSWORD=`	Sets the password for ECloudInternalUser (Windows only).

Command-line option Equivalent installer UI field	Variable set in the installer	Description
`--agentremovelogs <y or n>` Remove old Electric Agent logs	`EC_AGENT_REMOVE_LOGS=y or n`	Removes old agent log files. If not, the install appends to them. Default: n
`--agentsecureconsole <y or n>` Secure Agent Console port	`EC_AGENT_SECURE_CONSOLE_PORT= y or n`	y requires an agent-generated key to be entered before commands will run on the agent console port. Default: n
`--agentskipefs <1 or 0>` n/a	n/a	Prevents (y) the installer from installing EFS. Default: 0 (n)
`--agentskiplofs <1 or 0>` n/a	n/a	Prevents (y) the installer from installing LOFS. Default: 0 (n)
`--agenttempdir [ARG]` Agent temporary directory	`EC_AGENT_TEMPDIR=<dir>`	Sets the temporary directory for all agent files. Windows default: C:\WINDOWS\temp UNIX default: /tmp
`--agentuserlist [ARG]` Existing User	n/a	Sets a list of username/password pairs to be used as login accounts for agents (Windows only).
`--backupdb <y or n>` Back up local database	`EC_CM_BACKUPDB=y or n`	Backs up the local database only. Remote databases are not backed up. Default: n
`--baseinstallerunner <y or n>` Install Electric Runner client apps	`EC_BASE_INSTALL_ERUNNER=y or n`	Installs eRunner client applications on Cluster Manager and Electric Make machines. Default: y
`--cmaccelport [ARG]` Cluster Manager Accelerator Server port	`EC_CM_ACCELPORT=8030 (default)`	Sets the Cluster Manager Accelerator server port number for unencrypted traffic.
`--cmaccelsport [ARG]` Cluster Manager Secure Accelerator Server port	`EC_CM_ACCELSPORT=8031 (default)`	Sets the Cluster Manager Accelerator server port number for encrypted traffic.

Command-line option Equivalent installer UI field	Variable set in the installer	Description
`--cmdbhost [ARG]` Database Host/IP address	`EC_CM_DBHOST=localhost (default)`	Sets the Cluster Manager database host machine.
`--cmdbname [ARG]` Database Name	`EC_CM_DBNAME=ecloud (default)`	Sets the name of the database to create on the database server. If you do not use the default local database, you must provide the name of an existing database.
`--cmdbpassword [ARG]` Database Password	`EC_CM_DBPASSWORD=ecloud (default)`	Sets the user password to use when connecting to the database server.
`--cmdbport [ARG]` Database Port	`EC_CM_DBPORT=3306 (default)`	Sets the database listening port. Use 1433 for MS SQL.
`--cmdbtype <mariadb, mysql, oracle, or mssql>` Database Type	`EC_CM_DBTYPE=MariaDB (default), MySQL, Oracle, or MSSQL`	Sets the type of database to use for Cluster Manager.
`--cmdbuser [ARG]` Database User	`EC_CM_DBUSER=root (default)`	Sets the user name to use when connecting to the database server.
`--cmhttpport [ARG]` Cluster Manager HTTP port	`EC_CM_HTTPPORT=80 (default)`	Sets the Cluster Manager HTTP server port.
`--cmhttpsport [ARG]` Cluster Manager HTTPS port	`EC_CM_HTTPSPORT=443 (default)`	Sets the Cluster Manager HTTPS server port.
`--cmlogrotate <y or n>` Rotate Cluster Manager logs	`EC_CM_LOGROTATE=y or n`	Rotates Apache logs. Default: y
`--cmmigratedb <y or n>` Migrate existing database	`EC_CM_MIGRATEDB=y or n`	Migrates the database; this value is meaningful only if performing an upgrade. Default: n

Command-line option Equivalent installer UI field	Variable set in the installer	Description
`--cmremovelogs <y or n>` Remove old Cluster Manager logs	`EC_CM_REMOVE_LOGS=y or n`	Removes old Cluster Manager log files. If not, the install appends to them. Default: n
`--cmservername [ARG]` n/a	`EC_CM_SERVERNAME=`	Sets the Apache/SSLserver name.
`--cmserviceuser` Cluster Manager Service User	`EC_CM_SERVICEUSER=<existing LDAP user or local user>`	Sets the user to run the Cluster Manager service.
`--cmserviceuserpassword` Cluster Manager Service User Password	`EC_CM_SERVICEUSER_PASSWORD=`	Sets the password to use for the Cluster Manager service user. If not specified, the account that Cluster Manager runs as is used (Windows only).
`--debug` n/a	n/a	Runs the installer in debug mode.
`--debugconsole` n/a	n/a	Runs the installer with the debug console open.
`--erunnerdport` Electric Runner Server Port	`EC_ERUNNERD_PORT=2411 (default)`	Sets the port for eRunnerd to listen on.
`--erunnerdremovelogs <y or n>` Remove old Electric Runner logs	`EC_ERUNNERD_REMOVE_LOGS=y or n`	Removes old eRunnerd logs. If not, the install appends to them. Default: n
`--finalprefix [ARG]` n/a	n/a	Sets the location where the installed directory tree will be located. Use this option when --prefix is a temporary location that is not the final destination for the product.
`--ignoreoldconf <y or n>` n/a	n/a	Ignores the previous configuration.

Command-line option Equivalent installer UI field	Variable set in the installer	Description
`--localagentagentlicensefile` *<browse to location>*	n/a	Sets the location of the ElectricAccelerator Developer Edition license file.
`--mode <console, silent, or standard>` n/a	n/a	Sets the mode in which to run the installer. For a console login, standard and console are identical. For a GUI machine, standard brings up the UI. You can use console in a Unix X Window environment to force the use of console mode.
`--noredist <y or n>` Do not install the redistributables	`EC_BASE_NOREDIST=y or n`	Does not install the Microsoft Visual C++ 2005 SP1 Redistributable (Windows only). Default: n
`--pdfreader [ARG]` *<browse to location>*	`UnixPDFReader=<path to PDF reader>`	Sets the PDF reader to use.
`--prefix [ARG]` Destination Folder	`InstallDir=<dir>`	Sets the installation directory. Windows default: C:\ECloud UNIX default: /opt/ecloud
`--propertyfile [ARG]` n/a	n/a	Sets the property file from which to read installer options.
`--removezips <1 or 0>` n/a	n/a	Removes (1) zip files after installation.
`--rootrun <1 or 0>` n/a	n/a	Allows the installer to run when root privileges are not present and disables the execution of installer steps that would require root privileges. This option does not change access privileges. 1 (y) or 0 (n).

Command-line option Equivalent installer UI field	Variable set in the installer	Description
`--rwprefix [ARG]` n/a	n/a	Specifies the location for read/write files. When using this option for Developer Edition installation, place the license in `<rwprefixdir>/<arch>/tmp` manually if you did not import it during install.
`--skiprun <1 or 0>` n/a	n/a	Prevents (1) the installer from starting the agent or Cluster Manager
`--temp [ARG]` n/a	n/a	Sets the temporary directory used by this program.
`--test` n/a	n/a	Runs the installer without installing any files.
`--type <agent, emake, cm, erunner, erunnerd, localagent, or upgrade>` *<select install type>*	`InstallType=Electric Agent, Electric Make, Cluster Manager, Electric Runner Client, Electric Runner Server, Custom, or Local Agent`	Sets the type of installation to perform.
`--version` n/a	n/a	Displays installer version information.

Creating an Installer Properties File

An installer properties file is a text file that defines installation parameters. These parameters are the same values a user would normally set through the installer interface or command line.

To create an installer properties file:

1. Run an installation with your desired settings.

   This creates a properties file (`install.props`) in the top-level install directory.

2. Use the resulting properties file for subsequent silent installs of the same component type.

Components are installed individually so you must create an installer properties file for each ElectricAccelerator component you intend to install.

The table beginning on page 3-18 details the parameters within installer properties files.

Automating a Linux/Solaris Silent Install

1. Make sure you already have a properties file (`install.props`) created by a successful installation.
2. Log in to the remote machine as root.
3. Invoke the installer:

```
# ./<installer filename> --mode silent --propertyfile <properties file>
```

Automating a Windows Silent Install

1. Make sure you already have a properties file (`install.props`) created by a successful installation.
2. Log in to the remote machine as Administrator.
3. Invoke the installer in a DOS shell:

```
<installer filename> /mode silent /propertyfile <full path\properties file>
```

If you are performing a silent upgrade on an agent host by running the install on the host itself, you may be prompted before the machine is rebooted. This prompt occurs if others are logged in to the machine when you run the agent upgrade.

Additional Install Options

- Electric Runner Client - The Electric Runner Client component installs platform-independent tools (`clusterexec`, `clusterupload`, and `clusterdownload`) that allow you to perform operations across all hosts simultaneously . The tools allow you to:
  - Start and stop agents
  - Reboot hosts
  - Run commands on hosts
  - Upload files
  - Download files
- Electric Runner Server - The Electric Runner Server component installs the server side (erunnerd) on agents to allow them to serve requests from tools. If you decided not to install eRunner during Cluster Manager installation, you do not have access to these tools.
- Custom - A custom install allows you to install multiple components at once.

Migrating an eMake-only Installation to a Different Machine

On all platforms, you can relocate the complete eMake install directory (anything under the *install dir*, starting with the *arch* directory). A simple file copy is sufficient.

On Windows systems, you must ensure that the Visual Studio 2005 runtime libraries are installed. You can install the libraries by running `.../i686_win32/bin/install_redist32.bat` and/or `install_redist64.bat` on the new machine.

You can also choose to add the new location of the bin directory to the PATH.

Installing the Backward-Compatible eMake Package on Agents

Electric Cloud provides archives of eMake versions on the Electric Cloud FTP site.

There may be different reasons why you want to install the backward-compatibility package:

- You have new agents in the cluster that have only the most recent version of ElectricAccelerator, and you have old agents that were upgraded to the most recent version. To allow old eMakes to use the new agent machines, you must install the backward-compatibility package on the new agents.
- You encountered incompatibility error messages, indicating the eMake and agent software versions are not compatible:

  ERROR EC1050: 049300C4:
  The protocol version of Electric Make (45) is incompatible with the software installed on the Electric Cloud cluster 192.168.6.8 (55).
  You need to upgrade your copy of Electric Make; aborting build

  ERROR EC1050: 049300C4:
  The protocol version of this application (45) is incompatible with the software installed on the Electric Accelerator cluster 192.168.6.8 (55).
  You need to upgrade your client application; aborting build

Installing on a Single Agent Machine

1. Open a terminal window.
2. Change to the Electric Cloud installation directory (for example `c:/ECloud` or `/opt/ecloud`).
3. Download the appropriate set of archive files (for example, i686_win32_bcp.zip [all previous eMake versions] or i686_win32_7.0.0_bcp.zip [a single eMake version]) from the Electric Cloud FTP site: accelerator/emake_archive/<*platform*>.
4. Unpack the archive in the current directory (<*installDir*>). For Windows, make sure the extraction destination is <*installDir*> without any sub-directories, such as `c:/ECloud`.
5. Repeat the process for each new agent machine.

Installing on Multiple Agent Machines

1. Open a terminal window.
2. Change to a temporary directory (such as `/var/tmp`, or `c:/Windows/Temp`).
3. Download the appropriate set of archive files (for example, i686_win32_bcp.zip [all previous eMake versions] or i686_win32_7.0.0_bcp.zip [a single eMake version]) from the Electric Cloud FTP site: accelerator/emake_archive/<*platform*>.
4. Unpack the archive in the current directory. For Windows, make sure the extraction destination is the temporary directory (or the directory from step 2) without any sub-directories, such as `c:/Windows/Temp`.
5. Run (as an example): `clusterupload --cm=<cmname> i686_win32 c:/ECloud`.

Protocol Changes

Occasionally, Accelerator may contain protocol changes that make it necessary to update the eMake version on the build machines. An eMake-only installer is available for this type of update.

