

ElectricCommander 5.x Installation Guide

Electric Cloud, Inc.
www.electric-cloud.com

ElectricCommander 5.x

Copyright © 2002 – 2015 Electric Cloud, Inc. All rights reserved.

Published 5/29/2015

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

Electric Cloud products—ElectricAccelerator, ElectricCommander, ElectricInsight, and Electric Make—are commonly referred to by their “short names”—Accelerator, Commander, 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: Introduction to ElectricCommander	1-1
Unique Functionality	1-1
Challenges Solved by ElectricCommander	1-3
Architecture	1-3
Local Configuration	1-4
Remote Web Server Configuration	1-4
Clustered Configuration	1-6
Chapter 2: System Requirements and Platforms	2-1
Server Platforms	2-1
Agent Platforms	2-4
Server and Agent Compatibility	2-5
Hardware Requirements	2-5
Browser Requirements	2-7
Port Usage	2-7
Default Server Ports	2-7
Default Database Ports	2-7
Avoiding Port Conflicts	2-7
Database Requirements	2-8
Built-In Database	2-8
Supported Alternate Databases	2-8
Database Sizing	2-8
Disk Usage	2-9
Memory Usage	2-10
Checksum Utility	2-11
Software Licenses	2-11
Standard License	2-11
Enterprise License	2-11
Legacy Software Licenses	2-12
Chapter 3: Installing ElectricCommander	3-1
Choosing the Correct Installation Process	3-1
User Interface Installation	3-1
Interactive Command-Line Installation	3-2
Silent Unattended Installation	3-2
Non-Server Platform Agent Installation	3-2
Before You Install ElectricCommander	3-3
Default Installation Directories	3-3
Graphical User Interface Installation Methods	3-4

Running an Express Server Graphical User Interface Installation	3-4
Running an Advanced Graphical User Interface Installation	3-9
Running an Express Agent Graphical User Interface Installation	3-16
Interactive Command-line Installation Methods	3-21
Running an Express Server Command-Line Installation	3-21
Running an Advanced Command-Line Installation	3-22
Running an Express Agent Command-Line Installation	3-25
Silent Unattended Installation Method	3-27
Running a Silent Install	3-27
Silent Install Arguments	3-28
Linux Silent Installation Examples	3-31
Windows Silent Installation Examples	3-33
Non-Server Platform Agent Installation Method	3-34
Installing Agents with the UNIX Installer	3-34
UNIX Agent Silent Installation	3-36
Installing the MySQL JDBC Driver	3-37
Logging Into the ElectricCommander Web Interface	3-37
Chapter 4: Clustering	4-1
Benefits from Clustering	4-1
Architecture of a Commander Cluster	4-1
Resource, Agent, and Procedure Configuration Considerations	4-2
Default Local Resource Use	4-2
Separate Local Agents For Improved Performance	4-3
Pool Local Agents For Improved Reliability	4-3
Procedure Strategies	4-3
Agent Resource Strategies	4-3
Database Restriction	4-4
Unsupported Host	4-4
Software for Clustering	4-4
Apache Zookeeper	4-4
Load Balancer	4-4
Dependencies for Clustering	4-4
Running a Cluster in Single-Server Mode	4-5
Configuring Clustering	4-5
Separating Agents from Commander Servers	4-6
Preparing Your Cluster Resources	4-6
Installing and Configuring a Load Balancer	4-7
Installing ZooKeeper	4-7
Running ZooKeeper as a Service on Linux	4-8
Running ZooKeeper as a Service on Windows	4-9
Ensuring ZooKeeper Can Locate Java	4-10
Verifying that ZooKeeper is Running	4-10
Exhibitor Software	4-10
Configuring a Multi-ZooKeeper Cluster	4-11
ZooKeeper Requires a Majority of Nodes to Be Up	4-11
Installing Commander Software	4-11

New Commander Installation for Reliability	4-12
New Commander installation for Performance	4-12
Converting an Existing Commander installation for Reliability	4-13
Converting an Existing Commander Installation for Performance	4-14
Duplicating Repository Contents	4-14
Running a Cluster in Clustered Mode	4-15
Running a Cluster in Single-Server Mode	4-16
Adding the Configuration to ZooKeeper	4-17
Uploading Configuration Files to ZooKeeper	4-17
Prerequisites	4-18
Location	4-18
Command	4-18
Import Files	4-18
Next Steps	4-19
Configuring Web Server Properties	4-19
Configuring Repository Server Properties	4-19
Configuring Commander Agents	4-20
Configuring the Cluster Workspace	4-21
Configuring Commander Repositories	4-21
Adding Trusted Agents to Clusters	4-21
Separating Agents from Commander Servers	4-21
Verifying Commander Services	4-22
Accessing Commander with Clustering	4-22
Additional Ways to Improve a Commander Cluster	4-22
Third-Party Software	4-22
Commander Components	4-23
Chapter 5: Configuration	5-1
Applying an Enterprise License Key	5-1
External Database Configuration	5-1
Database Interactions	5-2
Database User	5-2
Configuring Commander to Use an Alternate Database	5-2
Setting the Database with the Web Interface	5-2
Setting the Database from a Command Line	5-3
Universal Access to the Plugins Directory	5-7
Configuring Universal Access for a Network Location	5-7
Replicating the Plugin Directory on Remote Systems	5-10
Environment Proxy Server Configuration	5-10
Configuring Proxy Settings for Servers	5-10
Testing Server Proxy Settings	5-11
Configuring Proxy Agents	5-11
Increasing File Descriptors on Linux	5-12
Adjusting Swappiness on Linux	5-12
Chapter 6: Roadmap to Upgrade ElectricCommander	6-1
Chapter 7: Upgrading from ElectricCommander 4.2.x to ElectricCommander 5.x	7-1
Upgrade Process	7-3

Properties Changed to UUIDs	7-6
Upgrading to ElectricCommander 5.x	7-7
Reasons for the Change	7-7
Impact for Commander 4.2.x Users	7-7
Actions to Take	7-9
Migration Tool in the Upgrade Tool	7-9
Default Job Name and Workflow Name Template Changes	7-10
Disadvantages of Using jobId and workflowId	7-10
Estimated Durations for Database Upgrades	7-10
Choosing the Correct Installation Method	7-11
User Interface Installation	7-11
Interactive Command-Line Installation	7-12
Silent Unattended Installation	7-12
Repository Server With a Commander Upgrade	7-12
Stand-Alone Repository Server or Web Server Upgrade	7-12
Non-Server Platform Agent Upgrade	7-13
Preparing for the Upgrade	7-13
Upgrade Testing	7-13
Backups	7-13
Upgrade Installer Preservation	7-14
MySQL Upgrades	7-14
External MySQL Database Characteristics	7-14
Upgrade Tool	7-14
Downloading and Extracting the Upgrade Tool	7-15
Troubleshooting the Upgrade Tool	7-15
Migration Tool	7-15
What the Migration Tool Does	7-15
Running the Migration Tool	7-16
jobCounter Considerations	7-17
Troubleshooting: upgrade.log File	7-18
Using the upgrade.properties File Templates	7-18
Database Upgrade Process	7-19
Before the Database Upgrade	7-20
What You Need to Do	7-20
What Your DBA Needs to Do	7-20
Running the Database Upgrade	7-21
Upgrade Tool Overview	7-22
Using the ElectricCommander 4.2.x Built-In Database in ElectricCommander 5.x	7-23
Troubleshooting the Database Upgrade	7-24
Troubleshooting: Using ectool Options	7-25
Troubleshooting: upgrade.log File	7-25
Troubleshooting: Checking the Status of the Database Upgrade	7-25
Troubleshooting: Error Messages	7-26
Warning Regarding the Commander 4.2.x Database Schema	7-26
Table or View Does Not Exist	7-26
Access Denied	7-27
Troubleshooting: When the Database Username and Password Change	7-27

Problem	7-27
Solution	7-27
Troubleshooting: When the Source and Target Databases Are on Different Servers	7-27
Problem	7-27
Solution	7-28
Troubleshooting: Using the --final Flag	7-28
Problem	7-28
Solution	7-28
Troubleshooting Oracle: Upgrade Fails	7-29
Troubleshooting: Not Enough Disk Space for the Database Upgrade	7-29
Problem	7-29
Solution	7-29
Troubleshooting: Upgrade Tool Can Fail to Detect Database Changes	7-30
User Interface Upgrade Method	7-31
Interactive Command-line Upgrade Method	7-34
Silent (Unattended) Upgrade Method	7-35
Copying Repository Contents	7-36
Chapter 8: Upgrading from ElectricCommander 5.0.x to ElectricCommander 5.x With Clustering	8-1
Upgrading Clusters	8-1
Choosing the Correct Upgrade Method	8-2
Preparing for Your Upgrade	8-4
User Interface Upgrade Method	8-5
Interactive Command-Line Upgrade Method	8-8
Silent (Unattended) Upgrade Method	8-9
Chapter 9: Upgrading from ElectricCommander 5.0.x to ElectricCommander 5.x Without Clustering	9-1
Choosing the Correct Upgrade Method	9-1
Preparing for Your Upgrade	9-3
User Interface Upgrade Method	9-4
Interactive Command-Line Upgrade Method	9-6
Silent (Unattended) Upgrade Method	9-7
Chapter 10: Uninstalling ElectricCommander	10-1
Uninstalling Commander on Windows	10-1
Uninstalling Windows 2008 or Windows 7	10-1
Uninstalling Windows XP or Windows 2003	10-1
Uninstalling Commander on UNIX	10-2
Chapter 11: Maintenance	11-1
Commander Server Backups	11-1
Data Backup Methods	11-1
Preparing for a Backup	11-3
Backing Up a Commander Server	11-3
Commander Server Restores	11-4
Preparing for a Restore	11-4
Restoring Your Commander Server	11-4

Apache Web Server or Agent Certificates	11-11
Generating a CA Request	11-11
Sending the CA Request	11-11
Installing the Signed Certificate	11-11
Using chkconfig	11-11
Starting and Stopping Servers and Agents Manually	11-12
Stop All Commander Agent Services	11-12
Stop All Commander Server Services	11-12
Start All Commander Agent Services	11-13
Start All Commander Server Services	11-13
Log File Locations	11-14
Agent Logs	11-14
Server Logs	11-14
Web Server Logs	11-14
Repository Server Logs	11-15
Installer Logs	11-15
Switching to an Alternate Database	11-15
Web Interface Online Help System	11-16
Chapter 12: Troubleshooting	12-1
Commander Server is Unresponsive and Displays an OutOfMemory Error	12-1
Commander Fails to Install on a Virtual Machine (VM)	12-2
Windows PHP Does not Handle Time Zones Correctly	12-3
Commander Server Certificate Fails Security Scan	12-3
Linux Upgrade Breaks Symbolic Links	12-5
Index	13-1

Chapter 1: Introduction to ElectricCommander

ElectricCommander (also referred to as Commander) is an enterprise-class solution for automating the software build and release process. ElectricCommander helps teams make software build, package, test, and deployment tasks more repeatable, more visible, and more efficient. At its core, ElectricCommander is a web-based system for automating and managing the build and release process. It provides a scalable solution, solving some of the biggest challenges of managing these “back-end” software development tasks.

Unique Functionality

ElectricCommander is the most scalable solution on the market. Only ElectricCommander provides enterprise-class scalability for build and release management. It is easy to install and use on a simple build, yet scales to support the largest and most complex build and test processes. The ElectricCommander multi-threaded Java server provides efficient synchronization even under high job volume.

Facilities provided by ElectricCommander:

- **Workflow functionality**

Use Workflows to design and manage processes at a higher level than individual jobs. Workflows allow you to combine procedures into processes to create build-test-deploy lifecycles (for example). A workflow contains states and transitions you define to provide complete control over your workflow process. The Commander Workflow feature allows you to define an unlimited range of large or small lifecycle combinations to meet your needs.

- **Continuous Integration Manager (CI Manager)**

This feature provides a front-end user interface for creating, managing, and monitoring continuous integration builds. The CI Manager dashboard provides:

- Visually see your running builds, build progress, build status, and historical build outcomes.
- Easily accessed “Actions” to configure a continuous integration build.
- Quick configuration of your preferred SCM system.
- A project can contain any number of continuous integration builds, depending on the work you have already set up for your procedures/steps to perform.

- **Resource management**

If a resource is over committed, Commander delays some jobs until others have finished with the resource. You can define pools of equivalent resources and Commander balances the load across the pool.

- **Access control**

Users log into the system and Commander uses their information to control activities. Privileges can be set for individuals or groups to ensure the security you need.

- **Preflight Build functionality**

Developers can build and test code changes in isolation on their local machines before those changes are committed to a production build.

- **Search, sort, and filter functions**

Minimize the display of information that is of no interest to you, and quickly retrieve the information you need.

- **Detailed job information**

Commander records a variety of information about each job. You can view jobs and see step run times, successes, and failures.

- **Email notifications**

Get important information or data to individuals or groups immediately and on a regular basis for a particular job or a specific job aspect.

- **Powerful and flexible reporting facilities**

Various statistics such as the number of compiles or test errors are collected after each step and recorded in the Commander database. A variety of reports can be generated from this information.

- **Artifact management**

Use artifacts to improve performance across builds, provide better reusability of components, and improve cross-team collaboration with greater traceability. For example, instead of each developer repeatedly downloading third-party packages from external source, these components can be published and versioned as an artifact. A developer then simply retrieves a specific artifact version from a local repository, guaranteeing a consistent package from build to build.

- **ElectricCommander command-line tool**

All Commander features are available from a command-line tool (ectool), a Perl API (ec-perl), and a web interface.

- **Plugin capability**

Commander has an extensible UI which enables easy development of plugins. You can integrate with other tools, use custom dashboards, and create unique user experiences based on roles.

- **Workspaces**

Commander creates a workspace for each job. A workspace is a disk area jobs can use for storage.

- **Data models based on properties**

Properties are used to store job input data such as the source code branch to use for the build, to collect data during a job (such as number of errors or warnings), and to annotate the job after it completes (for example, a build has passed QA).

- **Zones and gateways**

A zone (or top-level network) is a way to partition a collection of agents to secure them from use by other groups. A gateway is a secured connection between two zones when you want to share or transfer information to another zone. For example, you might want a developers zone and a test zone. The Commander server is a member of the default zone, created during Commander installation.

Challenges Solved by ElectricCommander

Traditional software build processes face the following challenges:

- Wasted time on script-intensive, manual, home-grown systems

These systems are error prone, do not scale well, and have little or no management visibility or reporting.

- Multiple, disconnected build and test systems across locations

Disconnected build and test systems result in redundant work and the inability to share/reuse code files across teams, making it painful to manage build and test data.

- Slow overall build and release cycles

Slow cycle times directly impact release predictability and time-to-market.

ElectricCommander addresses these problems with a three-tier architecture, AJAX-powered web interface, and first-of-its-kind build and release analytic capabilities for reporting and compliance. With this solution, your developers, release engineers, build managers, QA teams, and managers gain:

- A shared platform for disseminating best practices and reusing common procedures
- The ability to support geographically distributed teams
- Continuous integration and greater agility
- Faster throughput and more efficient hardware utilization
- Visibility and reporting for better project predictability
- Better software quality by integrating and validating against all target platforms and configurations

Architecture

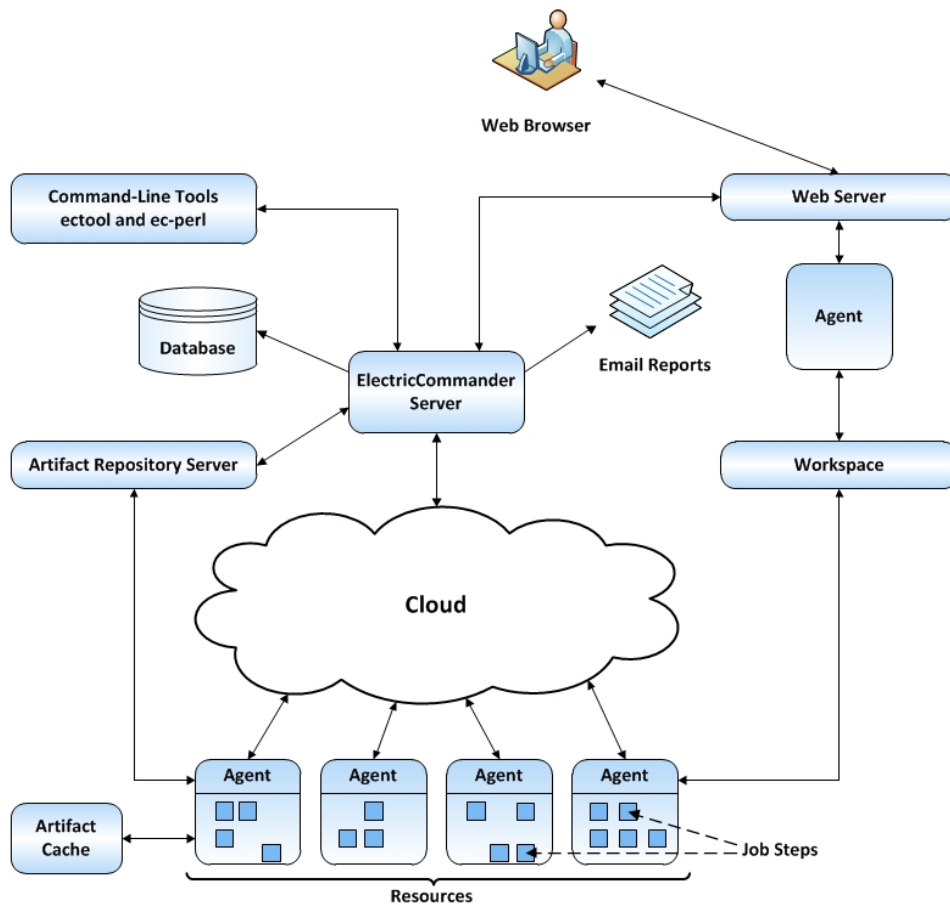
ElectricCommander was architected from the ground up to support enterprise scale software production. Based on a three-tier architecture, ElectricCommander scales to handle large, complex environments. Commander's multi-threaded Java server provides efficient synchronization even under high job volume.

Topics:

- [Local Configuration](#) on page 1-4
- [Remote Web Server Configuration](#) on page 1-4
- [Clustered Configuration](#) on page 1-6

Local Configuration

The following diagram shows an ElectricCommander architecture configuration at a single site.



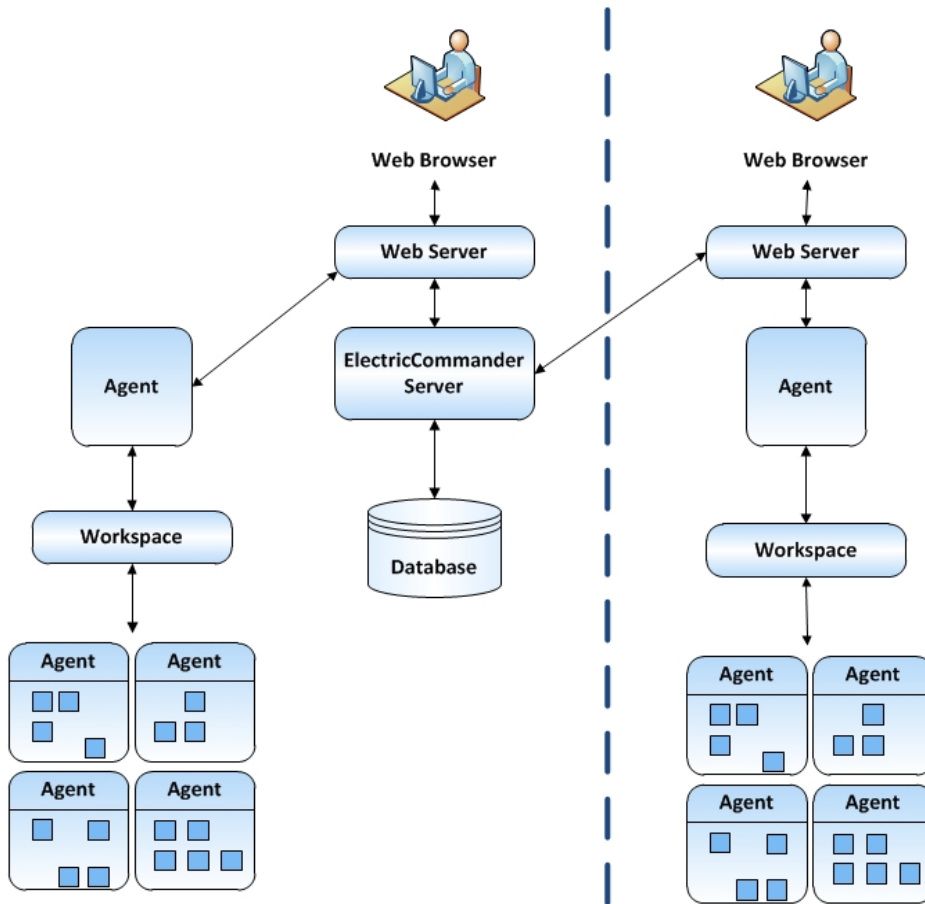
In the local configuration:

- The Commander server manages resources, issues commands, generates reports.
- An underlying database stores commands, metadata, and log files.
- Agents execute commands, monitor status, and collect results, in parallel across a cluster of servers for rapid throughput.

If you are only evaluating the ElectricCommander software, the database, Commander server, web server, and repository server can reside on the same machine. In a production environment, the database should reside on a separate machine from the Commander server to prevent performance issues. It is acceptable for the Commander server, web server, and repository server to reside on the same machine in a local configuration, but not required.

Remote Web Server Configuration

The following diagram shows an example remote web server architecture configuration.



In the example remote web server configuration:

- There are web servers at each site
- The database and Commander server is located at your headquarters
- Proxy resources exist at each site

This remote web server configuration helps prevent network latency. If you have multiple sites, Commander can be configured in numerous ways to help you work more efficiently.

Commander supports multiple workspaces, including those co-located on agents that use them. In this architecture, step log files are created locally so even the largest log files can be captured without a performance penalty. You can view the step log files remotely from the web UI, but there will be a performance penalty if the files must be retrieved across the WAN. This means that remote users will experience the performance penalty when the web server retrieves the step log file contents, and when the contents are sent back across the WAN to the browser.

To minimize these performance issues, install one central Commander server, and then install a Commander web server at each remote site. The remote Commander web servers should be co-located with the remote agents and workspaces so remote users to log in through their local web server. Any operations initiated from the remote location, including running a job, are completed by the central Commander server. Job data is retrieved from the central server when a remote user views the Job Details page. If the job is using a workspace at the remote user's site, the links to all step log files will refer to local paths. The log files are only accessed by the remote web server's agent, and not the Commander server. This eliminates both trips across the WAN

which improves performance. The Commander web server reads the log file locally (via its agent), and then displays the page to the user whose browser is also on the same side of the WAN.

Note: You can install Commander web servers on any Windows or Linux platform suitable for installing the Commander server.

Clustered Configuration

You can also add horizontal scalability and high availability to your Commander environment by adding additional machines to create a clustered Commander configuration.

A clustered Commander configuration has the following benefits:

- Increase the supported number of simultaneous jobs and corresponding API requests
- Expand capacity over time by adding additional Commander servers
- Distribute API requests across multiple Commander servers
- Add fault tolerance by re-routing jobs to running Commander servers
- Distribute Commander requests across multiple web servers

A clustered Commander configuration requires two additional software components:

- A centralized service for maintaining and synchronizing group services in cluster
- A load balancer for routing work to machines in the cluster

See [Clustering](#) on page 4-1 for additional details and clustered configuration set up procedures.

Chapter 2: System Requirements and Platforms

This section describes hardware and software specifications and configurations for installing and running ElectricCommander on Windows or UNIX systems. All version requirements for operating systems and databases are routinely tested and fully supported by Electric Cloud. Contact Electric Cloud Customer Support if you have any questions regarding newer software versions.

Server Platforms

This section describes the supported platforms for the ElectricCommander, web, or repository servers.

Windows Platforms

The following table lists all supported Microsoft Windows server platforms.

Platform	Notes
Windows 8.1 (32 and 64-bit)	—
Windows 7 (32 and 64-bit)	<ul style="list-style-type: none">• SP1 recommended• An administrator might need to disable User Account Control (UAC). If the installer runs under account X, but services will run under account Y, installation directories (both program and data) will probably have permissions that prevent Y's access. This applies particularly to data directories.
Windows Server 2012 Windows Server 2008 and Windows 2008 R2 (32 and 64-bit)	<ul style="list-style-type: none">• An administrator might need to disable User Account Control (UAC). If the installer runs under account X, but services will run under account Y, installation directories (both program and data) will probably have permissions that prevent Y's access. This applies particularly to data directories.
Windows Server 2003 (32 and 64-bit)	<ul style="list-style-type: none">• SP2 recommended but not required
Windows XP (32 and 64-bit)	<ul style="list-style-type: none">• SP2 or later required for hosts running Windows XP

Linux Platforms

The following table lists all supported Linux server platforms.

Platform	Notes
Red Hat Enterprise Linux 6 (32 and 64-bit)	<ul style="list-style-type: none"> Do not choose “nobody” for the RHEL user. RHEL does not allow a “<code>su - nobody -c foo.sh</code>” because it is not a shell account. Using “nobody” prevents the SQL server from starting, which means the default ElectricCommander database is inaccessible. If you plan to use Commander on RHEL 6.x, 64-bit, you must install some 32-bit libraries that were omitted by Red Hat. These libraries are required for the Commander installation executable file to work. We recommend running <i>all</i> of these commands on your RHEL 6, 64-bit machines. <ul style="list-style-type: none"> Run the <code>yum install libstdc++.i686</code> command. If you do not run this command, the Commander Apache server will not start and the installer silently fails for any type of Commander installation. Run the <code>yum install libuuid.i686</code> command if you are performing a Commander installation that includes an Apache server. If you are installing Commander agents only, without a web server, you do not need to run this command on each agent machine. Run the <code>yum install nss-pam-ldapd*.i686</code> command (to install 32-bit NSS packages) if using an LDAP account for ownership of the server/web/repository services. If you do not run this command, the Commander Apache server fails to start.
Red Hat Enterprise Linux 5 (32 and 64-bit)	<ul style="list-style-type: none"> Do not choose “nobody” for the RHEL user. RHEL does not allow a “<code>su - nobody -c foo.sh</code>” because it is not a shell account. Using “nobody” prevents the SQL server from starting, which means the default ElectricCommander database is inaccessible. If you plan to use Commander on RHEL5.x, the supported installation method requires the SeLinux configuration to be less restrictive. Either run the <code># setenforce 0</code> command as the root user to set the SeLinux operating mode to “Permissive”, or the file <code>/etc/selinux/config</code> and change the line “SELINUX=enforcing” to “SELINUX=permissive”. Either change requires rebooting the Linux server to have the changes take effect.

Platform	Notes
Ubuntu 10.04 – 12.04 (32 and 64-bit)	<ul style="list-style-type: none"> • Run <code>sudo apt-get update && sudo apt-get install ia32-libs</code> before installing ElectricCommander. This command installs Ubuntu's 32-bit compatibility layer. • Update <code>/etc/environment</code> to include the <code>ElectricCommander bin</code> directory in <code>PATH</code>. Steps running with impersonation on Ubuntu run with a <code>PATH</code> environment variable set in <code>/etc/environment</code>. As a side-effect, the <code>ElectricCommander bin</code> directory is not in the <code>PATH</code> in the impersonation context, so calls to tools like <code>ectool</code> and <code>postp</code> fail with a "not found" error. • If you plan to use an LDAP account for ownership of the <code>server/web/repository</code> services with Ubuntu 12.04 64-bit, you must run <code>sudo apt-get update && apt-get install libnss-ldap:i386</code>. This command installs 32-bit NSS packages. If you do not run this command, the Commander Apache server fails to start. • If you receive an error during installation similar to: <pre>File "/usr/lib/lsb/install_initd", line 3, in <module> import sys, re, os, initdutils File "/usr/lib/lsb/initdutils.py", line 18 raise ValueError, 'need a file or string' ^ SyntaxError: invalid syntax</pre> <p>This is a known Ubuntu bug. To fix it, run <code>sudo sed -i "s/python3/python/" /usr/lib/lsb/install_initd</code></p>

Platform	Notes
Ubuntu 12.10 – 14.04 (32 and 64-bit)	<ul style="list-style-type: none"> Run these commands before installing ElectricCommander. They install the Ubuntu 32-bit compatibility layer. <pre> sudo dpkg --add-architecture i386 sudo apt-get install lib32bz2-1.0 sudo apt-get install libuuid1:i386 </pre> Update <code>/etc/environment</code> to include the ElectricCommander <code>bin</code> directory in <code>PATH</code>. Steps running with impersonation on Ubuntu run with a <code>PATH</code> environment variable set in <code>/etc/environment</code>. As a side-effect, the ElectricCommander <code>bin</code> directory is not in the <code>PATH</code> in the impersonation context, so calls to tools like <code>ectool</code> and <code>postp</code> fail with a <i>not found</i> error. If you plan to use an LDAP account for ownership of the server/web/repository services with Ubuntu 14.04 64-bit, you must run <code>sudo apt-get update && apt-get install libnss-ldap:i386</code>. This command installs 32-bit NSS packages. If you do not run this command, the Commander Apache server fails to start. If you receive an error during installation similar to: <pre> File "/usr/lib/lsb/install_initd", line 3, in <module> import sys, re, os, initdutils File "/usr/lib/lsb/initdutils.py", line 18 raise ValueError, 'need a file or string' ^ SyntaxError: invalid syntax </pre> <p>This is a known Ubuntu bug. To fix it, run <code>sudo sed -i "s/python3/python/" /usr/lib/lsb/install_initd</code></p>

Agent Platforms

The following table lists all supported agent platforms.

Platform	Notes
All server machines specified for the Commander server	For more information, see Server Platforms on page 2-1.
Microsoft Windows 2000	–
Sun Solaris 9 (Architecture: Sparc only)	–
Sun Solaris 10 (Architecture: Sparc and Intel x86)	–

Platform	Notes
HP-UX 11i v1 (11.11) or later (Architecture: PA-RISC 2.0)	<ul style="list-style-type: none"> • Make sure the following patches (or patches superceding these patches) are installed: <ul style="list-style-type: none"> • PHKL_29243 • PHSS_39077 • HP-UX Secure Shell requires that a random number generator be located on the system. It searches for <code>/dev/urandom</code> and <code>/dev/random</code> (in that sequence) on the system and uses the first device it finds. If it fails to locate these two devices, HP-UX Secure Shell uses its own internal random number generator program. By default, HP-UX 11i v2 systems includes these random number devices. These devices can also be obtained for HP-UX 11i v1 by downloading and installing the HP-UX Strong Random Number Generator from http://software.hp.com. HP recommends that Secure Shell users on HP-UX 11i v1 systems install the Strong Random Number Generator product as it significantly speeds up program initialization and execution time for some commands.
SUSE Linux Enterprise Server 12.1 (32 and 64-bit)	–
SUSE Linux Enterprise Server 11.4 (32 and 64-bit)	–
SUSE Linux Enterprise Server 10.3 and 10.4 (32 and 64-bit)	–
Mac OS X 10.4 (Tiger) or later (Architecture: Intel)	–

Server and Agent Compatibility

Servers running ElectricCommander 5.0 or later no longer support pre-ElectricCommander 4.2 agents. The pre-ElectricCommander 4.2 ectool/Perl API no longer communicates with the ElectricCommander server. You must upgrade any pre-Commander 4.2 agents before you upgrade the ElectricCommander server.

Note: ElectricCommander 4.0 and earlier are no longer supported. Contact Electric Cloud Customer Support for manual instructions to upgrade your installation.

Hardware Requirements

This section lists the minimum requirements for any Windows or Linux machines installed with the ElectricCommander server software.

- Processor clock rate: 1.5 GHz or higher
- Memory: 2 GB available RAM or more (8 GB recommended for small to medium deployments)

- Cores: 2 or more (4 cores recommended for small to medium deployments)

Browser Requirements

ElectricCommander supports the following web browsers:

- Microsoft Internet Explorer 10.0 or later
- Mozilla Firefox 10.0 or later
- Chrome 13 or later

Note: Web browser extensions such as AdblockPlus for Google Chrome can interfere with the display of Commander web pages. We recommend disabling any ad blocking browser extensions or add an exclusion for ElectricCommander web pages.

Port Usage

ElectricCommander is pre-configured to use certain ports by default. This section contains the default port values and actions you can take to avoid port conflicts.

Default Server Ports

ElectricCommander servers use the following ports:

- 8000 - ElectricCommander Server
- 8443 - ElectricCommander Server (SSL port)
- 80 - ElectricCommander Web Server
- 443 - ElectricCommander Web Server (SSL port)
- 6800 - Port used by the ElectricCommander agent for HTTP communication on the localhost network interface.
- 7800 - ElectricCommander Agents (by default, this is an HTTPS port)
- 61613 - Preflight file transfer port, other file transfer, event notifications, or other messaging
- 8200 - Artifact repository server (by default, this is an HTTPS port)

Default Database Ports

Alternate databases use the following ports:

- 1521 - Oracle 11g Release 1, and R2
- 1433 - Microsoft SQL Server
- 3306 - MySQL

Avoiding Port Conflicts

If you are installing a ElectricCommander server and you have a web server or other application that uses the same ports as the ElectricCommander host, you can choose one of the following actions:

- Select different web server or application ports
- Uninstall the existing web server or application
- Disable the existing web server or application
- Reconfigure the existing web server or application to use another port

IMPORTANT: Make sure your firewall is open for the ports used by ElectricCommander.

Database Requirements

You are not able to log into ElectricCommander until a database is set up. During the ElectricCommander server installation, you can choose to install the built-in [default] database or an alternate database.

Note: If you are using two different ElectricCommander servers, the servers **cannot** point to the same database.

Built-In Database

If you install ElectricCommander using the built-in ElectricCommander database, be advised that this database is not supported for production systems. We strongly recommends using an alternate database in your production environment. **Using an alternate database requires a Commander enterprise license.**

You can switch to an alternate database at anytime. For more information, see [Configuration](#) on page 5-1.

Supported Alternate Databases

Alternate databases must be UTF-8 compliant, and configured to allow up to 200 open connections. ElectricCommander supports the following alternate databases:

- MySQL 5.0, 5.1, 5.5.12, 5.6 or later (5.5.12 or later is recommended)

Notes:

- For clean installs of the Commander server 4.2.3 or later, you must install the MySQL JDBC driver. See [Installing the MySQL JDBC Driver](#) on page 3-37.
- For upgrades, additions to my.cnf/my.ini are required. See [Installing the MySQL JDBC Driver](#) on page 3-37.
- MS SQL Server 2005 with Service Pack 2 or later, or MS SQL Server 2008, 2008 R2
- Oracle 10g R2 (including RAC)
- Oracle 11g Release 2, R2

Database Sizing

Expected database growth over time can be correlated with the number of job steps created. Database growth is NOT correlated with build log or build artifact sizes.

To create a reasonable database growth estimate per period:

1. Estimate the number of jobs per period.
2. Multiply the "estimated number of jobs" by the number of steps estimated per job. This will determine the estimated number of steps per period.
3. Multiply the "estimated number of steps per period" by 10 to determine the disk size (in Kbytes) required per period.

For example: If you run 500 jobs per day with an average of 200 steps per job, you would run 100K steps per day. This means your database would grow about 1 GB per day or 90 GB per quarter. Using this example, if you prune jobs older than 30 days, database size could be maintained at about 30 GB.

Disk Usage

Disk space usage varies and depends on the quantity and size of the jobs you run. We recommend starting with the following free space recommendations:

- **Server**—10 GB
- **Agents**—5 GB each

Sizing artifact cache directory space on resources

By default, artifacts are retrieved into the `<data-dir>/artifact-cache` directory of the agent installation. You can modify the `agent.conf` file to change the location, or you can specify the cache directory location on each resource known to Commander. Determining how much free space the cache partition needs to accommodate all of your artifact versions can be difficult. Here is one approach to approximate the disk/partition size you need:

For each artifact, estimate how large you think each version will be and how many versions you plan to keep. Compute the total required space to be the sum of `version-size * numVersions` for each artifact. Add a buffer of 50%. Using your end result, allocate a disk/partition that size and configure the cache as a directory on that disk/partition.

- **Repository server**—If using Artifact Management functionality, you could need 20-30 GB, or more disk space for your repository server.

Although a server install includes an artifact repository, We recommend that production repository servers be installed on different machines than the Commander server. The repository server might do a very large amount of disk and network I/O when transferring artifact versions to and from requesters and this might adversely affect Commander server performance.

Sizing the repository backingstore

For a repository installation, by default, the repository backingstore is the `<data-dir>/repository-data` directory. You can modify the `<data-dir>/conf/repository/server.properties` file or use `ecconfigure` to update the backingstore location. Determining exactly how much free space the backingstore disk/partition needs to accommodate your artifact versions can be difficult. Here is one approach to approximate the disk size you need:

For each artifact, estimate how large you think each version will be and how many versions you plan to keep. Compute the total required space to be the sum of `version-size * numVersions` for each artifact. Add a buffer of 50%. Using your end result, allocate a disk/partition that size and configure the repository backingstore as a directory on that disk/partition.

- **Logs**

You can set the following properties as Java system properties in `wrapper.conf`:

- `ec.logRoot` controls the location of the log output. The default location is the `logs/commander.log` directory.
- `ec.logHistory` controls the number of days of log history that is kept. The default is 30 days.
- `ec.logSize` controls the size of each log file before it is zipped up and a new log file started. The default is 100 MB.

To limit the amount of disk space for logging, the most effective approach is to use a lower `ec.logHistory` value.

Memory Usage

Memory usage varies depending on whether or not the Commander server is a dedicated machine.

- A Commander server running on a dedicated machine has a default minimum heap memory allocation of 20% and a maximum heap memory allocation of 40%. This applies to either a 32 or 64-bit system.
- A Commander repository server has a default minimum heap memory allocation of 10%.
- A Commander agent has a default minimum memory usage of 16 MB and a maximum memory usage of 64 MB.

There are two ways you can adjust the amount of memory for the Commander server.

- Modify the `wrapper.java.initmemory.percent` and `wrapper.java.maxmemory.percent` lines in `wrapper.conf`

Use the following table to determine the correct directory path.

Server Type	System	Path
Non-repository	Windows XP	c:\Documents and Settings\All Users\Application Data\Electric Cloud\ElectricCommander\conf\wrapper.conf
	Windows 2003	
	Windows 2008	c:\ProgramData\Electric Cloud\ElectricCommander\conf\wrapper.conf
	Windows 7	
	Linux	/opt/electriccloud/electriccommander/conf/wrapper.conf
Repository	Windows XP	c:\Documents and Settings\All Users\Application Data\Electric Cloud\ElectricCommander\conf\repository\wrapper.conf
	Windows 2003	
	Windows 2008	c:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\wrapper.conf
	Windows 7	
	Linux	/opt/electriccloud/electriccommander/conf/repository/wrapper.conf

- Use `ecconfigure` to set the initial and maximum memory settings.

For example, to set the CommanderServer initial memory percentage to 21% and the maximum memory percentage to 31%, enter the following command:

```
ecconfigure --serverInitMemory 21 --serverMaxMemory 31
```

To adjust the amount of memory for the Commander agent, modify the `wrapper.java.initmemory.percent` and `wrapper.java.maxmemory.percent` lines in `wrapper.conf` for the agent. Use the appropriate directory path:

- Windows: `c:\ProgramData\Electric Cloud\ElectricCommander\conf\agent\wrapper.conf`
- Linux: `/opt/electriccloud/electriccommander/conf/agent/wrapper.conf`

Checksum Utility

An MD5 checksum file is available on the Electric Cloud FTP site. If you choose to verify that Commander 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 Windows, Linux, and Mac operating systems.

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

To use the MD5 checksum utility on a Mac:

1. In Finder, browse to `/Applications/Utilities`.
2. Double-click the Terminal icon. A Terminal window appears.
3. In the Terminal window, type: “md5” (followed by a space).
4. Drag the downloaded file from the Finder into the Terminal window.
5. Click in the Terminal window, press the Return key, and compare the checksum displayed on the screen to the one on the download page.

Software Licenses

The Licenses page in the ElectricCommander web interface shows you the current software use entitlements. You can view your license information by going to **Administration > Licenses**.

Standard License

The standard license allows you to use only the built-in database. In ElectricCommander 4.2 and later, a *standard license* is packaged with the server and is automatically installed if no other license is present.

The following figure shows an example of an ElectricCommander server with a standard license.

The screenshot shows the ElectricCommander web interface. The top navigation bar includes links for Home, Projects, Jobs, Workflows, Cloud, Artifacts, Search, and Administration. The Administration section is expanded, showing sub-links like Event Log, Groups, Users, Licenses, Directory Providers, Email Configurations, Database Configuration, Plugins, Server, Source Control, and Defect Tracking. The Licenses page is active, displaying a table of licenses and a section for current usage.

Company	Feature	Expiration	Grace Period (Days)	Actions
Electric Cloud, Inc.	Server (standard)		0	View License Delete

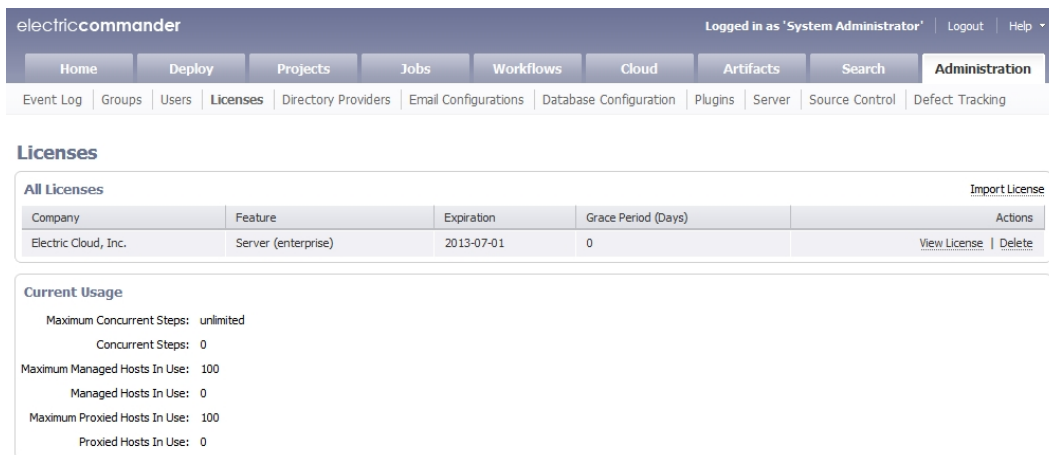
Current Usage

- Maximum Concurrent Steps: 2
- Concurrent Steps: 0
- Maximum Managed Hosts In Use: unlimited
- Managed Hosts In Use: 0
- Maximum Proxied Hosts In Use: unlimited
- Proxied Hosts In Use: 0

Enterprise License

The enterprise license allows you to use an external (alternate) database. In ElectricCommander 4.2 and later, you must apply an enterprise license key when you want to configure an alternate database.

The following figure shows an example of an ElectricCommander server with an enterprise license.



The screenshot shows the ElectricCommander web interface. The top navigation bar includes links for Home, Deploy, Projects, Jobs, Workflows, Cloud, Artifacts, Search, and Administration. The Administration section is active, showing a sub-menu with Event Log, Groups, Users, Licenses, Directory Providers, Email Configurations, Database Configuration, Plugins, Server, Source Control, and Defect Tracking. The Licenses page is displayed, showing a table of licenses and current usage statistics.

Licenses

All Licenses [Import License](#)

Company	Feature	Expiration	Grace Period (Days)	Actions
Electric Cloud, Inc.	Server (enterprise)	2013-07-01	0	View License Delete

Current Usage

- Maximum Concurrent Steps: unlimited
- Concurrent Steps: 0
- Maximum Managed Hosts In Use: 100
- Managed Hosts In Use: 0
- Maximum Proxied Hosts In Use: 100
- Proxied Hosts In Use: 0

Legacy Software Licenses

Before ElectricCommander 4.2, *all licenses were enterprise licenses*. If you are a legacy customer, and you perform a clean server installation, you can install your original (pre-ElectricCommander 4.2) license (which is an enterprise license) on the new server and configure the server to talk to the production database. This approach is possible only when *no other server* is running and connected to the production database.

Note: If you want to install a new server (for example, for test purposes), do *not* install your original license on the new server. Instead, install the standard license on the test server if your needs fit within the constraints of this license. You can then experiment with test servers without having to contact Electric Cloud Technical Support.

Chapter 3: Installing ElectricCommander

This section describes the various methods for installing the ElectricCommander software in a new environment.

IMPORTANT: If you are upgrading a previously installed Commander version, follow the instructions in [Roadmap to Upgrade ElectricCommander](#) on page 6-1.

Choosing the Correct Installation Process

This section describes the various installation methods and available options for specific platform types. For information about supported server platforms and supported non-server platforms, see [Server Platforms](#) on page 2-1 and [Agent Platforms](#) on page 2-4.

User Interface Installation

This method provides an installation Wizard you use to install the Commander software on a supported server platform. This installation method is generally preferred by Windows users, but is supported on Linux platforms with X installed.

Installation options:

- **Express Server**

This option installs the Commander server, web server, repository server, built-in database, one agent (for running jobs), and Commander tools on one machine with the default Commander server settings. This installation method is most useful for quickly installing the Commander software for evaluation purposes.

IMPORTANT: Running Commander on a single machine with the built-in database is generally not recommended for a production environment. Commander will connect to an external database in a typical production configuration.

If Commander was installed with the built-in database, you can reconfigure it to use an external database. Go to [External Database Configuration](#) on page 5-1 for configuration instructions.

- **Advanced**

This installation option allows you to select individual components, directories, or ports. You can use this option to install the Commander server, web server, and repository server on a machine without the built in database.

- **Express Agent**

This option quickly installs a Commander agent and Commander tools. Use this option for managed hosts where you intend to run job steps. This installation method is useful if you need to install an individual agent. If you need to install multiple agents, you might prefer to use the [Silent Unattended Installation](#) on page 3-2 method.

Interactive Command-Line Installation

This method provides an interactive command-line for installing the Commander software on a supported server platform. This installation method is only available for Linux platforms.

Installation options:

- **Express Server**

This option installs the Commander server, web server, repository server, built-in database, one agent (for running jobs), and Commander tools on one machine with the default Commander server settings. This installation method is most useful for quickly installing the Commander software for evaluation purposes.

IMPORTANT: Running Commander on a single machine with the built-in database is generally not recommended for a production environment. Commander will connect to an external database in a typical production configuration. If Commander was installed with the built-in database, you can reconfigure it to use an external database. Go to [External Database Configuration](#) for the configuration instructions.

- **Advanced**

This installation option allows you to select individual components, directories, or ports. You can use this option to install the Commander server, web server, and repository server on a machine without the built in database.

- **Express Agent**

This option quickly installs a Commander agent and Commander tools. Use this option for managed hosts where you intend to run job steps. This installation method is useful if you need to install an individual agent. If you need to install multiple agents, you might prefer to use the method called [Silent Unattended Installation](#) on page 3-2.

Silent Unattended Installation

This method provides a non-interactive command-line installation for supported server platforms. You might find this installation method preferable for installing multiple remote agents and servers.

Installation options:

- **Windows**

This option is only for Windows platforms.

- **Linux**

This option is only for Linux platforms.

Non-Server Platform Agent Installation

This method provides a command-line for installing the Commander agent and tool software on supported non-server platforms. This installation method is only supported on non-server platforms.

Installation options:

- **Command-Line Agent**

This option allows you to install an agent from a UNIX command-line installer.

- **Silent Agent**

This option allows you to run unattended (silent) installations with the UNIX installer.

Before You Install ElectricCommander

Review the following information before attempting to install any ElectricCommander software.

- You must install Commander on a local drive. Electric Cloud does not support installing the Commander server on a network volume.
- We recommend installing the ElectricCommander server first, before installing remote agents or web servers.
- Make sure you have completed any prerequisite platform setup. For more information, see [Server Platforms](#) on page 2-1 and [Agent Platforms](#) on page 2-4.
- The Commander-provided built-in database is **not** recommended for production systems. We strongly recommend using an alternate database in your production Commander environment. Additionally, the built-in database is **not** supported in a clustered Commander configuration. See [Supported Alternate Databases](#) on page 2-8 for alternate databases supported by Commander.
- If the X Window System is not running or not available, the Linux user interface installer will run in interactive command-line mode.
- When installing a 64-bit machine, the 64-bit version of the Java Runtime Environment is installed automatically.
- When installing an agent, repository server, and/or web server, you can enter information for a remote Commander server. That information can then be used to discover the server's plugins directory and set it accordingly so the local install is in sync with the remote Commander server. During an agent installation, you can create a resource object on the server automatically. Similarly, during a repository installation, you can create a repository object on the server automatically.
- If you plan to use a clustered Commander configuration, refer to [Clustering](#) on page 4-1 to read about additional requirements and considerations.

Default Installation Directories

Commander uses the following default installation directories:

Platform	Data Type	Default Path
Windows	Program files	C:\Program Files\Electric Cloud\ElectricCommander
	Data (database, logs, configuration files)	Windows 2008 or Windows 7 <ul style="list-style-type: none">• C:\ProgramData\Electric Cloud\ElectricCommander
		Windows XP or 2003 <ul style="list-style-type: none">• C:\Documents and Settings\All Users\Application Data\Electric Cloud\ElectricCommander
UNIX/Mac	All program files and data	/opt/electriccloud/electriccommander

Note: You can change the installation directories when you install the Commander software.

Graphical User Interface Installation Methods

The graphical user interface installation methods are supported by Windows platforms and Linux platforms running the X Window System.

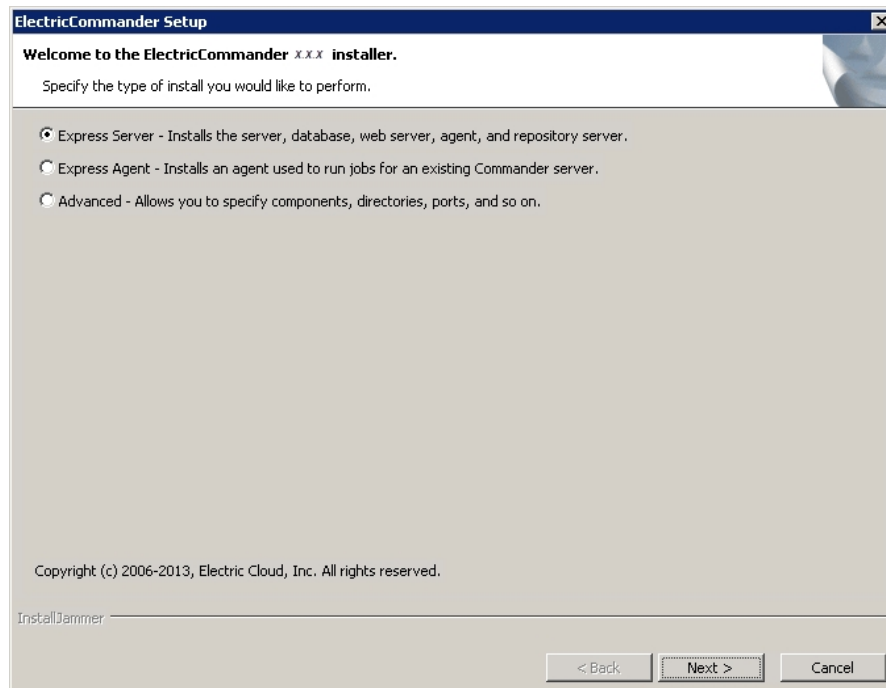
Running an Express Server Graphical User Interface Installation

The express server installation installs the Commander server, including the web server, built-in database, agent for running jobs, and Commander tools. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

Note: The built-in database is not supported in a clustered Commander configuration.

1. Double-click the `ElectricCommander-<version>` file to run the installer.

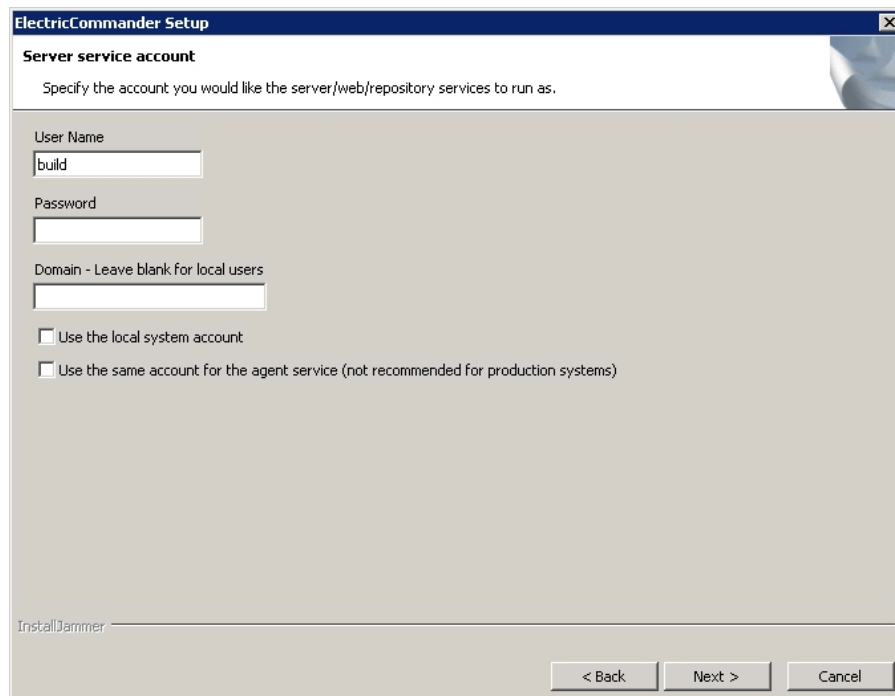
The Welcome to the ElectricCommander Installer screen appears.



Note: The screen examples in this procedure are from a Windows system. Different options will appear in some windows on a Linux system.

2. Select the **Express Server** installation option, and then click **Next** to continue.

The Server Service Account screen appears.



3. Select the appropriate step for your platform and complete the information for the server service account.

○ If you have a Windows system:

- **User Name** - Use this field to enter the name of the user who will run the Commander server, web server, and repository server services.
- **Password** - Use this field to enter the password of the user who will run the Commander server, web server, and repository server services.
- **Domain** - Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.
- **Use the local system account** - Select this check box if you want the Commander server, repository server, and web server services to run as the local Windows system account.

Note: The local system account cannot access network resources such as shared filesystems used for plugins or workspaces.

- **Use the same account for the agent service** - Select this check box if you want the agent on the Commander server machine to run as the same account.

For security reasons in production environments, you might want to use a separate account for the agent service because the server account has permission to read the key file (/opt/electriccloud/electriccommander/conf/passkey in Linux or C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey in Windows). The key file is used to decrypt passwords stored in Commander. Using a different account for the agent service ensures that a process running on the agent cannot gain access to the key file.

○ If you have a Linux system:

- **User Name** - Use this field to enter the name of the user who owns the Commander server, repository server, and web server processes.
- **Group Name** - Use this field to enter the name of the group who owns the Commander server, repository server, and web server processes.
- **Use the same account for the agent service** - Select this check box if you want the same user and group to own the agent process on the Commander server machine.

For security reasons in production environments, you might want to use a separate user and group for the agent service because the server service has permission to read the key file (/opt/electriccloud/electriccommander/conf/passkey in Linux or C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey in Windows). The key file is used to decrypt passwords stored in Commander. Using a different user and group for the agent service ensures that a process running on the agent cannot gain access to the key file.

4. Click **Next** to continue.

The Agent Service Account screen appears.

IMPORTANT: If you selected the **Use the same account for the agent service** check box on the previous screen, you will not see this screen.

The screenshot shows a window titled "ElectricCommander Setup" with a close button in the top right corner. The main heading is "Agent service account" with a subtitle "Specify the account you would like the agent service to run as." Below this are three text input fields: "User Name" (containing "build"), "Password", and "Domain - Leave blank for local users". There is a checkbox labeled "Use the local system account" which is currently unchecked. At the bottom left, there is a progress indicator labeled "InstallDammer" with a horizontal line. At the bottom right, there are three buttons: "< Back", "Next >", and "Cancel".

5. Select the appropriate step for your platform and complete the information for the agent service account.

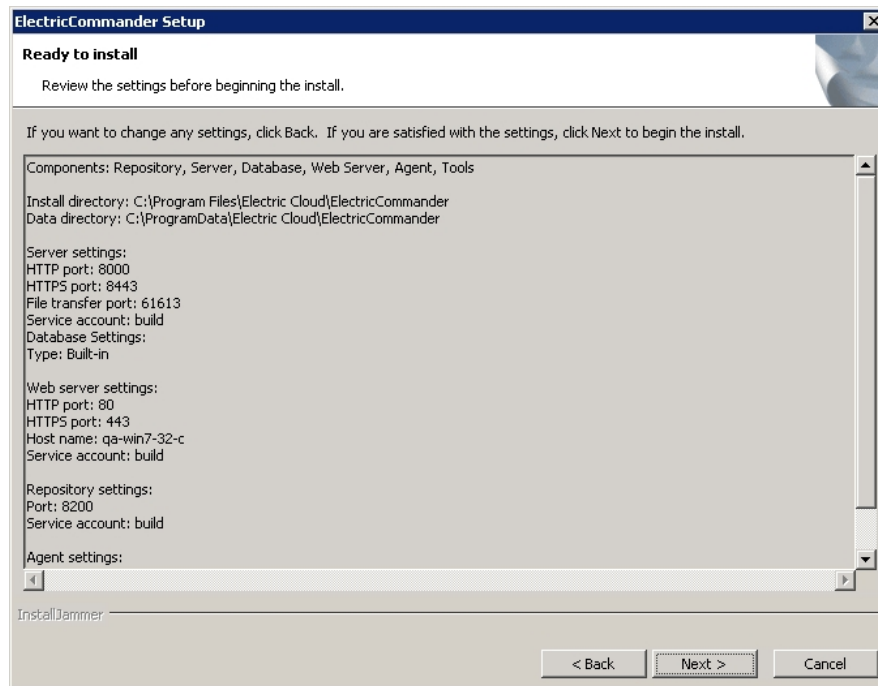
- If you have a Windows system:
 - **User Name** - Use this field to enter the name of the user who will run the Commander agent service.
 - **Password** - Use this field to enter the password of the user who will run the Commander agent service.
 - **Domain** - Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.
 - **Use the local system account** - Select this check box if you want the Commander agent service to run as the local Windows system account.

Note: The local system account does not have access to network shares.

- If you have a Linux system:
 - **User Name** - Use this field to enter the name of the user who owns the Commander agent process.
 - **Group Name** - Use this field to enter the name of the group that owns the Commander agent process.

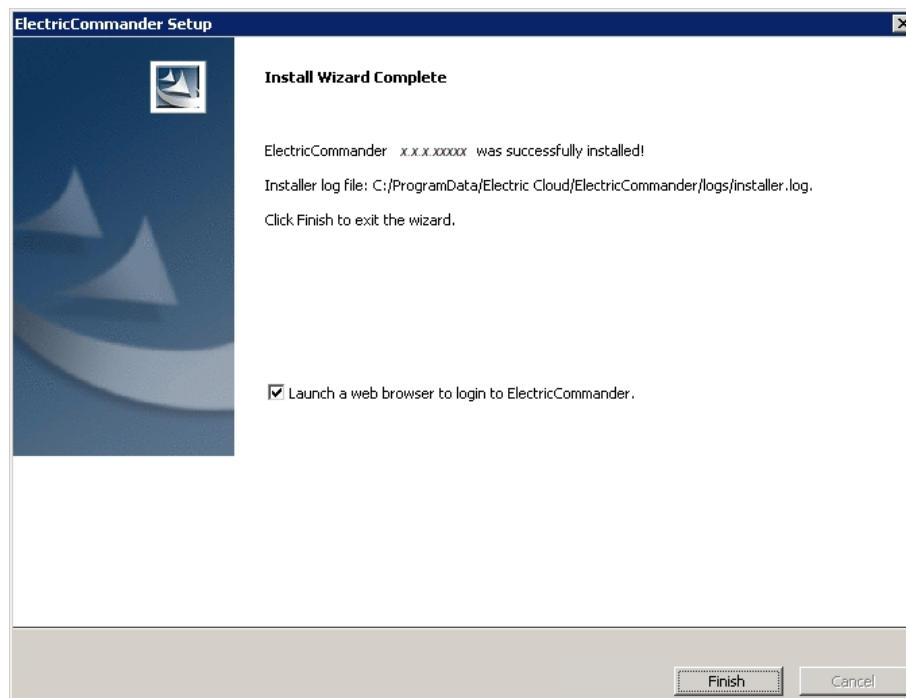
6. Click **Next** to continue.

The Ready to Install screen appears.



7. Review the default settings and your service account selections. Use the **Back** button to change your service account selections if necessary.
8. Click **Next** to continue.

The installer displays a status bar to show the progress of the installation process which can take fifteen minutes. When the install process is complete, the Install Wizard Complete screen appears.



Note: The Commander server will automatically start when installation is complete.

9. Select the **Launch a web browser to login to ElectricCommander** check box if you want Commander to open the login screen now.
10. Click **Finish** to close the wizard.

Running an Advanced Graphical User Interface Installation

The advanced graphical user interface installation allows you to install individual Commander components such as a Commander server, web server, repository server, or Commander tools on specific machines. You can also change the default installation settings to accommodate your environment. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

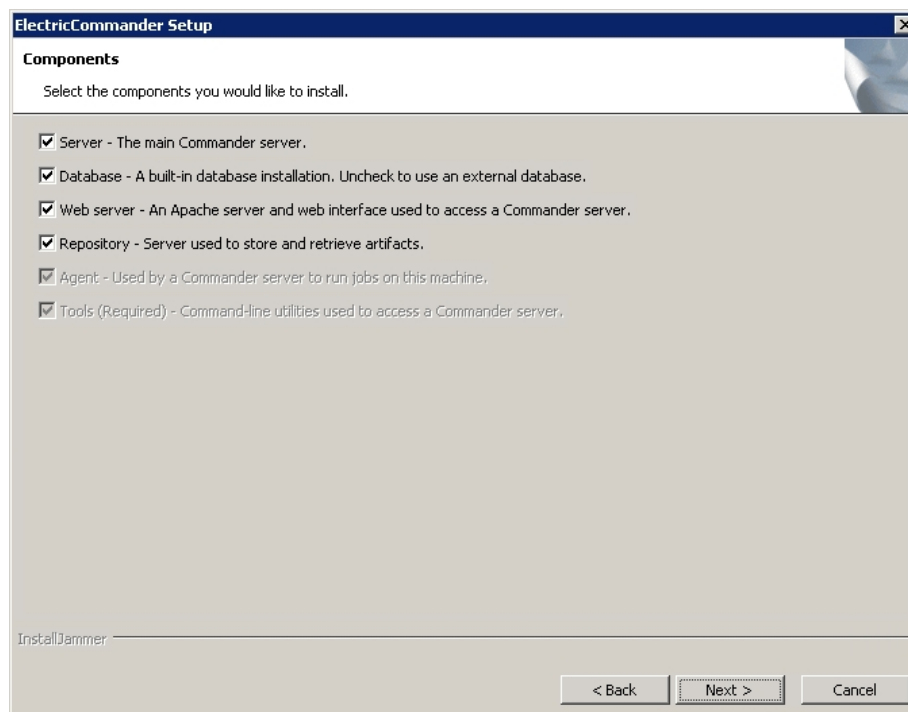
1. Double-click the `ElectricCommander-<version>` file to run the installer.

The Welcome to the ElectricCommander Installer screen appears.

Note: The screen examples in this procedure are from a Windows system. Different options will appear in some windows on a Linux system.

2. Select the **Advanced** installation option, and then click **Next** to continue.

The Components screen appears. All options are selected by default.



3. Clear the check boxes for any servers you do **NOT** want to install on the current machine. For more information, see [Architecture](#) on page 1-3.

Available options are:

Server - Select this check box if you want to install a Commander server.

Database - Only select this check box if you want to install the built-in database. This is not recommended for production systems. Additionally, the built-in database is **not** supported in a clustered

Commander configuration. Clear this check box if you plan to use an external database. If you plan to use MySQL, see [Installing the MySQL JDBC Driver](#) on page 3-37.

Web server - Select this check box if you want to install an Apache web server.

Repository - Select this check box if you want to install a Commander repository server. If you select this option, a single agent is automatically installed on the machine to run jobs.

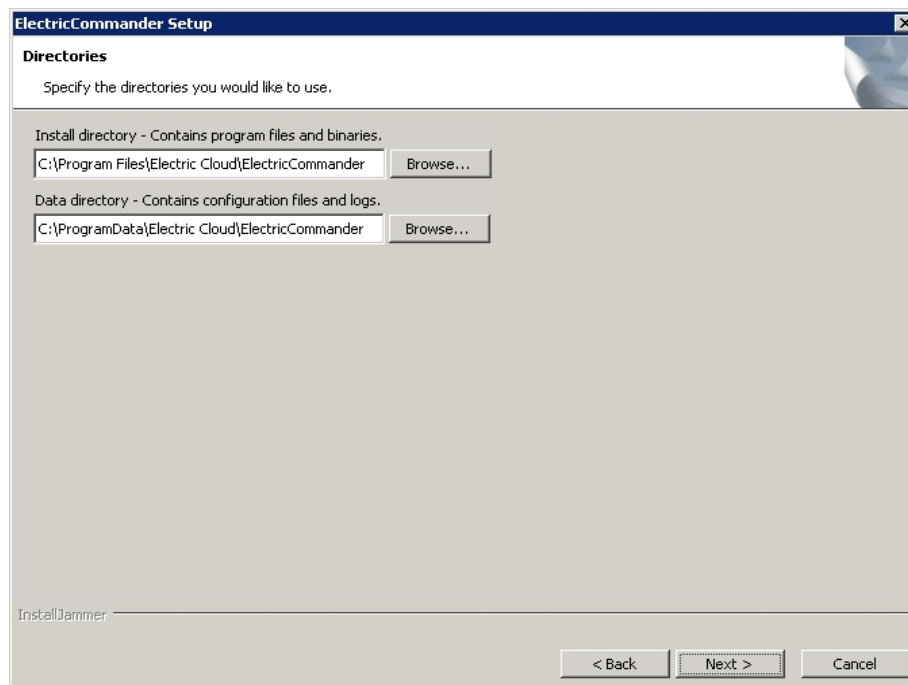
Agent - Select this check box if you want to install Commander agent software.

Tools - Select this check box if you want to install Commander tools. If you only want to install the Commander tools, clear all the check boxes. This option does not automatically install a Commander agent like the other options.

IMPORTANT: The following installation screens will vary depending on which servers you choose to install. **You might see all, some, or none of the screens in the following steps.**

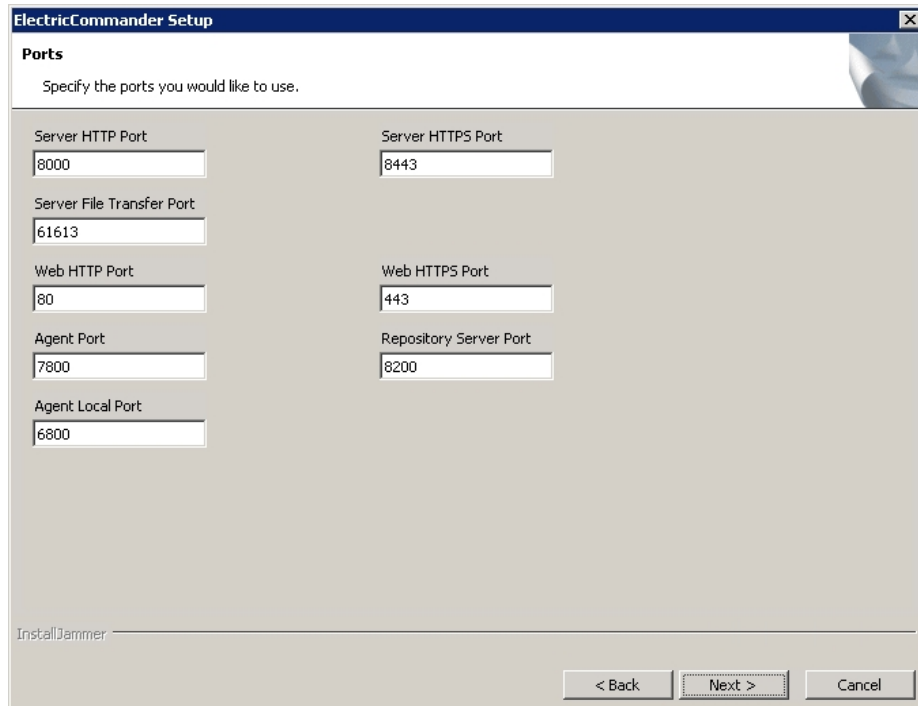
- Click **Next** to accept the component selections.

The Directories screen appears. Commander uses the default directories to install files and components.



- Click **Next** to continue, or click **Browse** to specify different directory locations.

The Ports screen with the default Commander port values appears if you are installing a Commander, web, or repository server.

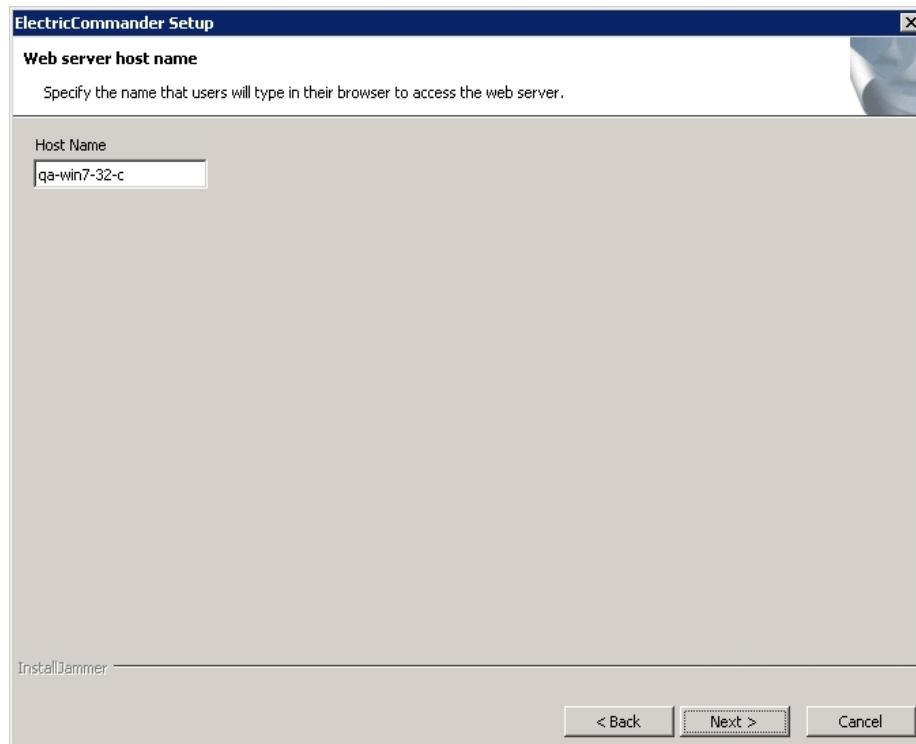


The screenshot shows the 'ElectricCommander Setup' window with the 'Ports' tab selected. The window title bar includes a close button. Below the title bar, the text 'Ports' is followed by the instruction 'Specify the ports you would like to use.' The main area contains ten text input fields arranged in two columns. The left column includes 'Server HTTP Port' (8000), 'Server File Transfer Port' (61613), 'Web HTTP Port' (80), 'Agent Port' (7800), and 'Agent Local Port' (6800). The right column includes 'Server HTTPS Port' (8443), 'Web HTTPS Port' (443), and 'Repository Server Port' (8200). At the bottom left, there is a label 'InstallJammer' followed by a horizontal line. At the bottom right, there are three buttons: '< Back', 'Next >', and 'Cancel'.

Port Type	Default Value
Server HTTP Port	8000
Server HTTPS Port	8443
Server File Transfer Port	61613
Web HTTP Port	80
Web HTTPS Port	443
Agent Port	7800
Repository Server Port	8200
Agent Local Port	6800

6. Complete the information for the Ports screen, and click **Next** to continue. You can enter alternate port numbers if you need to specify different port values.

The Web Server Host Name screen appears if you are installing a web server.



The screenshot shows a window titled "ElectricCommander Setup" with a close button (X) in the top right corner. The window has a white header area with the title "Web server host name" and a sub-instruction: "Specify the name that users will type in their browser to access the web server." Below this is a large beige area. In the top left of this area, the label "Host Name" is followed by a text input field containing the text "qa-win7-32-c". At the bottom left of the beige area, the text "InstallJammer" is visible. At the bottom right, there are three buttons: "< Back", "Next >" (which is highlighted with a dashed border), and "Cancel".

7. Complete the information for the Web Server Host Name screen, and click **Next** to continue. The host name is the name users must enter in their browser to access the Commander web server.

The Server Service Account screen appears if you are installing a Commander, web, or repository server.

8. Complete the information on the Server Service Account screen, and click **Next** to continue.

○ If you have a Windows system:

- **User Name** - Use this field to enter the name of the user who will run the Commander server, web server, and repository server services.
- **Password** - Use this field to enter the password of the user who will run the Commander server, web server, and repository server services.
- **Domain** - Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.
- **Use the local system account** - Select this check box if you want the Commander server, repository server, and web server services to run as the local Windows system account.

Note: The local system account cannot access network resources such as shared filesystems used for plugins or workspaces.

- **Use the same account for the agent service** - Select this check box if you want the agent on the Commander server machine to run as the same account.