Chapter 4: Configuration

Configuring ElectricAccelerator properly is important to using the software successfully. The following configuration information will help get you started.

Topics:

- Configuring Linux/Solaris
- Configuring Windows
  - Windows Notes
  - Registry Information
- Using the accelerator.properties File
- Agent Log Rotation
- Changing the Disk Cache Directory and Agent Temporary Storage Location
- Default Directories
- Logging In and Licensing
- Enabling Resource Management
- Installing an Apache Server Certificate
- Configuring LSF
- Enabling Grid Host Manager
- Documentation Roadmap

Configuring Linux/Solaris

RHEL 6 and Accelerator Services Started from a Shell

On RHEL 6, if Accelerator services are started from a shell (as opposed to at boot time), you may encounter errors such as

```
java.lang.OutOfMemoryError: unable to create new native thread
```

This type of error occurs because RHEL 6 greatly decreased the default maximum number of threads per user.

As a workaround, comment out the `nproc` line in the `/etc/security/limits.d/90-nproc.conf` file.

RHEL 5 and the kernel-devel Package

On RHEL 5, you must install the `kernel-devel` package version that matches the Linux kernel where the modules will be loaded.

Solaris 10 Configuration/Debugging

To get the maximum useful information from any core dumps that may occur under Solaris 10 systems (SPARC and x86), run `coreadm -G all -I all` (as superuser) at any time. This reconfigures the machine (persistently, in `/etc/coreadm.conf`) to include the maximum information in core dumps.

Configuring Windows

Agent/EFS Machines

Disable the Windows error reporting service (if applicable to your Windows version). This avoids popup windows for crashed applications.

Windows 8 eMake and Agent Machines

On Windows 8 eMake and agent machines, you must disable Admin Approval Mode. Follow these steps:

1. Type `secpol.msc` in the Start Menu and then press Enter.
2. Double-click **Local Policies**, then double-click **Security Options**.
3. Scroll to the bottom of the entry. Locate and double-click

   User Account Control: Run all administrators in Admin Approval Mode

4. Set it to disabled and click **OK**.
5. Reboot the machine.

Note: This configuration is required because of a change in the Windows 8 Admin Approval Mode defaults. Leaving Admin Approval Mode enabled may result in performance degradation.

Firewall Settings

Because the ports used to communicate with agents are dynamic, you may experience issues on Windows agent hosts depending on your firewall settings.

If you need to lock down communication with an agent host, exclude the following programs from the block list:

- `C:\Ecloud\i686_win32\bin\ecagent.exe`
- `C:\Ecloud\i686_win32\bin\erunnerd.exe`

Agent-Side Applications

It is important that applications running on agents are properly initialized for the users that run the agent processes. By default, the users ECloudInternalUser1, ECloudInternalUser2, and so on, own the processes run by agents. Agents can be run as any user, but each agent must have a unique user, which is a requirement for agents to function properly (contact technical support for additional information).

Note: The users ECloudInternalUser1, ECloudInternalUser2, and so on (or the users you choose to run agent processes), must be local administrators on the agent machines.

This requirement imposes additional setup steps for agent machines because some applications require per user setup (WinZip, Visual Studio 2005 and 2008, Microsoft Office, and so on). Setup is particularly important for applications that display a dialog if they are not properly initialized because this may result in stalled jobs (the application remains in interactive mode until the agent times it out for lack of activity) during the use of eMake.

To initialize applications, do one of the following:

- Use the `psexec`tool available from Microsoft (http://technet.microsoft.com/en-us/sysinternals/bb897553.aspx) to run the application as different users. This requires that you have the password for the account (for ECloudInternalUser1, and so on, contact technical support for the user password).
- Log in as each of the configured agent users and run the relevant application to initialize it.
- Identify which files/registry keys must be set for a user by using Microsoft's procmon tool (http://technet.microsoft.com/en-us/sysinternals/bb896645.aspx), creating a .reg file, and copying the user files to create those entities for all users.

Visual Studio

If you intend to use Accelerator to build Microsoft Visual Studio projects...

Running eccheckvsinst

Use the `eccheckvsinst` utility (located in `C:\ECloud\i686_win32\unsupported`) to check agent installations that will be involved in Visual Studio-based builds.

1. Invoke the utility on the target installations and redirect the output to a text file.
2. Repeat on all agents in the cluster.
3. Use a text diff tool to compare the various files.

   There should be no difference except the hostname (as noted in Ensuring a Consistent Build Environment).

   eccheckvsinst > eccheckvsinst.agent1
   eccheckvsinst > eccheckvsinst.agent2
   diff eccheckvsinst.agent1 eccheckvsinst.agent2

Before building the Visual Studio project

Before you can use Accelerator to build your Visual Studio project, you must install Visual Studio on each agent host, and then log in and run `devenv` as the user that owns the respective agent processes (usually `ECloudInternalUser1`, `ECloudInternalUser2`, and so on). You can use `psexec` to eliminate the need

to log in and log out multiple times. Using this tool is more efficient because Visual Studio stores user settings in the registry and creates files in “My Documents.” Also, ensure you can build your solutions/projects with the Visual Studio setup on each host.

If you install Visual Studio ***after*** installing Accelerator, register the Visual Studio add-in by running `install_ecaddin<N>.bat` where N is 70, 71, 80, 90, 100, or 110, depending on your Visual Studio version.

If you are using Visual Studio 2005 or later, reduce the number of parallel builds Visual Studio performs:

1. In Visual Studio, select Tools > Options.
2. In the Options dialog, open Projects and Solutions > Build and Run.
3. Set maximum number of parallel project builds to 1.

The *ElectricAccelerator Visual Studio Integration Guide* contains additional information about using Accelerator with Visual Studio.

Initializing Visual Studio

Use the `psexec` method to initialize Visual Studio as shown:

```
psexec -u ECloudInternalUser1 "C:\Program Files\Microsoft Visual Studio
8\Common7\IDE\devenv.exe"
```

As an alternative, disable profiles for Visual Studio by running this regedit script:

```
REGEDIT4
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\VisualStudio\8.0\Profile]
"AppidSupportsProfiles"="0"
```

Uninstalling the Visual Studio add-in

Uninstall the add-in by running `uninstall_ecaddin<N>.bat` where <*N*> is 70, 71, 80, 90, 100, or 110, depending on your Visual Studio version. These bat files are in the Accelerator `bin` directory.

Microsoft Office

You must run Microsoft Office by using `psexec` (or logging in directly) because there is no registry setting to initialize Microsoft Office easily.

Ensure that the Visual Basic setting for Security is set to Medium or Lower (assuming the build tries to run VB scripts). Find this under Tools > Options > Security > Macro Security.

WinZip

You must run WinZip by using `psexec` (or logging in directly) because there is no registry setting to initialize WinZip easily.

MSBuild

Specify configuration information for MSBuild under `C:\Program Files\MSBuild`. This information must be present on all agents, by either syncing that directory among all machines or by adding that directory to the eMake root.

Cygwin

Electric Cloud strongly recommends that the install location be the same on all agents. The install location must also be the same on all eMake machines. The mount mode must be the same for all machines.

Windows Notes

Accelerator may unset the JAVA_HOME environment variable

Installing Accelerator may unset the environment variable JAVA_HOME. Reset JAVA_HOME manually.

Automatic installer actions on Windows Server 2008 R2

- Disables the Windows error reporting service.
- Sets HKLM\SYSTEM\CurrentControlSet\Control\FileSystem\NtfsDisableLastAccessUpdate to 0. The default value for Windows Server 2008 R2 is 1.
- Disables User Account Control (UAC) for 64-bit versions. Disabling UAC avoids popup windows for applications that require administrator privileges. If UAC is enabled, application registry access is redirected to each user's virtual store, even if it runs under the Administrator account.

Windows kernel non-paged pool memory may become depleted

Under high-volume network traffic, the non-paged pool memory in the Windows kernel has the potential to become depleted. This issue in the Windows kernel can be triggered by applications such as MySQL, Java server applications, and so on. Over a period of time, this results in a machine crash.

The workaround is to use the 64-bit version of Windows. Though this does not completely resolve the issue, it increases the available memory to a point where crashes are unlikely and infrequent.

FileInfo and Superfetch services may affect EFS driver performance

The FileInfo and Superfetch services run on Windows 7 (Microsoft officially turned them off in Windows Server 2008 R2).

Because the FileInfo (used by Superfetch) filter driver issues a couple of calls for each file operation in the EFS driver, it has the potential to slow down the EFS driver.

Accelerator turns off the two services by default. You can choose to leave them running by removing the following two lines from runagent (located in `<ECloud install>\<arch>\bin`):

```
catch {service stop sysmain}
catch {service stop fileinfo}
```

and rebooting the machine.

Avoid real-time antivirus scans on agent machines

Real-time scans can slow down builds and can lead to unexpected failures due to issues with the antivirus software's dll injection. Generally, scans do not find anything relevant because all results are also checked on the eMake machine.

Terminating stale processes

Certain processes may continue to run on Windows Agent machines. You can choose to terminate all "stale" processes by adding the following line to runagent (located in `<ECloud install>\<arch>\bin`):

```
[efs connect] set terminateStaleProcess 1
```

Support for managed code/.NET

There are no known limitations with respect to building managed code or .NET code. There are, however, areas to keep in mind:

- Agents must have the same version of .NET installed.
- Agents must be on the same service pack level and have the same OS and tool hotfixes installed.
- The language bar must be enabled on all agent machines, or disabled on all agent machines.
- Including the Global Assembly Cache in the eMake root is *not* recommended. Contact technical support for more details.

Registry Information

To allow parallel building of Windows code, Accelerator virtualizes the registry and the file system. The following sections discuss important registry information.

Registry use under Windows

There are two relevant areas of registry use during an Accelerator build. By default, Accelerator virtualizes HKEY_CLASSES_ROOT (except HKEY_CLASSES_ROOT\Installer and HKEY_CLASSES_ROOT\Licenses).

- HKEY_CLASSES_ROOT
- All other keys

HKEY_CLASSES_ROOT

This key contains file name extensions and the COM class registration (http://msdn.microsoft.com/en-us/library/ms724475.aspx and http://technet2.microsoft.com/windowsserver/en/library/dd670c1d-2501-4f32-885b-0c6a1ae662f41033.mspx?mfr=true). Configuration data is stored under the program ids, CLSID, Interface, TypeLib, AppId, and so on.

For entities created during the build, this information must be virtualized to all involved agents.

The following information is registered for a type library:

http://msdn2.microsoft.com/en-us/library/ms221610(VS.85).aspx

```
\TypeLib\{libUUID}
\TypeLib\{libUUID}\major.minor = human_readable_string
\TypeLib\{libUUID}\major.minor\HELPDIR = [helpfile_path]
\TypeLib\{libUUID}\major.minor\Flags = typelib_flags
\TypeLib\{libUUID}\major.minor\lcid\platform = localized_typelib_filename
```

Other entities that are registered by UUID are registered in different places:

http://msdn2.microsoft.com/en-us/library/ms221150(VS.85).aspx

A ProgID("ApplicationName") maps to and from a CLSID(GUID). The CLSID maps to the actual ActiveX component ("APP.EXE"). The type library is available from the CLSID:

```
\CLSID\TypeLib = {UUID of type library}
\CLSID\{UUID} = human_readable_string
\CLSID\{UUID}\ProgID = AppName.ObjectName.VersionNumber
\CLSID\{UUID}\VersionIndependentProgID = AppName.ObjectName
\CLSID\{UUID}\LocalServer[32] = filepath[/Automation]
\CLSID\{UUID}\InProcServer[32] = filepath[/Automation]
```

http://msdn2.microsoft.com/en-us/library/ms221645(VS.85).aspx

Applications that add interfaces must register the interfaces so OLE can find the appropriate remoting code for interprocess communication. By default, Automation registers dispinterfaces that appear in the .odl file. It also registers remote Automation-compatible interfaces that are not registered elsewhere in the system registry under the label ProxyStubClsid32 (or ProxyStubClsid on 16-bit systems).

The syntax of the information registered for an interface is as follows:

```
\Interface\{UUID} = InterfaceName
\Interface\{UUID}\Typelib = LIBID
\Interface\{UUID}\ProxyStubClsid[32] = CLSID
```

All other keys

Other keys are likely not relevant to the build. HKEY_LOCAL_MACHINE, HKEY_CURRENT_USER, HKEY_USERS, and HKEY_CURRENT_USER are machine specific. If other areas must be virtualized, it is recommended that you add them to the `emake-reg-root` option.

Registry underlay

When a process in the build requests information from the registry, the EFS first checks if the requested key is already present in its cache. If the key is not present, the EFS relays the request to the agent, which in turn sends the request to eMake. After receiving the response from eMake, the agent loads the key into the EFS cache, subject to the following conditions:

- If the key does not exist at all in the local registry on the agent host, the value from the eMake response is used unconditionally.
- If the key exists in the local registry, the value from the local registry is given precedence over the initial value from eMake, but not any value set by prior commands in the build. That is, if the key is modified during the course of the build, the modified value is used in preference of any value from the local registry.

  The order of precedence is (lowest to highest):

  - Value from eMake host registry prior to the start of the build
  - Value from the agent host registry, if any
  - Value set by an earlier job in the build

  The additional checking of precedence enables Accelerator to interoperate with tools that store host-specific licensing information in the registry. If the agent simply used the value from eMake unconditionally in all cases, such tools would fail to operate correctly.

  Electric Cloud STRONGLY RECOMMENDS AGAINST running builds locally on agent host machines. Running builds locally on agent machines may add relevant keys to the local machine, which take precedence over the eMake machine's keys. If a key that should come from the eMake machine (such as the typelib information for a lib generated during the build) is already present on the agent due to a locally performed build, the wrong information is used, possibly causing a broken build.

  If an agent machine has locally created keys, remove the typelibs that are created during the build from the registry. Any typelib that has an invalid path name associated with it is a likely candidate for an "underlayed" lookup.

  Ideally, typelibs created by a build are known. At this point, it is recommended to check for their existence on the cluster. If an error occurs that indicates the direction of this problem (for example, a library/typelib cannot be found), investigate the failing agent's registry.

ExcludeProcessList registry entry

You can add a multi-string registry value to the agent host inside HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ElectricFS to exclude processes from interfering with EFS and causing a crash. The ExcludeProcessList entry can list processes from McAfee antivirus (for example, Mcshield.exe and mfevtps.exe) or other antivirus software.

Note: Make these registry changes only if the system crashed previously.

Using the accelerator.properties File

The `accelerator.properties` file allows you to modify parameters that affect a range of Accelerator behaviors. The file resides on the Cluster Manager in the following default location:

- Linux: `/opt/ecloud/i686_Linux/conf`
- Windows: `C:\ECloud\i686_win32\conf`
- Solaris SPARC: `/opt/ecloud/sun4u_SunOS/conf`
- Solaris x86: `/opt/ecloud/i686_SunOS.5.10/conf`

The following topics discuss parameters configurable through the `accelerator.properties` file.

- Changing Log Locations
- Disabling Agents by Default
- User Authentication
- Setting Up Email Notification
- Adding Custom Protocol Mismatch Error Text
- Using the Database Connection Monitor

Restart the Cluster Manager service after modifying the `accelerator.properties` file.

Changing Log Locations

Modify the following parameters to change the location of build logs and accelerator logs:

- `ACCELERATOR_BUILD_LOGS_PATH=build_logs`
- `ACCELERATOR_LOG=logs/accelerator.log`

Disabling Agents by Default

Agents are enabled by default. If you wish to disable them by default:

1. Change the `ACCELERATOR_AGENTS_STARTUP_ENABLED` property to `false`. This change affects agents installed after this point; existing agents on the Agents tab are not affected.
2. If you want the property to change the behavior of existing agents, you must restart the Cluster Manager process.

User Authentication

Accelerator uses account information from multiple sources. In most cases, the primary account information source is an external LDAP or Active Directory repository: both user and group information is retrieved from the repository. *Local* users and groups can be defined within Accelerator.

Getting Started

1. Go to the following directory on the Cluster Manager server:
   <ECloud install>/*<arch>*/conf

2. Make a copy of the `ldap_template.xml` file if you intend to use LDAP, or the `ad_template.xml` file if you prefer to use Active Directory. Save the copy in the `conf` directory as any name you choose or use the name: `securityContext.xml`.
3. Still working in `conf`, open the `accelerator.properties` file.
   - Locate the following commented-out text string:
     `#ACCELERATOR_SECURITY_CONTEXT_FILES=conf/securityContext.xml`
   - If you did not name your template copy `securityContext.xml`, replace `securityContext.xml` with the filename you chose.
   - Uncomment the `#ACCELERATOR_SECURITY_...` text string by removing the lead "#" sign.
   - Comment out `ACCELERATOR_SECURITY_CONTEXT_FILES=` (which immediately follows the line you uncommented)
4. Use the LDAP information and examples in the following sections to fill-in your own copy of the LDAP template.
5. Restart the Cluster Manager.
6. After the Cluster Manager is running, log in to the Cluster Manager UI as "admin".
7. Go to Administration > Permissions.
8. Click **Enable User** or **Enable Group**.
9. Search for the desired user or group (or search for "*" to see all). If you set it up correctly, the requested LDAP users are visible.
10. Select the appropriate users or groups to enable using the corresponding checkbox(es).
11. Ensure that the desired permissions are set for the users or groups. (The online help system contains additional information about permissions.)

Note: If you experience permissions issues while reading the `ldap_template.xml` or `ad_template.xml` copy, verify that the account that runs the Cluster Manager also has read permission for that file. To diagnose problems that cause Cluster Manager startup issues, including LDAP/Active Directory configuration problems, check
`<ECloud install>/<arch>/logs/accelerator.log`.

Configuring LDAP

A number of options must be customized for any LDAP configuration. The following sample configuration is from the `ldap_template.xml` file. After the sample, see a list of properties with a description.

Sample LDAP Configuration

```
<bean id="ldapDirectoryProvider"
                class="com.electriccloud.security.ldap. LdapDirectoryProviderImpl">
<property name="providerName" value="LDAP"/>
<property name="url" value="ldap://myldaphost/dc=company,dc=com"/>
<property name="userBase" value="ou=People"/>
<property name="userSearchFilter" value="uid={0}"/>
<property name="userSearchSubtree" value="false"/>
<property name="userNameAttribute" value="uid"/>
<property name="fullUserNameAttribute" value="gecos"/>
<property name="emailAttribute" value="mail"/>
<!--<property name="managerDn" value="uid=admin,ou=People,dc=company,dc=com"/> -
->
<!--<property name="managerPassword" value="password"/> -->
<!--<property name="groupBase" value="ou=Group"/> -->
<!--<property name="groupMemberFilter" value="(member={0})"/> -->
<!--<property name="groupMemberAttributes" value="member"/> -->
<!--<property name="groupSearchFilter" value="(objectClass=groupOfNames)"/> -->
<!--<property name="groupNameAttribute" value="cn"/> -->
<!--<property name="verifyCerts" value="true"/> -->
</bean>
```

The following properties configure LDAP mapping:

Property	Description
`emailAttribute`	(optional) The attribute in a user record that contains the user's email address. If the attribute is not specified, the account name and domain name are concatenated to form an email address.
`fullUserNameAttribute`	(optional) The attribute in a user record that contains the user's full name (first and last name) for display in the UI. If this attribute is not specified or the resulting value is empty, the user's account name is used instead.
`managerDn`	(optional) The DN of a user who has read-only access to LDAP user and group directories. If this property is not specified, the server attempts to connect as an unauthenticated user. Not all servers allow anonymous read-only access. **Note:** This user does not need to be an admin user with modify privileges.
`managerPassword`	(optional) If the managerDn property is set, this password is used to authenticate the manager user.
`providerName`	This human readable name is displayed in the user interface to identify users and groups that come from this provider.
`userBase`	This string is prepended to the `basedn` to construct the directory DN that contains user records.
`userNameAttribute`	The attribute in a user record that contains the user's account name.

Property	Description
`userSearchFilter`	This LDAP query is performed in the context of the user directory to search for a user by account name. The string “{0}” is replaced with the user’s login ID. Typically, the query compares a user record attribute with the substituted user login ID.
`userSearchSubtree`	This property, if true, searches the entire subtree as identified by context. If false, (the default) then only the level identified by the context is searched.
`url`	The LDAP server URL is in the form `protocol://host:port/basedn`. Protocol is either ldap or ldaps (for secure LDAP). The port is implied by the protocol, but can be overridden if it is not at the default location (389 for ldap, 636 for ldaps). The `basedn` is the path to the top level directory that contains users and groups at this site. This is typically the domain name where each part is listed with a `dc=` and separated by commas. **Note:** Spaces in the `basedn` must be URL encoded (%20).
`verifyCerts`	(optional) This property enables certificate verification for LDAPS connections if true.

In addition to user information, the LDAP server can be queried for **group information**. This query is optional because the local group mechanism can refer to LDAP users as well as local users. However, the following elements can be used to tell the server how to map groups in LDAP:

Property	Description
`groupBase`	(optional) This string is prepended to the `basedn` to construct the directory DN that contains group records.
`groupMemberAttributes`	(optional) A comma separated attribute names list that identifies a group member. Most LDAP configurations only specify a single value, but if there is a mixture of POSIX and LDAP style groups in the directory, multiple attributes might be required.
`groupMemberFilter`	(optional) This LDAP query is performed in the groups directory context to identify groups that contain a specific user as a member. There are two common forms of group record in LDAP directories: POSIX style groups where members are identified by account name, and `groupOfNames` or `uniqueGroupOfNames` records where members are identified by the full user DN. Both forms are supported, so the query is passed two parameters: “{0}” is replaced with the full user record DN, and “{1}” is replaced with the user's account name.
`groupNameAttribute`	(optional) The group record attribute that contains the name of the group.
`groupSearchFilter`	(optional) This LDAP query is performed in the context of the groups directory to enumerate group records.

Determining LDAP Mapping

A typical POSIX user record in LDAP looks similar to the example below. To set up a mapping for this record, it is necessary to identify various record components.

First, identify the path in the directory that contains user records. In this example, the build user has a distinguished name (dn) of `uid=build,ou=People,dc=mycompany,dc=com`. This name uniquely identifies the build user account and this path splits into three parts:

- base dn: `dc=mycompany,dc=com`
- user base: `ou=People`
- user element: `uid=build`

The `baseDn` is the parent of the directory that contains users. This value is combined with the protocol and server to form the URL. In this case, the URL ends up as `ldaps://dir/dc=mycompany,dc=com`. The `userBase` is the portion of the path that identifies the directory containing all user account records. This value is used directly as the `userBase` configuration element. The remaining portion identifies the user without the People directory: `uid=build`. The user name is replaced in this value with the string "{0}" to form the `userSearchFilter: uid={0}`. This query allows the server to search for a user's account name by looking for a record with a matching `uid` attribute. The final mapping step is to identify user record attributes that hold the account name, full user name, and (optionally) the user's email address. In this example, the account name is `uid` (identified earlier), the full user name attribute is `gecos`, and there is no email attribute.

At this point, the server is able to authenticate a user, look up a user by name, and determine the user's full name. For many installations this is sufficient.

Sample LDAP User Record

```
# build, People, electric-cloud.com
dn: uid=jdoe, ou=People, dc=mycompany,dc=com
loginShell: /bin/bash
uidNumber: 508
gidNumber: 508
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
uid: jdoe
gecos: John Doe
cn: John
homeDirectory: /net/filer/homes/build
```

Also, you can configure the server to look for **LDAP groups** that refer to user accounts. A typical group record is shown below. Like a user account, an LDAP group has a distinguished name with a baseDn, a group base, and a group element. In this case, the base dn is still `dc=mycompany,dc=com`. The `groupBase` configuration element is `ou=Group`, and the group name is `cn=build_users`.