For security reasons in production environments, you might want to use a separate account for the agent service because the server account has permission to read the key file (/opt/electriccloud/electriccommander/conf/passkey in Linux or C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey in Windows). The key file is used to decrypt passwords stored in Commander. Using a different account for the agent service ensures that a process running on the agent cannot gain access to the key file.

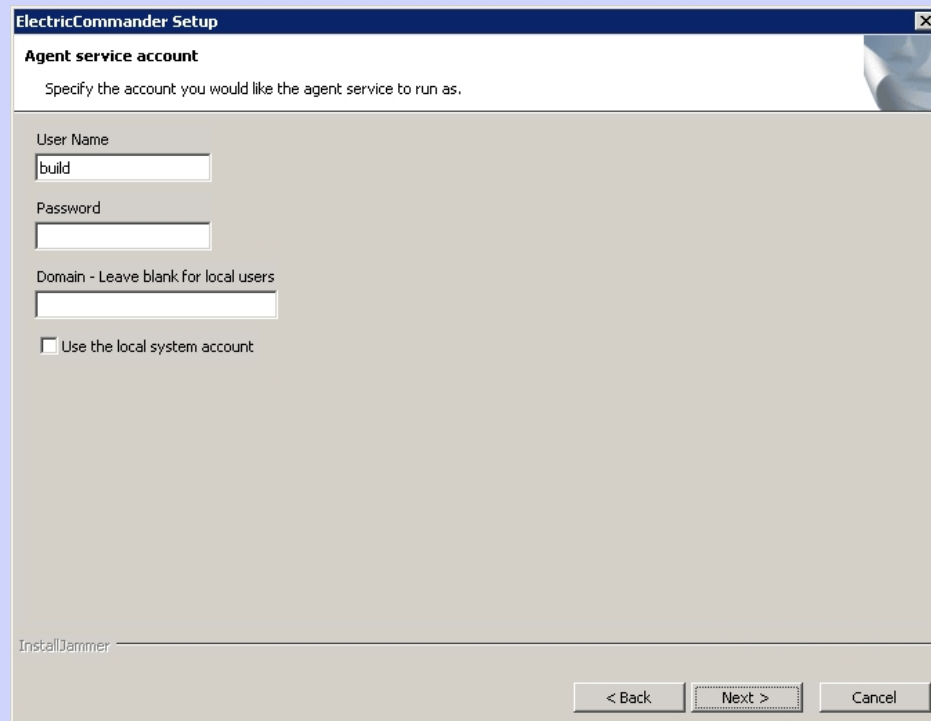
- If you have a Linux system:

- **User Name** - Use this field to enter the name of the user who owns the Commander server, repository server, and web server processes.
- **Group Name** - Use this field to enter the name of the group who owns the Commander server, repository server, and web server processes.
- **Use the same account for the agent service** - Select this check box if you want the same user and group to own the agent process on the Commander server machine.

For security reasons in production environments, you might want to use a separate user and group for the agent service because the server service has permission to read the key file (`/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows). The key file is used to decrypt passwords stored in Commander. Using a different user and group for the agent service ensures that a process running on the agent cannot gain access to the key file.

The Agent Service Account screen appears if you are installing an agent. An agent is automatically installed on the machine to run jobs if you are installing a web or repository server.

IMPORTANT: If you selected the **Use the same account for the agent service** check box on the previous screen, you will not see the fields to supply your agent service account information.



9. Complete the information on the Agent Service Account screen, and click **Next** to continue.

- If you have a Windows system:

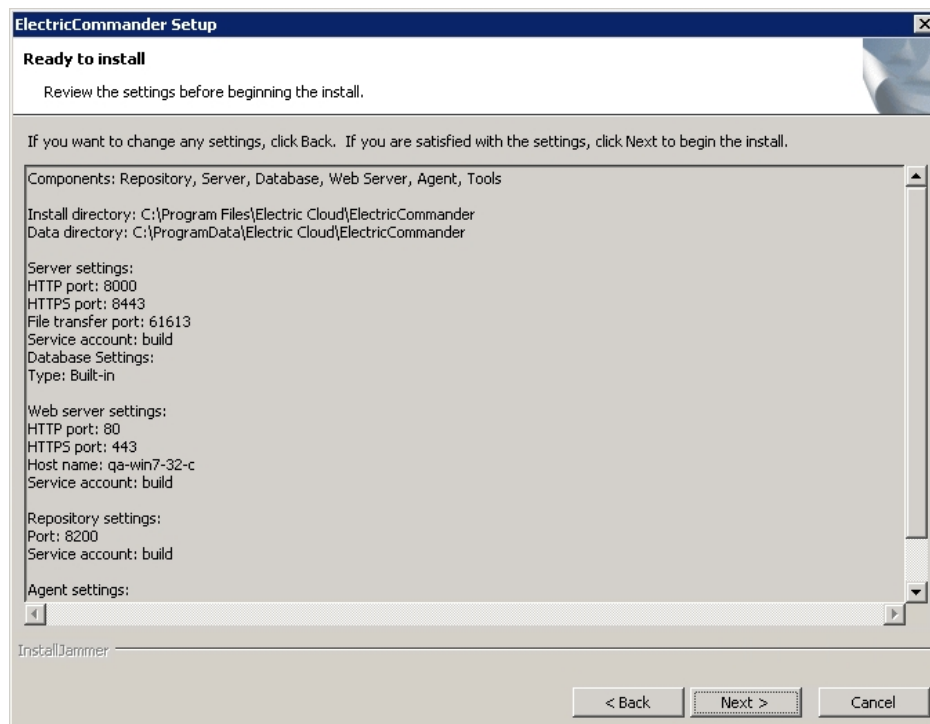
- **User Name** - Use this field to enter the name of the user who will run the Commander agent service.

- **Password** - Use this field to enter the password of the user who will run the Commander agent service.
- **Domain** - Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.
- **Use the local system account** - Select this check box if you want the Commander agent service to run as the local Windows system account.

Note: The local system account does not have access to network shares.

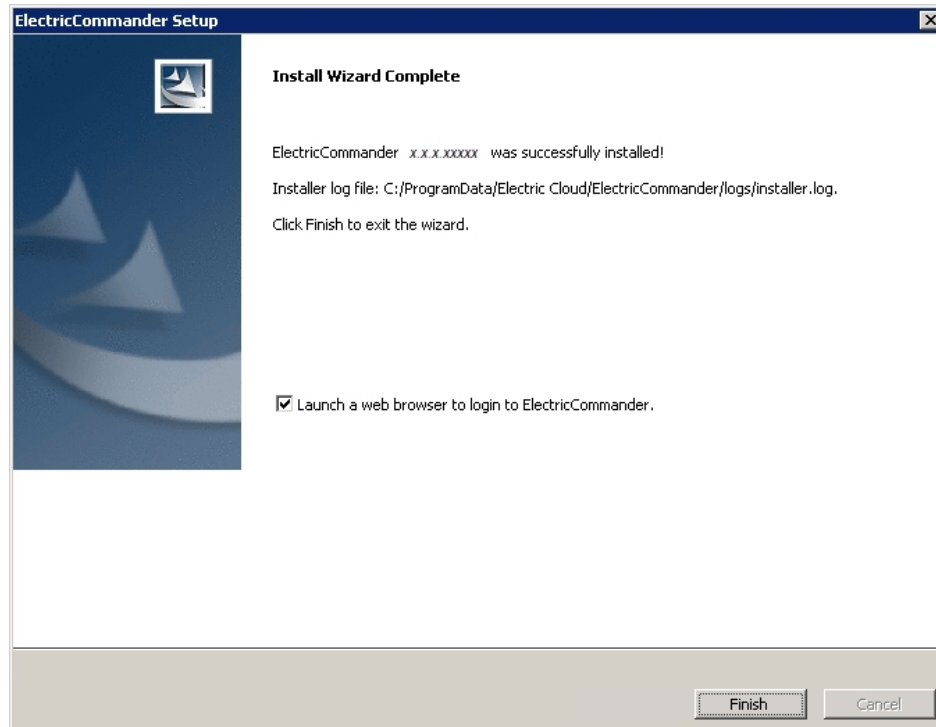
- If you have a Linux system:
 - **User Name** - Use this field to enter the name of the user who owns the Commander agent process.
 - **Group Name** - Use this field to enter the name of the group that owns the Commander agent process.

The Ready to Install screen appears.



10. Review your installation settings. Use the **Back** button to modify any information if necessary.
11. Click **Next** to continue.

The installer displays a status bar to show the progress of the installation process which can take fifteen minutes. When the install process is complete, the Install Wizard Complete screen appears.



Note: The Commander server will automatically start when installation is complete.

11. Select the **Launch a web browser to login to ElectricCommander** check box if you want the Commander login screen to open.
12. Click **Finish** to close the wizard.

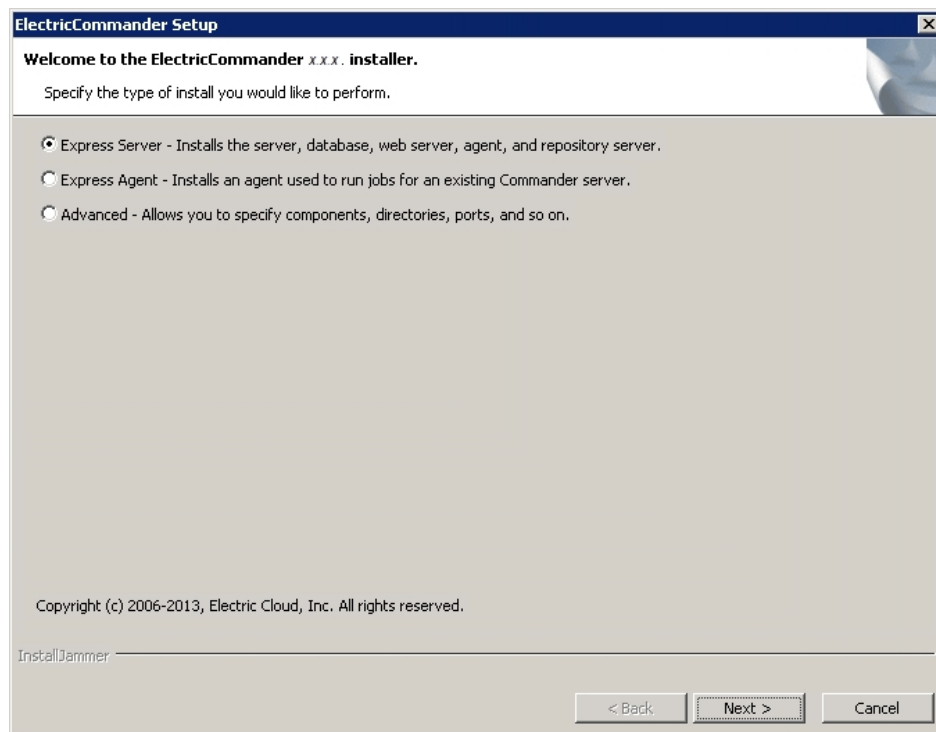
Running an Express Agent Graphical User Interface Installation

The Commander agent software must be installed on each agent machine you intend to use with Commander. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

Note: You might install Commander agent software on Windows or Linux with this installation method. For Solaris, HP-UX, Mac, or other supported UNIX agent machines, see [Non-Server Platform Agent Installation Method](#) on page 3-34.

1. Double-click the `ElectricCommander-<version>` file.

The Welcome to the ElectricCommander Installer screen appears.



Note: The screen examples in this procedure are from a Windows system. Different options might appear on a Linux system.

2. Select the **Express Agent** installation option, and then click **Next** to continue.

The Remote Commander Server screen appears.

3. Complete the following information on the Remote Commander Server screen.

- **Server Host Name** - Use this field to enter the name of the Commander server that will communicate with this agent. If the remote server is using a non-default HTTP port, you must specify the Server Host Name as `host:port`.
- **Commander User Name** - Use this field to enter the name of a Commander user on the Commander server who has sufficient privileges to create a resource. This field defaults to the Commander-supplied `admin` user.
- **Password** - Use this field to enter the password for the Commander user. The default password for the `admin` user is `changeme`.
- **Discover the plugins directory** - Select this check box if you want the agent machine to have access to the plugins directory. We recommend that you allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

Note: The plugins directory on the Commander server must be “shared” before the agent machine can use “discover” to find the directory. For more information, see [Universal Access to the Plugins Directory](#) on page 5-7.

- **Create a resource** - Select this check box if you want to create a resource on the remote Commander server for the agent you are installing.
- **Trusted** - Select this check box to restrict this agent to one Commander server. The agent will not respond to incoming communication from any other Commander server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.

- **Resource Name** - Use this field to enter the name of the resource you would like to use for the agent. This field is available for use when the Create a resource check box is selected.
 - **Create in default zone** - Select this check box if you want to create the agent in the default zone.
 - **Agent Gateway URL** - Use this field to enter the URL of the gateway used to communicate with the Commander server. This field is available for use when the Create in default zone check box is cleared.
 - **Zone Name** - Use this field to enter the name of the zone used during remote agent and/or remote repository creation. This field is available for use when the Create in default zone check box is cleared.
4. Click **Next** to continue.

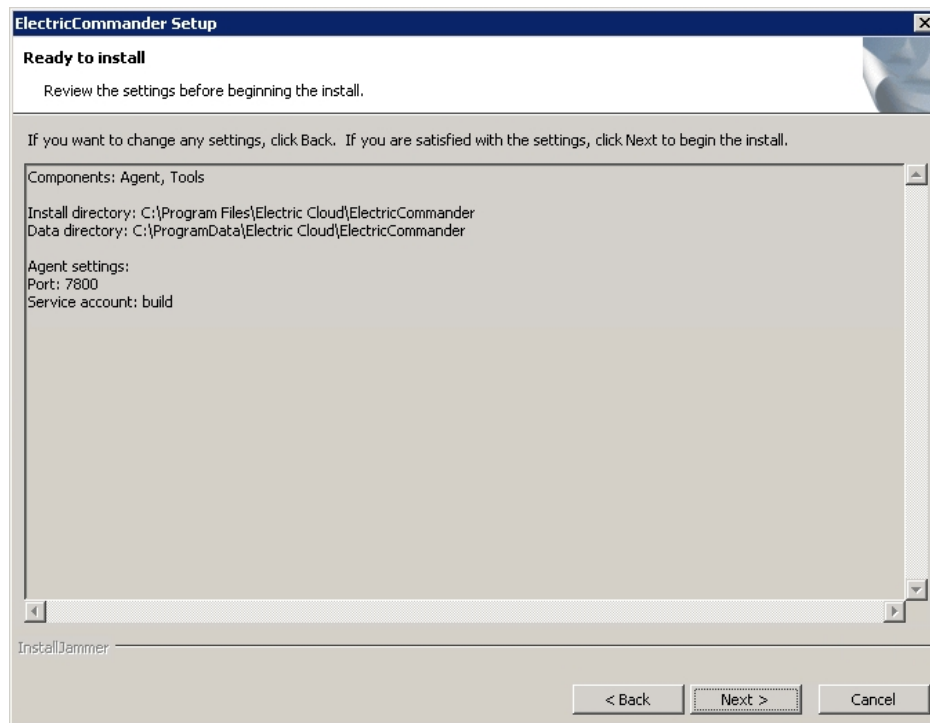
The Agent Service Account screen appears.

5. Select the appropriate steps for your platform and complete the following information on the screen.
- If you have a Windows system:
 - **User Name** - Use this field to enter the name of the user who will run the Commander agent service.
 - **Password** - Use this field to enter the password of the user who will run the Commander agent service.
 - **Domain** - Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.
 - **Use the local system account** - Select this check box if you want the Commander agent service to run as the local Windows system account.

Note: The local system account cannot access network resources such as shared filesystems used for plugins or workspaces.

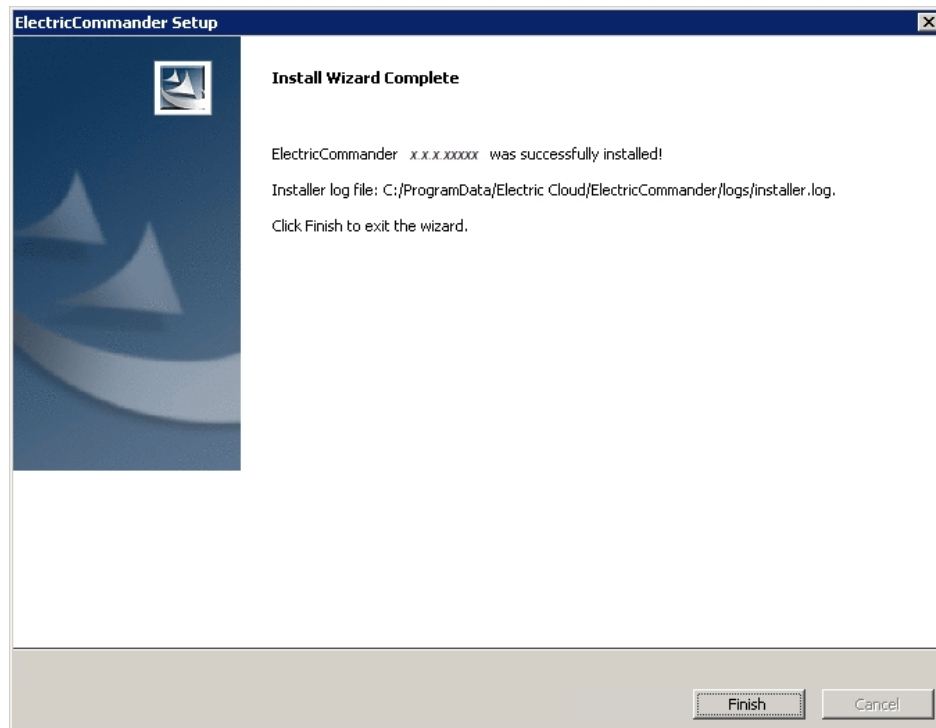
- If you have a Linux system:
 - **User Name** - Use this field to enter the name of the user who owns the Commander agent process.
 - **Group Name** - Use this field to enter the name of the group who owns the Commander agent process.
6. Click **Next** to continue.

The Ready to Install Screen appears.



7. Review this screen to verify your selections. Use the **Back** button to change any of your settings if necessary.
8. Click **Next** to continue.

Commander installs the agent and tools components. This process can take a few minutes.



9. Click **Finish** to complete the installation.

Interactive Command-line Installation Methods

The interactive command-line installation methods are only supported for Linux-only installs on a local Linux volume. Electric Cloud does not support installing the Commander server on a network volume.

Note: You might install Commander agent software on Linux with this installation method. For Solaris, HP-UX, Mac, or other supported UNIX agent machines, see [Non-Server Platform Agent Installation Method](#) on page 3-34.

Running an Express Server Command-Line Installation

The express server command-line installation installs the Commander server, web server, built-in database, agent for running jobs, and Commander tools. This installation uses the default Commander server settings. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure. **Note:** The built-in database is not supported in a clustered Commander configuration.

1. Choose one of the following commands to start the installer:
 - If you have a Linux platform, enter: `./ElectricCommander-<version>`
 - If you have a Linux platform with the X Window System, the installer will automatically bring up the graphical user interface. To override this behavior, enter: `./ElectricCommander-<version> --mode console`

The software displays the following message.

```
Copyright (c) 2006-2012, Electric Cloud, Inc. All rights reserved.
```

```
This will install ElectricCommander on your computer. Continue? [n/Y]
```

2. Continue the installation by entering `y`.

The software displays the following message.

```
Specify the type of setup you would like to perform: expressServer,  
expressAgent, or advanced. [expressServer]
```

3. Enter: `expressServer`

The software displays the following message.

```
Specify the user the server, web, and/or repository will run as. []
```

4. Enter a user name. This is the user who owns the Commander server, repository server, and web server processes. For example, you might enter `build`.

The software displays the following message.

```
Specify the group the server, web, and/or repository will run as. []
```

5. Enter a group name. This is the group who owns the Commander server, repository server, and web server processes. For example, you might enter `build`.

The software displays the following message.

```
Use the same service account for the agent (not recommended for production  
systems)? [y/N]
```

Note: For security reasons in production environments, you might want to use a separate user and group for the agent service because the server service has permission to read the key file (`/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows). The key file is used to decrypt passwords stored in Commander. Using a different user and group for the agent service ensures that a process running on the agent cannot gain access to the key file.

6. Choose one of the following options:

- Enter `y` to use the same user and group for the agent service. This is not recommended for production systems.
- Enter `n` to use a separate user and group for the agent service.

The software displays the following message.

```
Specify the user the agent will run as. []
```

- a. Enter a User Name. This is the user who owns the Commander agent process. For example, you might enter `build`.

The software displays the following message.

```
Specify the group the agent will run as. []
```

- b. Enter a Group Name. This is the group that owns the Commander agent process. For example, you might enter `build`.

The commander software is installed on the machine. When the installation completes successfully, a message that contains the line `"ElectricCommander <version> was successfully installed!"` appears.

Running an Advanced Command-Line Installation

The advanced command-line installation allows you to install individual Commander components such as a Commander server, web server, repository server, or Commander tools on specific machines. You can also

change the default installation settings to accommodate your environment. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

1. Run the following command to start the installer.
 - If you have a Linux platform, enter: `./ElectricCommander-<version>`
 - If you have a Linux platform with the X Window System, the installer will automatically bring up the graphical user interface. To override this behavior, enter: `./ElectricCommander-<version> --mode console`

The software displays the following message.

```
Copyright (c) 2006-2012, Electric Cloud, Inc. All rights reserved.

This will install ElectricCommander on your computer. Continue? [n/Y]
```

2. Continue the installation by entering `y`.

The software displays the following message.

```
Specify the type of setup you would like to perform: expressServer,
expressAgent, or advanced. [expressServer]
```

3. Enter: `advanced`

The software displays the following message.

```
Install a Commander server? [n/Y]
```

4. Select the servers you want to install on the current machine. For more information, see [Architecture](#) on page 1-3.

Note: If you only want to install the Commander tools, enter `n` for all of the server options. The commander tools are automatically installed even if you choose not to install any server software.

- a. Enter `y` to install a Commander server.

The software displays the following message.

```
Install a built-in database? [n/Y]
```

- b. Choose one of the following options:

- Enter `y` to install a built in database. This is not recommended for production systems. Additionally, the built-in database is **not** supported in a clustered Commander configuration.
- Enter `n` if you plan to use an external database. For more information, see [External Database Configuration](#) on page 5-1. If you plan to use MySQL, see [Installing the MySQL JDBC Driver](#) on page 3-37.

The software displays the following message.

```
Install an Apache web server? [n/Y]
```

- c. Enter `y` to install an Apache web server.

The software displays the following message.

```
Install a Commander repository server? [n/Y]
```

- d. Enter `y` to install a Commander repository server.

The software displays the following message.

Specify the install directory (for program files and binaries).
[/opt/electriccloud/electriccommander]

IMPORTANT: The following installation options will vary depending on which servers you choose to install. You might see all, some, or none of the prompts in the following steps.

5. Press enter to accept the default installation directory, or enter a new installation directory path for program files and binaries.

The software displays the following message.

Specify the data directory (for configuration files and logs).
[/opt/electriccloud/electriccommander]

6. Press enter to accept the default installation directory, or enter a new installation directory path for configuration files and logs.

The software displays prompts for server port values. The prompts that appear will vary depending on the sever software you previously selected to install.

7. Press enter to accept the default port values, or enter alternate port numbers if you need to specify a different port value.

The following message only appears if you are installing an Apache web server. If you are not installing a web server, you will see a prompt to enter a user name.

Specify the host name that users will type in their browser to access the web server. [hostName]

8. Enter a web server host name if you are installing an Apache web server. This is the host name users need to type into their browser to access the Commander web server.

The software displays the following message.

Specify the user the server, web, and/or repository will run as. []

9. Enter a user name if you are installing a Commander, web, or repository server. This is the user who owns the Commander server, repository server, and web server processes. For example, you might enter build.

The software displays the following message.

Specify the group the server, web, and/or web repository will run as. []

10. Enter a group name if you are installing a Commander, web, or repository server. This is the group who owns the Commander server, repository server, and web server processes. For example, you might enter build.

The following message only appears if an agent is installed on this machine. An agent is automatically installed with a web or repository server.

Use the same service account for the agent (not recommended for production systems)? [y/N]

Note: For security reasons in production environments, you might want to use a separate user and group for the agent service because the server service has permission to read the key file (/opt/electriccloud/electriccommander/conf/passkey in Linux or C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey in Windows). The key file is used to decrypt passwords stored in Commander. Using a different user and group for the agent service ensures that a process running on the agent cannot gain access to the key file.

11. Choose one of the following options if an agent is automatically installed with the server:
 - Enter `y` to use the same user and group for the agent service. This is not recommended for production systems.
 - Enter `n` to use a separate user and group for the agent service.

The software displays the following message.

```
Specify the user the agent will run as. []
```

- a. Enter a User Name. This is the user who owns the Commander agent process.

The software displays the following message.

```
Specify the group the agent will run as. []
```

- b. Enter a Group Name. This is the group that owns the Commander agent process..

The commander software is installed on the machine. When the installation completes successfully, a message that contains the line "ElectricCommander <version> was successfully installed!" appears.

Running an Express Agent Command-Line Installation

The agent software must be installed on each machine you intend to use with Commander. An agent is a Commander component that runs on a machine resource. The agent executes Commander job steps, monitors step progress, and records job completion information.

Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

1. Choose one of the following commands to start the installer:
 - If you have a Linux platform, enter: `./ElectricCommander-<version>`
 - If you have a Linux platform with the X Window System, the installer will automatically bring up the graphical user interface. To override this behavior, enter: `./ElectricCommander-<version> --mode console`

The software displays the following message.

```
Copyright (c) 2006-2015, Electric Cloud, Inc. All rights reserved.
```

```
This will install ElectricCommander on your computer. Continue? [n/Y]
```

2. Continue the installation by entering `y`.

The software displays the following message.

```
Specify the type of setup you would like to perform: expressServer,
expressAgent, or advanced. [expressServer]
```

3. Enter: `expressAgent`

The software displays the following message.

```
Discover the plugins directory from a remote Commander server? [y/n]
```

4. Enter `y` if you want the agent machine to have access to the plugins directory. We recommend that you allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

IMPORTANT: The plugins directory on the Commander server must be “shared” before the agent machine can use “discover” to find the directory. For more information, see [Universal Access to the Plugins Directory](#) on page 5-7.

The software displays the following message.

```
Create a resource for the installed agent on a remote Commander server? [y/n]
```

5. Enter `y` to automatically create a resource object for the agent on a remote Commander server. This option is recommended to save time configuring new Commander resources for **pre-existing** Commander servers.

The software displays the following message.

```
Register as trusted agent (required for gateway)? [y/n]
```

Note: Making an agent trusted restricts the agent to one Commander server. The agent will not respond to incoming communication from any other Commander server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.

6. Choose one of the following options:
 - If a gateway used to communicate with the Commander server, you must select `y`. This option allows you to create a trusted network connection between the agent and server under the same certificate authority. This will allow the agent and the Commander server to communicate across the network.
 - If there is no gateway between the agent and commander server, enter `n`.

IMPORTANT: If you deviated from the recommended agent options, you will see variations in the installation options that appear on your system.

The software displays the following message.

```
Create repository and/or agent in the default zone? [y/n]
```

7. Enter `y` to create the agent in the default zone.

The software displays the following message.

```
Specify the host:port of a remote Commander server the agent, repository server  
and/or web server being installed can link to. The port is only required if it  
is not the default. [] <hostName:port>
```

8. Enter the Server Host Name of the Commander server that will communicate with this agent. You must specify the Server Host Name as `hostName:port` if the remote server is using a non-default HTTP port.

The software displays the following message.

```
Specify the user name with which to login to <hostName:port>. [admin]
```

9. Enter the user name of a user on the Commander server who has sufficient privileges to create a resource. The default is the Commander-supplied `admin` user.

The software displays the following message.

```
Specify the password for "<commander_user>" on <hostName:port>. []
```

10. Enter the password for the commander user. The default password for the `admin` user is `changeme`.

The software displays the following message.

```
Specify the name of the resource to create on <hostName:port>. [<resource_name>]
```

11. Enter the following information if the agent must be registered as a trusted agent. These options only appear if you entered `y` for `Register as trusted agent (required for gateway)? [y/n]`.

- a. Enter a resource name to use on the Commander server.

The software displays the following message.

Specify the agent gateway URL in the form of 'ipOrHostname:port' []

- b. Enter an agent gateway URL. This is the URL of the gateway used to communicate with the Commander server.

The software displays the following message.

Specify the zone name for the agent and/or repository []

- c. Enter the Zone Name. This is the zone used during remote agent and/or remote repository creation.

The software displays the following message.

Specify the user the agent will run as. []

12. Enter a user name. This is the user who owns the Commander agent process. For example, you might enter `build`.

The software displays the following message.

Specify the group the agent will run as. []

13. Enter a Group Name. This is the group that owns the Commander agent process. For example, you might enter `build`.

The commander software is installed on the machine. When the installation completes successfully, a message that contains the line "ElectricCommander <version> was successfully installed!" appears.

Silent Unattended Installation Method

You can run the Commander installer in unattended (silent) mode with no user interface for either Windows or Linux. Use the arguments in the following list to construct the commands you need for the particular installation you need to perform. For example: server, agent, web server, and so on.

Running a Silent Install

- Choose one of the following commands to begin a silent install:

- If you have a Linux system, go to a command line and enter:

```
ElectricCommander-<version> --mode silent <--arguments>
```

- If you have a Windows system, go to a command line and enter:

```
ElectricCommander-<version>.exe --mode silent <--arguments>
```

Where:

- `<version>` is your Commander version number
- `<arguments>` represents any additional silent install arguments you add to the command. For more information, see [Silent Install Arguments](#) on page 3-28.

Silent Install Arguments

The following argument table is an excerpt from the installer help text. You can view the full installer help by entering the `ElectricCommander-<version> --help` command.

IMPORTANT: Only limited validity checking is performed on these values during an unattended installation, which means typing errors or other mistakes might manifest themselves in strange ways.

<pre>ElectricCommander-<version> --help</pre> <p>Usage: <code>ElectricCommander-<version> [options ...]</code></p> <p>Available Options:</p>	
<code>--agentGatewayURL [ARG]</code>	The URL of the gateway used to communicate with the Commander server.
<code>--agentLocalPort [ARG]</code>	Port used by the Commander agent for HTTP communication on the localhost network interface.
<code>--agentPort [ARG]</code>	Port used by the installed Commander agent for HTTPS communication on any network interface.
<code>--dataDirectory [ARG]</code>	Directory used to store configuration files, logs, and database artifacts.
<code>--force32Bit</code>	Force a 32-bit install, even if the machine is 64-bit.
<code>--haveRunMigrationUtility</code>	<p>When you upgrade from ElectricCommander 4.2.x to ElectricCommander 5.0 and later, the database is modified to support modifications to universally unique identifiers (UUIDs).</p> <p>The upgrade process includes using a migration tool to add support for UUIDs. For more information, go to Properties Changed to UUIDs on page 7-6.</p>
<code>--help</code>	Display this information.
<code>--installAgent</code>	Install the Commander agent.
<code>--installDatabase</code>	Install a local built-in database to use with the main Commander server. This database is not recommended for production systems.
<code>--installDirectory [ARG]</code>	Directory used to store program files and binaries.
<code>--installRepository</code>	Install a Commander artifact repository server.
<code>--installServer</code>	Install the main Commander server.
<code>--installWeb</code>	Install the local web server and Commander web interface.

<p>ElectricCommander-<version> --help</p> <p>Usage: ElectricCommander-<version> [options ...]</p> <p>Available Options:</p>	
--mode [ARG]	The mode in which the installer will run. Available values: console, silent, or standard
--remoteServer [ARG]	The <code>host:port</code> for the remote Commander server. The port is optional and can be omitted if the server is using the default HTTP port.
--remoteServerCreateRepository	Create a repository object on the remote Commander server.
--remoteServerCreateResource	Create a resource for the installed agent on the remote Commander server.
--remoteServerDiscoverPlugins	Set the plugins directory for the installed agent and/or web server to the shared plugins directory defined on the remote Commander server.
--remoteServerPassword [ARG]	The password to use when logging in to the remote Commander server.
--remoteServerRepository [ARG]	The name of the repository to create on the remote Commander server.
--remoteServerResource [ARG]	The name of the resource to create on the remote Commander server.
--remoteServerUser [ARG]	The username to use when logging in to the remote Commander server.
--repositoryPort [ARG]	Port used by the Commander artifact repository server (the default is 8200).
--response-file [ARG]	A file containing installer responses. ARG is a file with or without a path. The file contains name/value pairs separated by "=" (an equal sign) representing command line argument values. Any value on the command line supersedes the value in the response-file. The response-file can be automatically created during the install using the <code>save-response-file</code> command-line argument.
--save-response-file [ARG]	A file for writing installer responses to when the installer exists.
--serverFileTransferPort [ARG]	File transfer port used by the installed Commander server.
--serverHttpPort [ARG]	HTTP port used by the installed Commander server.
--serverHttpsPort [ARG]	HTTPS port used by the installed Commander server.

```
ElectricCommander-<version> --help
```

```
Usage: ElectricCommander-<version> [options ...]
```

Available Options:

<code>--temp [ARG]</code>	Directory used to store temporary files used by the installer.
<code>--trustedAgent</code>	Restricts the agent to one Commander server. The agent will not respond to incoming communication from any other Commander server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.
<code>--unixAgentGroup [ARG]</code>	The group the installed Commander agent runs as.
<code>--unixAgentUser [ARG]</code>	The user the installed Commander agent runs as.
<code>--unixServerGroup [ARG]</code>	The group the installed Commander/web/repository server runs as.
<code>--unixServerUser [ARG]</code>	The user the installed Commander/web/repository server runs as.
<code>--useSameServiceAccount</code>	Use the same account for server and agent services. Not recommended for production systems.
<code>--version</code>	Display installer version information.
<code>--webHostName [ARG]</code>	The name users need to type in their browser to access the web server.
<code>--webHttpPort [ARG]</code>	HTTP port used by the installed web server.
<code>--webHttpsPort [ARG]</code>	HTTPS port used by the installed web server.
<code>--windowsAgentDomain [ARG]</code>	The domain of the account the installed Commander agent runs as.
<code>--windowsAgentLocalSystem</code>	Run the Commander agent as the local system account.
<code>--windowsAgentPassword [ARG]</code>	The password of the account the installed Commander agent runs as.
<code>--windowsAgentUser [ARG]</code>	The user name of the account the installed Commander agent runs as.
<code>--windowsServerDomain [ARG]</code>	The domain of the account the installed Commander/web/repository server runs as.

ElectricCommander-<version> --help Usage: ElectricCommander-<version> [options ...] Available Options:	
--windowsServerLocalSystem	Run the Commander/web/repository server as the local system account. Note: The local system account cannot access network resources such as shared filesystems used for plugins or workspaces.
--windowsServerPassword [ARG]	The password of the account the installed Commander/web/repository server runs as.
--windowsServerUser [ARG]	The user name of the account the installed Commander/web/repository server runs as.
--windowsSkipAdminCheck	Do not check that the user running the installer is a direct member of group Administrators.
--zoneName [ARG]	The zone name used during remote agent and/or remote repository creation.

Linux Silent Installation Examples

The following examples are command strings to use for unattended (silent) installations. In many instances, the command text with the associated options wraps to the following lines.

IMPORTANT: You must enter the installation command and all options on a single line.

Complete Commander Installation

This installation installs the Commander server, including the web, repository, and database servers, one agent, and Commander tools. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

- Enter:

```
./ElectricCommander-<version> --mode silent --installServer --installAgent --installDatabase --installWeb --installRepository --unixServerUser <server user> --unixServerGroup <server group> --unixAgentUser <agent user> --unixAgentGroup <agent group>
```

Where:

- <server user> is the user who owns the Commander server, repository server, and web server processes.
- <server group> is the group who owns the Commander server, repository server, and web server processes.
- <agent user> is the user who owns the Commander agent process.
- <agent group> is the group that owns the Commander agent process.

Repository Server and an Agent Installation

This installation example installs an ElectricCommander repository server, an agent, and Commander tools. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

- Enter:

```
./ElectricCommander-<version> --mode silent --installRepository --installAgent -  
--unixAgentUser <agent user> --unixAgentGroup <agent group> --unixServerUser <ser  
ver user> --unixServerGroup <server group> --remoteServer <your existing Command  
er server>
```

Where:

- *<server user>* is the user who owns the Commander server, repository server, and web server processes.
- *<server group>* is the group who owns the Commander server, repository server, and web server processes.
- *<agent user>* is the user who owns the Commander agent process.
- *<agent group>* is the group that owns the Commander agent process.

Commander Agent Installation

The Commander agent software must be installed on each agent machine you intend to use with Commander. This installation also installs Tools. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

- Enter:

```
./ElectricCommander-<version> --mode silent --installAgent --unixAgentUser <agen  
t user> --unixAgentGroup <agent group>
```

Where:

- *<agent user>* is the user who owns the Commander agent process.
- *<agent group>* is the group that owns the Commander agent process.