The server must identify records in the directory that correspond to groups—it does this by applying the `groupMemberFilter` to the records in the `groupBase` directory. In this case, group records are members of the posixAccount object class, so the filter can be set to `objectClass=posixGroup`. To display a group by its name, the server must know which attribute represents the group name. In this case, set the `groupNameAttribute` to `cn`.

Finally, the server needs a filter to determine which accounts belong to the group and the attribute name that represents a single group member. Group membership can be identified in one of two ways. Either the members are listed by account name, or by their LDAP distinguished name. In this case, POSIX group membership is determined by account name, so set the `groupMemberAttributes` property to `memberUid`, and set the `groupMemberFilter` to `memberUid={1}`.

Sample LDAP Group Record

```
# build_users, Group, mycompany.com
dn: cn=build_users,ou=Group,dc=mycompany,dc=com
objectClass: posixGroup
objectClass: top
gidNumber: 100
memberUid: jdoe
memberUid: mary
cn: build_users
```

Sample Active Directory Configuration

The following XML element defines parameters needed to connect to an Active Directory server and the query to use for looking up user information.

```
<bean id="adDirectoryProvider"
                class="com.electriccloud.security.ldap.ActiveDirectory  Provider">
 <property name="providerName" value="ActiveDirectory"/>
        <property name="url" value="ldap://server/dc=company,dc=com"/>
        <property name="managerDn"value="cn=myuser,cn=Users,dc=company,dc=com"/>
        <property name="managerPassword" value="mypw"/>
        <property name="userBase" value="cn=Users"/>
        <property name="userSearchFilter" value="(sAMAccountName={0})"/>
        <property name="userSearchSubtree" value="false"/>
        <property name="userNameAttribute" value="sAMAccountName"/>
        <property name="fullUserNameAttribute" value="name"/>
        <property name="emailAttribute" value="userPrincipalName"/>
<!-- <property name="groupBase" value=""/> -->
<!-- <property name="groupMemberFilter" value="member={0}"/> -->
<!-- <property name="groupMemberAttributes" value="member"/> -->
<!-- <property name="groupSearchFilter"value="(objectClass=group)"/> -->
<!-- <property name="groupNameAttribute" value="cn"/> -->
        <property name="pageSize" value="500"/>
</bean>
```

Setting Up Email Notification

To configure the Cluster Manager to send email notifications through SMTP, define your SMTP connection properties. SMTP connection properties are contained in `accelerator.properties`. The following properties must be defined:

Property	Description
ACCELERATOR_MAIL_HOST	The name of the SMTP mail server.
ACCELERATOR_MAIL_PORT	The mail server port. Typically, this is ‘25’ for SMTP or ‘465’ for Secure SMTP.
ACCELERATOR_MAIL_PROTOCOL	The mail server protocol. This can be ‘smtp’ for SMTP or ‘smtps’ for Secure SMTP.

If your mail server requires an authenticated user to send mail, you must also set the following properties:

Property	Description
ACCELERATOR_MAIL_USERNAME	The user name required for authentication.
ACCELERATOR_MAIL_PASSWORD	The password to use for authentication.

For example:

```
ACCELERATOR_MAIL_HOST=smtp.electric-cloud.com
ACCELERATOR_MAIL_PORT=25
ACCELERATOR_MAIL_PROTOCOL=smtp
ACCELERATOR_MAIL_USERNAME=cm
ACCELERATOR_MAIL_PASSWORD=mypass
```

This example sets up the Cluster Manager to send mail as the 'cm' user through the `smtp.electric-cloud.com` server listening for SMTP connections on port 25.

After making changes to the mail configuration, you must restart the Cluster Manager service for the changes to take effect.

Configuring the Mail Policy

On the Administration > Server Settings page, you can configure various aspects of the global email notification policy. Cluster Manager sends mail notifications when messages are added to the message log. The following settings control how and when mail is sent:

Setting	Description
Email Interval	The number of minutes between email notifications.
Email Item Limit	The maximum number of messages that will be sent in a single email at the end of each interval. If more messages than the limit arrive, they are ignored.
Email From	The name you use in the ‘From’ line in the email header.
Email Prefix	A string prepended to every subject line of each outgoing email. This string can be used to assist with mail filtering.

User Email Preferences

Each user that logs in to the Cluster Manager web interface can configure which mail notifications to receive. By default, a user receives no email notifications.

For users to configure their own mail notifications (for messages and build completions), they must have Modify permission defined in the User field on the Permissions page. The administrator can edit a user’s permission on the Administration > Permissions page. By default, the User field is set to None.

To receive notifications about messages in the message log:

1. Go to the Messages > Message Policies page.
2. Select the Watch Messages checkbox to enable message notifications. Watch Level controls the minimum severity level of messages that generate a notification.

To receive notifications when builds complete, go to the Builds > Build Classes page. For each build class in the list, select the checkbox in the Notify column to control whether the current user is notified when a build of that class completes. Users can selectively enable notifications for a set of build classes that interest them.

Adding Custom Protocol Mismatch Error Text

You can insert custom error text that is displayed when there is a protocol mismatch.

1. Go to the `accelerator.properties` file.
2. Add your custom text to the `ACCELERATOR_CLIENT_MISMATCH_DETAIL` property.
3. Restart the Cluster Manager service.

Using the Database Connection Monitor

You can configure the Cluster Manager to monitor the connection to the database. The following properties in `accelerator.properties` control the monitor's behavior:

- `ACCELERATOR_DB_WATCHDOG_FAILURE_THRESHOLD=3`

  By default, if the database connection fails three times in a row, the server is restarted. You can change the threshold to a number that meets your needs. Changing the threshold to 0 turns off database connection monitoring.

- `ACCELERATOR_DB_WATCHDOG_INTERVAL=20000`

  By default, the monitor check the database connection every 20000 milliseconds (20 seconds). You can change the threshold to an interval that meets your needs.

Agent Log Rotation

Accelerator automatically rotates session logs using default settings. Information about session logs:

- stdout/stderr logs go to `consoleN.log` (where N is the agent number).
- Normal session trace messages go to `ecagentN.log` (where N is the agent number).
- Only session trace logs will be rotated.
- The default size limit for log files before being rotated is 1 GB.
- The default number of rotated files to keep is 20. When that limit is reached, the oldest log files will begin to be deleted to make space for newer logs.
- The most recent log messages are in `ecagentN.log`. When the log file reaches its size limit, that file is renamed to `ecagentN.log.1`. If `ecagentN.log.1` already exists, then it is renamed to `ecagentN.log.2` before `ecagentN.log` is renamed to `ecagentN.log.1`. If `ecagentN.log.2` exists, it is first renamed to `ecagentN.log.3` and so on. This rotation continues until the retain count is met, at which point file deletion begins.

Configuring Log Rotation

The preferred method for configuring log rotation is to use the `runagent.local` file:

1. Navigate to the `<ECloud install>\<arch>\bin` directory.
2. Open the `runagent.local` file.
3. Add the desired options using this format: `session logger <option>`

   For example: `session logger retaincount 10`

Available log rotation options:

Option	Description
`maxfilesize <bytes>`	Get/set the maximum file size for log files before they are rotated. The default is 1 GB.
`filename <path>`	Get/set the file name to log to. The default for UNIX is `/var/log/ecagentN.log`. The default for Windows is `<install dir>\ecagentN.log` (for example, `C:\ECloud\ecagentN.log`).
`retaincount <count>`	Get/set the number of rotated log files to retain before starting to delete them. The default is 20. To turn off log rotation, set to 0.
`clear`	Clear the contents of the ecagentN.log file. This is useful when log rotation is turned off.

Alternate Methods for Configuring Log Rotation

If you do not use the `runagent.local` file, you can

- use cmtool (for example, `cmtool --cm=<YOURCM> runAgentCmd "session logger maxfilesize 5000000000"`)

  Note: cmtool is not available for local agents (ElectricAccelerator Developer Edition).

or

- telnet to the agent (for example, see the following information)

```
-bash-3.00# telnet localhost 2419
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
qa_solx10_64_1@192.168.6.8: session logger retaincount 5

qa_solx10_64_1@192.168.6.8: session logger retaincount
5

qa_solx10_64_1@192.168.6.8: session logger maxfilesize
1073741824
qa_solx10_64_1@192.168.6.8: session logger maxfilesize 5000
qa_solx10_64_1@192.168.6.8: session logger maxfilesize
5000

qa_solx10_64_1@192.168.6.8: session trace all
all
```

Changing the Disk Cache Directory and Agent Temporary Storage Location

The same location is used for the disk cache directory and agent temporary storage.

- Use the agent's `ecconfig` command to change one agent:

```
ecconfig -tempdir <newtempdir>
```

Where *<newtempdir>* is the new directory.

- Run the `clusterexec` command on either the Cluster Manager or the build server to change all agents at once:

  ```
  clusterexec --cm=<YOURCM> ecconfig -tempdir <newtempdir>
  ```

  Where:

  - *<YOURCM>* is your Cluster Manager host name
  - *<newtempdir>* is the new directory. Specify a full `PATH` to the directory you want to use. Each agent on the host creates a unique subdirectory within the disk cache/temporary storage location, so they do not conflict with each other. The path specified for `-tempdir` must already exist. On Linux/UNIX, the path must have permissions that allow build users to create temporary files. On Windows, agent users must have permission to create temporary files because the eMake client user is not impersonated.

Restart the agents for your changes to take effect:

```
clusterexec --cm=<YOURCM> restartAgent
```

After specifying a different disk cache/temporary storage location, you can switch back to the default location whenever you choose. To do this, use `ecconfig -tempdir` with an empty string [""] as the location.

Default Directories

Installation Location

Platform	Location
Linux	`/opt/ecloud/i686_Linux`
Windows	`C:\ECloud\i686_win32`
Solaris SPARC	`/opt/ecloud/sun4u_SunOS`
Solaris x86	`/opt/ecloud/i686_SunOS5.10`

Log Files

Platform	Install Log	Agent Log
Linux	`/opt/ecloud/install_$timestamp.log`	`/var/log/ecagent?.log`
Windows	`C:\ECloud\install_ $timestamp.log`	*<ECloud Install>*`\ecagent?.log`
Solaris SPARC	`/opt/ecloud/install_$timestamp.log`	`/var/log/ecagent?.log`
Solaris x86	`/opt/ecloud/install_$timestamp.log`	`/var/log/ecagent?.log`

Where:

- ? is the agent number
- *<ECloud Install>* is the installation directory

Disk Cache Directory and Agent Temporary Storage Location

The same location is used for the disk cache directory and agent temporary storage.

Platform	Location
Linux	`/tmp`
Windows	`C:\WINDOWS\Temp`
Solaris SPARC	`/tmp`
Solaris x86	`/tmp`

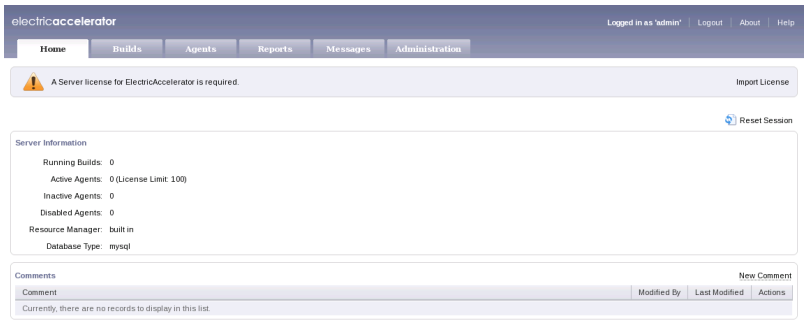
The most common reason to change this location is due to insufficient disk cache space. To change the location, see Changing the Disk Cache Directory and Agent Temporary Storage Location.