Web Server and an Agent

This installation example installs a Commander web server, an agent, and Commander tools. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

- Enter:

```
./ElectricCommander-<version> --mode silent --installWeb --unixAgentUser <agent  
user> --unixAgentGroup <agent group> --remoteServer <your existing Commander ser  
ver> --unixServerUser <server user> --unixServerGroup <server group>
```

Where:

- *<server user>* is the user who owns the Commander server, repository server, and web server processes.
- *<server group>* is the group who owns the Commander server, repository server, and web server processes.

- *<agent user>* is the user who owns the Commander agent process.
- *<agent group>* is the group that owns the Commander agent process.

Tools Only Installation

This installation example installs only the Commander tools. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

- Enter:

```
./ElectricCommander-<version> --mode silent
```

Windows Silent Installation Examples

The following examples are command strings to use for unattended (silent) installations. In many instances, the command text with the associated options wraps to the following lines.

IMPORTANT: You must enter the installation command and all options on a single line.

Complete Commander Installation

This installation installs the Commander server, including the web, repository, and database servers, one agent, and Commander tools. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

- Enter:

```
ElectricCommander-<version>.exe --mode silent --installServer --installAgent --installDatabase --installWeb --installRepository --windowsServerUser <server user> --windowsServerDomain <domain> --windowsServerPassword <password> --windowsAgentUser <agent user> --windowsAgentDomain <domain> --windowsAgentPassword <password>
```

Where:

- *<server user>* is the user who owns the Commander server, repository server, and web server processes.
- *<agent user>* is the user who owns the Commander agent process.

Repository Server and an Agent Installation

This installation example installs an ElectricCommander repository server, an agent, and Commander tools. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

- Enter:

```
ElectricCommander-<version>.exe --mode silent --installRepository --installAgent --windowsAgentUser <agent user> --windowsAgentDomain <domain> --windowsAgentPassword <password> --windowsServerUser <server user> --windowsServerDomain <domain> --windowsServerPassword <password> --remoteServer <your existing Commander server>
```

Where:

- *<server user>* is the user who owns the Commander server, repository server, and web server processes.
- *<agent user>* is the user who owns the Commander agent process.

Commander Agent Installation

The Commander agent software must be installed on each agent machine you intend to use with Commander. This installation also installs Tools. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

- Enter:

```
ElectricCommander-<version>.exe --mode silent --installAgent --windowsAgentUser  
<agent user> --windowsAgentDomain <domain> --windowsAgentPassword <password>
```

Where:

- *<agent user>* is the user who owns the Commander agent process.

Web Server and an Agent

This installation example installs a Commander web server, an agent, and Commander tools. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

- Enter:

```
ElectricCommander-<version>.exe --mode silent --installWeb --windowsAgentUser <a  
gent user> --windowsAgentDomain <domain> --windowsAgentPassword <password> --rem  
oteServer <your existing Commander server> --windowsServerUser <server user> --w  
indowsServerDomain <domain> --windowsServerPassword <password>
```

Where:

- *<server user>* is the user who owns the Commander server, repository server, and web server processes.
- *<agent user>* is the user who owns the Commander agent process.

Tools Only Installation

This installation example installs only the Commander tools. Review [Before You Install ElectricCommander](#) on page 3-3 before performing this procedure.

- Enter:

```
ElectricCommander-<version>.exe --mode silent
```

Non-Server Platform Agent Installation Method

To install agents and tools on machines that are not supported Commander server platforms, you must use the appropriate UNIX installer instead of the `./ElectricCommander-<version>` installer file. For more information about supported agent platforms, see [Agent Platforms](#) on page 2-4.

You must use the `commander_<OSType>.bin` file to install the Commander agent software on a Windows 2000, Solaris, HP-UX, or Mac OS agent machine. You can find copies of the various agent UNIX installers on the Electric Cloud FTP site.

Installing Agents with the UNIX Installer

Use the `commander_<OSType>.bin` file to install the Commander agent software on a Windows 2000, Solaris, HP-UX, or Mac OS agent machine.

1. Obtain a copy of `commander_<OSType>.bin`
2. Log in as **root**.

3. Run `chmod +x ./commander_<OSType>.bin` to ensure the installer is executable.
4. Run `./commander_<OSType>.bin` to begin the Commander agent installation on a UNIX machine.

Where `<OSType>` is the agent platform.

The following information is displayed.

```
Checking install integrity, please wait...
ElectricCommander 4.2.0 for Linux Installer
Copyright 2006-2012 ElectricCloud, Inc. All rights reserved.
Press CTRL-C to exit the installation at any time.
Press Enter to accept default settings.
log file: /tmp/commander_install_20121203_120130.log
This suite installer can install several different product options.
NOTE: The default is to install everything.
Which products would you like to install (server, web server, agent, tools):
```

5. Enter: agent

Note: When you enter the “agent” option, an agent and tools will be installed.

6. Press **Enter** to begin the Commander UNIX agent installation.

The following information is displayed.

```
OK, installing agent.
Where would you like the software to be installed?
Note: The destination should not be an NFS filesystem.
If no directory is specified, by default an “electricCloud” directory is created
in /opt.
Enter destination directory (default is /opt): <directory path>
```

7. Enter the destination directory path.

The following information is displayed.

```
Enter a pre-existing user to own installed agent files and run agent processes:
```

8. Enter the name of the user who owns the Commander agent process.

The following information is displayed.

```
Enter a pre-existing group to own installed agent files and run agent processes:
```

9. Enter the name of the group who owns the Commander agent process.

The following information is displayed.

```
Enter the agent port (default is 7800):
```

10. Accept the default port or specify a different port to eliminate any conflicts with your existing system configuration.
11. Press **Enter** to start the agent software installation.

When the installation is complete, the following information is displayed.

OK: Installation successful!

UNIX Agent Silent Installation

The UNIX installers also have a command-line interface for unattended (silent) installations. You can specify configuration options on the command line or in a configuration file.

UNIX Silent Installation Command Arguments

The following table lists the arguments for running a silent install with the UNIX installer.

-q	This option causes the installer to operate in silent mode. The default installation options will be used unless overridden in a configuration file or on the command line.
-f	This option forces the installer to remove and replace any existing files in the destination directory. This option does not uninstall the previous version, it completely removes the directory and writes a new one. For upgrades, see Roadmap to Upgrade ElectricCommander on page 6-1.
--config	This option can be used to specify a file containing configuration variables and values.

Running a Silent UNIX Installation

Use the following procedure to run a silent UNIX installation.

- Enter the following command to begin a silent install:

```
commander_<OSType>.bin -q <arguments>
```

Where:

- <OSType> is the agent platform.
- <arguments> represents any additional silent install arguments you add to the command.

The following example command-line string uses a configuration file to override default values on a silent install:

```
commander_sun4u_SunOS.bin -q -f --config myconfig
```

Example Configuration File Parameters

The following examples are parameter values to use for an unattended (silent) installation configuration file.

Agent Install Parameters

```
EC_INSTALL_TYPE=agent
DESTINATION_DIR="/opt"
AGENT_USER_TO_RUN_AS=(user for agent files and processes)
AGENT_GROUP_TO_RUN_AS=(group for agent)
EC_AGENT_PORT=(agent port number)
```

Tools Install Parameters

```
EC_INSTALL_TYPE=tools
DESTINATION_DIR="/opt"
USER_TO_RUN_AS="user"
GROUP_TO_RUN_AS="group"
```

Installing the MySQL JDBC Driver

Starting with Commander 4.2.3, Commander no longer includes the MySQL JDBC driver. If you are upgrading a Commander server, this change has no effect (the existing driver is saved).

If you plan to perform a clean Commander server installation that will connect to a MySQL database, you must obtain and install the MySQL JDBC driver.

Follow these steps:

1. Run the installer. Make sure that you do not install the built-in database.
2. After the installation completes, download the MySQL JDBC driver from <http://dev.mysql.com/downloads/connector/j/>.
3. Install the driver in the <install-dir>/server/lib directory.
4. Restart the server.
5. Go to the Database Configuration page and configure the server to use a MySQL database.

Logging Into the ElectricCommander Web Interface

This task describes how to log into the ElectricCommander web interface. If you chose during installation to configure an external database, you will NOT be able to log into ElectricCommander until you set up a database configuration.

1. Enter the URL of the ElectricCommander server in a browser window. For example, <https://123.123.1.222>

The login screen appears.



IMPORTANT: For a new installation, the default admin account user name is *admin* and the password is *changeme*. You should change the default admin account as soon as possible.

2. Enter a user name and password, and click **Login**.

Chapter 4: Clustering

This section provides you with guidelines and procedures for adding horizontal scalability and high availability to your ElectricCommander environment. Horizontal scalability and high availability is achieved by adding additional machines to a Commander configuration to create a server cluster. A clustered configuration of Commander servers also requires these software components:

- A centralized service for maintaining and synchronizing group services in a cluster called Apache ZooKeeper.
- A software or hardware load balancer for routing work to machines in the cluster.

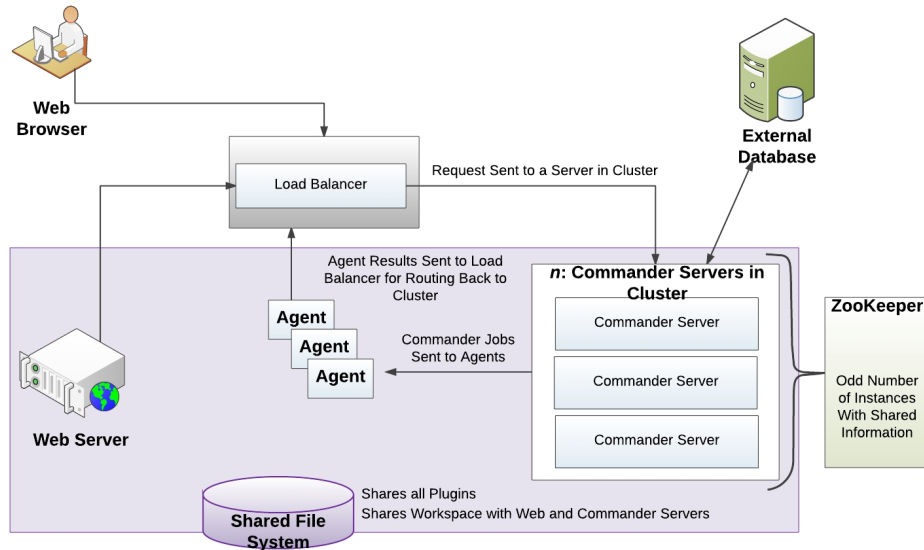
Benefits from Clustering

A clustered Commander configuration has the following benefits:

- Increase the supported number of simultaneous jobs and API requests
- Distribute API requests across multiple Commander servers
- Expand capacity over time by adding additional Commander servers
- Add fault tolerance by rerouting jobs and API requests to running Commander servers
- Distribute Commander requests across multiple web servers

Architecture of a Commander Cluster

The following figure shows an example of the Commander architecture in a clustered configuration.



In the clustered configuration, a node refers to the machine on which an agent is installed, and a server refers to the Commander server cluster. In the *server status*, the load balancer reports only status and error messages about a specific server.

As shown in the previous diagram, we strongly recommend that you point the web browser to the load balancer, not to the web server.

Resource, Agent, and Procedure Configuration Considerations

This section describes various Commander behaviors and possible modifications that you may want to make in your Commander configuration.

Default Local Resource Use

In a default installation, an agent is installed alongside the Commander server, and a resource named *local* is automatically created for the agent. In a cluster, the local resource points to only node. By default, some of the plugins and sample code installed with Commander automatically use the local resource because it is usually present.

Many Commander users also write procedures that use the local resource for these reasons:

- A user knows the local resource exists.
- The user needs local access to the file system, such as access to the log files or configuration files.

- The user needs local access to resources on the Commander server to perform tasks such as checking the CPU, memory, or disk space usage.

IMPORTANT: Any existing procedures for local access to Commander server file systems or resources need to be modified when you change your installation to run in a server cluster. There are now multiple Commander server nodes, each with their own log files, configuration files, and local resources.

The name of the Commander log file on a Commander server node in a cluster has also been changed to `commander-<hostname>.log` to facilitate collecting the multiple logs together without them overwriting each other.

Separate Local Agents For Improved Performance

If you are using clustering for performance reasons, you must manage your machine resources efficiently. You use additional machine resources if you run a Commander agent and Commander server on the same machine. If you are concerned about performance, remove any agents from your Commander server node machines. You should also verify that any agents that may have been installed as part of an earlier configuration are also removed. *An agent is installed automatically with any Commander service by default.*

Pool Local Agents For Improved Reliability

If you are using clustering only for reliability reasons, you must reduce or eliminate single points of failure. In this configuration, having a single server node that runs the local agent is counterproductive. If you need a more reliable configuration, you can install agents on all of your Commander server node machines and put them in a resource pool named *local*.

Procedure Strategies

If you have a local agent on each Commander server, it may be appropriate for some procedures to have one or more procedure steps that are broadcasted across all the resources in the local resource pool. These procedure steps are probably followed by a step that aggregates the resulting data in an appropriate way. For other procedures, it may be more appropriate to use the shared file system to which all the Commander servers have access, and still have the step run on a single agent.

Agent Resource Strategies

If you do not have any agents local to any of your server machines, a local resource is not automatically created. There are two possible strategies to handle this:

- Do not have a local resource and remove any local resource on your system. For everything that is currently configured to use the local resource, reconfigure them to use other resources.
- Create a resource pool or resource called *local* for agents on non-Commander-server machines. Everything that is configured to use the local resource now runs on an agent of the local pool. However, this agent is no longer local to any of your Commander server nodes.

If anything relies on the local resource being local to the server, it must be modified to work across multiple server nodes, to function remotely from an agent running on another machine, or to be both. In some configurations, you may need to use the shared file system between the Commander servers and the remote agent.

For example, if you want to write log-parsing procedures using nonlocal agents, you can configure all of your Commander servers to write their logs to different locations in this shared file system. A procedure step running on a remote agent machine with access to this shared file system can then parse all the server logs.

Database Restriction

The built-in database is not supported in a clustered Commander configuration. You must use an alternate database listed in [Supported Alternate Databases](#) on page 2-8: Oracle, MS SQL Server, or MySQL.

Unsupported Host

ElectricCommander does not support the local host IP address (127.0.0.1) for any ElectricCommander configuration, because it is ambiguous when multiple servers are used.

Software for Clustering

Apache Zookeeper

Apache ZooKeeper is a centralized service required for clustering.

- [Apache ZooKeeper](#) is a critical part of the clustering architecture. You must use ZooKeeper Version 3.4.5 or later to maintain and synchronize group services in the Commander cluster. Electric Cloud has also provided a tool called ZKConfigTool, which you can use to quickly populate ZooKeeper with Commander configuration information.
- [Exhibitor](#) can be used in conjunction with ZooKeeper to monitor the synchronization between the ZooKeeper nodes. This software is not required to implement a Commander cluster configuration, but can provide instance monitoring, backup, recovery, cleanup and visualization services. For more information, see the [Exhibitor documentation](#).

Load Balancer

You must use a load balancer in a Commander cluster. You can choose to use any hardware load balancer or load-balancing software for a cluster configuration as long as the load balancer can act as an SSL endpoint and support returning HTTP redirections.

Dependencies for Clustering

A clustered configuration has the following minimum requirements:

- Two or more copies of the Commander server. Clustering is supported starting with Commander 5.0.
- Two or more Commander agents.
- At least one Commander web server on its own machine, or two or more servers if you are using clustering for reliability improvements.
- An external Commander database for each cluster of Commander servers.

Note: Multiple Commander clusters can use the same database server but not the same database schema instance.

- An enterprise license. The license is required by Commander to connect to an external database.
- Apache ZooKeeper as the centralized service for maintaining configuration information.
 - ZooKeeper should be installed on a machine without a Commander server or load balancer. This separation of services is advised to optimize the performance and reliability of your configuration.
 - ZooKeeper must be installed on an odd number of machines. For example, you may need 1, 3, 5, or more instances of the software depending on your environment.
- The Commander servers must be configured to appear as a single instance in ZooKeeper.

- A hardware load balancer or load-balancing software installed on one machine.
- The web servers, agents, and Commander servers should share a common file system for plugin information.

Running a Cluster in Single-Server Mode

A couple of rarely-used Commander operations are not supported in clustered mode. If you need to perform any of the following operations, you must restart the server in single-server mode:

- Changing the database configuration
This operation can be done in the web UI using Administration > Database Configuration or with ectool. After performing this operation, you must rerun ZKConfigTool to upload the changed configuration from the local database.properties file to ZooKeeper before switching back to clustered mode.
- Doing a full import using ectool
- Doing a full export using the safe mode with either of these options:
 - Restart
 - Shutdown

To run a Commander machine in single-server mode:

1. Identify the machine you need to work on and shut down the other machines in the Commander cluster.
2. Verify that the database.properties, keystore, passkey, and commander.properties configuration files on the machine you are working on are up to date and match those that were uploaded to ZooKeeper.
3. Switch this remaining Commander machine to single-server mode by running the following command:

`ecconfigure --serverEnableClusteredMode=0`
4. Restart that Commander machine.
5. Complete your work on the Commander machine.
6. Switch this remaining Commander machine back to clustered mode by running the following command:

`ecconfigure --serverEnableClusteredMode=1`
7. If you changed the database configuration, use ZKConfigTool to upload the updated database.properties file to ZooKeeper. See [Uploading Configuration Files to ZooKeeper](#) on page 4-17 for more information.
8. Restart all machines in the Commander cluster.

Configuring Clustering

There are two different approaches you can take when you configure your Commander software for horizontal scalability. The approach you choose depends on the needs for your particular Commander environment for reliability versus performance.

IMPORTANT: Whichever of the following approaches you choose, we recommend that you install multiple Commander services (agent, web server, Commander server, and repository) on more than one physical machine (for example, not just virtual machines) to eliminate single points of failure.

- **Reliability** - Choose this type of configuration if your only concern is redundancy for the Commander application. This approach requires only the addition of multiple Commander machines to the server cluster. Multiple Commander services can reside on a machine, but multiple instances of the service software should exist. For example, a Commander server and agent can reside on the same machine as long as other instances of the components exist on different physical machines.
- **Performance** - Choose this type of configuration if your Commander cluster will be in a high load environment. This approach requires the installation of the Commander software services on a sufficient number of dedicated machines. You should install the Commander server, repository, web server, and agent services on separate machines from the server nodes to avoid competition for system resources.

Note: You can change a reliability configuration to a performance configuration at a later time, but additional configuration of your Commander software will be required. For more information, see [Separating Agents from Commander Servers](#) on page 4-21.

Separating Agents from Commander Servers

Use this procedure if you need to separate Commander services and agents. By default, a Commander agent is installed with the Commander server, web server, and repository. For more information, see [Resource, Agent, and Procedure Configuration Considerations](#) on page 4-2 and [Verifying Commander Services](#) on page 4-22.

1. Verify that no Commander agents are installed on any of the Commander server nodes. If necessary, remove the agent software from the Commander server nodes.
2. Verify that none of the Commander utilities use a local resource. If you are not sure if a local resource is in use, create an agent resource called */ocal/* and monitor the system.
3. Remove the local resource.
4. Create a new agent resource with a new name for each agent on each Commander server node machine.
5. Create a resource pool named */ocal/* containing all these resources.

Preparing Your Cluster Resources

Before you install any Commander software, you must complete the following tasks:

1. Identify all machines to be used in the horizontally scalable configuration. It is helpful to have all the network information and machine descriptions available before you begin any work so IP addresses can be used consistently throughout a cluster.
 - a. Identify which systems will have a new installation of the Commander software and which pre-existing systems will be converted to operate in a cluster. Because traffic between the load balancer and the Commander server nodes is not encrypted, for security reasons all the Commander server nodes should be located on the same private network as the load balancer, preferably in the same data center.
 - b. Record the IP addresses of:
 - The load balancer machine
 - The machines that will run ZooKeeper
 - The web server machines
 - The Commander server you will use to import configuration information into ZooKeeper
 - The remaining Commander server machines that will make up the cluster

- c. Record the fully qualified domain name of the load balancer machine. This name will be used in several stages of the configuration process, and should be used consistently throughout the process.
2. Install the load balancer on a machine. For more information, see the instructions from the manufacturer.
3. Install ZooKeeper on an odd number of machines. To eliminate a single point of failure, three or five instances of the software are recommended. For more information, go to [Installing ZooKeeper](#) on page 4-7.

Installing and Configuring a Load Balancer

You must use a load balancer in a Commander cluster. You can choose to use any hardware load balancer or load-balancing software for a cluster configuration as long as the load balancer can act as an SSL endpoint and support returning HTTP redirections.

When configuring your load balancer, follow these general guidelines.

- You must configure a load-balancer IP address for each node in your cluster.
- Load-balance traffic on port 8000 across the Commander servers on port 8000
- Act as an SSL endpoint for port 8443 and load-balance the traffic on that port across the Commander servers on port 8000.
- Act as an SSL endpoint for port 61613 and load-balance the traffic on that port across the Commander servers on port 61613.
- The load balancer should be able to perform frequent health check HTTP GET requests for a specific URL and take servers temporarily out of rotation if they receive an HTTP status 503 response.
- There are no requirements for the state associated with a user session to be replicated across the cluster.

For an example of how to configure a widely used load balancer such as HAProxy, go to the knowledge-based article called "Configuring Load Balancers in ElectricCommander Clusters" on the Electric Cloud Support Web Site:

1. Go to <https://electriccloud.zendesk.com/hc/en-us/categories/200176553-ElectricCommander>.
2. In the Commander KB section, enter **KBEC-00281** in the Search field.
3. Click **Search**.
4. Click the link to the knowledge-based article called "KBEC-00281 - Configuring Load Balancers in ElectricCommander Clusters" to open it.

You can use this example as a model for the load balancer configuration in your system and modify it to meet the system requirements of your particular model of load balancer and system configuration.

Electric Cloud does not support any specific load balancer. For information using about HAProxy, go to the [HAProxy](#) website. For more information about using another load balancer, go to the website for that load balancer.

Installing ZooKeeper

Use Apache Zookeeper Version 3.4.5 or later to maintain and synchronize group services in a clustered Commander configuration. For more information, go to the [Apache ZooKeeper website](#).

For your convenience, ZooKeeper 3.4.5 is bundled in your Commander installation here: `<install dir>/utils`.

To install ZooKeeper:

1. Use either the bundled ZooKeeper or download ZooKeeper from the [ZooKeeper website](#).
2. Extract and install the files into an appropriate location. For example, `/opt/zookeeper-<release_version>`

IMPORTANT: You must install ZooKeeper on an odd number of machines. The number of machines will determine if you install ZooKeeper in standalone mode (for one machine) or in replicated mode (for three or five machines).

3. Create a `zoo.cfg` configuration file for each machine with an instance of ZooKeeper. For example, `zookeeper-<release_version>/conf/zoo.cfg`.

- For standalone mode on a single ZooKeeper machine, the file has these values:

```
tickTime=2000
dataDir=/var/lib/zookeeper
clientPort=2181
```

- For replicated mode across multiple ZooKeeper machines, the `zoo.cfg` file on each server should have these values:

```
tickTime=2000
dataDir=/var/lib/zookeeper
clientPort=2181
initLimit=5
syncLimit=2
server.1=<ZooKeeper_hostname_1>:2888:3888
server.2=<ZooKeeper_hostname_2>:2888:3888
server.3=<ZooKeeper_hostname_3>:2888:3888
server.4=<ZooKeeper_hostname_4>:2888:3888
server.5=<ZooKeeper_hostname_5>:2888:3888
```

Where `<ZooKeeper_hostname_1>` through `<ZooKeeper_hostname_5>` are the hostnames of the servers for the ZooKeeper service.

Note: The file has only three server value lines for a three-ZooKeeper configuration.

Create `/var/lib/zookeeper/myid` files on each ZooKeeper server, with each containing a single ASCII digit: 1 for the first server, 2 for the second server, and so on, corresponding to their `server.<digit>` values in the `zoo.cfg` files

4. Create `/var/lib/zookeeper/myid` files on each ZooKeeper server, with each containing a single ASCII digit as follows:
 - 1 for the first server,
 - 2 for the second server
 - Up to 5, corresponding to the appropriate `server.<digit>` value in the `zoo.cfg` file.

Running ZooKeeper as a Service on Linux

Install ZooKeeper on machines other than those running ElectricCommander. ZooKeeper must not be run on the same machines as those running the ElectricCommander servers.

To run ZooKeeper as a service, follow these steps:

1. For each machine on which you want to install ZooKeeper, navigate to the `zookeeper-wrapper.zip` file in the Commander `<install_dir>/utils` directory.

2. Copy or move the `zookeeper-wrapper.zip` file to each of the machines that you want to use for ZooKeeper.
3. Extract `zookeeper-wrapper.zip` to a directory of your choice.
For example, `/opt/zookeeper-<release_version>`.
4. Using a text editor, open the `wrapper.conf` file in the extracted `zookeeper-wrapper/conf` directory.
For example, `/opt/zookeeper-<release_version>/zookeeper-wrapper/conf`.
5. Add the path to the ZooKeeper home directory. For example:


```
# Path to unpacked zookeeper
set.default.ZOOKEEPER_HOME=/opt/zookeeper-<release_version>
```
6. Add the path to the JRE. For example:


```
set.default.JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64/jre
```

 Now you are ready to start ZooKeeper as a service.
7. Run `zookeeper start`

Running ZooKeeper as a Service on Windows

Install ZooKeeper on machines other than those running ElectricCommander. ZooKeeper must not be run on the same machines as those running the ElectricCommander servers.

To run ZooKeeper as a service, follow these steps:

1. For each machine on which you want to install ZooKeeper, navigate to the `zookeeper-wrapper.zip` file in the Commander `<install_dir>\utils` directory.
2. Copy or move the `zookeeper-wrapper.zip` file to each of the machines that you want to use for ZooKeeper.
3. Extract `zookeeper-wrapper.zip` to a directory of your choice.
For example, `C:\Users\Administrator\zooservice`
4. Using a text editor, open the `wrapper.conf` file located in the extracted `zookeeper-wrapper/conf` directory.
For example, `C:\Users\Administrator\zooservice\zookeeper-wrapper\zookeeper-wrapper\conf`
5. Add the path to the ZooKeeper home directory.
For example:


```
# Path to unpacked zookeeper
set.default.ZOOKEEPER_HOME=C:\Users\Administrator\zooservice\zookeeper-3.4.5
```

 Now you are ready to install and start ZooKeeper as a service.

6. Navigate to and click `InstallZooKeeper-NT.bat` to install ZooKeeper as a service.

The file is in the `zookeeper-wrapper\bin` directory.

For example, `C:\Users\Administrator\zooservice\zookeeper-wrapper\zookeeper-wrapper\conf`.

7. Navigate to and click `StartZooKeeper-NT.bat` to start ZooKeeper as a service.

The file is located in the `zookeeper-wrapper\bin` directory.

For example, `C:\Users\Administrator\zooservice\zookeeper-wrapper\zookeeper-wrapper\conf`.

If you choose to use the command-line interface or a script to start the service, enter `ZooKeeperCommand.bat start`.

Ensuring ZooKeeper Can Locate Java

Because ZooKeeper is a Java application, ensure Java is installed and ZooKeeper can locate it.

The default value for `JAVA_HOME` setting (in the `zookeeper-wrapper/conf/wrapper.conf` file) is

```
set.default.JAVA_HOME=/opt/electriccloud/electriccommander/jre
```

if Commander is not installed or not installed in its default directory, set `JAVA_HOME` in `wrapper.conf` to the location for Java. For example:

```
set.default.JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
```

Verifying that ZooKeeper is Running

To quickly check that the ZooKeeper software is running, follow these steps:

1. Log in to each ZooKeeper machine and run the command:

```
echo ruok | nc 127.0.0.1 2181
```

2. Confirm that you get the following response from each ZooKeeper instance:

```
imok
```

Note: If you get no response, or a 'broken pipe' error, ZooKeeper is not running.

3. Obtain more information about the status of Zookeeper by logging into each ZooKeeper machine and running the command:

```
echo status | nc 127.0.0.1 2181
```

Exhibitor Software

After installing ZooKeeper you might want to install the optional Exhibitor software on every machine with an instance of Zookeeper. The Exhibitor software provides a web interface that allows you to monitor the status of ZooKeeper. It also keeps the configurations of all the ZooKeeper nodes in sync when any of them are changed, and provides tools to rotate and prune the ZooKeeper logs, to prevent them from growing indefinitely. For more information, see the [Exhibitor documentation](#).

If you choose to install Exhibitor, you must configure your Commander server nodes so they know how to contact Exhibitor. If you have already set up your Commander server cluster and ZooKeeper servers and are later adding Exhibitor to it, this can be done using the `ecconfigure` tool, which is normally found at `/opt/electriccloud/electriccommander/bin/ecconfigure` on Linux or `C:\Program Files\Electric Cloud\ElectricCommander\bin\ecconfigure.exe` on Windows. Bring down all nodes

in your cluster, and run `ecconfigure` on each Commander node in the cluster with the option (as the user that Commander runs as, or with administrative privileges):

```
ecconfigure --serverExhibitorConnection <Exhibitor_servers>
```

where `<Exhibitor_servers>` is a comma-separated (no spaces) list of the IP addresses or fully-qualified domain names of your three or five (or for a test system possibly just one) Exhibitor servers (the port number that Commander uses to connect to Exhibitor is always port 80). For example, `10.0.2.1,10.0.2.2,10.0.2.3` for a three-ZooKeeper/Exhibitor cluster.

Note: It is not necessary to use the command with a single exhibitor and single ZooKeeper server.

Configuring a Multi-ZooKeeper Cluster

If you plan to use a multi-ZooKeeper cluster, you must configure each ZooKeeper with a unique number from the range 1, 2, 3 for a 3-ZooKeeper cluster, or 1, 2, 3, 4, 5 for a 5-ZooKeeper cluster. You must include this number in the following file:

```
<dataDir>/myid
```

where `<dataDir>` is the path you set in your `zoo.cfg` file.

For example, you can run these commands:

```
sudo touch /var/lib/zookeeper/myid
sudo -- sh -c 'echo <number> > /var/lib/zookeeper/myid'
```

where `<number>` is the appropriate number between 1 and 3, or 1 and 5.

ZooKeeper Requires a Majority of Nodes to Be Up

ZooKeeper requires a majority of its nodes to be up in order for it to be functional. A majority is:

- 1 of 1
- 2 of 3
- 3 of 5

If a majority of nodes is not up, the expected behavior is a "not currently serving requests" error from ZooKeeper.

Installing Commander Software

You must install the Commander software components on all the nodes in your Commander cluster. Where you install the individual components depends on the type of cluster configuration you need to create. For more information on how to install the Commander software, see [Installing ElectricCommander](#) on page 3-1.

Choose one of the following four installation approaches for your environment:

- [New Commander Installation for Reliability](#) on page 4-12
- [New Commander installation for Performance](#) on page 4-12
- [Converting an Existing Commander installation for Reliability](#) on page 4-13
- [Converting an Existing Commander Installation for Performance](#) on page 4-14

For any of these approaches, when you install agent, repository, and web server services, you can save time by configuring the software to point to a remote server location. You must also register all of these service agents as resources on the Commander server. For more information, see [Duplicating Repository Contents](#) on page 4-14.

- For a command-line installation, set the option `--remoteServer <load_balancer_FQDN>`. If you are doing an advanced installation on Linux, when prompted for the remote commander server, enter the `<load_balancer_FQDN>`.
- For a graphical user interface installation, set the Server Host Name field of the "Remote commander server" installer page to `<load_balancer_FQDN>:8000`.
The `load_balancer_FQDN` is the fully qualified domain name of your load balancer machine.

New Commander Installation for Reliability

The reliability approach allows multiple Commander services to run on a machine, but multiple instances of the service should exist to prevent single points of failure.

1. Install the Commander server and agent software on one node in the Commander cluster.

Note: If you do not already have a Commander web server that you can temporarily point at this Commander server node, you might want to also install a Commander web server that can be used for the following two steps in this section. Before you install the Commander server and agent software on the remaining nodes in the Commander cluster, turn off the web server. You turn off the web server on Linux by using the command `/etc/init.d/commanderApache stop`, or on Windows by stopping the service and setting the Startup Type to Manual.

2. Configure Commander to use an external database. At this time, the Commander node is in a single-server configuration.

For more information, see [Switching to an Alternate Database](#) on page 11-15.

3. Move the plugins directory on the Commander server software node to a location on the shared file system.

For more information, see [Universal Access to the Plugins Directory](#) on page 5-7.

4. Install the Commander server and agent software on the remaining nodes in the Commander cluster.
5. Install the Commander repository service on one or more machines.
6. Register agents on these machines as resources on the Commander server.
7. Install the Commander web server service on one or more machines.

New Commander installation for Performance

The performance approach requires separate machines for each Commander service.

1. Install just the Commander server software on all the nodes in the Commander cluster.

For more information, see [Silent Unattended Installation Method](#) on page 3-27.

Note: If you do not already have a Commander web server that you can temporarily point at this Commander server node, you may also want to install a Commander web server that can be used for the following two steps in this section. Before you install the Commander server and agent software on the remaining nodes in the Commander cluster, turn off the web server. You turn off the web server on Linux by using the command `/etc/init.d/commanderApache stop`, or on Windows by stopping the service and setting the Startup Type to Manual.

2. Configure one instance of the Commander server software to use an external database.

At this time, the Commander node is in a single-server configuration.

For more information, see [Switching to an Alternate Database](#) on page 11-15.

3. Move the plugins directory on the Commander server software node to a location on the shared file system.

For more information, see [Moving the Plugins Directory to a Pre-Configured Network Location](#) on page 5-7.

4. Install the following software services on one or more individual machines.

Each service should not be installed with any other Commander software components.

- Commander agent
- Commander repository server
- Commander web server

5. Remove any agents that were automatically installed with the Commander server.

For more information, see [Separating Agents from Commander Servers](#) on page 4-21 and [Verifying Commander Services](#) on page 4-22.

Converting an Existing Commander installation for Reliability

Because this is a conversion of an existing Commander system, one or more machines with the Commander server, agent, web server, and repository software already exist. The reliability approach allows multiple Commander services to run on a machine, but multiple instances of the service should exist to prevent single points of failure.

1. Upgrade the existing Commander software according to the instructions in [Roadmap to Upgrade ElectricCommander on page 6-1](#).

Horizontal scalability is supported starting with Commander 5.0.

2. Verify that the Commander software is pointing to an external database.

To verify which database is in use:

- a. Log in to Commander.
- b. Select **Administration > Database Configuration** to see the current database.

The database connection is successfully configured if you can log into Commander.

See [Switching to an Alternate Database](#) on page 11-15 if additional configuration is required.

3. Verify that the Commander software is configured to use a plugins directory located on the shared file system.

For more information, see [Universal Access to the Plugins Directory](#).

4. Install the Commander server and agent software on the remaining nodes for the Commander cluster.
5. Install the Commander repository server on one or more machines.
6. Install the Commander web server on one or more machines.
7. Register the machine agents as resources on the Commander server.

Converting an Existing Commander Installation for Performance

Because this is a conversion of an existing Commander system, one or more machines with the Commander server, agent, web server, and repository software already exist. The performance approach requires separate machines for each Commander service.

1. Upgrade the existing Commander software according to the instructions in [Roadmap for the Upgrade Process](#).

Horizontal scalability is supported starting with Commander 5.0.

2. Verify that the Commander software is pointing to an external database.

To verify which database is in use:

- a. Log in to Commander.
- b. Select **Administration > Database Configuration** to see the current database.

The database connection is successfully configured if you can log into Commander.

See [Switching to an Alternate Database](#) on page 11-15 if additional configuration is required.

3. Verify that the Commander software is configured to use a plugins directory located on the shared file system.

For more information, see [Universal Access to the Plugins Directory](#).

4. Remove any web server or agent software that is installed with the original Commander machine.

This software will be reinstalled on a separate system.

5. Install the Commander server software on the nodes for the Commander cluster.

6. Install the following software services on one or more individual machines.

Each machine should not be installed with any other Commander software services.

- Commander agent
- Commander repository server
- Commander web server

7. Remove any agents that were automatically installed with the Commander server, web server, and repository services.

The original Commander machine as well as the new installations should be checked to verify that the agent software is removed.

For more information, see [Separating Agents from Commander Servers](#) on page 4-21 and [Verifying Commander Services](#) on page 4-22

8. Remove any repository server software that is installed with the original Commander machine after you duplicate the repository server contents.

For more information, see [Installing Commander Software](#) on page 4-11.

If necessary, install the repository software on additional machines.

Duplicating Repository Contents

To duplicate the contents of an existing repository server into a newly installed repository server:

1. Stop the repository servers.
2. Copy the entire contents of the repository backingstore directory from the existing repository server to the corresponding location on the newly installed repository server.

The default location for the backingstore directory (`<datadir>/repository-data`) is:

- UNIX - `/opt/electriccloud/electriccommander/repository-data`
- Windows - `C:\ProgramData\Electric Cloud\ElectricCommander\repository-data`

3. Start both repository servers.

Running a Cluster in Clustered Mode

You must configure the machines installed with the Commander software to operate as a cluster.