Logging In and Licensing

1. From the machine where Cluster Manager is installed, open a browser window and type-in `localhost` to display the Login screen. If you are accessing the Cluster Manager machine remotely, type-in the machine name, ***not*** localhost. If you are not using port 80, use `host:port`.

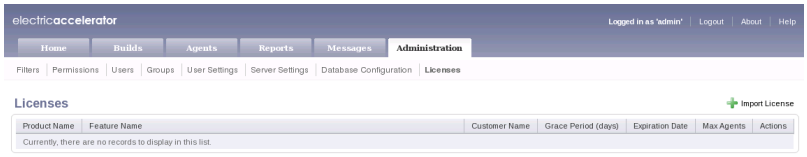
2. Type-in your user name and password. For a new installation, the default admin account user name and password is admin/changeme; for an upgrade, the default admin account user name and password is admin/*<previous password>*. Click **Login**.
3. When the Security Alert screen appears, click **Yes** to proceed or **View Certificate** to install the certificate.

   If you clicked **Yes**, the ElectricAccelerator Home page appears next. If you have not installed your ElectricAccelerator license, an advisory message appears in the top-left portion of the web page as shown in the following screen.



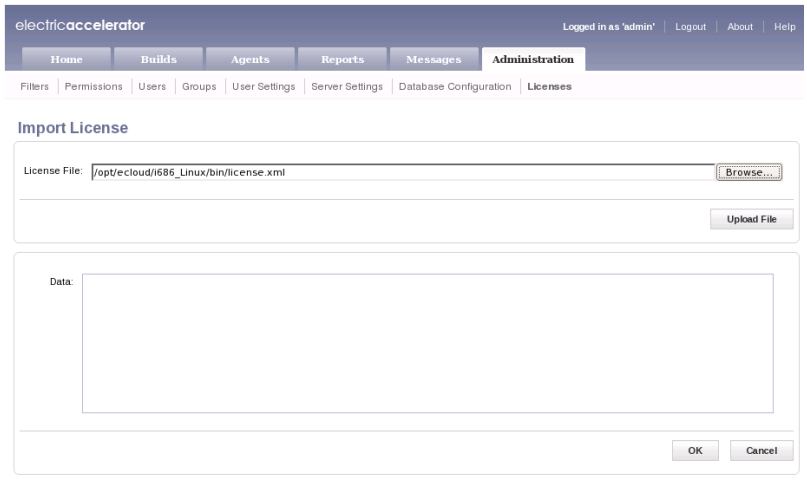
4. On the right-side of this page, click **Import License** and the Import License page appears.

   You should have already received your license file. If not, contact Electric Cloud technical support.



5. Use one of the following methods to install your license:

   - **Browse** for your license file and click **Upload File**.
   - Open your Accelerator license file, copy and paste the text to the Data field and click **OK**.



After the ElectricAccelerator license is installed, you can get familiar with home page features, complete remaining Configuration tasks, and begin using Accelerator.

Enabling Resource Management

ElectricAccelerator resource managers allow you to select a subset of available agents to run builds. The following resource management choices are available:

- none - do not use a resource manager (default).
- built in - use Accelerator's internal resource manager. For example, you might have a build that requires a certain hardware feature. If so, you can instruct Accelerator to use agents on hosts only containing that hardware.
- priority pool - use the priority pool resource manager, which allows you to group resources into pools that can be prioritized differently among groups.
- grid - for use with grid management software, such as LSF or Oracle Grid Engine.

Note: You may choose not to use the resource manager feature, but if you use this feature, note that choosing one resource manager precludes using the other one—resource managers are mutually exclusive.

To enable a resource manager for your build:

1. Log in to the Cluster Manager.
2. Go to Administration > Server Settings.
3. Select a resource manager.

Enabling resource management provides an additional subtab in the web UI. From any Cluster Manager page, click the Agents tab to see the newly activated Resources subtab. Accelerator's online help system contains more resource manager information and help for using the Resources and the Resource Details topics.

If you selected grid...
After you complete the installation process and perform the steps in Configuring LSF (if applicable) and Enabling Grid Host Manager:

1. Restart the Cluster Manager.
2. Log in as an administrator.
3. Go to Administration > Server Settings.

   If everything is available, the grid radio button will be accessible. If this button is not available, review the Accelerator Log file—LSF, for example, may not be available to the process running the ElectricAccelerator server.

4. If the grid radio button is available, select it.

   The Cluster Manager will dynamically request hosts as needed to support running Electric Make.

   IMPORTANT: If using LSF, the LSF client must be installed on the Cluster Manager host.

Installing an Apache Server Certificate

After installation, you may want to create a new Apache certificate.

By default, Accelerator generates a temporary self-signed certificate during installation. This certificate is used whenever a browser makes an HTTPS connection to the Apache server. During Accelerator installation, Apache is configured to look for a private key file named `$HOSTNAME.key` and a certificate named `$HOSTNAME.crt`. These files are in `$DATADIR/apache/conf/ssl.key` and `$DATADIR/apache/conf/ssl.crt` respectively. `$DATADIR` is the directory where Accelerator data files were installed. To find these files on Windows, go to `C:/ECloud/i686_win32`.

Because the certificate is self-signed, browsers complain that it is an untrusted certificate. Most organizations will want to generate a new certificate signed by a recognized certificate authority (CA) to get rid of the browser warnings. The following list summarizes the process:

- Generate a new certificate and private key
- Send the request to the CA
- Install the signed certificate

Generate a New Certificate and Private Key

1. Locate `openssl` binary and `openssl.cnf` in `$DATADIR/apache/bin/ssl`.
2. Copy `openssl.cnf` into a temporary directory.
3. Generate a new private key and certificate.
4. Enter the appropriate information for your organization when prompted.

   The most important field is the Common Name, which is the fully qualified name of the host running the Apache server where you want the certificate. This name must match the host portion of the URL used to connect to the Cluster Manager.

```
$ openssl req -config openssl.cnf -new -out $HOSTNAME.csr
Loading 'screen' into random state - done
Generating a 1024 bit RSA private key
.......++++++
.....................................................++++++
writing new private key to 'privkey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated into
your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields, but you can leave some blank.
For some fields, there will be a default value, if you enter '.', the field will
remain blank.
-----
Country Name (2 letter code) []:US
State or Province Name (full name) []:California
Locality Name (eg, city) []:Sunnyvale
Organization Name (eg, company) []:Electric Cloud
Organizational Unit Name (eg, section) []:
Common Name (eg, your websites domain name) []:myserver.mycompany.com


Email Address []:user@mycompany.com
```

   Note: This is the email address the user would contact if there are issues concerning the certificate.

```
Please enter the following 'extra' attributes to be sent with your certificate
request.
A challenge password []:
```

This information generates a new private key in `privkey.pem` and a signing request in `$HOSTNAME.csr`.

If you want to use the private key without having to enter a challenge password each time the server starts, issue the following command to strip out the password:

```
$ openssl rsa -in privkey.pem -out $HOSTNAME.key
Enter pass phrase for privkey.pem:
writing RSA key
```

This creates a PEM encoded private key file named `$HOSTNAME.key` without the password.

Send the Request to the CA

The `$HOSTNAME.csr` file generated in the previous section is a request for a certificate authority to sign the certificate. When you send this file to the CA, the CA verifies the information inside and sends you a signed certificate in response. The signed certificate includes the original certificate and the signature of the CA. Name the signed certificate '`$HOSTNAME.crt`'.

Install the Key and Signed Certificate

1. Copy the two files, `$HOSTNAME.key` and `$HOSTNAME.crt`, into `$DATADIR/apache/conf/ssl.key` and `$DATADIR/apache/conf/ssl.crt`.
2. Restart the Apache server.

   Ensure the `$HOSTNAME.key` file is readable only by the user running the Apache server process.
3. Delete the contents of the temporary directory you created because this directory contains the cryptographic information used to generate the key.

Configuring LSF

After you install the Cluster Manager:

1. Log in with the Cluster Manager Service User ID (fully enabled to start and stop LSF jobs).
2. Run a sample LSF job to make certain the user ID is ready.

   ```
   bsub -u myemail@mycompany.com -x -q <queueName> -J <jobName> ls
   ```

   If the job runs and results are returned, the user ID should be appropriate to use for Cluster Manager. (If the appropriate result is not returned, you may need to restart the server after making the necessary changes.)
3. Update the Accelerator properties file in the `conf` directory. Edit the following entries in the `accelerator.properties` file:

Agent Connect Script Initial Wait Interval

Use this option when the `agentconnect` script is run. The value is the number of seconds a host waits for agents to be activated and assigned to a build. The typical setting is between 10 and 60 seconds. A short setting might not allow enough time for LSF to schedule the job and start the agents. A setting too long and hosts that will never be used can tie up resources.

```
HM_LSF_INITIAL_WAIT=30
```

Agent Connect Script Idle Host Interval

This option identifies LSF hosts that are no longer needed. When the ElectricAccelerator Cluster Manager sees a host unused for the number of seconds specified in this setting, it cancels the corresponding LSF job. Typical values for this setting are between 3 and 30 seconds. Use a longer setting if your LSF grid is continuously using ElectricAccelerator.

```
HM_LSF_IDLE_HOST_TIME=5
```

Agent Connect Script NotFound Job Cleanup Interval

Use this option to clean up LSF jobs reported as “NOTFOUND” for the number of seconds specified as the value. There is a case in LSF that queries the LSF job against `mbatchd` and returns a “job is not found” result when actually the job is still running. Without this setting, Cluster Manager treats “NOTFOUND” as an

invalid status and removes the job on the Cluster Manager side. This action leads to LSF job synchronization issues.

The typical value for this setting is a few seconds more than the value of `MBD_REFRESH_TIME` if you are using multi-threading `mbatchd` mode. If you do not encounter the “NOTFOUND” problem, Zero value for this setting means the “NOTFOUND” status is treated as an invalid status and the job is removed from the Cluster Manager immediately.

```
HM_LSF_NOTFOUND_KILL_TIME=15
```

Agent Connect Script Max Pending Jobs

Use this option to limit the number of jobs the Cluster Manager will leave in a pending state. This option has a distinct impact on your overall build throughput. While builds are running and could use more Accelerator agents for parallel processing, Cluster Manager leaves jobs in the LSF queue in case more hosts become available. The typical setting is 2. If software builds are important and hosts become available quickly, a setting of 5 might be appropriate.

```
HM_LSF_MAX_PENDING_REQUESTS=2
```

Creating a Queue

There must be a queue that allows the Cluster Manager to submit host exclusive jobs or a queue that specifies `HJOB_LIMIT=1`.

1. Create a queue that limits jobs to the host group.
2. Edit the `conf/lsbatch/xxxx/configdir/lsb.queues` file.

   Example queue definition that allows exclusive jobs and limits those jobs to Agent/EFS hosts:

   ```
   Begin Queue
   QUEUE_NAME = eaQueue
   EXCLUSIVE = Y
   HOSTS = eaGroup
   DESCRIPTION = Queue to control ElectricAccelerator agents.
   End Queue
   ```

   Note: If you use the exclusive method, the -x option after the bsub command is required in submitJob.tcl. The -x option is not required when using HJOB_LIMIT = 1.

   Example queue definition that is NOT exclusive:

   ```
   Begin Queue
   QUEUE_NAME = eaQueue
   EXCLUSIVE = N
   HJOB_LIMIT = 1
   HOSTS = eaGroup
   DESCRIPTION = Queue to control ElectricAccelerator agents.
   End Queue
   ```

   If `EXCLUSIVE = N`, jobs from another queue may also run on the same host.

You may want to consider imposing limits on Accelerator’s use of the grid. LSF queue definitions provide a variety of controlling features. For example, you can limit how many hosts are available to Accelerator during certain time periods. See your LSF Administrator’s Guide for more information.