To configure them in clustered mode:

1. Share the plugins directory across the Commander servers, agents, and web servers if you have not done so already. For more information, see [Universal Access to the Plugins Directory](#).

2. Configure all the Commander servers for clustered mode operation, giving them a unique identifier that points to the load balancer, the location of the ZooKeeper servers, and optionally also the Exhibitor servers if you are using Exhibitor.

You must use the `ecconfigure` commands on each Commander server, web server, and agent on which the software component is installed. You must enter the commands on each server that you add. There are no problems if you rerun the commands on a machine. Use one of these methods:

- Use the `ecconfigure` tool.

- a. Locate the `ecconfigure` tool.

For Linux, it is usually at

```
/opt/electriccloud/electriccommander/bin/ecconfigure.
```

For Windows, it is usually at `C:\Program Files\Electric Cloud\ElectricCommander\bin\ecconfigure.exe`.

- b. Run it with the following options on each Commander server node (as the user that Commander runs as, or with administrative privileges):

```
ecconfigure --serverName <load_balancer_FQDN> --  
serverZooKeeperConnection <ZooKeeper_servers>
```

- Use Exhibitor and enter:

```
ecconfigure --serverName <load_balancer_FQDN> --  
serverZooKeeperConnection <ZooKeeper_servers> --  
serverExhibitorConnection <Exhibitor_servers>
```

where

- `<load_balancer_FQDN>` is the fully-qualified domain name of your load balancer machine
- `<ZooKeeper_servers>` is a comma-separated (no spaces) list of the `IP_address_or_FQDN:port_number` for each of your three or five (or for a test system possibly just one) ZooKeeper servers (the port number for ZooKeeper is normally 2181).
- If you are using Exhibitor, `<Exhibitor_servers>` is a comma-separated (no spaces) list of the IP addresses or fully-qualified domain names of your three or five (or for a test system possibly just one) Exhibitor servers (the port number that Commander uses to connect to Exhibitor is always port 80). For example, `10.0.2.1,10.0.2.2,10.0.2.3` for a three-ZooKeeper/Exhibitor cluster.

Example for a three-ZooKeeper cluster:

```
ecconfigure --serverName machine.company.com --serverZooKeeperConnection  
10.0.2.1:2181,10.0.2.2:2181,10.0.2.3:2181
```

3. Configure the load balancing software for the Commander server cluster and Commander web servers.

For more information, see the instructions from the manufacturer.

Running a Cluster in Single-Server Mode

A couple of rarely-used Commander operations are not supported in clustered mode. If you need to perform any of the following operations, you must restart the server in single-server mode:

- Changing the database configuration
This operation can be done in the web UI using Administration > Database Configuration or with ectool. After performing this operation, you must rerun ZKConfigTool to upload the changed configuration from the local database.properties file to ZooKeeper before switching back to clustered mode.
- Doing a full import using ectool
- Doing a full export using the safe mode with either of these options:
 - Restart
 - Shutdown

To run a Commander machine in single-server mode:

1. Identify the machine you need to work on and shut down the other machines in the Commander cluster.
2. Verify that the database.properties, keystore, passkey, and commander.properties configuration files on the machine you are working on are up to date and match those that were uploaded to ZooKeeper.
3. Switch this remaining Commander machine to single-server mode by running the following command:

```
ecconfigure --serverEnableClusteredMode=0
```
4. Restart that Commander machine.
5. Complete your work on the Commander machine.
6. Switch this remaining Commander machine back to clustered mode by running the following command:

```
ecconfigure --serverEnableClusteredMode=1
```
7. If you changed the database configuration, use ZKConfigTool to upload the updated database.properties file to ZooKeeper. See [Uploading Configuration Files to ZooKeeper](#) on page 4-17 for more information.
8. Restart all machines in the Commander cluster.

Adding the Configuration to ZooKeeper

After setting the cluster in clustered mode, you must populate ZooKeeper with your configuration files.

Use ZKConfigTool to upload configuration files from your pre-existing Commander server to ZooKeeper.

Uploading Configuration Files to ZooKeeper

Before starting the ElectricCommander server cluster, you need to populate an Apache ZooKeeper server with configuration files that all ElectricCommander server nodes will use in a clustered configuration. This command-line tool imports your ElectricCommander database configuration files into your ZooKeeper server.

The following minimum set of files is imported:

- database.properties
- keystore
- passkey
- commander.properties

Prerequisites

- The ElectricCommander server package must be installed on the system.
- The system must be running a version of Java supported by ElectricCommander. Java is automatically installed on a system with the ElectricCommander software as part of the Tools installation.
- The ZooKeeper software must be installed on the network.

Location

The Commander installer adds ZKConfigTool to the following default locations:

- **Windows:** C:\Program Files\Electric Cloud\ElectricCommander\server\bin\zk-config-tool-jar-with-dependencies.jar
- **Linux:** /opt/electriccloud/electriccommander/server/bin/zk-config-tool-jar-with-dependencies.jar

Command

ZKConfigTool is best run from the `<Commander install dir>/conf` directory.

```
$ java -jar zk-config-tool-jar-with-dependencies.jar -<options>
```

Option	Description
<code>-commanderPropertiesFile <path_to_file></code>	Import the ElectricCommander commander.properties file.
<code>-databasePropertiesFile <path_to_file></code>	Import the ElectricCommander database.properties file.
<code>-help</code>	Show the command help.
<code>-keystoreFile <path_to_file></code>	Import the ElectricCommander keystore file.
<code>-passkeyFile <path_to_file></code>	Import the ElectricCommander passkey file.
<code>-writeFile <path_on_zookeeper> <path_to_file></code>	Write the specified file to the ZooKeeper server.

Import Files

Run ZKConfigTool to completely populate the ZooKeeper server with configuration information. The system must have the Tools installation and can communicate with ZooKeeper.

- If you run the command with defined file options, the tool immediately attempts to contact the specified ZooKeeper system and import the specified files.

Example command run from the conf directory:

```
COMMANDER_ZK_CONNECTION=<ZooKeeper_Server_IP>:2181 ../jre/bin/java -jar
../server/bin/zk-config-tool-jar-with-dependencies.jar
com.electriccloud.commander.cluster.ZKConfigTool --databasePropertiesFile
database.properties --keystoreFile keystore --passkeyFile passkey
--commanderPropertiesFile commander.properties
```

- If you run the command without defined file options, the tool displays a series of prompts for the required Commander information. After the correct information is entered, the tool contacts the specified ZooKeeper system to import the specified files.

Example command run from the conf directory:

```
COMMANDER_ZK_CONNECTION=<ZooKeeper_Server_IP>:2181 ../jre/bin/java -jar
../server/bin/zk-config-tool-jar-with-dependencies.jar
com.electriccloud.commander.cluster.ZKConfigTool
```

Next Steps

After you have uploaded the new configuration files, you should shut down all the Commander servers in your cluster if any are running. Then restart all the Commanders servers in the cluster.

Configuring Web Server Properties

You must update the `httpd.conf` file on each web server in the cluster. The `httpd.conf` file is usually in `apache/conf` on a Linux machine and `ProgramData\Electric Cloud\ElectricCommander\apache\conf` on a Windows machine.

To configure all the web servers for clustered mode operation and give them the name of the load balancer:

1. Locate the `ecconfigure` tool.

On Linux, it is usually at `/opt/electriccloud/electriccommander/bin/ecconfigure`.

On Windows, it is usually at `C:\Program Files\Electric Cloud\ElectricCommander\bin\ecconfigure.exe`.

2. Run the tool with the following option on each web server. You may need to do this as root or with administrative privileges.

```
ecconfigure --webTargetHostName <load_balancer_FQDN>
```

where `<load_balancer_FQDN>` is the fully qualified domain name of your load balancer machine.

Configuring Repository Server Properties

Additional configuration is required for any repository servers that existed before being converted to operate in a cluster, or were not configured to point to a remote server location during installation.

To configure all the repository servers for clustered mode operation and give them the name of the load balancer:

1. Locate the `ecconfigure` tool.

On Linux, it is usually at `/opt/electriccloud/electriccommander/bin/ecconfigure`.

On Windows, it is usually at C:\Program Files\Electric Cloud\ElectricCommander\bin\ecconfigure.exe.

2. Run the tool with the following option on each repository server, You may need to do this as root or with administrative privileges.

```
ecconfigure --repositoryTargetHostName <load_balancer_FQDN>
```

where <load_balancer_FQDN> is the fully qualified domain name of your load balancer machine.

Configuring Commander Agents

You must configure Commander agents to function within a resource pool.

1. Start and log in to Commander.
2. Go to the **Cloud > Resources** page.
3. Delete any resource named *local*.
4. Create a resource pool named *local*.
5. Create resources for all the machines that had the Commander agent software installed.
6. Add agent resources to the local resource pool.

Choose the appropriate step for your approach.

- If you are creating a reliability configuration where each Commander server machine also has an agent installed, the local resource pool should consist of the set of agents local to the Commander server machines.
- If you are creating a performance configuration where none of the Commander server machines have agents installed, the local resource pool (which in this configuration is not actually local to the servers) should contain several agents to handle any work that may be assigned. For example, work may be assigned due to old default resource settings.

7. Verify that the default resource pool contains two or more resources for reliability.

You must create and add resources to this pool if none exist.

8. Go to the **Administration > Server** page.
9. Click on the **Settings** link near the top right to open the Edit Server Settings form.
10. Set the **Server IP address** entry in the form to the fully qualified domain name of your load balancer.

This setting controls how agents contact the Commander server when they send results from jobs and similar messages.

11. Set the **Stomp Client URI** entry to

```
stomp+ssl://<load_balancer_FQDN>:61613
```

where <load_balancer_FQDN> is the fully qualified domain name of your load balancer.

12. Click to clear the **Use SSL for Stomp** check box.

Note: If you add additional agents, you must configure the agents to use the shared plugins directory. For more information, see [Universal Access to the Plugins Directory](#).

Configuring the Cluster Workspace

You must edit the default workspace for log files across the Commander servers, agents, and web servers.

IMPORTANT: As you need and create more workspaces over time, each workspace should be in a shared network location that all machines in the Commander cluster can access.

To edit the default workspace:

1. Select **Cloud > Workspaces**.
2. Edit the default workspace entry to reference a shared network location.

For more information, see the “Workspaces and Disk Space Management “ topic in the Commander Help.

Configuring Commander Repositories

You must configure Commander repositories.

1. Go to the **Commander Artifacts > Repositories** page.
2. Verify that the repository server URL points to the load balancer machine.

Adding Trusted Agents to Clusters

Perform the following procedure in these situations:

- Every time that you create trusted agents.
- Every time that you revoke the certificates of existing trusted agents.
- To create a gateway to a zone with trusted agents at both ends. You have to do this only a few times.
- When you want all agents in cluster to be trusted.

Follow these steps to add a trusted agent to a cluster:

1. Shut down all but one node in the cluster.
2. On the machine that is now the only node running, enter these commands to create a trusted agent:

```
ectool --server 10.168.33.12 login admin changeme

/opt/electriccloud/electriccommander/bin/eccert initAgent --remote --force

ectool createResource <agent_name> --hostName <agent_FQDN_or_IP> --trusted true
--port 78
```

3. Enter `scp -r` to copy the `/opt/electriccloud/electriccommander/conf/security/` directory subtree from the only node currently running to the other nodes' machines.

Separating Agents from Commander Servers

Use this procedure if you need to separate Commander services and agents. By default, a Commander agent is installed with the Commander server, web server, and repository. For more information, see [Resource, Agent, and Procedure Configuration Considerations](#) on page 4-2 and [Verifying Commander Services](#) on page 4-22.

1. Verify that no Commander agents are installed on any of the Commander server nodes. If necessary, remove the agent software from the Commander server nodes.

2. Verify that none of the Commander utilities use a local resource. If you are not sure if a local resource is in use, create an agent resource called */oca/* and monitor the system.
3. Remove the local resource.
4. Create a new agent resource with a new name for each agent on each Commander server node machine.
5. Create a resource pool named */oca/* containing all these resources.

Verifying Commander Services

You can verify what services are on a machine installed with Commander software by the following methods:

- If you have a Linux system
 1. Go to the `/etc/init.d/` directory.
 2. Look for scripts starting with "commander". For example, `commanderAgent`, `commanderApache`, `commanderRepository`, `commanderServer`.
 3. As root, from any directory, use this command format:


```
/etc/init.d/<service_name> status
```


where `<service_name>` is the Commander service you are interested in, such as


```
/etc/init.d/commanderServer status
```
- If you have a Windows system:
 1. Go to the Services control panel.
 2. Look for services starting with the name "ElectricCommander". For example, ElectricCommander Agent, ElectricCommander Database, ElectricCommander Server.
 3. If the services have a status of Started, they are installed and running.

Accessing Commander with Clustering

You access a Commander server in a clustered configuration the same way you would for a single-server configuration.

Enter the address of one of the web servers into your browser address bar.

If you are using `ectool`, use the `--server` option to direct your request to the fully qualified domain name of the load balancer.

Additional Ways to Improve a Commander Cluster

Clustering your Commander configuration does not necessarily remove all single points of failure or potential performance bottlenecks from your system. There are other components that are part of the Commander environment that you must consider to eliminate single points of failure or prevent performance bottlenecks.

Third-Party Software

The following items are widely-used third-party commercial products that are used in conjunction with Commander. A variety of solutions and strategies to increase the reliability and scalability of these products and eliminate remaining single-points-of-failure are available from other vendors and sources.

- Network
- Load balancer
- External database
- File server used for the shared file system

Commander Components

You can address some or all of the following potential issues with the following Commander components. The issues you address depends on the level of reliability and performance you need for your system.

- Repository server - You can mirror your artifacts across multiple repository servers.
- Commander procedures or steps - You can specify a resource pool of agents rather than a single agent.

Chapter 5: Configuration

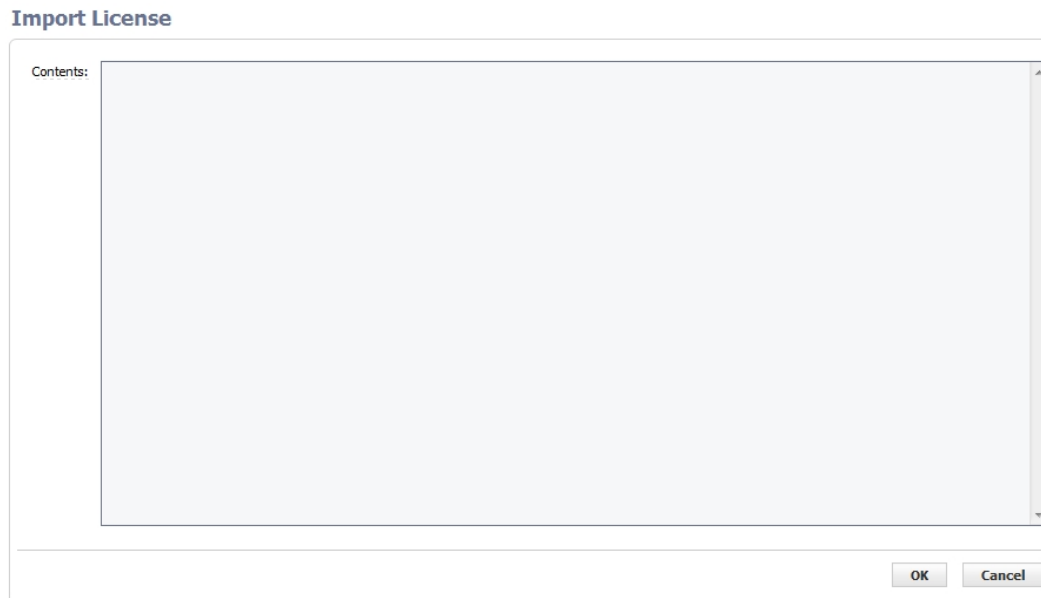
This section contains the configuration tasks you must perform after you install ElectricCommander .

Applying an Enterprise License Key

Use the following task to add an enterprise license to a Commander server.

1. Log into the Commander server. For more information, see [Logging Into the ElectricCommander Web Interface](#) on page 3-37.
2. Go to **Administration > Licenses**.
3. Click **Import License**.

The Import License text box appears.



4. Open the license file in a text editor and copy and paste the entire license text in the Import License text box.
5. Click **OK** to import the Commander license.

External Database Configuration

During the Commander server installation, if you elected *not* to install the built-in [default] database, you need to configure your alternate database choice. A Commander enterprise license is required to configure an alternate

database. For more information about supported databases, see [Database Requirements](#) on page 2-8. For installation instructions, see [Configuring Commander to Use an Alternate Database](#) on page 5-2.

Database Interactions

Your database administrator (DBA) must create a database for use specifically by Commander. The Commander server interacts with the database using a JDBC driver for each of the databases Commander supports.

For MySQL, the JDBC driver is not installed by the Commander installer (for licensing reasons). It must be downloaded and installed separately from the MySQL website. For more information, go to [Installing the MySQL JDBC Driver](#) on page 3-37.

The first step in any interaction is to present user credentials to the database. This information is stored in the Commander `database.properties` file as a user name plus a password. The password is stored as an encrypted string, using the “passkey” generated by the server.

Database User

For MS SQL Server and MySQL, your DBA should create a database user for use specifically by Commander.

For Oracle, your DBA must create a database user for use specifically by Commander.

The Commander database user must have permissions to add or delete rows from the database at all times. The database user must also have rights to create or delete tables, and add or remove a columns, indices, and constraints to a table at certain defined times.

IMPORTANT:

Electric Cloud only supports installations where the database user has rights to create and delete tables at all times.

To upgrade the MySQL databases, the database user must also have permissions to create and delete schema (databases) for the entire duration of the upgrade process.

For security reasons, permissions granted to database users to create and delete schema in multi-tenant MySQL databases may be revoked after the upgrade process is completed.

When the database first starts up, Commander creates a schema in the specified database so the database user should be the owner of the Commander database. This will allow the Commander software to make required schema changes.

Configuring Commander to Use an Alternate Database

If you deselected the “database” check box during installation, you will NOT be able to log into Commander until you set up a database configuration pointing to an external database. A Commander enterprise license is required, see [External Database Configuration](#) on page 5-1. You can change the database configuration through the Commander web interface or the Commander command-line tool.

If you are currently using Commander in clustered mode and need to change the database configuration, you must reset Commander to single-server mode before changing the database configuration. For more information, go to [Running a Cluster in Single-Server Mode](#) on page 4-16.

Setting the Database with the Web Interface

Use this procedure to set the database with the Commander web interface. You will NOT be able to log into Commander until you set up a database configuration pointing to an external database, but you can use the Commander web interface to connect to an external database.

1. Go to the Administration tab in the Commander UI and select Database Configuration.

The Database Configuration screen appears.

The screenshot shows the 'Database Configuration' screen in the Commander UI. The top navigation bar includes tabs for Home, Deploy, Projects, Jobs, Workflows, Cloud, Artifacts, Search, and Administration. The Administration tab is active, and the 'Database Configuration' sub-tab is selected. The form contains the following fields:

- Database Type:** A drop-down menu with a required field indicator.
- Database Name:** A text input field with a required field indicator.
- Host Name:** A text input field with a required field indicator.
- Port:** A text input field.
- Database Credentials:** A section containing:
 - User Name:** A text input field with a required field indicator.
 - Password:** A text input field.
 - Retype Password:** A text input field.

At the bottom right of the form are two buttons: 'Save and Restart Server' and 'Cancel'.

2. Select your **Database Type** from the drop-down menu.
3. Fill-in your **Database Name**.
4. Fill-in the **Host Name** for your database server.
5. Accept the default **Port** or supply the port number you need for your database.
6. Supply the database **User Name** the Commander server will use to access your database.
7. Fill-in and confirm the **Password** for the database user you specified.
8. Click **Save and Restart Server** after supplying information in all fields.

Setting the Database from a Command Line

This section contains topics related to setting an alternate database for ElectricCommander through a command line.

SQL Server Authentication

SQL Server supports two types of user authentication:

- SQL Server Authentication
- Windows Authentication

You must find out from your DBA which authentication type is required for the Commander user because, the authentication type influences how information is provided in the `ectool` command `setDatabaseConfiguration`. See [setDatabaseConfiguration Command Examples](#) on page 5-4 for example command syntax.

Setting the Database with `ectool`

Use the `ectool` command, `setDatabaseConfiguration` to change the database configuration from a command line.

1. Determine if your database is SQL Server. The type of user authentication utilized by the database impacts the syntax of the `setDatabaseConfiguration` command you use in this procedure. For more information, see [SQL Server Authentication](#) on page 5-3 and [setDatabaseConfiguration Command](#)

[Examples](#) on page 5-4.

2. Go to a command line and enter:

```
setDatabaseConfiguration <--options>
```

Where *<options>* represents the options you add to the command. For more information, see [setDatabaseConfiguration Command Options](#) on page 5-4.

After changing the ElectricCommander database configuration, the server attempts to connect to the database to do the initial schema setup.

3. Use `ectool getServerStatus` to see if there are problems logging into the database or creating the schema. This command shows log messages from the server bootstrap process.

Note: Before the server is completely up, `getServerStatus` does not require a login session, but after the server is up, it does. Thus, if you issue a `getServerStatus` call and get a “session expired” error, the server is up.

setDatabaseConfiguration Command Examples

The following text shows some examples of the `setDatabaseConfiguration` command.

SQL Server Authentication:

```
ectool setDatabaseConfiguration --databaseType sqlserver --databaseName commander  
--hostName localhost --port 1433 --userName commander --password commander
```

The `--userName` and `--password` options must be included in the `setDatabaseConfiguration` command.

SQL Server with NTLM login:

```
ectool setDatabaseConfiguration --databaseType sqlserver --databaseName commander  
--hostName localhost --port 1433 --userName commander@domain.com --password commander
```

The user name must include the domain name, for example, `user@domain.com` or `domain\user`.

SQL Server with SSO login:

```
ectool setDatabaseConfiguration --databaseType sqlserver --databaseName commander  
--hostName localhost --port 1433 --userName ""
```

The commander service must be run as the domain user, and the user name must be set to “blank” (for example, `--userName ""`). Do not specify a password if using SSO; the server will forward credentials from the service itself.

setDatabaseConfiguration Command Options

The following options are available for use with the `ectool` command `setDatabaseConfiguration`. The option command syntax is:

```
setDatabaseConfiguration  
    [--databaseType <mysql|sqlserver|Oracle|builtin>]  
    [--databaseName <database name>]  
    [--hostName <host name>]  
    [--ignorePasskeyMismatch <Boolean flag>]  
    [--ignoreServerMismatch <Boolean flag>]  
    [--password <password>]  
    [--port <port number>]  
    [--preserveSessions <Boolean flag>]
```

```
[--userName <user name>]  
[--customDatabaseDialect <custom database dialect>]  
[--customDatabaseDriver <custom database driver>]  
[--customDatabaseUrl <custom database URL>]
```

The following table describes the command options:

Option	Description
databaseType	Selects the database type—supported options are <code>mysql sqlserver Oracle builtin</code>
databaseName	The name of your alternate database—this is not the host name, but the name the DBA gave the database object.
hostName	The host name where your database is running
ignorePasskeyMismatch	<Boolean flag - 0 1 true false> - If the server is started with a different passkey, ignore the mismatch if “true”. Note: This action discards all saved passwords.
ignoreServerMismatch	<Boolean flag - 0 1 true false> - If the server is started on a different host than where the server previously started, ignore the mismatch if “true”.
port	The port number used by the database Server—if omitted, port 1433 is used
preserveSessions	<Boolean flag - 0 1 true false> - If ignoring a server mismatch, default behavior invalidates all sessions. Setting this flag to “true” saves all sessions, allowing the server to reconnect to running jobs. This option is used in combination with <code>ignoreServerMismatch</code> .
userName	The user name to use when connecting to the database—For MS SQL Server, see the information on the next page regarding support for SQL Server Authentication and Windows Authentication.
password	The password to use to connect to the database—For MS SQL Server, see the information on the next page regarding support for SQL Server Authentication and Windows Authentication.
customDatabaseDialect	Internal option - use only at the request of Electric Cloud support
customDatabaseDriver	Internal option - use only at the request of Electric Cloud support
customDatabaseUrl	Internal option - use only at the request of Electric Cloud support

Enabling Full Database URL Control

If you want to enable full control over the database URL:

1. Go to the `database.properties` file.
2. Create a property called `COMMANDER_CUSTOM_DB_URL`.
3. Add the explicit connect string to that property.

Note: `COMMANDER_DB_URL` is a “write-only” property that it is never read by Commander. It is generated based on other settings in that file and serves as information only.

If you were using Commander in clustered mode, you should now upload your modified `database.properties` file to ZooKeeper and then change Commander back to clustered mode. For more information, go to [Running a Cluster in Clustered Mode](#) on page 4-15.

Universal Access to the Plugins Directory

A plugin is a collection of one or more features, or a third-party integration or tool that can be added to Commander. The Commander server installs all plugins into a configurable location named the plugins directory. This directory must be readable by the web server and any agents that need access to the content of one or more plugins. There are two ways to make the plugins directory readable by the web server and any agents. You can configure the Commander server, agents, and web servers to point to a central network location, or you can replicate the contents of the plugins directory on remote agents and web servers.

Configuring Universal Access for a Network Location

Use these procedures to configure Commander server, agents, and web servers to point to a universally accessible network location. This is the recommended approach because newly installed plugins are immediately available to all agents and web servers. We strongly recommend that you do this when you are running Commander in clustered mode, because it allows all the server nodes to share a common plugins directory. You also avoid the overhead of managing multiple plugins directories.

Note: Windows local system accounts cannot access remotely mounted plugin directories.

Ways to Configure Universal Access to Plugins

A network location for the plugins directory can be set up in one of two ways:

- [Moving the Plugins Directory to a Pre-Configured Network Location](#) on page 5-7 - This approach is recommended if you already have a network file system accessible to the Commander server and all remote agents and web servers.
- [Leaving the Plugins Directory on the Commander Server](#) on page 5-8 - This approach leaves the plugins directory in the current location on the Commander server and shares the location across the network. *This approach is only recommended if you do not already have a network location available.*

Moving the Plugins Directory to a Pre-Configured Network Location

Use this task to move the plugins directory to a pre-configured network location. You must have a network file system accessible to the Commander server and all remote agents and web servers to perform this task.

1. Create an empty directory in the network accessible location.
2. Move the contents of the plugins subdirectory from the Commander server's data directory to this new directory.
3. Run the following commands on the Commander server:
 - a. `ectool setProperty /server/settings/pluginsDirectory "<PLUGINS>"`
 - b. Run this command ONLY if a web server was installed on this machine (by default during an Express Server install):
`ecconfigure --webPluginsDirectory "<PLUGINS>"`

- c. Run this command ONLY if an agent was installed on this machine (by default during an Express Server install):

```
ecconfigure --agentPluginsDirectory "<PLUGINS>"
```
- d. Run this command ONLY if you plan on installing remote Windows agents or web servers:

```
ectool setProperty "/server/Electric Cloud/windowsPluginsShare" "<PLUGINS>"
```
- e. Run this command ONLY if you plan on installing remote UNIX agents or web servers:

```
ectool setProperty "/server/Electric Cloud/unixPluginsShare" "<PLUGINS>"
```

Where <PLUGINS> is either the Windows path to the shared directory for Windows machines, or the UNIX path to the shared directory for UNIX machines.

Note: When installing remote agents or web servers, you will be prompted to enter information about the Commander server. Select the checkbox to "discover the plugins directory", and the correct location is automatically picked up from the server and set during installation.

4. Run the following command on remote agents that were already installed:

```
ecconfigure --agentPluginsDirectory "<PLUGINS>"
```

Where <PLUGINS> is either the Windows path to the shared directory for Windows machines, or the UNIX path to the shared directory for UNIX machines.

5. Run the following command on remote web servers that were already installed:

```
ecconfigure --webPluginsDirectory "<PLUGINS>"
```

Where <PLUGINS> is either the Windows path to the shared directory for Windows machines, or the UNIX path to the shared directory for UNIX machines.

Leaving the Plugins Directory on the Commander Server

Use this task to leave the plugins in the current location on the Commander server and share that location across the network so remote agents and web servers can obtain access.

IMPORTANT: This approach is recommended only if you do not already have a network location available to the Commander server and all remote agents and web servers. See [Moving the Plugins Directory to a Pre-Configured Network Location](#) on page 5-7 if you do have a universally available network location.

1. Choose the appropriate step based on the Commander server platform:
 - If your Commander server is a Windows machine, the plugins directory is automatically shared by the name "commander-plugins" during installation. When you install remote agents and web servers on Windows, they will discover this location and be configured to use it.
 - If you are installing remote agents or web servers on UNIX machines, follow these steps:
 - a. Create a Samba mount on a UNIX machine pointing to the plugins share on the Windows machine,

```
//<COMMANDER_SERVER_HOST_NAME>/commander-plugins
```
 - b. Export the Samba mount as a network file system share on the same UNIX machine used in the previous step.
 - i. Add the following entry to `/etc/exports`:

```
/opt/electriccloud/electriccommander/plugins by default): <EXPORT> *  
(ro, sync)
```

Where <EXPORT> is the directory you want to export.
 - ii. Start/restart the NFS server.

- c. Mount the network file system share to an available directory before installation on UNIX remote agents and web servers. Make sure to mount the share to the same directory across all machines, henceforth referred to as <UNIX_PLUGINS>.
 - i. Create <UNIX_PLUGINS>
 - ii. Add the following entry to /etc/fstab:


```
<HOST>:<EXPORT> <UNIX_PLUGINS> nfs defaults 0 0
```

Where <HOST> is the host name of the machine on which the directory has been exported and <EXPORT> is the directory being exported on that machine.
 - iii. Call: `mount -a`
 - d. Run the following command on the Commander server machine:


```
ectool setProperty "/server/Electric Cloud/unixPluginsShare" "<UNIX_PLUGINS>"
```
2. Perform the following steps if your Commander server is a Linux machine and you are installing remote agents or web servers on other UNIX machines. The plugins directory is not automatically shared as on Windows.
 - a. Export the local plugins directory as a network file system share on the Commander server machine:
 - i. Add the following entry to /etc/exports (


```
/opt/electriccloud/electriccommander/plugins by default):
```

```
<EXPORT> *(ro, sync)
```

Where <EXPORT> is the directory you want to export.
 - ii. Start/restart the NFS server.
 - b. Mount the network file system share to an available directory before installation on UNIX remote agents and web servers. You must mount the share to the same directory across all machines. This share is referred to as <UNIX_PLUGINS> in the following steps:
 - i. Create <UNIX_PLUGINS>
 - ii. Add the following entry to /etc/fstab (replace <HOST> with the host name of the machine on which the directory has been exported and <EXPORT> with the directory being exported on that machine): <HOST>:<EXPORT> <UNIX_PLUGINS> nfs defaults 0 0
 - iii. Call: `mount -a`
 - c. On the Commander server machine, run the following command:


```
ectool setProperty "/server/Electric Cloud/unixPluginsShare" "<UNIX_PLUGINS>"
```
 - d. If you are installing remote agents or web servers on Windows machines, perform the following steps:
 - i. Create a Samba share on the Commander server that is accessible to Windows machines under the name <WINDOWS_PLUGINS>.
 - ii. Run the following command:


```
ectool setProperty "/server/Electric Cloud/windowsPluginsShare" "<WINDOWS_PLUGINS>"
```
3. Choose the appropriate step to configure remote agents or web servers.
 - If you need to install the software for a new remote agent or web server, you must select the **"discover the plugins directory"** option. This will allow the installer to automatically detect and set

the correct location from the server.

- If you need to configure remote agents that were previously installed, run the following command:
`ecconfigure --agentPluginsDirectory "<PLUGINS>"`
Where <PLUGINS> is the Windows path to the shared directory for Windows machines, or the UNIX path to the shared directory for UNIX machines.
- If you need to configure a remote web servers that were already installed, run the following command:
`ecconfigure --webPluginsDirectory "<PLUGINS>"`
Where <PLUGINS> is the Windows path to the shared directory for Windows machines, or the UNIX path to the shared directory for UNIX machines.

Replicating the Plugin Directory on Remote Systems

Use this procedure to keep the Plugins directory in its default server location and replicate the contents to remote agents and web servers.

IMPORTANT: This approach requires you to manage multiple plugin directories. Every time a new plugin is installed on the Commander server, you must synchronize the changes across all remote copies of the directory. **Only replicate the Plugins directory if you cannot use or configure a central network location.**

- Copy the plugins directory to remote agents and web servers using any file copy mechanism.
 - The copied plugins directories must be readable by only the remote agents and web servers.
 - Plugins should be copied to a plugins subdirectory within the data directory for each remote agent and web server.
 - Every time the Commander server Plugins directory is updated, you must synchronize the changes across all remote copies.

Environment Proxy Server Configuration

In your environment proxy servers might exist between an intranet and internet. Because proxy servers can inhibit certain types of internet access, you will need to set proxy settings for each impacted machine in your installation. Commander servers or web servers can be deployed behind a proxy server.

It is not a common practice to place repository servers or agent systems behind a proxy server since these systems communicate with Commander through an intranet connection.

Configuring Proxy Settings for Servers

Use `ecconfigure` to set proxy settings for any web server or Commander server in your configuration that is deployed behind a proxy server. Repository servers are not typically placed behind a proxy server.

1. Select the appropriate perl scripts to run depending on the server type.

- To set Commander server proxy settings, enter:

```
ec-perl src/ecconfigure.pl --serverProxyHost <IP_ADDRESS_PROXY> --  
serverProxyPort <PORT> --serverNoProxyHosts "<HOST1,HOST2>"
```

- To set web server proxy settings, enter:

```
ec-perl src/ecconfigure.pl --webProxyUrl http://<IP_ADDRESS:PORT> --  
webNoProxyHosts <HOST1,HOST2,HOST3>
```

Where:

<IP_ADDRESS_PROXY> is the IP address of the proxy server,

<PORT> is the server port for the proxy server, and

<HOST1,HOST2> is one or more comma separated host names for the servers in the configuration.

2. Restart all the servers where you have applied a proxy setting

IMPORTANT: If you do not restart the servers, the proxy settings will not work.

Testing Server Proxy Settings

Use the following task to verify your proxy server settings.

- Perform the following steps depending on your server type.
 - If you have a web server, perform the following steps:
 1. Go to the **Plugin Manager** web page.
 2. Verify the catalog can be viewed and no errors are reported when accessing the catalog URL.
 - If you have a Commander server
 1. Go to the **Plugin Manager** web page.
 2. Verify you can install a plugin from the catalog.

Configuring Proxy Agents

Use `ecconfigure` to set proxy settings for any agent system that is deployed behind a proxy server. A proxy server is not usually placed between agents and a Commander server.

IMPORTANT: When you use a proxy agent, the proxy target *must* run an SSH v2 server. For more information on proxy target agents, see the Configuration and Resource online help topics.

1. Run the following command to set Agent proxy settings: `ec-perl src/ecconfigure.pl --agentProxyHost <IP_ADDRESS_PROXY> --agentProxyPort <PORT> --agentNoProxyHosts "<HOST1,HOST2>"`

Where:

<IP_ADDRESS_PROXY> is the IP address of the proxy server,

<PORT> is the server port for the proxy server, and

<HOST1,HOST2> is one or more comma separated host names for the servers in the configuration.

2. Set the cygwin 1.7 privilege by running the following commands.

Note: Certain commands require administrator privileges to run (for example, `net stop xxx`) using cygwin 1.7 `sshd`. These commands can fail with “access denied” errors. These errors did not occur in cygwin 1.5. The Commander proxy agent relies on `sshd` being privileged. To set this privilege on cygwin 1.7, you need to run an additional setup script (in addition to `ssh-host-config`).

- a. `cyglsa-config`
- b. `reboot`

Increasing File Descriptors on Linux

A file descriptor is an object a process uses to read or write to an open file and open network sockets (although there are other uses).

Operating systems place limits on the number of file descriptors a process can open. In addition to per-process limits, an OS also has a global limit on the number of file descriptors that all its processes, together, might consume.

A common bottleneck in the default Linux operating system configuration is a lack of file descriptors.

ElectricCommander Server

An ElectricCommander Server uses approximately one file descriptor per running job step and three per uncompleted job.

The following example configures Commander to use a new limit of 32768:

1. Add the following line to the `init` script for the Commander Server (in `/etc/init.d/commander`) before the `su -` command:

```
ulimit -n 32768
```

2. Restart the Commander server:

```
/etc/init.d/commanderServer restart
```

ElectricCommander Agent

An ElectricCommander Agent uses at least two file descriptors per running job step.

It is important to make sure that operating systems on high traffic sites are configured to provide sufficient numbers of file descriptors to Commander.

The following example describes how to raise the maximum number of file descriptors to 32768 for the ElectricCommander process on the Red Hat Linux distribution:

1. Allow all users to modify their file descriptor limits from an initial value of 1024 up to the maximum permitted value of 32768 by changing `/etc/security/limits.conf`. The following two lines should be part of the file contents:

- `soft nfile 1024`
- `hard nfile 32768`

2. In `/etc/pam.d/login`, add the following line if it does not already exist:

```
session required pam_limits.so
```

3. Configure Commander to use the new limits. Add the following line to the `init` script for the Commander Agent (in `/etc/init.d/ecmdrAgent` or `/etc/init.d/commanderAgent`):

```
ulimit -n 32768
```

4. Restart the Commander agent:

```
/etc/init.d/ecmdrAgent restart
```

Adjusting Swappiness on Linux

For Java-based machines (Commander server, repository server, agent), We recommend that you adjust the swappiness kernel parameter to favor applications over disk cache.

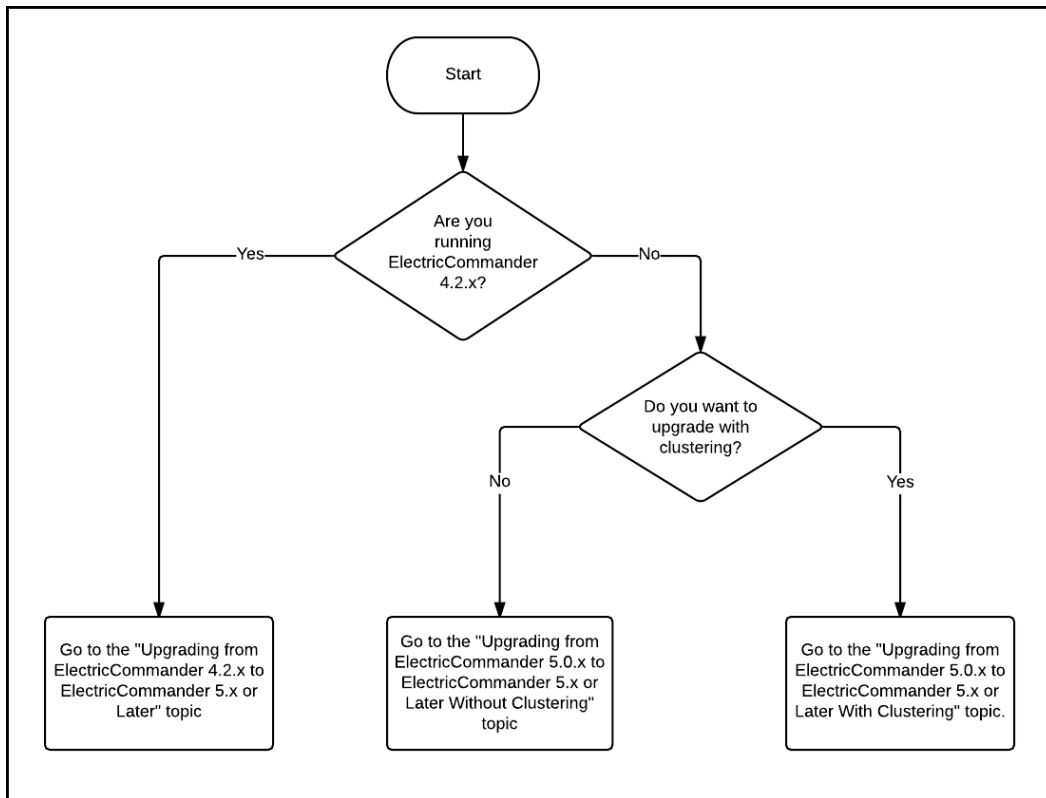
- To favor applications 100% over disk cache use:

```
sysctl -w vm.swappiness=0
```

The default of 60 can result in significant delays during garbage collection if any i/o intensive process runs on the machine.

Chapter 6: Roadmap to Upgrade ElectricCommander

To upgrade to ElectricCommander 5.x:



Upgrade		
From	To	Go to
ElectricCommander 4.2.x	ElectricCommander 5.x	Upgrading from ElectricCommander 4.2.x to ElectricCommander 5.x on page 7-1
ElectricCommander 5.0.x without clustering	ElectricCommander 5.x without clustering	Upgrading from ElectricCommander 5.0.x to ElectricCommander 5.x Without Clustering on page 9-1
ElectricCommander 5.0.x with clustering	ElectricCommander 5.x with clustering	Upgrading from ElectricCommander 5.0.x to ElectricCommander 5.x With Clustering on page 8-1
ElectricCommander 5.0.x with clustering	ElectricCommander 5.x without clustering	Upgrading from ElectricCommander 5.0.x to ElectricCommander 5.x Without Clustering on page 9-1
ElectricCommander 5.0.x without clustering	ElectricCommander 5.x with clustering	Upgrading from ElectricCommander 5.0.x to ElectricCommander 5.x With Clustering on page 8-1

IMPORTANT: If your system is running a version earlier than ElectricCommander 4.2, you first need to upgrade to ElectricCommander 4.2.x. You should be able to safely upgrade from ElectricCommander 3.10.x, 4.0.x, or 4.1.x to ElectricCommander 4.2.x, and then upgrade to ElectricCommander 5.x.

IMPORTANT: If you are installing new hardware when performing the upgrade from ElectricCommander 4.2.x to ElectricCommander 5.x, be aware that you need to do an additional step for the background upgrader.

When performing an upgrade on existing hardware, you run the background upgrader against your source database, and then run the installer to install the latest version. Running the installer will run the background upgrader with the `--final` flag, ensuring that your database is finalized for the upgrade. However, if you install the software on a new server (specifically, the new server that is not pointing at the newly upgraded database), the `--final` flag will not run automatically for the background upgrader, and you will need to run it manually.

You need to run the background upgrader one more time with `--final` flag as an option in this command:

```
java -jar commander-upgrade-jar-with-dependencies.jar --final
```

This will mark the database as being finalized and allow ElectricCommander to work properly.

Your order of upgrade operations might be as follows:

1. Run the background upgrader against the ElectricCommander 4.2.x database.
2. Install the ElectricCommander 5.x software on the new server.
3. Shut down the Commander 4.2.x server, and run the background upgrade script with the `--final` flag.
4. Point the new ElectricCommander 5.x server at the new (target) database, either through the UI or using the `ectool setDatabaseConfiguration` command.

If you are running the standard installation on an existing Commander environment, you do not need to manually run with the `--final` flag, because it is run when you execute the installer to upgrade your existing Commander server to the new version.

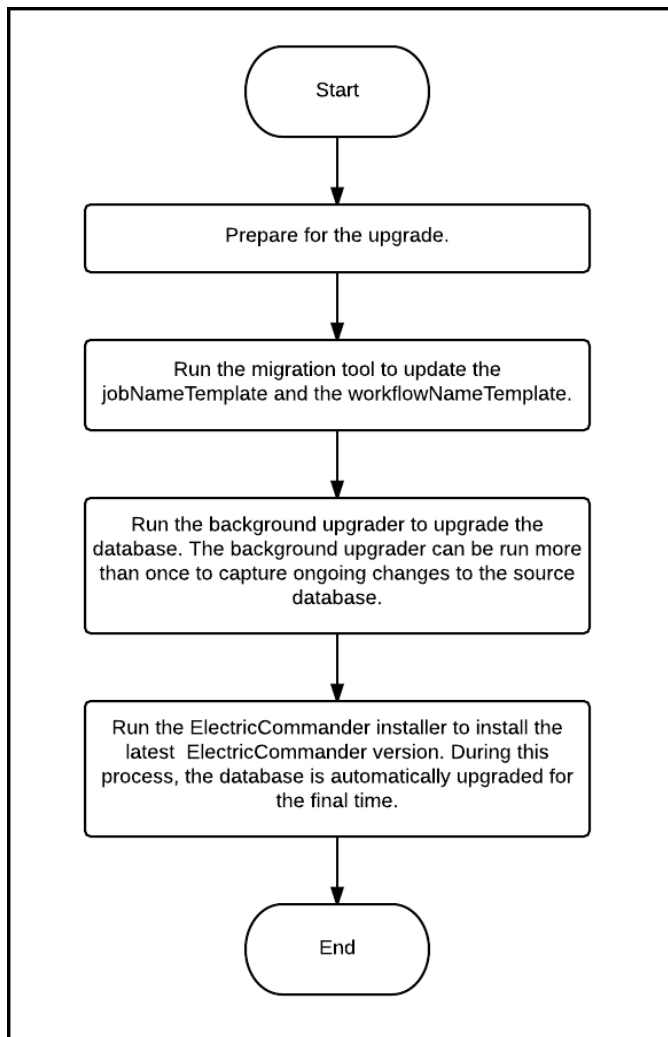
Chapter 7: Upgrading from ElectricCommander 4.2.x to ElectricCommander 5.x

Upgrading from ElectricCommander 4.2.x to ElectricCommander 5x or later is more complex and extensive than for previous releases.

- The upgrade requires a database upgrade that is more extensive than in previous releases.
- The upgrade process has changed. The database is modified to use universally unique identifiers (UUIDs) for entity IDs.

The upgrade process includes using a *migration tool* to add support for UUIDs, an *upgrade tool* to upgrade the database, and the ElectricCommander installer.

- Running the ElectricCommander installer is the last step in the upgrade process. The installer works the same way as the installers in previous releases. The `ElectricCommander-<x.x.x.xxxxx>` installer, where `x.x.x.xxxxx` is the software version, collects the ElectricCommander service account credentials, uninstalls the current release, installs the latest ElectricCommander release, configures the system with all property values mined, and restores custom files and data.



The upgrade process also includes these tasks:

- You must have a source (original) database and a target (Commander 5.x) database in the same database instance.
- Before running the Commander 5.x installer, you can prepopulate your target (Commander 5.x) database by using the *background upgrader*, which can replicate and upgrade data from your database while the ElectricCommander 4.2.x server is still running.

While this can be a slow process, if your database is large, it reduces the downtime between turning off your Commander 4.2.x server and starting your Commander 5.x server by copying most of the data across while the Commander 4.2.x server is still running.

If a large number of jobs have run against (or other changes have been made to) the source (original) database between the last pass of the background upgrader and running the ElectricCommander installer, this will increase the downtime required to complete the upgrade process. This downtime can be minimized by rerunning the background upgrader again to incrementally replicate recent changes from the source (original) database to the target database shortly before running the ElectricCommander installer.

- You can run the background upgrader more than once to ensure that the source (original) and target (Commander 5.x) databases remain synchronized.
- The database upgrade can be slow. To estimate the time that the upgrade may take, see [Estimated Durations for Database Upgrades](#) on page 7-10

IMPORTANT: Change tracking is enabled when you upgrade to ElectricCommander 5.3 and later. This can significantly increase the time to complete the upgrade. If you want to upgrade without change tracking, add this line to the database.properties file before starting the upgrade:

```
COMMANDER_DB_AUDITING_ENABLED=false
```

IMPORTANT: If you are installing new hardware when performing the upgrade from ElectricCommander 4.2.x to ElectricCommander 5.x, be aware that you need to do an additional step for the background upgrader.

When performing an upgrade on existing hardware, you run the background upgrader against your source database, and then run the installer to install the latest version. Running the installer will run the background upgrader with the `--final` flag, ensuring that your database is finalized for the upgrade. However, if you install the software on a new server (specifically, the new server that is not pointing at the newly upgraded database), the `--final` flag will not run automatically for the background upgrader, and you will need to run it manually.

You need to run the background upgrader one more time with `--final` flag as an option in this command:

```
java -jar commander-upgrade-jar-with-dependencies.jar --final
```

This will mark the database as being finalized and allow ElectricCommander to work properly.

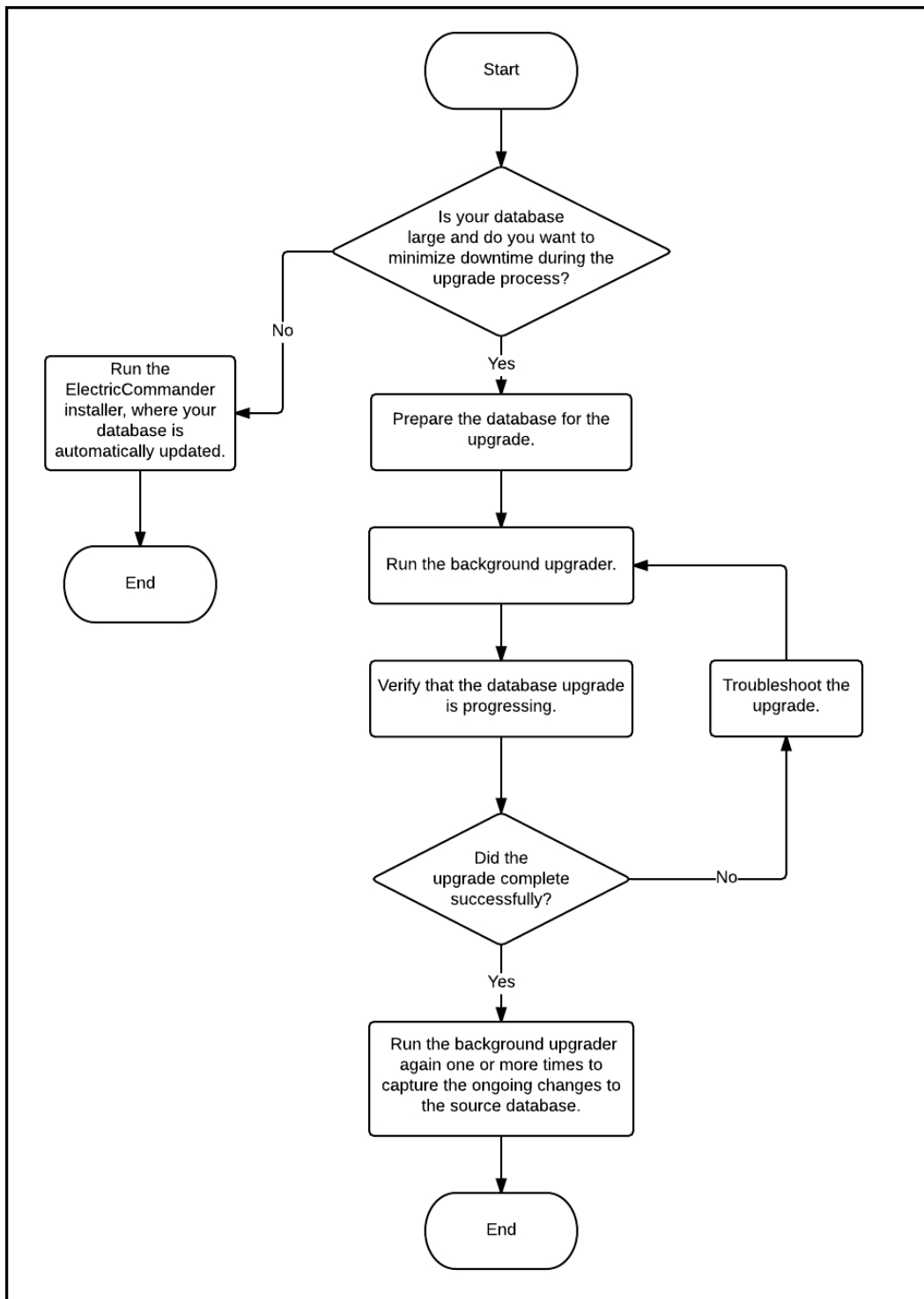
Your order of upgrade operations might be as follows:

1. Run the background upgrader against the ElectricCommander 4.2.x database.
2. Install the ElectricCommander 5.x software on the new server.
3. Shut down the Commander 4.2.x server, and run the background upgrade script with the `--final` flag.
4. Point the new ElectricCommander 5.x server at the new (target) database, either through the UI or using the `ectool setDatabaseConfiguration` command.

If you are running the standard installation on an existing Commander environment, you do not need to manually run with the `--final` flag, because it is run when you execute the installer to upgrade your existing Commander server to the new version.

Upgrade Process

IMPORTANT: If your system is running a version earlier than ElectricCommander 4.2, you first need to upgrade to ElectricCommander 4.2.x. You should be able to safely upgrade from ElectricCommander 3.10.x, 4.0.x, or 4.1.x to ElectricCommander 4.2.x, and then upgrade to ElectricCommander 5.x.



You can upgrade from ElectricCommander 4.2.x to ElectricCommander 5.x or later using one of these methods:

- If your database is large and you want to reduce downtime when the ElectricCommander software is upgraded, follow these steps to perform the bulk of the database upgrade while your Commander 4.2.x

server is still running.

1. Prepare for the upgrade.

Go to [Preparing for the Upgrade](#) on page 7-13 for more information.

2. Download and extract the upgrade tool.

Go to [Upgrade Tool on page 7-14](#) for more information.

3. Run the *migration tool* to update the `jobNameTemplate` and the `workflowNameTemplate`.

The migration tool may not convert every instance of `jobId` to `jobCounter` and of `workflowId` to `workflowCounter`. You need to review your Commander configuration to verify that all the jobIds and workflowIds are properly converted.

Go to [Migration Tool on page 7-15](#) for how to use the migration tool.

4. Upgrade the database with the background upgrader.

Go to [Database Upgrade Process on page 7-19](#) for how to prepare, run, and troubleshoot the database upgrade.

5. Before running the ElectricCommander installer:

- a. Make sure that the database upgrade is as complete as possible.
- b. Shut down the Commander 4.2.x server.

IMPORTANT: The Commander 4.2.x server must be shut down.

6. Run the ElectricCommander Installer.

During the installation, the background upgrader automatically runs automatically to complete the database scheme upgrade.

Select one of these methods:

- [User Interface Upgrade Method on page 7-31](#)
- [Interactive Command-line Upgrade Method on page 7-34](#)
- [Silent \(Unattended\) Upgrade Method on page 7-35](#)

7. (Recommended) Fully test the upgrade to verify that it was successful completed.

8. (Recommended) After verifying that the upgrade was successfully completed, archive and remove the source (original) database.

9. (Optional) Copy the contents of an existing repository server into a newly installed repository server.

Go to [Copying Repository Contents on page 7-36](#) for instructions.

10. Restart the Commander 5.x server.

11. Connect the Commander 5.x server to the target (Commander 5.x) database.

- If your database is small or downtime is not an issue for you, follow these steps:
 1. Prepare for the upgrade.

Go to [Preparing for the Upgrade on page 7-13](#) for more information.
 2. Download and extract the upgrade tool.

Go to [Upgrade Tool on page 7-14](#) for more information.
 3. Create the target (Commander 5.x) database.

Go to [Database Upgrade Process](#) on page 7-19 for more information.
 4. Run the *migration tool* to update the `jobNameTemplate` and the `workflowNameTemplate` in the source (original) database.

The migration tool may not convert every instance of `jobId` to `jobCounter` and of `workflowId` to `workflowCounter`. You need to review your Commander configuration to verify that all the jobIds and workflowIds are properly converted.

Go to [Migration Tool on page 7-15](#) for how to use the migration tool.
 5. Before running the ElectricCommander installer, shut down the Commander server.
 6. Run the ElectricCommander installer to install Commander 5.x.
 7. Restart the Commander 5.x server.
 8. Connect the Commander 5.x server to the target (Commander 5.x) database.

IMPORTANT: You cannot upgrade the ElectricCommander 4.2.x H2 database. If you want to continue using a built-in database in ElectricCommander 5.x, follow the procedure in [Using the ElectricCommander 4.2.x Built-In Database in ElectricCommander 5.x](#) on page 7-23.

IMPORTANT: You must ensure that the database has sufficient disk space for the upgrade, which includes space for the data copied during the upgrade in addition to space for temporary usage during the upgrade for log files, redo files, and undo files. We recommend that you have at least three times (3x) the size of your current database available during the upgrade process.

IMPORTANT: Change tracking is enabled when you upgrade to ElectricCommander 5.3 and later. This can significantly increase the time to complete the upgrade. If you want to upgrade without change tracking, add this line to the `database.properties` file before starting the upgrade:

```
COMMANDER_DB_AUDITING_ENABLED=false
```

Properties Changed to UUIDs

Starting in Commander 5.0, entity types in ElectricCommander changed. ElectricCommander now uses a universally unique identifier (UUIDs) for all entity types. The most common entity changes that affect you are the `jobId` and `workflowId` properties. In earlier releases, these properties were monotonically incrementing integers that were created directly by the database. The new auto-generated UUID is displayed as an 36-character string. This UUID is stored internally in the database as a 128-bit binary type.

For example:

- A new job in Commander 4.2.x may show 1809 as a `jobId`, but launching this same job in Commander 5.x may result in a `jobId` like this: 4fa765dd-73f1-11e3-b67e-b0a420524153.

- When you migrate a `jobId` from Commander 4.2.x to Commander 5.x, ElectricCommander converts the `jobId` to hexadecimal and adds zeroes before creating the 36-character string.

If you migrate a Commander 4.2.x job with the `jobId` 42978, the `jobId` becomes 00000000-0000-0000-0000-00000000e7e2 in Commander 5.x.

You most commonly reference `jobId` and `workflowId` in the Job name template and Workflow name template fields in these locations:

- Administration > Server > Settings
- Edit procedure or Edit workflow definition

Upgrading to ElectricCommander 5.x

You use an upgrade tool to upgrade from ElectricCommander 4.2.x to ElectricCommander 5.x. The upgrade tool has a migration tool that automates and eases the transition to `jobCounter` and `workflowCounter`. Go to [Upgrade Tool](#) on page 7-14 and [Migration Tool](#) on page 7-15 for more information.

You can then use the new Commander 5.x project-level job counters to count and identify individual jobs.

Reasons for the Change

Commander 5.x changed IDs to UUIDs for several reasons:

- Commander 5.x allows you to configure a cluster of Commander servers, which adds horizontal scalability and high availability to your Commander implementation. Using UUIDs for all primary keys enables multiple Commander servers to run against the same database without having to coordinate primary key generation across the servers or through the database, which could cause a performance bottleneck.
- In Commander 4.2.x, the database generated IDs directly. This would create confusion if you moved from one database to another, such as moving from MySQL to Oracle, which would reset the IDs after you uploaded the new database. Using custom counters avoids this ID reset problem and is convenient when moving to a production database such as MS SQL Server or Oracle.

Impact for Commander 4.2.x Users

Dependence on jobId or workflowId

Many customer implementations have relied on job and workflow names that depend on `$(jobId)` or `$(workflowId)`. Although jobs and workflows with these system-generated identifiers do not fail after upgrading to Commander 5.x, the readability of this information will become more difficult for users because previously sequential numbers are replaced by 36-character random hexadecimal UUIDs.

Order of Preference for Name Template Values

Whenever a job is run, Commander names the job according to the job name template value in these locations in descending order of preference:

1. The procedure
2. The property sheet in Administration > Server > Settings
3. The server's hard-coded default

Job Name Templates

Upgrading to Commander 5.x automatically updates Commander hard-coded default job and workflow name templates. However, you may not see this change in your job or workflow names.

Commander 5.x *does not automatically update* your name templates if you have ever saved your Commander server settings for the template default names. It does not matter if you made any changes; the only action that matters is saving the server settings. Saving the server settings creates a custom property sheet, even though it might contain all default values. The creation of the property sheet means job and workflow names are derived from the property sheet's name templates and not the hard-coded name templates. The hard-coded name templates were updated in Commander 5.x or later, but because your job and workflow names are not derived from hard-coded name templates, their names are not updated automatically the next time you run a job or workflow. Similarly, your name templates remain as they were in Commander 4.2.x if they are located in an individual procedure. You need to run the migration tool to update occurrences of `$(jobId)` or `$(workflowId)`.

Commander 5.x *automatically updates* your name templates if you have never saved your Commander server settings. Because you have never saved the server settings, a custom property sheet does not exist. This means that Commander uses the default hard-coded name templates, which were updated in Commander 5.x or later. Your job or workflow names automatically use the new default name templates the next time you run a job or workflow.

To determine which name templates Commander is using, follow these steps:

1. In the Commander web interface go to Administration > Server.
2. Check if more than one property is listed in the Custom Server Properties section.

If there are multiple properties listed, you are using a property sheet's templates, not hard-coded templates.

Default Job Name Template Impact

The Commander 4.2.x default job name template contains the `jobId`, as seen in:

```
job_$/myJob/jobId]_$/timestamp]
```

If you are using a property sheet's default job name template and you leave the default job name template as-is, job names for new jobs on the ElectricCommander 5.x server will display with a UUID entry instead of the sequential counter.

If you are using the hard-coded default job name template and you leave the default job name template as-is, your job names will have a job counter instead of the job ID number after you upgrade.

Example 1

You are using the property sheet's default job name template in Commander 4.2.x. This means you do not have a procedure-level job name template. You do not change anything after upgrading to Commander 5.x.

Assume you have previously run 2500 jobs. In Commander 4.2.x, your next job name might be

```
job_2501_201401021303
```

In Commander 5.x that job name would now look like

```
job_4fa765dd-73f1-11e3-b67e-b0a420524153_201401021303
```

Example 2

You are using the hard-coded default job name template in Commander 4.2.x. This means you do not have a custom property sheet nor a procedure-level job name template. You do not change anything after upgrading to Commander 5.x.

Assume you have previously run 2500 jobs. In Commander 4.2.x, your next job name might be

```
job_2501_201401021303
```

In Commander 5.x that job name would now look like

```
Default_2501_20140102130343
```

Customized Job Name Template Impact

If you customized your job name template but still used the `jobId`, the UUID now appears instead of the previous `jobId`. This is also true if you used `jobId` in a procedure-level job name template.

We strongly recommend running the migration tool.

Workflow Name Template Impact

The Commander 4.2.x default workflow name template contains the `workflowId`, as shown in:

```
workflow_${/myWorkflow/workflowId}_${/timestamp}
```

The impact to your workflow names would be the same as what is shown in the job name example.

Actions to Take

Note: This section references `jobId` to `jobCounter` changes, but also consider the same for `workflowId` to `workflowCounter` changes.

- We strongly recommend that you convert your naming templates to reference an incrementing job or workflow counter that is tied to the project. Go to the next section ([Migration Tool in the Upgrade Tool](#) on page 7-9) for a tool to help you with the conversion.

Examples of naming templates containing properties that increment automatically:

```
`${/projectName}_${/increment /myproject/jobCounter}_${/timestamp yyyMMddHHmmss}`
`${/projectName}_${/increment /myproject/workflowCounter}_${/timestamp yyyMMddHHmmss}`
```

- If you have any scripts that refer to object IDs and assume that they are numeric, you must update these scripts to handle the IDs as strings.

The migration tool is not able to make the same changes to any locations in your scripts or step commands that reference `jobId` or `workflowId`. You need to manually make these changes to `jobCounter` and `workflowCounter`.

For example, a Perl script with `my $jobId = ${/jobId};` should be modified by quoting the job ID as follows: `my $jobId = '${/jobId}';`. This should be done to accommodate the change from a numeric value to an alphanumeric value, the UUID.

Migration Tool in the Upgrade Tool

Electric Cloud provides the migration tool to ease the transition from `jobId` and `jobCounter`.

This script does the following:

- Examines every procedure in your system and makes the following changes to any `jobNameTemplate` it finds
 - `${/jobId}` becomes `${/increment /myProject/jobCounter}` or `${/increment /server/ec_counters/jobCounter}`
 - Within `${/javascript myJob.jobId}`, `myJob.jobId` becomes `getProperty('/increment /myJob/project/jobCounter')` or `getProperty('/increment /server/ec_counters/jobCounter')`

- Examines every procedure in your system and makes the following changes to any workflowNameTemplate it finds
 - ``${workflowId}`` becomes ``${/increment /myProject/workflowCounter}`` or ``${/increment /server/ec_counters/workflowCounter}``
 - Within ``${/javascript myWorkflow.workflowbId}``, `myWorkflow.workflowbId` becomes `getProperty('/increment /myWorkflow/project/workflowCounter')` or `getProperty('/increment /server/ec_counters/workflowCounter')`
- Determines the current value of the global job ID counter by querying the most recent job. This is used as the starting value for a jobCounter property (you can override this value).

By default, the migration tool runs in *dry run* mode. Dry run mode lists the changes the script would make (showing the *before* and *after* values). You must run the script with an explicit option to actually make the changes. [Migration Tool](#) on page 7-15 contains detailed instructions and explanations. Be sure to read that information before you run the migration tool. [jobCounter Considerations on page 7-17](#) contains important information for defining a starting jobCounter value.

You can run this script either before or after you upgrade to Commander 5.x.

Rather than waiting to upgrade to Commander 5.x before using jobCounters, we recommend that you begin exploring the use of jobCounters in your existing production environment. Changing your existing projects to reference a counter rather than `jobId` can be implemented at any time and allows you to start recording your project-level job-count metrics. Applying these changes to your system before upgrading to Commander 5.x allows you to make these changes at your own pace and avoid concerns being raised by users encountering UUIDs in their job/workflow names after the upgrade.

Default Job Name and Workflow Name Template Changes

Users who upgraded from Commander 4.2.x may notice that Commander updated the hard-coded job name and workflow name templates for Commander 5.x. They no longer contain `jobId` or `workflowId`.

The new default name templates are:

```
`${projectName}_${/increment /myproject/jobCounter}_${/timestamp yyyMMddHHmmss}`  
`${projectName}_${/increment /myproject/workflowCounter}_${/timestamp yyyMMddHHmmss}`
```

Changing from a system `jobId` to a project-based `jobCounter` allows you to recognize the number of jobs that each project launches. The previous default of using a system `jobId` provided only a single system-level metric.

Disadvantages of Using `jobId` and `workflowId`

We strongly recommend using `jobCounter` and `workflowCounter` instead of `jobId` and `workflowId`. These are a few reasons to use `jobCounter` and `workflowCounter` instead of `jobId` and `workflowId`:

- The `jobId` and `workflowId` are no longer human-readable integers so they do not provide any identifiable information and cannot be used as counters.
- Because of the increased length of `jobId` and `workflowId`, you might encounter limitations such as maximum Windows path name length, maximum URL length, and maximum command line length.
- The default job name and workflow name templates no longer contain `jobId` and `workflowId`.

Estimated Durations for Database Upgrades

The database upgrade process can be time-consuming. To minimize the duration of this process, the upgrade tool has a background upgrader to allow your ElectricCommander system to build the Commander 5.x database while the Commander 4.2.x server is still running. You can run the background upgrader before you

run the installer, which can minimize downtime for the ElectricCommander environment and reduce the time required for the upgrade process.

It may take many hours to several days to upgrade the database, depending on the database hardware, database size, and the load on your database server. In our internal testing on moderately powerful database servers, database upgrades in the size range of 10 GB to several tens of GB usually took many hours on MySQL or Microsoft SQL Server, and several days on Oracle.

If you have a large database, you can run the background upgrader more than once to synchronize the source (original) and target (Commander 5.x) databases while the Commander 4.2.x server is still running.

To estimate the duration of the next database upgrade pass:

1. Find the log file called upgrade.log.

When you run the background upgrader, the log file is in a subdirectory called *logs* of the current working directory from which the background upgrader was run.

2. In the log file, search for the phrase "End of pass <number> through tables."
3. Calculate the duration of a database upgrade pass.

Look for the timestamp when the current upgrade ended and the timestamp when the previous upgrade ended.

The duration is the difference between the time when the last pass ended and the time when the previous pass ended.

The downtime for upgrading Commander 5.x is expected to take at least as long as the last pass, because it runs one more pass of the background upgrader. On MySQL, it can take many hours more than that, because MySQL requires entire tables to be copied when new columns are added to them.

IMPORTANT: Change tracking is enabled when you upgrade to ElectricCommander 5.3 and later. This can significantly increase the time to complete the upgrade. If you want to upgrade without change tracking, add this line to the database.properties file before starting the upgrade:

```
COMMANDER_DB_AUDITING_ENABLED=false
```

Choosing the Correct Installation Method

This topic has the various installation upgrade methods and options for specific platform configurations. For information about supported server platforms and supported non-server platforms, see [Server Platforms on page 2-1](#) and [Agent Platforms on page 2-4](#).

User Interface Installation

This method provides a wizard for upgrading the Commander software on a supported server platform. This upgrade method is generally preferred by Windows users, but is supported on Linux platforms with the X Window System installed.

Upgrade options:

- **Upgrade Existing Installation**

This option uninstalls the current release, installs the latest Commander release, collects the Commander service account credentials, configures the system with all property values, and restores custom files and data.

- **Clean Install**

This option allows you to specify a different installation directory for the new version. The files from your previous Commander version will not be removed or modified and will remain in their original directories.

Interactive Command-Line Installation

This method provides an interactive command-line for upgrading the Commander software on a supported server platform. This upgrade method is available only for Linux platforms.

Upgrade options:

- **Upgrade Existing Installation**

This option uninstalls the current release, installs the latest Commander release, collects the Commander service account credentials, configures the system with all property values, and restores custom files and data.

- **Clean Install**

This option allows you to specify a different installation directory for the new version. The files from your previous Commander version are not be removed or modified and remain in their original directories.

Silent Unattended Installation

This method provides a non-interactive command-line upgrade for supported server platforms. You might prefer this installation method when upgrading multiple remote agents and servers.

Upgrade options:

- **Upgrade Existing Installation**

This option uninstalls the current release, installs the latest Commander release, collects the Commander service account credentials, configures the system with all property values, and restores custom files and data.

IMPORTANT: You cannot add a new repository server with this upgrade method.

Repository Server With a Commander Upgrade

After Commander 4.x, the only way to install a repository server on the same machine as other services is to uninstall and reinstall Commander. You can install the repository server on a different machine to avoid uninstalling and reinstalling Commander.

Stand-Alone Repository Server or Web Server Upgrade

You cannot directly upgrade a stand-alone repository server or stand-alone web server. You must uninstall and then reinstall these servers using the Commander installer. The uninstall and reinstall process is required to install an agent on the server machine. Starting with Commander 4.2, an agent is required on the machine with the stand-alone repository server or web server.

For more information, see [Uninstalling ElectricCommander on page 10-1](#), [Installing ElectricCommander on page 3-1](#), and [Copying Repository Contents on page 7-36](#).

Note: Commander 5.x web servers must use a Commander 5.x agent to retrieve log files to be displayed in the web interface. This *workspace proxy* agent is typically set to *localhost* but can be modified with the config.php file.

Non-Server Platform Agent Upgrade

You cannot directly upgrade a non-server platform agent. You must uninstall and then reinstall these machines using the Commander installer. For more information, see [Uninstalling ElectricCommander](#) on page 10-1 and [Non-Server Platform Agent Installation Method](#) on page 3-34.

Preparing for the Upgrade

Review the following information before you upgrade the ElectricCommander software.

Topics:

- [Upgrade Testing](#) on page 7-13
- [Backups](#) on page 7-13
- [Upgrade Installer Preservation](#) on page 7-14
- [MySQL Upgrades](#) on page 7-14

Upgrade Testing

In most implementations, ElectricCommander is being used in an environment that affects many users. We recommend that you test your upgrade on a separate test server to understand all aspects of the upgrade process. This minimizes the potential impacts to downstream users.

We also strongly recommend that you make a copy of your database and walk through the upgrade process on a test environment before performing the actual upgrade on your production environment. The result of this is to understand the impact of performing the upgrade in your environment and to understand the effect that the upgrade has on your environment. If you need a test license to perform such a test, contact support@electric-cloud.com to request a temporary license.

Backups

If you are upgrading a Commander server, it is *extremely* important that you back up your existing Commander data *before* upgrading. For more information about backups, see [Commander Server Backups](#) on page 11-1.

- Always back up the Plugins Directory. The default location is the `plugins` subdirectory within the data directory.
- Always back up the files that contain your configuration and custom settings. To ensure all important settings are saved, back up the following subdirectories in the data directory:
 - Commander server and agent configuration files in the `conf` subdirectory
 - The `passkey` file in one of these directories:
`/opt/electriccloud/electriccommander/conf/passkey` in Linux or
`C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows
 - Apache web server configuration files in the `apache/conf` subdirectory
 - The local MySQL database configuration file, `my.ini`, in the `mysql` subdirectory (if applicable)
- Always back up any other files where you have created custom configurations, specified other custom information, or created any type of modification.

Note: The number of Commander files you may have modified is too numerous to list, so you may want to back up the entire Commander data directory and other miscellaneous files that may have changed.

- If you use an artifact repository, back up your Commander repository configuration files in the `conf/repository` subdirectory.
- Determine if any changes were made to the custom editor or preflight driver script properties (installed by default). Back up those files if changes were made.

These properties are stored in the server-level property sheet, which can be viewed in the web UI by accessing the Administration tab/ Server subtab.

Custom editors are stored in the nested sheet named *ec_customEditors*. Preflight driver scripts are stored in the nested sheet named *ec_preflight*. The upgrade process overwrites default custom editor and preflight driver scripts with current versions. We recommend backing up any custom properties you created by renaming the property. For example, change `ec_preflight/clientDrivers/perforce` to `ec_preflight/clientDrivers/perforce_modified`.

Upgrade Installer Preservation

After you back up your ElectricCommander server, create a folder where you can download the `ElectricCommander-x.x.x.xxxxx` installation file.