Additional Information

Ensure you have a standard way to construct the LSF resource query string. If not, you may encounter an issue illustrated by the following example:

If the LSF query condition is CPU > 8 and OS = windows:

User1 may put `--emake-resource="cpu>8&&OS==windows"` while User2 may put `--emake-resource="OS==windows&&cpu>8"` (which is the same resource defined in a different order).

The Cluster Manager handles these as two different resources, but LSF handles them as the same resource.

The issue manifests in this manner: If you submit two builds with the same LSF resource query string (the same CPU arch OS ...), the builds do not appear to share the started agents and they act as totally different builds with two separate resources. In the UI, you can identify the issue by hovering the cursor over the Builds page agent bars and seeing a different “Allocated” number for these two builds. If they are on the same resource, the “Allocated” value for both builds would be the same and they would share the agents fairly.

Enabling Grid Host Manager

Grid Host Manager allows Accelerator to communicate and integrate with a host grid such as LSF or Oracle Grid Engine. You must enable Grid Host Manager using cmtool:

```
cmtool modifyServer --resourceManagerType cloud
```

After you enable Grid Host Manager, the following scripts become available so you can communicate with the host grid server.

Script	Description	Input argument	Output
`checkResource.tcl`	Checks whether a resource string is valid.	Resource string in "`bsub -R`" (same as in `--emake-resource`)	"`true`" if the string is valid, "`false`" otherwise.
`initialize.tcl`	Checks whether the grid setup is ready (for example, already "`source profile.lsf`" from LSF, can run `b*` commands, and so on)	none	"`true`" if the grid setup is ready and initialized, "`false`" otherwise.
`killJob.tcl`	Kills a grid job based on job ID.	Grid job ID	none
`queryJobs.tcl`	Queries the status of all grid jobs	none	A string format: `job1Id:host1:status, job2Id:host2:status, job3Id:host:status...`
`submitJob.tcl`	Submits a grid job	Resource string, OS, host:port of the Cluster Manager	Grid job ID

Grid Host Manager Examples

checkResource.tcl

```
# returns "true" if the resource is correct, "false" otherwise.

# define all the variables
set executable "lshosts"
set resource [lindex $argv 0]
set timestamp [clock format [clock seconds] -format {%Y-%m-%d %I:%M:%S %p}]
set log "$timestamp: $executable $resource"

# main program
if { ![catch { exec $executable -R $resource } output] } {

  # logging the output
  exec echo "$log\n$output" >> cloud/cloud.log

  if [regexp {Unknown resource} $output match id] {
     puts -nonewline "false"
  } else {
  puts -nonewline "true"
  }

} else {
  # logging the error
  exec echo "$log\n$output\n$::errorInfo" >> cloud/cloud.log

        puts "error in executing $executable"
  puts $::errorInfo
  exit 1
}
```

initialize.tcl

```
# returns "true" if the cloud is setup correctly, "false" otherwise.

set timestamp [clock format [clock seconds] -format {%Y-%m-%d %I:%M:%S %p}]
set log "$timestamp: initial cloud setup check"

exec echo "$log" >> cloud/cloud.log
puts -nonewline "true"
```

submitJob.tcl

```
# submitJob.tcl - submit jobs in the cloud setup e.g. LSF/SGE
#
# this script accepts three arguments:
#   1. resource - for bsub -R <resource>
#   2. os - for LSF to invoke agentConnect in different paths under different
OSes.
#   3. host:port - the argument for agentconnect

# define all the variables
set executable "bsub"
set resource [lindex $argv 0]
set hostport [lindex $argv 2]
set queue "eaQueue"
switch [lindex $argv 1] {
  "linux" { set agentConnect "/opt/ecloud/i686_Linux/bin/agentconnect" }
   "windows" { set agentConnect "c:\\ecloud\\i686_win32\\bin\\agentconnect" }
   "solaris" { set agentConnect "/opt/ecloud/sun4u_SunOS/bin/agentconnect" }
   "solarisx86" { set agentConnect "/opt/ecloud/i686_
SunOS.5.10/bin/agentconnect" }
```

```
   default { exit 1 }
}
set timestamp [clock format [clock seconds] -format {%Y-%m-%d %I:%M:%S %p}]
set log "$timestamp: $executable $resource [lindex $argv 1] $hostport"

# main program
if { ![catch { exec $executable -R $resource -q $queue $agentConnect
$hostport } output] } {
   # logging the output
   exec echo "$log\n$output" >> cloud/cloud.log

   if [regexp {Job <(\d+)>} $output match id] {
     puts $id
   } else {
     puts $output
     exit 1
   }

} else {
   # logging the error
   exec echo "$log\n$output\n$::errorInfo" >> cloud/cloud.log

   puts "error in executing $executable"
   puts $::errorInfo
   exit 1
}
```

queryJobs.tcl

```
# queryJobs.tcl - query jobs status in the cloud setup e.g. LSF/SGE
#
# this script needs to return the following format:
#   job1Id:host1:status,job2Id:host2:status,job3Id:host:status...

# define all the variables
set executable "bjobs"
set queue "eaQueue"
set status(PEND)  "pending"
set status(PSUSP) "pending"
set status(SSUSP) "pending"
set status(USUSP) "pending"
set status(RUN)   "running"
set status(DONE)  "end"
set status(EXIT)  "failed"
set status(UNKWN) "failed"
set status(ZOMBI) "end"

set timestamp [clock format [clock seconds] -format {%Y-%m-%d %I:%M:%S %p}]
set log "$timestamp: $executable"

# main program
set result ""
if { ![catch { exec $executable -q $queue } output] } {
  # logging the output
  exec echo "$log\n$output" >> cloud/cloud.log

  if [regexp {JOBID\s+USER} $output] {
    # split the lines into a list
    set data [split $output "\n"]
```

```
    # parse the first line to get all the indices
    set line [lindex $data 0]
    set idStart 0
    set idEnd [expr [string first USER $line] - 1]
    set statStart [string first STAT $line]
    set statEnd [expr [string first QUEUE $line] - 1]
    set hostStart [string first EXEC_HOST $line]
    set hostEnd [expr [string first JOB_NAME $line] - 1]

    # remember to ignore the first line
    foreach line [lrange $data 1 end] {
       # do some line processing here
       append result [string trim [string range $line $idStart $idEnd]]:
       append result [string trim [string range $line $hostStart $hostEnd]]:
       append result [set status([string trim [string range $line $statStart
$statEnd]])],
    }

    # trim the last comma
    set result [string trimright $result ,]
  } else {
    exit 1
  }

} else {
  exec echo "$log\n$output" >> cloud/cloud.log

  if { ![regexp {No unfinished job found} $output] } {
    # logging the error
    exec echo $::errorInfo >> cloud/cloud.log

    puts "error in executing $executable"
    puts $::errorInfo
    exit 1
  }
}

puts $result
```

killJob.tcl

```
# killJob.tcl - kill jobs in the cloud setup e.g. LSF/SGE
#
# this script accepts one argument:
#   id: for bkill <id>

# define all the variables
set executable "bkill"
set id [lindex $argv 0]
set timestamp [clock format [clock seconds] -format {%Y-%m-%d %I:%M:%S %p}]
set log "$timestamp: $executable $id"

# main program
if { ![catch { exec $executable $id } output] } {
  # logging the output
  exec echo "$log\n$output" >> cloud/cloud.log

  if [regexp {Job <(\d+)> is being terminated} $output match id] {
    exit 0
  } else {
```

```
    puts $output
    exit 1
  }
} else {
  # logging the error
   exec echo "$log\n$output\n$::errorInfo" >> cloud/cloud.log

   puts "error in executing $executable"
   puts $::errorInfo
   exit 1
}
```

Documentation Roadmap

The following list is an overview of more product information and help.

- *ElectricAccelerator Electric Make Users Guide* includes:
  - Configuring Accelerator, defining your build, configuring sources/tools
  - Starting a build, Electric Make settings and options
  - Organizing builds, build classes
  - Make compatibility, output, and support
  - Third-party integration products, including Cygwin, Rational ClearCase, and Eclipse
  - Dependency management, including eDepend, the Ledger file, and managing the history data file
  - Annotation
  - Performance tuning, managing files, running multiple agents
  - Electric Make command-line options and environment variables
  - Using Electrify
- Help available within the Accelerator product includes:
  - Online help available from the Help link on each web page
  - Including access to the *ElectricAccelerator Electric Make Users Guide* and *cmtool Reference Guide* PDF files
  - Inline help describes what to type in web page fields (accessed by clicking the Show/Hide links)
- *ElectricAccelerator cmtool Reference Guide* includes:
  - An introductory chapter to familiarize you with cmtool—the Accelerator command-line tool application
  - A complete list of cmtool commands and arguments
- *ElectricAccelerator Visual Studio Integration Guide* includes:
  - Building Visual Studio solutions and projects from within the Visual Studio IDE using Electric Make
- *ElectricAccelerator Release Notes* includes:
  - Latest feature changes, fixes, and known issues
  - Install/upgrade notes
- Information on the Electric Cloud Support Web Site (https://electriccloud.zendesk.com/home) includes:

- Knowledge base articles
- User forums

Chapter 5: Upgrade

You can upgrade the software using a GUI or an interactive command-line interface (for Linux and Solaris), or by using a "silent" installation. Not all instructions are the same for each platform. Follow the instructions carefully for your particular platform.

Topics:

- Upgrading ElectricAccelerator on Linux/Solaris
- Upgrading ElectricAccelerator on Windows
- ecclusterupgrade Options

Upgrading ElectricAccelerator on Linux/Solaris

IMPORTANT: Installing any component completely replaces the current installation on that machine. Only one Accelerator installation can exist on a machine and it is whatever combination of components were chosen from the most recent install. For example, installing the Cluster Manager on a machine with the Electric Agent, results in a machine with Cluster Manager only, not a machine with the Electric Agent and Cluster Manager.

If you plan to install Electric Agent on a machine that has ElectricAccelerator Developer Edition installed, Electric Cloud recommends doing a full uninstall of ElectricAccelerator Developer Edition first.

Cluster Manager

Notes:

- Ensure no builds are running before beginning a Cluster Manager upgrade.
- The Cluster Manager server and agent hosts must run the same version of Accelerator.
- If you are changing Cluster Manager ports during the upgrade, Electric Cloud recommends closing your browser windows to avoid potential conflicts and to ensure your settings take effect.
- If you want to upgrade the Cluster Manager ***and*** migrate the database to a different machine, you must first upgrade the Cluster Manager (to update the existing database's schema), then migrate the database to the new machine.
- Remote database configuration upgrade triggers warning.

  The Cluster Manager can connect to a remote database, and the user can shut down the local database (if applicable).

  During the upgrade, the installer cannot reach the local database (because it is shut down already), so it gives this prompt:

  ```
  Unable to contact the database. Please enter a correct username/password:
  Root User (empty=no database migration):
  ```

  Leaving the value empty allows the installer to finish. Upon completion, a warning message is displayed. This is the expected behavior and does not indicate an actual issue.

- Duration of database schema upgrade triggers extraneous failure message.

  During the ElectricAccelerator upgrade process, the installer attempts to connect to the Cluster Manager. Because the Cluster Manager handles the database schema upgrade, having a large database may cause the Cluster Manager to take a long time to finish the upgrade. A failure message stating that the Cluster Manager failed to come up may be displayed. This dialog does not indicate an actual failure, it means only that the Cluster Manager is still processing the schema upgrade. The Cluster Manager will load after the schema upgrade completes.

- To back up and restore your database, use the `exportData` and `importData` cmtool commands as described in the *ElectricAccelerator cmtool Reference and Users Guide*.

- Electric Cloud no longer bundles MySQL as the default local database for ElectricAccelerator. Starting with ElectricAccelerator v7.1, Electric Cloud bundles MariaDB and installs it as the default local database on the Cluster Manager. When you perform a Cluster Manager upgrade, MariaDB replaces the local MySQL database (if applicable) and migrates all information from it.

  MariaDB is an open-source database that is fully compatible with MySQL. As open-source software, there is no enterprise license support model available with MariaDB. If using an open-source database is of concern to your organization, you continue to have the option to use any of the following databases as a replacement: MySQL, Oracle, or SQL Server. See the online help topic "Database Configuration" for information about configuring Accelerator to use an external database.

Using the GUI to Upgrade the Cluster Manager

1. Log in as root.
2. Double-click `ElectricAccelerator-<version>` to start the upgrade.

   The Welcome screen appears after the installation packages are extracted.
3. Proceed with the upgrade in the same way you initially installed Cluster Manager.
4. Ensure you type-in the correct value for your Cluster Manager admin password. The encrypted default value is *changeme*. The upgrade fails if the correct password is not used. You can change the Cluster Manager admin password during the upgrade ***only*** if the current password is the default password.
5. Accept the existing Cluster Manager configuration or update your configuration settings.
6. When the Start Copying Files screen appears, click **Next**.
7. The Installing screen displays while the installation proceeds. When installation is finished, the Complete screen displays. Click **Finish**.

   Installation is complete. The installation log file is in the install directory's root by default.

Note: If Apache fails to start properly after the Cluster Manager installs, reboot the system.

Using the Interactive Command-Line to Upgrade the Cluster Manager

1. Log in as root.
2. Run `chmod +x` on the installer (`ElectricAccelerator-<version>`) to ensure it is executable.
3. Run `./<installer filename> --mode console` to start the upgrade.
4. Enter **3** to select the **Cluster Manager** package. Press Enter.
5. Enter configuration details for the Cluster Manager upgrade.
   - Accept the default listening port you specified during initial Cluster Manager installation.
   - Accept the default database listening port you specified during initial Cluster Manager installation.
   - You can change the Cluster Manager admin password during the upgrade ***only*** if the current password is the default.
   - Migrate data from the current Cluster Manager version to preserve previously generated data. However, if the database schema has changed significantly, migration is not possible and a warning message is displayed.

If you want to upgrade the Cluster Manager ***and*** migrate the database to a different machine, you must first upgrade the Cluster Manager (to update the existing database's schema), then migrate the database to the new machine.

The installer upgrades Cluster Manager using the configuration details you entered, followed by "Installation complete" when the upgrade completes.

The installation log file is in the install directory's root, `/opt/ecloud`, by default.

Electric Agent/EFS

Notes:

- The Cluster Manager server and agent hosts must run the same version of Accelerator.
- Ensure no builds are running before beginning an Agent/EFS upgrade.
- Do ***not*** run the installer from a directory path that contains spaces.
- After agents on the grid are upgraded, they do not re-register with the Cluster Manager until a build runs. This means the Cluster Manager has old information about the agents and EFS (including outdated version information) until the next build runs and "wakes up" the agents. Initially, after an upgrade, the install log may show the upgrade has failed, but only until the next build is run.

Upgrading Electric Agent/EFS on Multiple Hosts Simultaneously

If you have used `ecconfig` to change the Cluster Manager/port any time after you initially ran the installer, you must manually update the Cluster Manager/port (where the agent hosts point to) in `/opt/ecloud/install.props` before running `ecclusterupgrade`.

1. Copy the installer executable to the Cluster Manager machine to a location such as `/var/tmp`.
2. On the Cluster Manager machine, run:

   ```
   ecclusterupgrade
   ```

3. Depending on your setup, you may require the following options as well:

   - `--agentinstalldir=<path>` indicates the agents' install location. The upgrade expects that all agents you want to upgrade have the same install location. If this is not true, you must run the script multiple times using the relevant `--agentinstalldir` path names. You must specify the `--hosts` option also to limit the upgrade to those machines that use the "non-standard" agent install directory.
   - `--targetos=<os>` limits which operating system to upgrade. <*os*> indicates which operating system to update. Available values are windows, linux, and solaris. This allows you to update any target operating system type from any operating system.
   - `--hosts="host1 host2 ..."` specifies a list of hosts to upgrade. You can also use a pattern such as `host[1-3], host[2,5]`.
   - If you have multiple installer versions, use:
     - `--installerversion=<version>` to select the version you want to use. You need this option only if there are multiple installer versions in the `local/--installerdir` location.

     or

     - `--installer=<filename>` to explicitly select an installer. Use an absolute file name for this option.

   See ecclusterupgrade Options for all ecclusterupgrade options.

Example ecclusterupgrade Command

This command upgrades only agent1 and agent2:

```
/opt/ecloud/i686_Linux/bin/ecclusterupgrade --installer=/tmp/ElectricAccelerator-
7.1.1.48690-Linux-x86_64-Install --hosts="agent1 agent2"
```

Using the GUI to Upgrade Electric Agent/EFS on a Single Cluster Host

1. Log in as root.
2. Double-click `ElectricAccelerator-<version>` to start the upgrade. The Welcome screen appears after the installation packages are extracted.

   Click **Next** to continue.
3. Select Electric Agent, and click **Next**.
4. The Electric Agent screen is filled-in with your previous configuration information. The maximum number of Agents allowed on one 32-bit machine is 32; the maximum for one 64-bit machine is 64. Click **Next** to continue to the next screen.
5. When the Start Copying Files screen appears, click **Next**.
6. The Installing screen displays while the upgrade progresses. When the upgrade is finished, the Complete screen displays. Click **Finish**.
7. Reboot the agent machine after the upgrade.

The installation log file is in the install directory's root, `/opt/ecloud`, by default.

Using the Interactive Command-Line to Upgrade Electric Agent/EFS on a Single Cluster Host

1. Log in as root.
2. Run `chmod +x` on the installer (`ElectricAccelerator-<version>`) to ensure it is executable.
3. Run `./<installer filename> --mode console` to start the upgrade.
4. Enter **1** to select the **Electric Agent/EFS** package. Press Enter.
5. Enter configuration details for the Agent/EFS upgrade.

   Note: Use the Cluster Manager default port 8030, or type-in the alternative port you specified during the initial installation.

   The installer upgrades Electric Agent/EFS using the configuration details you entered, followed by "Installation complete" when the upgrade completes.
6. You may receive a message to reboot the agent machine after upgrading Agent/EFS—rebooting may not be required.

Note: The installer dynamically builds the EFS kernel module if it detects it does not have a prebuilt version matching your Linux kernel version.

The installation log file is in the install directory's root, `/opt/ecloud` by default.

Electric Make

Upgrading Electric Make

To upgrade an Electric Make installation, simply overwrite the existing Electric Make version using the same procedure as your initial eMake installation.

Note: If you are considering uninstalling Electric Make prior to upgrading your current version, be advised that if Electric Make and Cluster Manager are installed on the same machine, you will remove Cluster Manager when you uninstall Electric Make.

Upgrading ElectricAccelerator on Windows

IMPORTANT: Installing any component completely replaces the current installation on that machine. Only one Accelerator installation can exist on a machine and it is whatever combination of components were chosen from the most recent install. For example, installing the Cluster Manager on a machine with the Electric Agent, results in a machine with Cluster Manager only, not a machine with the Electric Agent and Cluster Manager.

If you plan to install Electric Agent on a machine that has ElectricAccelerator Developer Edition installed, Electric Cloud recommends doing a full uninstall of ElectricAccelerator Developer Edition first.

Cluster Manager

Notes:

- Ensure no builds are running before beginning a Cluster Manager upgrade.
- The Cluster Manager server and agent hosts must run the same version of Accelerator.
- If you are changing Cluster Manager ports during the upgrade, Electric Cloud recommends closing your browser windows to avoid potential conflicts and to ensure your settings take effect.
- If you want to upgrade the Cluster Manager ***and*** migrate the database to a different machine, you must first upgrade the Cluster Manager (to update the existing database's schema), then migrate the database to the new machine.
- Remote database configuration upgrade triggers warning.

  The Cluster Manager can connect to a remote database, and the user can shut down the local database (if applicable).

  During the upgrade, the installer cannot reach the local database (because it is shut down already), so it gives this prompt:

  ```
  Unable to contact the database. Please enter a correct username/password:
  Root User (empty=no database migration):
  ```

  Leaving the value empty allows the installer to finish. Upon completion, a warning message is displayed. This is the expected behavior and does not indicate an actual issue.

- Duration of database schema upgrade triggers extraneous failure message.

  During the ElectricAccelerator upgrade process, the installer attempts to connect to the Cluster Manager. Because the Cluster Manager handles the database schema upgrade, having a large database may cause the Cluster Manager to take a long time to finish the upgrade. A failure message stating that the Cluster Manager failed to come up may be displayed. This dialog does not indicate an actual failure, it means only that the Cluster Manager is still processing the schema upgrade. The Cluster Manager will load after the schema upgrade completes.

- To back up and restore your database, use the `exportData` and `importData` cmtool commands as described in the *ElectricAccelerator cmtool Reference and Users Guide*.
- Electric Cloud no longer bundles MySQL as the default local database for ElectricAccelerator. Starting with ElectricAccelerator v7.1, Electric Cloud bundles MariaDB and installs it as the default local database on the Cluster Manager. When you perform a Cluster Manager upgrade, MariaDB replaces the local MySQL database (if applicable) and migrates all information from it.

  MariaDB is an open-source database that is fully compatible with MySQL. As open-source software, there is no enterprise license support model available with MariaDB. If using an open-source database is of concern to your organization, you continue to have the option to use any of the following databases as a replacement: MySQL, Oracle, or SQL Server. See the online help topic "Database Configuration" for information about configuring Accelerator to use an external database.

Upgrading the Cluster Manager

1. Log in as Administrator.
2. Double-click `ElectricAccelerator-<version>` to start the upgrade.

   The Welcome screen appears after the installation packages are extracted.
3. Proceed with the upgrade in the same way you initially installed Cluster Manager.
4. Ensure you type-in the correct value for your Cluster Manager admin password. The encrypted default value is *changeme*. The upgrade fails if the correct password is not used. You can change the Cluster Manager admin password during the upgrade ***only*** if the current password is the default password.
5. Accept the existing Cluster Manager configuration or update your configuration settings.
6. When the Start Copying Files screen appears, click **Next**.
7. The Installing screen displays while the installation proceeds. When installation is finished, the Complete screen displays. Click **Finish**.

   Installation is complete. The installation log file is in the install directory's root by default.

Note: If Apache fails to start properly after the Cluster Manager installs, reboot the system.

Electric Agent/EFS

Notes:

- The Cluster Manager server and agent hosts must run the same version of Accelerator.
- Ensure no builds are running before beginning an Agent/EFS upgrade.
- Do ***not*** run the installer from a directory path that contains spaces (such as the Windows desktop).
- After agents on the grid are upgraded, they do not re-register with the Cluster Manager until a build runs. This means the Cluster Manager has old information about the agents and EFS (including outdated version information) until the next build runs and "wakes up" the agents. Initially, after an upgrade, the install log may show the upgrade has failed, but only until the next build is run.

Upgrading Electric Agent/EFS on Multiple Hosts Simultaneously

If you run `ecclusterupgrade` on a Windows 7 Cluster Manager under a user account other than the "administrator" account, you may encounter a "permission denied" error. This is due to a Windows 7 issue. The workaround is to run cmd using "Run as administrator" and then run `ecclusterupgrade`.

1. Copy the installer executable to the Cluster Manager machine to a location such as `c:\temp`.
2. On the Cluster Manager machine, cd to the Accelerator `bin` directory and run:

   ```
   ecclusterupgrade.exe
   ```

3. Depending on your setup, you may require the following options as well:

   - `--agentinstalldir=<path>` indicates the agents' install location. The upgrade expects that all agents you want to upgrade have the same install location. If this is not true, you must run the script multiple times using the relevant `--agentinstalldir` path names. You must specify the `--hosts` option also to limit the upgrade to those machines that use the "non-standard" agent install directory.
   - `--targetos=<os>` limits which operating system to upgrade. *<os>* indicates which operating system to update. Available values are windows, linux, and solaris. This allows you to update any target operating system type from any operating system.
   - `--hosts="host1 host2 ..."` specifies a list of hosts to upgrade. You can also use a pattern such as `host[1-3], host[2,5]`.
   - If you have multiple installer versions, use:
     - `--installerversion=<version>` to select the version you want to use. You need this option only if there are multiple installer versions in the `local/--installerdir` location.

     or

     - `--installer=<filename>` to explicitly select an installer. Use an absolute file name for this option.