MySQL Upgrades

Even if you use the background upgrader to prepopulate the target (Commander 5.x) database, the MySQL server database is upgraded during the Commander upgrade process. This may take many hours to complete if you have a significant database.

IMPORTANT: After starting the installer to upgrade from ElectricCommander 4.2.x to ElectricCommander 5.x, do not interrupt the upgrade process.

- If the upgrade process is interrupted, you might corrupt your database. You have to restore your database from a previous backup.

The following applies when the database is updated:

- Upgrading saves the existing local MySQL database, if applicable.
- Use the `ectool` to view the upgrade progress. On a command line, enter `ectool getServerStatus`.

External MySQL Database Characteristics

When upgrading Commander, ensure that `default-storage-engine=INNODB` is in the `[mysqld]` section in `my.cnf/my.ini`.

An install and upgrade log file named `installer.log` is created in the `logs` subdirectory in the data directory.

Upgrade Tool

You use the `upgrade-bundle.zip` to upgrade ElectricCommander from Commander 4.2.x to Commander 5.x.

The `upgrade-bundle.zip` has these files:

- Background upgrader to update the database—`commander-upgrade-jar-with-dependencies.jar`
You can run the background upgrader before you install the latest ElectricCommander software.
- ElectricCommander installer to install the latest Commander software—`ElectricCommander-x.x.x.xxxxx`.

The Windows upgrade tool for Commander 5.x has an installer called *ElectricCommander-x.x.x.xxxxx.exe*. The corresponding Linux installer is *ElectricCommander-x.x.x.xxxxx*.

- Migration tool to convert `$(jobId)` and `$(workflowId)` to `jobCounter` and `workflowCounter` in the `jobNameTemplate` and `workflowNameTemplate` properties—`migrateTemplates.pl`

For more information, see [Migration Tool on page 7-15](#).

- `README.txt` - README file for Windows and Linux

IMPORTANT: Only use the background upgrader when upgrading from ElectricCommander 4.2.x to ElectricCommander 5.x. Do not use it when you are updating ElectricCommander 5.0.x to a later ElectricCommander version.

Downloading and Extracting the Upgrade Tool

After downloading `upgrade-bundle.zip` from the Electric Cloud site, extract these files:

- Background upgrader: `commander-upgrade-jar-with-dependencies.jar` to the `utils` subdirectory.
- ElectricCommander installer: `ElectricCommander-x.x.x.xxxxx` to the `src` subdirectory.
- Migration tool for UUIDs: `migrateTemplates.pl` to the `bin` subdirectory.

Troubleshooting the Upgrade Tool

If the upgrade process fails, check `upgrade.log` and examine the file for potential reasons for the failure.

- Look at the status entries and error messages in the log file.
- Look for incorrect, invalid, or missing values and update the `upgrade.properties` file with the correct values.
- Look for Java stack traces in the log file and on the screen of your workstation.
- Look for connection issues with the database server.
- If the installer lost its connection to the database server, check its physical and network connections to the database server and try to reconnect.
- If the `upgrade.properties` file has incorrect or missing information, an error occurs and the installation fails.
- Compare the `upgrade.properties` file with the current `database.properties` file to find incorrect or missing values.
- Look for properties that could affect communication with the database server.

After resolving all the issues, restart the upgrade process.

Migration Tool

The migration tool is an ec-perl script, `migrateTemplates.pl`, which converts `jobNameTemplate` and `workflowNameTemplate` properties containing `$(jobId)` and `$(workflowId)` to use a counter instead.

See [Properties Changed to UUIDs on page 7-6](#) for more information about the change to UUIDs that occurred in ElectricCommander 5.0 and later.

What the Migration Tool Does

This script does the following:

- Examines every procedure in your system and makes the following changes to any jobNameTemplate it finds
 - `$(jobId)` becomes `[/increment /myProject/jobCounter]` or `[/increment /server/ec_counters/jobCounter]`
 - Within `[/javascript myJob.jobId]`, `myJob.jobId` becomes `getProperty('/increment /myJob/project/jobCounter')` or `getProperty('/increment /server/ec_counters/jobCounter')`
- Examines every procedure in your system and makes the following changes to any workflowNameTemplate it finds
 - `$(workflowId)` becomes `[/increment /myProject/workflowCounter]` or `[/increment /server/ec_counters/workflowCounter]`
 - Within `[/javascript myWorkflow.workflowbId]`, `myWorkflow.workflowbId` becomes `getProperty('/increment /myWorkflow/project/workflowCounter')` or `getProperty('/increment /server/ec_counters/workflowCounter')`
- Determines the current value of the global job ID counter by querying the most recent job. This is used as the starting value for a jobCounter property (you can override this value).

Running the Migration Tool

By default, the migration tool runs in *dry run* mode. Dry run mode lists the changes the script would make (showing *before* and *after* values). Before you make the actual changes, make sure you examine the proposed changes thoroughly to avoid job name collision.

After you download the upgrade tool, upgrade-bundle.zip, extract migrateTemplates.pl and put it in the bin subdirectory.

To invoke the migration tool:

```
ec-perl migrateTemplates.pl [option] ...
```

The following options are available:

Option	Description
<code>--server <server></code>	Specifies the Commander server to migrate. Default: localhost.
<code>--dryRun <1 0></code>	1 shows prospective changes; 0 actually makes them. Default: 1.
<code>--project <project></code>	Limits migration to the specified project.
<code>--counters <server project></code>	Whether job counters are server-wide or project-specific. Default: project.
<code>--startingJobId <n></code>	Initializes the job counter to the specified number. Default: the current greatest job ID.

Example 1

You want a unique job counter across the entire Commander system and want the next job number to be 1001:

```
ec-perl migrateTemplates.pl --counters server --startingJobId 1000
```

The above example will change `$(jobId)` to `[/increment /server/jobCounter]` and initialize the job counter to 1000. This means the next job run on the server will be number 1001. The following actually makes the proposed changes:

```
ec-perl migrateTemplates.pl --counters server --startingJobId 1000 --dryRun 0
```

Example 2

You want a unique job counter for each project and want the counters to continue from the current greatest job ID.

```
ec-perl migrateTemplates.pl --counters project
```

The above example will change `$(jobId)` to `[/increment /myJob/project/jobCounter]` and use the current greatest job ID as the initial value for the job counter.

The following actually makes the proposed changes:

```
ec-perl migrateTemplates.pl --counters project --dryRun 0
```

jobCounter Considerations

When using the migration tool to define a starting jobCounter value, keep in mind the following scenarios and their respective actions:

My job names will keep the same general format, replacing jobId with a new jobCounter

You must set your starting counter at a number higher than the most recent job that was used for this project.

However, you might find it more useful to use a more recognizable base point to allow you to differentiate jobs that began after making this change

Example

The last job run under this project as job 32348.

Technically, setting your counter to “32349” would avoid any possible job name conflicts.

However, it would be probably be more effective to set your counter to “40000”, “100000” or “1000000” to help clarify that any new jobs were jobs that were launched after making this change.

My job names will use a completely different format than in the past and will rely on a new jobCounter

You might simply want to set your jobCounter to “1”.

My job names did not rely on jobId previously; they relied on timestamps for unique naming

No change is necessary, but you might want to consider using a jobCounter in the future to help capture the relative number of jobs occurring under each project.

My job names already use a project-based jobCounter property

No changes are necessary.

My job names already use a server-based jobCounter property

Although you do not have to change this, be aware that using a project-based property instead might provide slight performance improvements during large bursts of job launches.

Troubleshooting: upgrade.log File

Review the upgrade.log file when you want to do the following:

- Check the status of the database upgrade because it does not seem like it is doing anything.
- Determine why a database upgrade failed.

The upgrade.log file is located at `../logs/upgrade.log`, where `..` represents your current working directory for running the database upgrade tool.

When the background upgrader is run by the ElectricCommander installer, the current working directory will not be a subdirectory of the temp directory.

Using the upgrade.properties File Templates

1. Copy the template content corresponding to your target (ElectricCommander 5.x) database type.
2. Open a new empty file in a text editor.
3. Paste the entire template contents into the new file.
4. Replace the template values (after the = symbols) with the values from your existing Commander deployment.

Except for the `COMMANDER_DB_PASSWORD`, `COMMANDER_DB_BATCH_SIZE`, `COMMANDER_DB_NAME` (MySQL or Microsoft SQL Server), and `COMMANDER_DB_USER` (Oracle) values, use your existing database value in one of these these locations:

- Administration > Database Configuration tab in your existing Commander server
- `<install_dir>/conf/database.properties` file in your existing Commander deployment

Make sure that you maintain the formatting of the values in the database.properties file.

5. Save the file as `upgrade.properties`.
6. Because the upgrade.properties file contains the plaintext password for your Commander database, restrict read access to the file, and make sure that it is deleted when you no longer need it.

MySQL

```
# Example configuration file for mysql
COMMANDER_DB_TYPE=mysql

# The ip or dns of the mysql server
COMMANDER_DB_HOST=myDbHost.example.com

# The mysql port (defaults to 3306)
COMMANDER_DB_PORT=3306

# The name of the target (Commander 5.x) database
COMMANDER_DB_NAME=commander50

# The mysql username
COMMANDER_DB_USER=mysqlUser

# The mysql password
COMMANDER_DB_PASSWORD=mySqlPassword

# The mysql batch size
COMMANDER_DB_BATCH_SIZE=100
```



```
# The existing 4.2 database.  
# Must be on the same server as the new  
# database and readable by the mysqlUser  
OLD_COMMANDER_DB_NAME=commander42
```

Microsoft SQL Server

```
# Example configuration file for sqlserver  
COMMANDER_DB_TYPE=sqlserver  
  
# The ip or dns of the sqlserver server  
COMMANDER_DB_HOST=myDbHost.example.com  
  
# The sqlserver port (defaults to 1433)  
COMMANDER_DB_PORT=1433  
  
# The name of the target (Commander 5.x) database  
COMMANDER_DB_NAME=commander50  
  
# The sqlserver username  
COMMANDER_DB_USER=sqlserverUser  
  
# The sqlserver password  
COMMANDER_DB_PASSWORD=sqlserverPassword  
  
# The sqlserver batch size  
COMMANDER_DB_BATCH_SIZE=100  
  
# The existing 4.2 database.  
# Must be on the same server as the new  
# database and readable by the sqlserverUser  
OLD_COMMANDER_DB_NAME=commander42
```

Oracle

```
# Example configuration file for oracle  
COMMANDER_DB_TYPE=oracle  
  
# The ip or dns of the TNS listener  
COMMANDER_DB_HOST=myDbHost.example.com  
  
# The TNS listener port (defaults to 1521)  
COMMANDER_DB_PORT=1521  
  
# The sid of database  
COMMANDER_DB_NAME=commanderSID  
  
# The target (Commander 5.x) username/database  
COMMANDER_DB_USER=commander50  
  
# The oracle password  
COMMANDER_DB_PASSWORD=oraclePassword  
  
# The oracle batch size  
COMMANDER_DB_BATCH_SIZE=100  
  
# The existing 4.2 user/database.  
# Must be on the same server as the new  
# database and readable by the target (Commander 5.x) user  
OLD_COMMANDER_DB_USER=commander42
```

Database Upgrade Process

When you upgrade from Commander 4.2.x to Commander 5.x, the database must be upgraded. The database upgrade process includes creating a new database for Commander 5.x and copying the data from the Commander 4.2.x database to the new database.

This process is more involved than in previous releases because the database changes are more extensive than in previous ElectricCommander upgrades.

IMPORTANT: You cannot upgrade the ElectricCommander 4.2.x H2 database. If you want to continue using a built-in database in ElectricCommander 5.x, follow the procedure in [Using the ElectricCommander 4.2.x Built-In Database in ElectricCommander 5.x](#) on page 7-23.

IMPORTANT: After you have run the Commander 5.x upgrade, you cannot rerun the background upgrader against the database because Commander 5.x makes further database upgrades.

Before the Database Upgrade

What You Need to Do

Download the upgrade tool (upgrade-bundle.zip) from the Electric Cloud site and extract the upgrade tool.

Go to these topics and review the information in them:

- [Upgrade Tool on page 7-14](#)
- [Downloading and Extracting the Upgrade Tool on page 7-15](#)
- [Upgrade Tool Overview on page 7-22](#)
- [Upgrade Tool Overview on page 7-22](#)

What Your DBA Needs to Do

Your DBA should review each of the following recommendations to determine which of these are necessary for your environment:

How long the database upgrade may take

- The amount of time that it takes to upgrade the database depends on the database hardware, database size, and the load on database server.
- For large databases, the upgrade tool could run many hours, or even several days. The upgrade tool might put significant extra load on your database server while it runs in the background.
- For an estimate of how long the upgrade may take, go to [Estimated Durations for Database Upgrades on page 7-10](#).

Source (original) and target (ElectricCommander 5.x) databases

- Before you start the database upgrade, you must create a new, empty database that is the same type as the existing source (original) database and is the target (ElectricCommander 5.x) database for the upgrade.
- Your source (original) and target (ElectricCommander 5.x) databases must be on the same database server.
- The text encoding settings of the source (original) and target (ElectricCommander 5.x) databases must match exactly.

Disk space for the database upgrade

- You need to make sure that the database server has sufficient disk space for the database upgrade, including disk space for the data copied during the upgrade, log files, redo files, and undo files. The log space depends on the database type and the logging strategy.

- We recommend that you have at least three times (3x) the size of your current Commander 4.2.x source (original) database.
- Some of this space is needed temporarily for the upgrade process to run. After the upgrade process is completed, the disk space for the new target (ElectricCommander 5.x) database is only a little larger than before.

Deleting old data

IMPORTANT: If your Commander 4.2 .x source (original) database contains a lot of old job data, we *strongly* recommend that you delete as much of this data as possible before running the background upgrader, which will help to speed up the eventual upgrade. It will take time for this deletion effort to complete and for the background upgrader to run.

Starting the database upgrade

- Once you start the database upgrade process, ensure that the old Commander server is not shut down. This avoids a situation where the upgrade tool can fail to detect source database changes when trying to keep it synchronized with the target (ElectricCommander 5.x) database.
- See [Troubleshooting: Upgrade Tool Can Fail to Detect Database Changes](#) on page 7-30 for more details about the circumstances that can trigger this condition.

MySQL databases only

- You must give the database user permissions to create and delete tables at all times.
- You must also give the user DROP SCHEMA and CREATE SCHEMA permissions to create and delete databases during the duration of the upgrade process.

Oracle databases only

- Oracle refers to a database instance as a *User*.
- You must have a new user for the new empty target (ElectricCommander 5.x) database tables.
- We recommend putting the new user tables in their own workspace.
- Ensure that you have allocated adequate table space, temp space, and undo space.
- You must explicitly grant read access permissions to the target (ElectricCommander 5.x) database to read information in the source (original) database.

For example, you can enter the following SQL commands:

```
GRANT "CONNECT" TO "TESTART"
```

```
GRANT "RESOURCE" TO "TESTDRT"
```

Multiple commands such as `GRANT SELECT ON "TESTART"."EC_AGENT" TO "TESTDRT"`

- The target (Commander 5.x) database user must have the same password as the source (original) database user.
- Ensure that the database UNDO_RETENTION parameter is set large enough (at least 6 to 12 hours).

Running the Database Upgrade

To upgrade the database:

1. Create or edit an upgrade.properties file one of these ways:
 - a. Edit the file manually.
 - b. Use the wizard to help you edit the file.
 - c. Create a new upgrade.properties file with the upgrade.properties file template. Go to [Using the upgrade.properties File Templates](#) on page 7-18 for more information.

The upgrade.properties file provides information to the database upgrade process about the target (ElectricCommander 5.x) and source (original) databases.

In Linux, the file is in

```
/opt/electriccloud/electriccommander/conf/database.properties
```

In Windows, the file is at

```
C:\Program Files\Electric Cloud\ElectricCommander\conf\database.properties
```

2. Run the background upgrader.

Note: If you are using MySQL, make sure you have put `mysql-connector-java.jar` in the `$INSTALL_DIRECTORY/server/lib` directory before you run the background upgrader.

For example, in Linux, run:

```
/opt/electriccloud/electriccommander/bin/commander-upgrade <path to your upgrade .properties file>
```

The background upgrader begins copying your existing database information into the new database.

During the upgrade process, the upgrader does not send output to the console. Depending on the size of your database, the database upgrade process may take a long time.

Depending on the size of your database, the database upgrade process may take a long time.

Use the procedure in [Troubleshooting: Checking the Status of the Database Upgrade](#) on page 7-25 to verify that the upgrade is progressing.

3. Verify that the database upgrade is progressing by checking the end of the log (also referred to as tailing the log).

You should also check if the upgrade process succeeded. If the process succeeds, it returns a zero (0) exit code, and the last few lines of its log file show that the process has succeeded.

4. (Optional) Run the background upgrader again one or more times.

Running the background upgrader again after its initial pass allows you to capture the ongoing changes to the database because the Commander 4.2 server is still running. The background upgrader will replicate additions, deletions, and updates from the source (original) database to the target (ElectricCommander 5.x) database incrementally.

Upgrade Tool Overview

You can run the background upgrader to upgrade the database from ElectricCommander 4.2.x to ElectricCommander 5.x before you run the installer.

IMPORTANT: Only use the background upgrader when upgrading from ElectricCommander 4.2.x to ElectricCommander 5.x. Do not use it when you are updating ElectricCommander 5.0.x to a later ElectricCommander version.

- This minimizes downtime during the ElectricCommander software installation.
- When you run the installer without running the background upgrader first, the database is automatically updated as part of the installation process. However, this could take a long time and increase the downtime to several days.

IMPORTANT: Before you start the database upgrade, you must create a new, empty database that is the same type as the existing source (original) database and is the target (ElectricCommander 5.x) database for the upgrade.

To use the upgrade tool, perform the following steps. The tool prompts you for information if it is missing parameters required to upgrade ElectricCommander.

1. Run the migration tool to update the `jobNameTemplate` and `workflowNameTemplate` properties.

Go to [Migration Tool](#) on page 7-15 for more information.

You can also run the migration tool after you use the installer.

2. Run the background upgrader to upgrade the database.

To create or edit the upgrade properties file, see [Using the ElectricCommander 4.2.x Built-In Database in ElectricCommander 5.x](#) on page 7-23 or more information.

This may take a long time depending on the size of your database. If you have a very large database, it may take several days to copy your existing database information to the new database.

IMPORTANT: During the database upgrade, this step can be interrupted or stopped due to errors. If this happens, the background upgrader restarts the process to copy the database information, which may increase the time it takes to complete this step.

You can repeat this step more than once.

3. To complete the upgrade and install the Commander software, do one of the following:
 - Run the installer to update the Commander software.
 - If you have previously run the Commander installer, before you run the background upgrader, you need to run the background upgrader with the `--final` flag in the `java -jar commander-upgrade-jar-with-dependencies.jar --final` command. This completes the upgrade process. Go to [Troubleshooting: Using the --final Flag](#) on page 7-28 to understand the prerequisites before running the `--final` flag.

During this step, the data from the existing source (original) database for Commander 4.2.x is completely copied to the new target (Commander 5.x) database.

Result of the upgrade process:

- If the installation is successful, the installer returns a value of zero (0) in the `upgrade.log` file.
- If the installation fails or is incomplete, the installer returns a nonzero value in the `upgrade.log` file.

Using the ElectricCommander 4.2.x Built-In Database in ElectricCommander 5.x

In Commander 5.0, the built-in database changed from H2 to HSQLDB (also referred to as HyperSQL Database).

If you are using the built-in database in Commander 4.2.x and want to continue using the built-in database in Commander 5.x, you should review the following methods to create a backup of the Commander 4.2.x built-in database and determine which method is appropriate for your environment:

1. Create a complete XML database backup of your Commander 4.2.x database using `ectool export`.

We recommend that you run `ectool export` on an inactive system to prevent this process from failing due to conflicts.

You can also create a backup of your Commander 4.2.x database by exporting without jobs to limit the time the system is inactive. However, the backup will not have history data.

See [Data Backup Methods on page 11-1](#) and the “Using ectool and the Commander API” help topic in the Commander online help system for more information about this process.

2. Install a Commander 5.x server that includes the built-in database.
3. Import the Commander 4.2.x database into Commander 5.x using `ectool import`.

Troubleshooting the Database Upgrade

Use the following information to troubleshoot the database upgrade:

Issue	What to Do	Links to Help topics
What ectool options to use for troubleshooting	Use <code>--help</code> and <code>--version</code> .	Troubleshooting: Using ectool Options on page 7-25
Check the status of the database upgrade or determine why the upgrade failed.	Review the <code>upgrade.log</code> file.	Troubleshooting: upgrade.log File on page 7-25 Troubleshooting: Checking the Status of the Database Upgrade on page 7-25
The database username and password have changed.	Go to Troubleshooting: When the Database Username and Password Change .	Troubleshooting: When the Database Username and Password Change on page 7-27
When the source (original) and target (ElectricCommander 5.x) databases are on different servers.	Go to Troubleshooting: When the Source and Target Databases Are on Different Servers on page 7-27 .	Troubleshooting: When the Source and Target Databases Are on Different Servers on page 7-27
Using the <code>--final</code> flag with the background upgrader.	Go to Troubleshooting: The --final Flag .	Troubleshooting: The --final Flag
The Oracle database upgrade fails.	Go to Troubleshooting Oracle: Upgrade Fails on page 7-29 .	Troubleshooting Oracle: Upgrade Fails on page 7-29

Issue	What to Do	Links to Help topics
The database does not have enough disk space for the database upgrade.	Go to Troubleshooting: Not Enough Disk Space for the Database Upgrade .	Troubleshooting: Not Enough Disk Space for the Database Upgrade on page 7-29
The upgrade tool can fail to detect database changes.	Go to Troubleshooting: Upgrade Tool Can Fail to Detect Database Changes .	Troubleshooting: Upgrade Tool Can Fail to Detect Database Changes on page 7-30
Error messages	Go to Troubleshooting: Error Messages on page 7-26.	Troubleshooting: Error Messages on page 7-26

Troubleshooting: Using ectool Options

Use these ectool options to troubleshoot the database upgrade:

Option	Description
<code>--help</code>	Displays this help information.
<code>--version</code>	Displays the version information

Troubleshooting: upgrade.log File

Review the upgrade.log file when you want to do the following:

- Check the status of the database upgrade because it does not seem like it is doing anything.
- Determine why a database upgrade failed.

The upgrade.log file is located at `../logs/upgrade.log`, where `..` represents your current working directory for running the database upgrade tool.

When the background upgrader is run by the ElectricCommander installer, the current working directory will not be a subdirectory of the temp directory.

Troubleshooting: Checking the Status of the Database Upgrade

You can verify that the upgrade is progressing by using the `tail -f` command on the upgrade.log file in a separate terminal window. For example:

```
tail -f ../logs/upgrade.log
```

where `..` represents your current working directory for running the database upgrade tool.

Depending on the amount of data to copy from the source to the target (ElectricCommander 5.x), you might not see changes in the upgrade.log file for anything up to six to ten hours.

Troubleshooting: Error Messages

These error messages also apply when you are upgrading ElectricCommander 4.2.x to ElectricCommander 5.x.

Warning Regarding the Commander 4.2.x Database Schema

When the background upgrader is running, you need to check the upgrade.log file for errors. If you see the following warning near the end of the log, the upgrade did not complete successfully.

```
2014-11-12T16:19:17.085 | WARN | main | | | | MigrateDatabaseImpl | Due to issues with Oracle support for the SERIALIZABLE isolation level, it is not possible on Oracle to efficiently do an exact sync of the 5.0 schema to the 4.2 schema while the 4.2.x Commander server is still running. The 5.0 schema is currently NOT fully synchronized and NOT safe to use with 5.0, and its foreign key constraints are still disabled. However, the bulk of the data should now have been copied across, so future upgrader runs should be relatively quick. The 5.0 installer will automatically run the upgrader again after shutting down the 4.2.x server: as long as that upgrader run succeeds, it will bring the 5.0 schema into exact sync with the state of the 4.2 schema and reenables the constraints.
```

Note that the error states that it is not possible on Oracle to efficiently do an exact sync of the 5.0 schema to the 4.2 schema while the 4.2.x Commander server is still running.

If you see this warning, do the following:

1. Ensure that the Commander 4.2.x server is not accessing the database.

If it is, stop all Commander services on the Commander 4.2.x server.

2. Do one of these actions:

- a. Run the background upgrader with the `--final` flag using this command:

```
java -jar commander-upgrade-jar-with-dependencies.jar --final
```

- b. Run the Commander 5.x installer, which automatically runs the background upgrader with the `--final` flag one last time.

For more information on the `--final` flag, go to the following:

- [Upgrade Tool Overview](#) on page 7-22
- [Troubleshooting: Using the --final Flag](#) on page 7-28

Table or View Does Not Exist

See also: [Troubleshooting: When the Source and Target Databases Are on Different Servers](#) on page 7-27

When the background upgrader is running, this error is in the upgrade.log file:

```
Unexpected error:org.hibernate.exception.SQLGrammarException: could not execute statement : java.sql.SQLException: ORA-00942: table or view does not exist
```

Generally, this error occurs when the owner of the Commander 5.x database table owner does not have read access to the Commander 4.2.x database tables. If the user permissions that you are using to install Commander 5.x do not have read access to the Commander 4.2.x tables, the user (you) cannot run the background upgrader. It cannot read the information to which it needs from the Commander 4.2.x tables to do the conversion.

To resolve this error:

1. Have your DBA give the Commander 5.x user access to the Commander 4.2 database tables.
2. Rerun the background upgrader.

Access Denied

This error may occur when the background upgrader is running:

```
2014-11-03T07:21:34.344 | DEBUG | CP-pool-watch-thread-commander | | | |
EcConnectionHook | Failed to acquire connection: java.sql.SQLException: Access denied
for user 'commander'@'localhost' (using password: YES).
```

To resolve this error:

1. Define the real password (not hash) for COMMANDER_DB_PASSWORD in upgrade.properties file.
2. Restart background upgrader.

For more information about the upgrade.properties file, go to [Using the upgrade.properties File Templates](#) on page 7-18.

Troubleshooting: When the Database Username and Password Change

Problem

While you are upgrading ElectricCommander with clustering, the database username and password changes. The database upgrade cannot be successfully completed.

Solution

Do one of the following tasks:

- Update the `database.properties` file on the ElectricCommander server.
If this does not resolve the issue, find where the file is stored and update that file.
- If the `database.properties` file is stored with other configuration files on the Apache ZooKeeper server,
 1. Update the username and password in the `database.properties` file.
 2. Run the following command to populate the ZooKeeper server with the new username and password:

```
COMMANDER_ZK_CONNECTION=<ZooKeeper_Server_IP>:2181 ../jre/bin/java -jar
../server/bin/zk-config-tool-jar-with-dependencies.jar
com.electriccloud.commander.cluster.ZKConfigTool --databasePropertiesFile
database.properties --keystoreFile keystore --passkeyFile passkey
--commanderPropertiesFile commander.properties
```

Troubleshooting: When the Source and Target Databases Are on Different Servers

Problem

The upgrade from ElectricCommander 4.2.x to ElectricCommander 5.x was successfully completed at a site. After the upgrade, the data center is moved to a new site. However, the new MS SQL database cannot be brought up at the new site, and a SQL error message about a "table or view not found" appears.

Solution

This is an permissions issue. Assign the user bringing up the database at the new site the proper permissions for this task.

At the new site, the system interprets the user bringing up the new database as a new user who does not have the proper permissions to bring up the database. The database is the same database instance that was at the original site.

Troubleshooting: Using the `--final` Flag

Problem

When you use the `--final` flag in the `java -jar commander-upgrade-jar-with-dependencies.jar --final` command to complete the database upgrade, one of the following occurs:

- While the ElectricCommander 4.2.x database server is running, the background upgrader has been running more than once to capture the ongoing updates in the source (ElectricCommander 4.2.x server) and to prepopulate the target (ElectricCommander 5.x) database for ElectricCommander 5.x.

Before running the ElectricCommander installer, the background upgrader is run with the `--final` flag for the last time. The upgrade fails, and an error message appears that there are issues with MS SQL server support for the SNAPSHOT isolation level.

You may also notice that the size of the database has decreased.

- The target (ElectricCommander 5.x) database is on a different server than the source (original) database.

When you install ElectricCommander 5.x, the database upgrade fails, and an error message appears that there are issues with MS SQL server support for the SNAPSHOT isolation level.

You may also notice that the size of the database has decreased.

Solution

The error message means that while the Commander 4.2.x server is still running, the background upgrader was successfully run without the `--final` flag against the database.

However, it is not possible to perform an exact synchronization of the ElectricCommander 5.x database to the ElectricCommander 4.2.x database while the Commander 4.2.x server is still running. The ElectricCommander 5.x database is not yet fully synchronized and is not safe to use with ElectricCommander 5.x. The database foreign key constraints are still disabled. Most of the data should now have been copied from the source and target (ElectricCommander 5.x) database, and future upgrade runs should be take less time.

The size of the database is expected to decrease before the last run of the background upgrader. It is common for an active database with many create and delete updates to have unused space in it. The upgrade from ElectricCommander 4.2.x to ElectricCommander 5.x compacts the database. To verify that the ElectricCommander 4.2.x database has been upgraded to the ElectricCommander 5.x database, you can compare row counts for various tables in the ElectricCommander 4.2.x and ElectricCommander 5.x databases. Most of the tables should match exactly.

Do one of the following to complete the database upgrade:

- When you are installing the Commander 5.x server on the same machine as the Commander 4.2.x server, we recommend that you shut down the Commander 4.2.x server and run the ElectricCommander installer with the appropriate options. During the installation, the background upgrader runs automatically with the `--final` flag to complete the database upgrade.

- Shut down the Commander 4.2.x server before running the background upgrader the last time. Then run the background upgrader with the `--final` flag in the `java -jar commander-upgrade-jar-with-dependencies.jar --final` command. If the background upgrader runs successfully, the ElectricCommander 5.x database and Commander 4.2.x database are completely synchronized and the constraints are enabled again.

Troubleshooting Oracle: Upgrade Fails

If the Oracle database upgrade fails, try to fix the problem as follows:

- Edit the `upgrade.properties` file.

For example, decrease the `COMMANDER_DB_BATCH_SIZE` from 100 to a smaller value, such as 50.

- Edit the timeout settings in Oracle.

The batch size affects how long the upgrade process takes.

If an error occurs and the upgrade process stops, you may have to use a smaller batch size, which results in it taking longer to run the upgrade process.

Troubleshooting: Not Enough Disk Space for the Database Upgrade

Problem

When the database does not have enough disk space for the database upgrade, one of the following occurs:

- The upgrade tool runs for at least 15 minutes and then fails and a log message about the database ran out of temporary or undo space (such as ORA-00001, ORA-01652, ORA-08176, ORA-08177, or ORA-15555 on Oracle) appears.
- The upgrade tool hangs for 12 or more hours without SQL messages being added to the log during that time. The database server has used up all its available storage space. It could have run out of disk space, or have run out of RAM and be disk-thrashing.

Solution

Your DBA may need to do the following tasks:

- Make more space available in the database, use a lower value for the `COMMANDER_DB_BATCH_SIZE` setting in the `upgrader.conf` file, or do both tasks. Then rerun the upgrade tool.
- In general, smaller batches consume less database temporary and undo space. The lower `COMMANDER_DB_BATCH_SIZE` values, however, are likely to make transferring a large database take longer, because smaller batches incur more overhead by resynchronizing to the updated state of the ElectricCommander 4.2 database more frequently.
- We recommend an initial value of 100 for `COMMANDER_DB_BATCH_SIZE`. If that value fails, try 30, 10, 5, 2, and 1 in that order.
- For most tables, the default batch row limit used is `COMMANDER_DB_BATCH_SIZE` x 100,000 rows per SQL query. The upgrade tool might temporarily reduce the row limit for some tables if it detects a problem.

- In general, it is not necessary (and is counterproductive) to wipe the ElectricCommander 5.x database before rerunning the upgrade tool. The upgrade tool copies differences from the ElectricCommander 4.2 to the ElectricCommander 5.x database to synchronize them (inserting, updating, or deleting rows from the ElectricCommander 5.x database as required). This means that a new run of the upgrade tool continues building on any data that has already been transferred. However, if the upgrade tool repeatedly fails after only a few minutes with no data transferred and an error about the database being invalid, try wiping the ElectricCommander 5.x database and rerunning the upgrade tool.
- If the upgrade tool stops, or if you have to kill it, check that:
 - You are connected to your database.
 - The temporary resources that the upgrade tool was using have been released. If not, have your DBA kill the upgrade tool's connection.

Troubleshooting: Upgrade Tool Can Fail to Detect Database Changes

The following sequence of events rarely occurs, but may cause the upgrade tool to fail to detect existing source database changes when the upgrade process is creating a new target (ElectricCommander 5.x) database copy and keeping it synchronized with the source (original) database.

1. A run of one or more source (original) database entries with the greatest ID numbers in use have already been copied to the target (ElectricCommander 5.x) database.
2. This run of entries is deleted from the source (original) database, which reduces the greatest ID number in use.
3. Some or all of the deletions are not yet successfully propagated to the target (ElectricCommander 5.x) database, and the old Commander version is shut down.
4. While the old Commander version is down, the background upgrader process does not complete successfully, so the deletions still are not copied to the target (ElectricCommander 5.x) database.
5. The old Commander version restarts and checks the source (original) database, finds the greatest ID currently in use, and sets that ID plus one as the ID of the next entry to be created.
6. The old Commander server creates an entry reusing the ID of a deleted object that was not copied to the source (original) database, and by chance the entry:
 - Is in the same table as the deleted entry with the same ID
 - Has the same revision number (which is most likely if both were revision 1)
 - Is a type that does not have a name, or else it has the same name as the deleted entry, but
 - Is not completely identical to the deleted entry

Based on all criteria used by the upgrade tool, the two objects appear identical, so the upgrade tool does not know that it must delete the target (ElectricCommander 5.x) database entry and replace it with the new entry from the source (original) database. However, the two are not actually identical, and these differences are not propagated to the target (ElectricCommander 5.x) database.

7. During the remaining background upgrader process, ElectricCommander does not further update the entry in the source (original) database.

To ensure that this does not happen, after you start the database upgrade process, do not shut down the old Commander server during the upgrade process. If, for some reason, you must do so, then do one of the following:

- Run the upgrade tool to completion before restarting the old Commander server.
- If this is not practicable (for example, it requires too much downtime for the old Commander server), wipe the partially-copied target (ElectricCommander 5.x) database and start the entire upgrade process again.

User Interface Upgrade Method

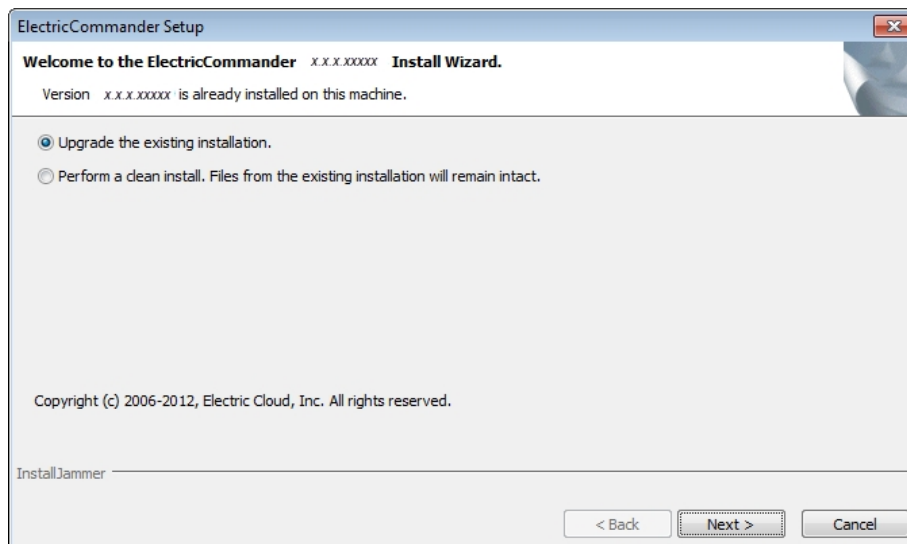
Use the following procedure to upgrade the Commander software. Review [Preparing for the Upgrade on page 7-13](#) before performing this procedure.

When you upgrade ElectricCommander from ElectricCommander 4.2.x to ElectricCommander 5.x, we recommend that you run the background upgrader and complete the database upgrade before upgrading to Commander 5.x because the upgrade process requires an extensive database upgrade.

Note: Upgrading to Commander 4.2 or later saves the existing database, if applicable.

1. Make sure that you have run the background upgrader and that the database upgrade is complete, and then connect your Commander server to the database.
2. Open the installer and double-click the `ElectricCommander-<version>` file to begin the installation process.

The Welcome to the ElectricCommander Installer screen appears.