   See ecclusterupgrade Options for all ecclusterupgrade options.

Example ecclusterupgrade Command

This command upgrades only agent1 and agent2:

```
ecclusterupgrade.exe --installer=c:\tmp\ElectricAccelerator-7.1.1.48690-Windows-
Install.exe --hosts="agent1 agent2"
```

Upgrading Electric Agent/EFS on a Single Cluster Host

The Agent/EFS upgrade process for a single host machine is similar to a new Agent/EFS installation, with one *very important* exception, detailed below.

1. Log in as Administrator.
2. Double-click `ElectricAccelerator-<version>` to start the upgrade. The Welcome screen appears after the installation packages are extracted.

   Click **Next** to continue

3. Select Electric Agent, and click **Next**.
4. The Electric Agent screen is filled-in with your previous configuration information. The maximum number of Agents allowed on one 32-bit machine is 32; the maximum for one 64-bit machine is 64. Click **Next** to continue to the next screen.
5. If you previously specified an Agent to run as a specific user, the selection is filled-in as you specified.

   At this point, you can revert to the ElectricAccelerator default user if you need the CSRSS, svchost, or OLE support service(s) for your builds—these services are not available to you because you previously

specified a specific user. On a single host, use of these services requires each agent service to run as a different user.

To change the service user to allow access to the CSRSS, svchost, and OLE support services, select ECloud Internal User. This means the ECloud user account is used to run agent services. If you do not want to use ElectricAccelerator default accounts, use `ecconfig` on each agent host to assign distinct users to each agent.

To use the `ecconfig` option *after* upgrading your agent machines, refer to Using the ecconfig Command-Line Option.

6. When the Start Copying Files screen appears, click **Next**.
7. The Installing screen displays while the upgrade progresses. When the upgrade is finished, the Complete screen displays. Click **Finish**.
8. Reboot the agent machine after the upgrade.

The installation log file is in the install directory's root by default.

Electric Make

Upgrading Electric Make

To upgrade an Electric Make installation, simply overwrite the existing Electric Make version using the same procedure as your initial eMake installation.

Note: If you are considering uninstalling Electric Make prior to upgrading your current version, be advised that if Electric Make and Cluster Manager are installed on the same machine, you will remove Cluster Manager when you uninstall Electric Make.

Using the `ecconfig` Command-Line Option

Open a command-line window to use `ecconfig`. The following table is the output from running `#ecconfig.exe -help`.

Argument	Description
`-cm value`	Host:port of the Cluster Manager the agent host should point to.
`-numagents value`	This is the number of agents to start <*xx*>. If you modify `numagents`, you must restart the agents for your change to take effect. After the restart, the server database is updated and the agents become active.
`-agentusers value`	Specify agent usernames and passwords: `-agentusers "username1/password1 username2/password2...usernameN/passwordN."` **Note:** You must specify the same number of username/password pairs as the number of agents. The argument must be encapsulated in quotes.

Argument	Description
`-agentuser value`	Specify a username and password for a specific agent: `-agentuser "agentID username/password"` where agent ID is the numerical ID of the agent to configure (between 1 and `numagents`).
`-lsfhost value`	y or n. Type “y” to specify this machine is part of an LSF grid or “n” for no.
`-secureconsole value`	y or n. Type “y” to secure the agent console port or “n” for no.
`-tempdir value`	This is the location where agents should store temp files. Blank for system `tempdir`.
`-help or -?`	Print this message.

ecclusterupgrade Options

Option	Description
`--admin-password=`*value*	The Cluster Manager admin user's password. Default is 'changeme'.
`--agentinstalldir=`*path*	The agent side install directory. You must specify the `--hosts` option also to limit the upgrade to those machines that use the “non-standard” agent install directory.
`--batchsize=`*int*	The number of hosts to process per iteration. This allows you to break the upgrade of a large cluster into more manageable smaller batches to avoid timeouts. Default is 20, the recommended maximum.
`--cm=host[:`*port*`[:`*securePort*`]]`	Cluster Manager whose hosts are to be upgraded. Default is localhost:8030:8031.
`--cm-secure-port=`*port*	Cluster Manager secure port. Default is 8031.
`--gen-hostlist`	Query the Cluster Manager to get the list of hosts, write it to a file, and exit.
`--help`	Display this table's help text.
`--hosts="host1 host2 ..."`	List of hosts to upgrade. Can be a pattern like host[1-3], host[2,5]. This switch supersedes `--cm`.

Option	Description
`--installer=`*`filename`*	The installer executable to use. Use an absolute filename for this option. If the installer is not in the expected directory, you must provide the full path to the installer, even when you specify `--installerdir`.
`--installerdir=`*`dirname`*	The directory containing the installer executables.
`--installerversion=`*`version`*	The installer version (for example, 7.0.0.12345). This option is needed only if there are multiple installer versions in the *`local`*`/--installerdir` location.
`--installtimeout=`*`value`*	The number of minutes to wait for the agent install to finish. Default is 20 minutes.
`--logfile=`*`value`*	Log file to write messages to. By default, messages are printed to: `/var/tmp` (Linux and Solaris) or `C:\` (Windows) in the form of ecloud_node_upgrade_ YYYMMDD_HHMMSS.log
`--newcm=`*`hostname/ip`*	The host name or IP address of the new Cluster Manager where you want to migrate the upgraded agents.
`--remotetmpdir=`*`path`*	The location where the remote install is run.
`--targetos=`*`os`*	Agent machine OS (windows, linux, solaris).

Chapter 6: Uninstall

If you wish to uninstall the software, follow the procedures described in the following topics.

Topics:

- Uninstalling Accelerator on Linux/Solaris
- Uninstalling Accelerator on Windows

Uninstalling Accelerator on Linux/Solaris

Note: The uninstaller removes all ElectricAccelerator components from a machine at the same time. If you are uninstalling as part of an upgrade procedure, follow the instructions in Upgrading ElectricAccelerator on Linux/Solaris.

1. Log in as root.
2. Change to the `tmp` directory by entering `# cd /tmp`
3. Copy the uninstaller `uninstall-accelerator` to the `/tmp` directory by entering:
   `# cp /opt/ecloud/uninstall-accelerator /tmp`
4. Invoke the uninstaller in console mode by entering `./uninstall-accelerator /mode console`

   The system displays the following:

   ```
   This will completely remove ElectricAccelerator from your system.  Are you sure
   you want to do this? [n/Y]
   ```

5. Enter `y` to confirm the uninstall.

   The system displays the following:

   ```
   A full uninstall will remove all leftover files, including other packages such
   as ElectricInsight.

   Perform a full uninstall [y/N]
   ```

6. Enter `y` or `n`.

   The system displays the following:

   ```
   Back up local database [y/N]
   ```

7. Enter `y` or `n`.

   No second opportunity to confirm the uninstall request is displayed. The uninstall begins immediately.

   You will see an "uninstall complete" message when Accelerator software is removed.

8. Check the `/etc/sysconfig/ecagent.conf` directory (for Linux) or `/etc/ecagent.conf` directory (for Solaris) and remove the file `ecagent.conf` if you want to delete *all* Accelerator files.

Uninstalling Accelerator on Windows

Note: The uninstaller removes all ElectricAccelerator components from a machine at the same time. If you are uninstalling as part of an upgrade procedure, follow the instructions in Upgrading ElectricAccelerator on Windows.

1. Go to the Electric Cloud installation directory and run `uninstall-accelerator.exe`.

   Note: For Windows systems running Windows Server 2008 or later, the user must right-click the uninstaller and select Run as administrator.

2. Select any of the following before continuing with the uninstall process:

- Perform a full uninstall - Removes all leftover files, including other packages such as Visual Studio Integration and ElectricInsight.
- Back up local database - Retains a copy of your local database.

You will see an "uninstall complete" message when Accelerator software is removed. If Electric Agent/EFS was installed, you may be prompted to reboot.

Index

A

Accelerator
- cluster usage 1-4
- compatibility matrix 2-12
- component interactions 1-3
- functionality 1-2
- ports 2-10

accelerator.properties file 4-8

Active Directory 4-8

agent
- disabling by default 4-8
- installation 3-5, 3-13
- log rotation 4-15
- logs 4-17
- temporary storage location 4-16

antivirus 3-2

Apache server certificates 4-20

B

backward compatibility package 3-25

browser requirements 2-10

build environment consistency 2-13

C

certificates
- Apache server 4-20

checksum utility 2-14

Cluster Manager
- installation 3-2, 3-11
- Linux requirements 2-11

compatibility matrix 2-12

configuration
- agent log rotation 4-15
- agent temporary storage location 4-16
- disk cache directory 4-16
- email notification 4-13
- grid 4-22
- Linux 4-2
- LSF 4-22
- Solaris 4-2
- Windows 4-2

Cygwin 2-12

D

database
- connection monitor 4-15
- supported versions 2-11

default
- agent temporary storage location 4-18
- disk cache directory location 4-18
- install directories 4-17
- log file locations 4-17

document roadmap 4-28

E

ecclusterupgrade 5-4, 5-7, 5-10
- options 5-10

email notification 4-13

eMake
backward compatibility package 3-25
installation 3-9, 3-17
migration 3-25

G

grid 1-5
configuration 4-22
host manager 4-24
planning 2-13
resource management 4-19

H

hardware requirements 2-9

I

installation
additional options 3-25
agent 3-5, 3-13
backward compatibility package 3-25
Cluster Manager 3-2, 3-11
command-line options 3-18
eMake 3-9, 3-17
eMake only 3-25
from Cygwin shell 3-11
Linux 3-2, 3-5, 3-9
location limitations 3-1
logs 4-17
properties file 3-24
silent 3-18
Solaris 3-5, 3-9
Windows 3-11, 3-13, 3-17
integration
with the grid 1-5

L

LDAP 4-8
license 4-18
Linux
configuration 4-2
file descriptor limits 3-5
install information 3-2
installation 3-2, 3-5, 3-9
known kernel issue 2-4
supported versions 2-2
upgrading 5-2
log in 4-18
log rotation 4-15
logs
Accelerator 4-8
agent 4-17
build 4-8
changing locations 4-8
install 4-17
LSF 1-5, 2-13, 4-24
configuration 4-22

M

multiple agents
upgrading 5-4, 5-7, 5-10
mysqlcheck 3-2

P

path settings 3-10
port use 2-10
properties file 3-24
protocol mismatch error text 4-15

R

registry
information 4-6
requirements
browser 2-10
database 2-11
hardware 2-9
resource management 4-19

S

silent install 3-18

Solaris
configuration 4-2
install information 3-2
installation 3-5, 3-9
supported versions 2-7
upgrading 5-2
supported
Accelerator versions 2-12
browsers 2-10
build tools 2-8
ClearCase 2-8
Cygwin 2-12
databases 2-11
GNU Make 2-8
Linux platforms 2-2
NMAKE 2-8
RHEL platforms 2-2
SLES platforms 2-3
Solaris platforms 2-7
Ubuntu platforms 2-4
Visual Studio 2-8
Windows platforms 2-5

T

toolchain 2-11

U

umask 3-2
uninstall
on Linux 6-2
on Windows 6-2
upgrading
Linux 5-2
multiple agents 5-4, 5-7, 5-10
Solaris 5-2
Windows 5-6
user authentication 4-8

W

Windows
configuration 4-2
important notes 4-5
install information 3-10
installation 3-11, 3-13, 3-17
installer activity 3-10
registry information 4-6
supported versions 2-5
upgrading 5-6