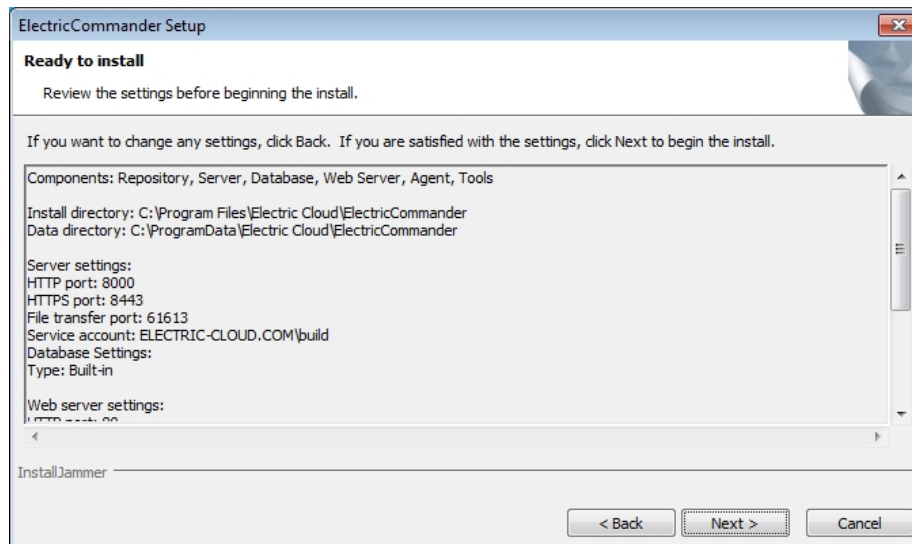
3. Choose one of the following options:
 - Select **Upgrade the existing installation** if you want to upgrade your current Commander installation directory.
 - Select **Perform a clean install** if you want to specify a different installation directory for the new version.

IMPORTANT: During a clean installation, current services remain running until you click **Next** on the Ready to Install screen. This means the new installation cannot use the same ports and

directories as the existing installation. To use the same ports and directories, you must manually stop the existing services. This will free the existing ports and directories.

3. Click **Next**.
4. Provide your **New Database Name** (non-Oracle) or **New Database Username** (Oracle) and the corresponding **Password** and click **Next** to continue.
5. Click **Next** to upgrade the existing installation.

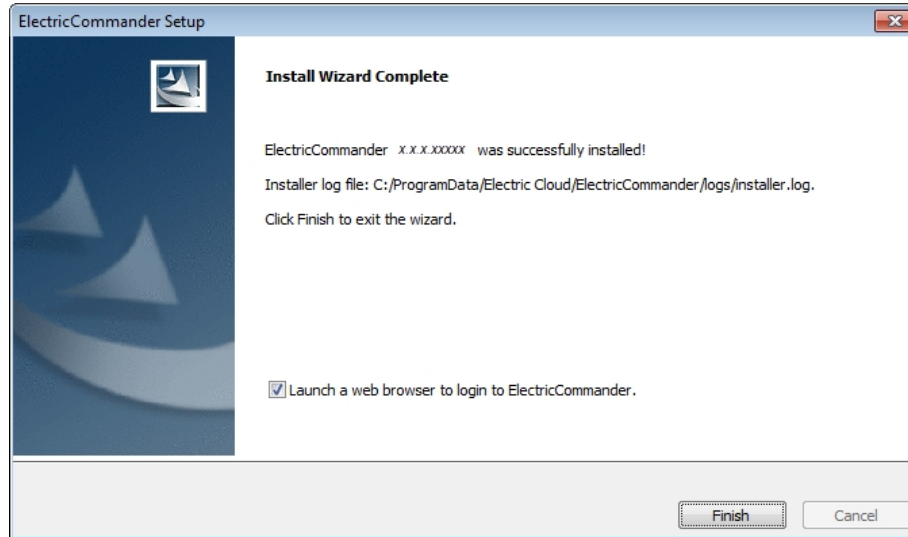
The Ready to Install screen appears.



6. Review the upgrade settings. Use the Back button to change your selections if necessary.
7. Click **Next** to continue.

The installer displays a status bar to show the progress of the upgrade process. This process can take up to fifteen minutes to complete. After this process is complete, the new Commander version is

installed.

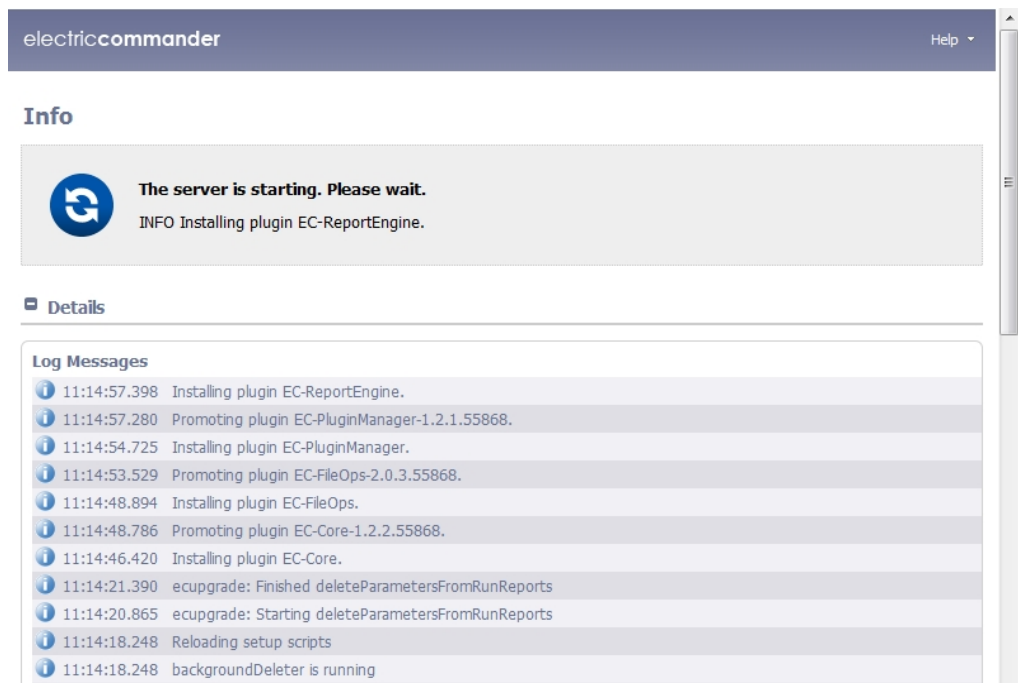


8. Select the **Launch a web browser to login to ElectricCommander** check box if you want Commander to open the login screen now.
9. Click **Finish** to complete the upgrade.

If a Commander server is being upgraded, when the installation is complete the server will continue to upgrade the database (if applicable). You will not be able to log in to the Commander server until the database upgrade is complete. You can view the upgrade status by using `ectool` from a command line:
`ectool getServerStatus.`

After clicking **Finish**, you might see a web page similar to the following screen if the upgrade is still in

progress.



Interactive Command-line Upgrade Method

Use the following procedure to complete the command-line upgrade of a Linux platform. Review [Preparing for the Upgrade on page 7-13](#) before performing this procedure.

When you upgrade ElectricCommander from ElectricCommander 4.2.x to ElectricCommander 5.x, we recommend that you run the background upgrader and complete the database upgrade before upgrading to Commander 5.x because the upgrade process requires an extensive database upgrade.

Note: Upgrading to Commander 4.2 or later saves the existing database.

1. Make sure that you have run the background upgrader and that the database upgrade is complete.
Then connect your Commander server to the database.
2. Go to the command line and enter one of the following commands to begin the upgrade:
 - If you have a Linux platform, enter `./ElectricCommander-<version> .`
 - If you have a Linux platform with the X Window System, the installer automatically brings up the graphical user interface. To override this behavior, enter `./ElectricCommander-<version> --mode console.`

The system displays the following message:

```
Copyright (c) 2006-2014, Electric Cloud, Inc. All rights reserved.
This will install ElectricCommander on your computer. Continue? [n/Y]
```

3. Enter `y`.

The system displays the following message:


```
Upgrade the existing <version> installation to version  
<version>? [n/Y]
```

4. Choose one of the following options:

- If you want to upgrade your current Commander installation directory, enter: `y`.
- If you want to specify a different installation directory (clean install) for the new version, perform the following steps:

a. Enter `n`.

The system displays a prompt for a clean installation. During a clean installation, current services remain running until the install process starts. This means the new installation cannot use the same ports and directories as the existing installation. To use the same ports and directories, you must manually stop the existing services. This will free the existing ports and directories.

b. Enter `y` and complete the information for the subsequent prompts.

The system displays the following message:

For a non-Oracle database: New database name and database password.

For an Oracle database: New database username and database password.

Then the system displays the following message:

```
Installing agent...
```

```
Installing server...
```

```
Copied log file to "/opt/electriccloud/electriccommander/logs"
```

```
ElectricCommander x.x.x.xxxxx was successfully installed!
```

```
Installer log file: /opt/electriccloud/electriccommander/logs/installer.log.
```

Silent (Unattended) Upgrade Method

You can run the Commander upgrade in silent (unattended) mode with no user interface for either Windows or Linux.

When you upgrade from ElectricCommander 4.2.x to ElectricCommander 5.x, we recommend that you run the background upgrader and complete the database upgrade before upgrading to Commander 5.x because the upgrade process requires an extensive database upgrade.

Note: Upgrading to Commander 5.x saves the existing database.

To upgrade Commander in silent mode:

1. Make sure that you have run the background upgrader and that the database upgrade is complete.
2. Connect your Commander server to the database.

3. Go to the command line and enter one of the following commands from a command line to begin the silent upgrade (for non-Oracle and Oracle databases, respectively):

```
./ElectricCommander-<version> --mode silent --upgradeNewDBName <newdatabasename>  
--upgradeDBPassword <password> --haveRunMigrationUtility
```

```
./ElectricCommander-<version> --mode silent --upgradeNewDBUserName <newdatabaseu  
sername> --upgradeDBPassword <password>
```

where <newdatabaseusername> is your new database username, <newdatabasename> is your new database name, and <password> is the corresponding password.

Copying Repository Contents

Perform the following steps to copy the contents of an existing repository server into a newly installed repository server:

1. Install the new repository server software.
2. Stop the existing and new repository servers.
3. Copy the entire contents of the repository backingstore directory from the existing repository server to the corresponding location on the newly installed repository server.

The default location for the backingstore directory (<datadir>/repository-data) is:

- **UNIX** - /opt/electriccloud/electriccommander/repository-data
- **Windows** - C:\ProgramData\Electric Cloud\ElectricCommander\repository-data

Chapter 8: Upgrading from ElectricCommander 5.0.x to ElectricCommander 5.x With Clustering

IMPORTANT: If your system is running a version earlier than ElectricCommander 4.2, you first need to upgrade to ElectricCommander 4.2.x. You should be able to safely upgrade from ElectricCommander 3.10.x, 4.0.x, or 4.1.x to ElectricCommander 4.2.x, and then upgrade to ElectricCommander 5.x.

IMPORTANT: This upgrade process does not work if your system is running ElectricCommander 4.2.x. You must use the upgrade process described in [Upgrading from ElectricCommander 4.2.x to ElectricCommander 5.x on page 7-1](#).

To upgrade ElectricCommander from Version 5.0.x to Version 5.x, use the `ElectricCommander-<version>` installer, which collects the Commander service account credentials, uninstalls the current release, installs the latest Commander release, configures the system with all property values mined, and restores custom files and data.

After preparing for the upgrade, make sure to shut down the Commander server before installing ElectricCommander 5.x.

Upgrading Clusters

In a cluster configuration, make sure to review the tasks in [Preparing for the Upgrade on page 7-13](#). You must make backup copies of these files:

- `httpd.conf` in `<data directory>/apache/conf` subdirectory
- `wrapper.conf` in `<data directory>/conf` subdirectory

Before starting the upgrade process, shut down the Commander server before installing ElectricCommander 5.x.

During the upgrade process, ElectricCommander is not available after the web server is stopped or after the last Commander server node is stopped.

1. Perform either of these actions:

- Stop the Commander server on all nodes.
- Stop the Commander server on all nodes except the primary Commander server node.

You can keep your primary Commander server node up at this point because the installer automatically stops the primary node when it upgrades that node.

2. On the Commander server, do the following:
 - a. Upgrade the Commander server. This will also do the following:
 - Connect the server to the database.
 - Upgrade the plugins.
 - Start the server.
 - b. Check and restore the `wrapper.conf` settings. For example, the settings for the line `"wrapper.java.additional.600="`.
 - c. Restart the Commander server.
3. On the Commander web server, do the following:
 - a. Upgrade the node on the web server.
 - b. Check and restore the `httpd.conf` settings.

4. Upgrade any nodes on the repository servers.

Your ElectricCommander system is now available.

5. Upgrade the remaining Commander server nodes.

During the upgrade process, some settings may be lost. Verify the following settings before connecting to the ElectricCommander system:

- `httpd.conf` settings for redirecting—These lines should be commented out:

```
# Redirect http to https
# RewriteCond %{HTTPS} !=on
# RewriteRule ^/commander/(.*) https://%{SERVER_NAME}:443%{REQUEST_URI} [NC,R,L]
```
- `httpd.conf` setting for `COMMANDER_SERVER`—This should point to the load balancer:

```
SetEnv COMMANDER_SERVER "<FQDN of your load balancer>"
```
- `wrapper.conf` contains the line pointing at your Zookeeper instances

For example:

```
wrapper.java.additional.600=-DCOMMANDER_ZK_CONNECTION=192.168.7.20:2181
```

Choosing the Correct Upgrade Method

This section describes the various upgrade methods and options for specific platform configurations. For information about supported server platforms and supported non-server platforms, see [Server Platforms](#) on page 2-1 and [Agent Platforms](#) on page 2-4.

User Interface Upgrade

This method provides a wizard for upgrading the Commander software on a supported server platform. This upgrade method is generally preferred by Windows users, but is supported on Linux platforms with the X Window System installed.

Upgrade options:

- **Upgrade Existing Installation**

This option uninstalls the current release, installs the latest Commander release, collects the Commander service account credentials, configures the system with all property values, and restores custom files and data.

- **Clean Install**

This option allows you to specify a different installation directory for the new version. The files from your previous Commander version will not be removed or modified and will remain in their original directories.

Interactive Command-Line Upgrade

This method provides an interactive command-line for upgrading the Commander software on a supported server platform. This upgrade method is only available for Linux platforms.

Upgrade options:

- **Upgrade Existing Installation**

This option uninstalls the current release, installs the latest Commander release, collects the Commander service account credentials, configures the system with all property values, and restores custom files and data.

- **Clean Install**

This option allows you to specify a different installation directory for the new version. The files from your previous Commander version will not be removed or modified and will remain in their original directories.

Silent Unattended Upgrade

This method provides a non-interactive command-line upgrade for supported server platforms. You may find this installation method preferable for upgrading multiple remote agents and servers.

Upgrade options:

- **Upgrade Existing Installation**

This option uninstalls the current release, installs the latest Commander release, collects the Commander service account credentials, configures the system with all property values, and restores custom files and data.

IMPORTANT: You cannot add a new repository server with this upgrade method.

Repository Server With a Commander Upgrade

After Commander 4.x, the only way to install a repository server on the same machine as other services is to uninstall and reinstall Commander. You can install the repository server on a different machine to avoid uninstalling and reinstalling Commander.

Non-Server Platform Agent Upgrade

You cannot directly upgrade a non-server platform agent. You must uninstall and then reinstall these machines using the Commander installer. For more information, see [Uninstalling ElectricCommander](#) on page 10-1 and [Non-Server Platform Agent Installation Method](#) on page 3-34.

Stand-Alone Repository Server or Web Server Upgrade

You cannot directly upgrade a stand-alone repository server or stand-alone web server. You must uninstall and then reinstall these servers using the Commander installer. The uninstall and reinstall process is required to install an agent on the server machine. Starting in Commander 4.2, an agent is required on the machine with the stand-alone repository server or web server. For more information, see [Copying Repository Content](#), [Uninstalling ElectricCommander](#) on page 10-1, and [Installing ElectricCommander](#) on page 3-1.

Non-Server Platform Agent Upgrade

You cannot directly upgrade a non-server platform agent. You must uninstall and then reinstall these machines using the Commander installer. For more information, see [Uninstalling ElectricCommander](#) on page 10-1 and [Non-Server Platform Agent Installation Method](#) on page 3-34.

Preparing for Your Upgrade

Review the following information before you upgrade the Commander software.

Upgrade Testing

In most implementations, ElectricCommander is being used in an environment that affects many users. We recommend that you test your upgrade on a separate test server to understand all aspects of the upgrade process. This minimizes the potential impacts to downstream users.

Backups

If you are upgrading a Commander server, it is *extremely* important that you back up your existing Commander data *before* upgrading. See [Commander Server Backups](#) on page 11-1 for more information about backups.

- Always back up the Plugins Directory. The default location is the `plugins` subdirectory within the data directory.
- Always back up the files that contain your configuration and custom settings. To ensure all important settings are saved, back up the following subdirectories in the data directory:
 - Commander server and agent configuration files in the `conf` subdirectory
 - The `passkey` file in one of these directories:
`/opt/electriccloud/electriccommander/conf/passkey` in Linux or
`C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows
 - Apache web server configuration files in the `apache/conf` subdirectory
 - The local MySQL database configuration file, `my.ini`, in the `mysql` subdirectory (if applicable)
- Always back up any other files where you have created custom configurations, specified other custom information, or created any type of modification.

Note: The number of Commander files you might have modified is too numerous to list, so you might want to back up the entire Commander data directory and other miscellaneous files that might have changed.

- If you use an artifact repository, back up your Commander repository configuration files in the `conf/repository` subdirectory.
- Determine if any changes were made to the custom editor or preflight driver script properties (installed by default). Back up those files if changes were made.

These properties are stored in the server-level property sheet, which can be viewed in the web UI by accessing the Administration tab/ Server subtab.

Custom editors are stored in the nested sheet named `ec_customEditors`. Preflight driver scripts are stored in the nested sheet named `ec_preflight`. The upgrade process overwrites default custom editor and preflight driver scripts with current versions. We recommend backing up any custom properties you created by renaming the property. For example, change `ec_preflight/clientDrivers/perforce` to `ec_preflight/clientDrivers/perforce_modified`.

Upgrade Installer Preservation

After you back up your ElectricCommander server, create a folder where you can download the `ElectricCommander-<version>` installation file.

MySQL Upgrades

Commander upgrades involving a MySQL database can take several hours to complete if you have a significant data set. *Do not interrupt the upgrade process.* You can corrupt your database if the upgrade process is interrupted. A restore from a previous database backup will be required.

- Use the `ectool` to view the upgrade progress. On a command line, enter

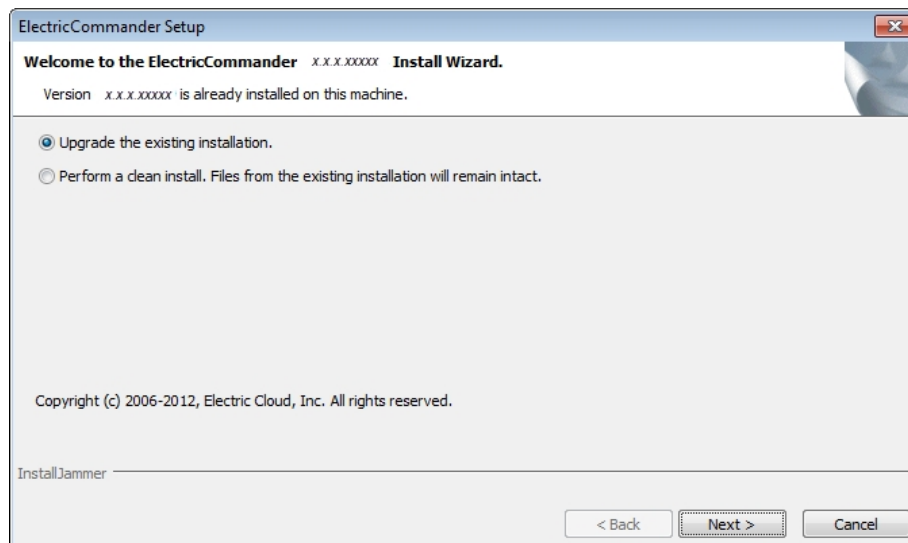
```
ectool getServerStatus
```

Note: An install/upgrade log file named `installer.log` is created in the `logs` subdirectory in the data directory.

User Interface Upgrade Method

Use this procedure to upgrade the Commander software. Review [Preparing for the Upgrade on page 7-13](#) before performing this procedure.

1. Double-click the `ElectricCommander-<version>` file to begin installation. The Welcome to the ElectricCommander Installer screen appears.



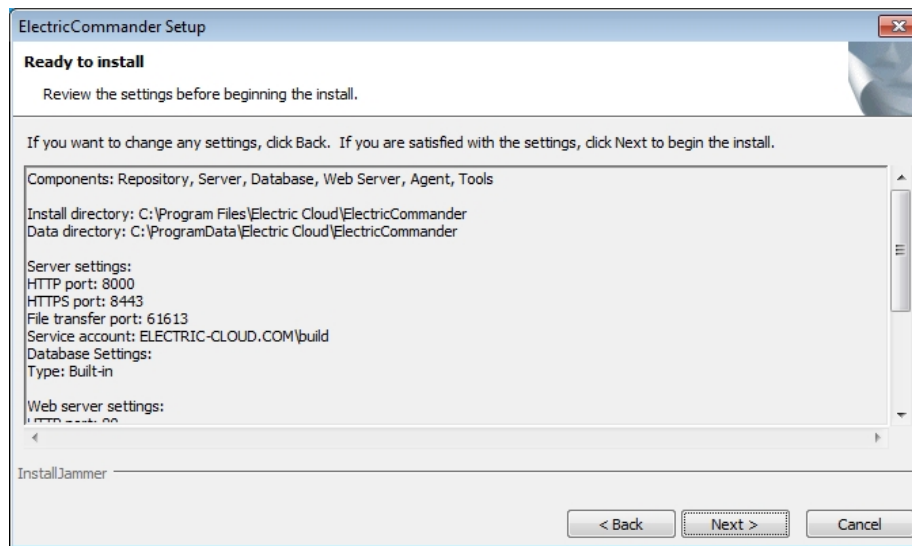
2. Choose one of the following options:
 - Select **Upgrade the existing installation** if you want to upgrade your current Commander installation directory.

- Select **Perform a clean install** if you want to specify a different installation directory for the new version.

IMPORTANT: During a clean installation, current services remain running until you click **Next** on the Ready to Install screen. This means the new installation cannot use the same ports and directories as the existing installation. To use the same ports and directories, you must manually stop the existing services. This will free the existing ports and directories.

3. Click **Next** to upgrade the existing installation.

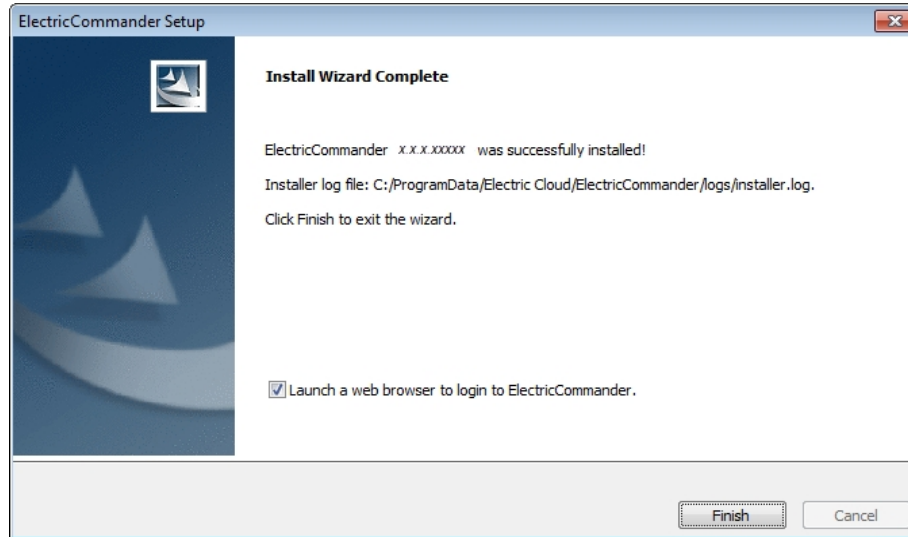
The Ready to Install screen appears.



4. Review the upgrade settings. Use the Back button to change your selections if necessary.
5. Click **Next** to continue.

The installer displays a status bar to show the progress of the upgrade process. The time that it takes to complete this process depends on the size of the database. It may take fifteen minutes or longer to

complete. Once this process is complete, the new Commander version is installed.



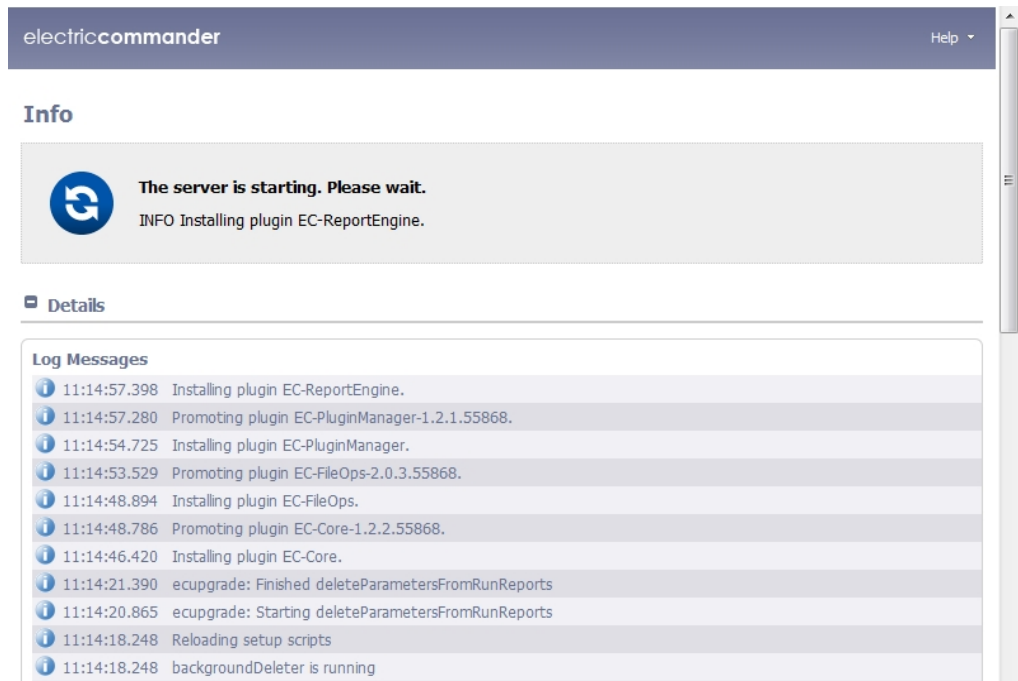
6. Select the **Launch a web browser to login to ElectricCommander** check box if you want Commander to open the login screen now.
7. Click **Finish** to complete the upgrade.

If a Commander server is being upgraded, when the installation is complete the server will continue to upgrade the database (if applicable). You will not be able to log in to the Commander server until the database upgrade is complete. You can view the upgrade status by using `ectool` from a command line:

```
ectool getServerStatus.
```

After clicking **Finish**, you may see a web page similar to the following screen if the upgrade is still in

progress.



Interactive Command-Line Upgrade Method

Use the following procedure to complete a command-line upgrade of a Linux platform. Review [Preparing for the Upgrade on page 7-13](#) before performing this procedure.

1. Choose one of the following commands to begin the upgrade:
 - If you have a Linux platform, enter `./ElectricCommander-<version> .`
 - If you have a Linux platform with the X Window System, the installer will automatically bring up the graphical user interface. To override this behavior, enter `./ElectricCommander-<version> --mode console .`

The system displays the following message:

```
Copyright (c) 2006-2014, Electric Cloud, Inc. All rights reserved.
This will install ElectricCommander on your computer. Continue? [n/Y]
```

2. Enter: `y`

The System displays the following message:

```
Upgrade the existing <version> installation to version <version>? [n/Y]
```

3. Choose one of the following options:
 - If you want to upgrade your current Commander installation directory, enter `y`.
 - If you want to specify a different installation directory (clean install) for the new version, perform the following steps:

- a. Enter `n`.

The system displays a prompt for a clean installation. During a clean installation, current services remain running until the install process starts. This means the new installation cannot use the same ports and directories as the existing installation. To use the same ports and directories, you must manually stop the existing services. This will free the existing ports and directories.

- b. Enter `y` and complete the information for the subsequent prompts.

The system displays the following message:

```
Installing agent...
Installing server...
Copied log file to "/opt/electriccloud/electriccommander/logs"
ElectricCommander <version> was successfully installed!
Installer log file: /opt/electriccloud/electriccommander/logs/installer.log.
```

Silent (Unattended) Upgrade Method

You can run the Commander upgrade in unattended (silent) mode with no user interface for either Windows or Linux.

Enter the following command from a command line to begin a silent upgrade:

```
./ElectricCommander-<version> --mode silent
```

Chapter 9: Upgrading from ElectricCommander 5.0.x to ElectricCommander 5.x Without Clustering

IMPORTANT: If your system is running a version earlier than ElectricCommander 4.2, you first need to upgrade to ElectricCommander 4.2.x. You should be able to safely upgrade from ElectricCommander 3.10.x , 4.0.x, or 4.1.x to ElectricCommander 4.2.x, and then upgrade to ElectricCommander 5.x.

IMPORTANT: This upgrade process does not work if your system is running ElectricCommander 4.2.x. You must use the upgrade process described in [Upgrading from ElectricCommander 4.2.x to ElectricCommander 5.x on page 7-1](#).

To upgrade ElectricCommander from Version 5.0.x to Version 5.x, use the `ElectricCommander-<version>` installer, which collects the Commander service account credentials, uninstalls the current release, installs the latest Commander release, configures the system with all property values mined, and restores custom files and data.

After preparing for the upgrade, make sure to shut down the Commander server before installing ElectricCommander 5.x.

Choosing the Correct Upgrade Method

This section describes the various upgrade methods and options for specific platform configurations. For information about supported server platforms and supported non-server platforms, see [Server Platforms](#) on page 2-1 and [Agent Platforms](#) on page 2-4.

User Interface Upgrade

This method provides a wizard for upgrading the Commander software on a supported server platform. This upgrade method is generally preferred by Windows users, but is supported on Linux platforms with the X Window System installed.

Upgrade options:

- **Upgrade Existing Installation**

This option uninstalls the current release, installs the latest Commander release, collects the Commander service account credentials, configures the system with all property values, and restores custom files and data.

- **Clean Install**

This option allows you to specify a different installation directory for the new version. The files from your previous Commander version will not be removed or modified and will remain in their original directories.

Interactive Command-Line Upgrade

This method provides an interactive command-line for upgrading the Commander software on a supported server platform. This upgrade method is only available for Linux platforms.

Upgrade options:

- **Upgrade Existing Installation**

This option uninstalls the current release, installs the latest Commander release, collects the Commander service account credentials, configures the system with all property values, and restores custom files and data.

- **Clean Install**

This option allows you to specify a different installation directory for the new version. The files from your previous Commander version will not be removed or modified and will remain in their original directories.

Silent Unattended Upgrade

This method provides a non-interactive command-line upgrade for supported server platforms. You may find this installation method preferable for upgrading multiple remote agents and servers.

Upgrade options:

- **Upgrade Existing Installation**

This option uninstalls the current release, installs the latest Commander release, collects the Commander service account credentials, configures the system with all property values, and restores custom files and data.

IMPORTANT: You cannot add a new repository server with this upgrade method.

Repository Server With a Commander Upgrade

After Commander 4.x, the only way to install a repository server on the same machine as other services is to uninstall and reinstall Commander. You can install the repository server on a different machine to avoid uninstalling and reinstalling Commander.

Non-Server Platform Agent Upgrade

You cannot directly upgrade a non-server platform agent. You must uninstall and then reinstall these machines using the Commander installer. For more information, see [Uninstalling ElectricCommander](#) on page 10-1 and [Non-Server Platform Agent Installation Method](#) on page 3-34.

Stand-Alone Repository Server or Web Server Upgrade

You cannot directly upgrade a stand-alone repository server or stand-alone web server. You must uninstall and then reinstall these servers using the Commander installer. The uninstall and reinstall process is required to install an agent on the server machine. Starting in Commander 4.2, an agent is required on the machine with the stand-alone repository server or web server. For more information, see [Copying Repository Content](#), [Uninstalling ElectricCommander](#) on page 10-1, and [Installing ElectricCommander](#) on page 3-1.

Non-Server Platform Agent Upgrade

You cannot directly upgrade a non-server platform agent. You must uninstall and then reinstall these machines using the Commander installer. For more information, see [Uninstalling ElectricCommander](#) on page 10-1 and [Non-Server Platform Agent Installation Method](#) on page 3-34.

Preparing for Your Upgrade

Review the following information before you upgrade the Commander software.

Upgrade Testing

In most implementations, ElectricCommander is being used in an environment that affects many users. We recommend that you test your upgrade on a separate test server to understand all aspects of the upgrade process. This minimizes the potential impacts to downstream users.

Backups

If you are upgrading a Commander server, it is *extremely* important that you back up your existing Commander data *before* upgrading. See [Commander Server Backups](#) on page 11-1 for more information about backups.

- Always back up the Plugins Directory. The default location is the `plugins` subdirectory within the data directory.
- Always back up the files that contain your configuration and custom settings. To ensure all important settings are saved, back up the following subdirectories in the data directory:
 - Commander server and agent configuration files in the `conf` subdirectory
 - The `passkey` file in one of these directories:
`/opt/electriccloud/electriccommander/conf/passkey` in Linux or
`C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows
 - Apache web server configuration files in the `apache/conf` subdirectory
 - The local MySQL database configuration file, `my.ini`, in the `mysql` subdirectory (if applicable)
- Always back up any other files where you have created custom configurations, specified other custom information, or created any type of modification.

Note: The number of Commander files you might have modified is too numerous to list, so you might want to back up the entire Commander data directory and other miscellaneous files that might have changed.

- If you use an artifact repository, back up your Commander repository configuration files in the `conf/repository` subdirectory.
- Determine if any changes were made to the custom editor or preflight driver script properties (installed by default). Back up those files if changes were made.

These properties are stored in the server-level property sheet, which can be viewed in the web UI by accessing the Administration tab/ Server subtab.

Custom editors are stored in the nested sheet named `ec_customEditors`. Preflight driver scripts are stored in the nested sheet named `ec_preflight`. The upgrade process overwrites default custom editor and preflight driver scripts with current versions. We recommend backing up any custom properties you created by renaming the property. For example, change `ec_preflight/clientDrivers/perforce` to `ec_preflight/clientDrivers/perforce_modified`.

Upgrade Installer Preservation

After you back up your ElectricCommander server, create a folder where you can download the `ElectricCommander-<version>` installation file.

MySQL Upgrades

Commander upgrades involving a MySQL database can take several hours to complete if you have a significant data set. *Do not interrupt the upgrade process.* You can corrupt your database if the upgrade process is

interrupted. A restore from a previous database backup will be required.

- Use the `ectool` to view the upgrade progress. On a command line, enter

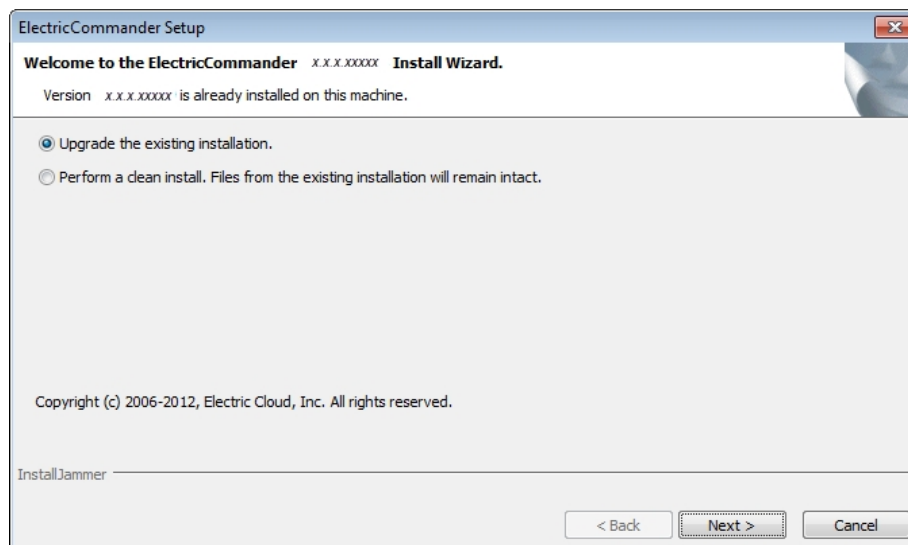
```
ectool getServerStatus
```

Note: An install/upgrade log file named `installer.log` is created in the `logs` subdirectory in the data directory.

User Interface Upgrade Method

Use this procedure to upgrade the Commander software. Review [Preparing for the Upgrade on page 7-13](#) before performing this procedure.

1. Double-click the `ElectricCommander-<version>` file to begin installation. The Welcome to the ElectricCommander Installer screen appears.

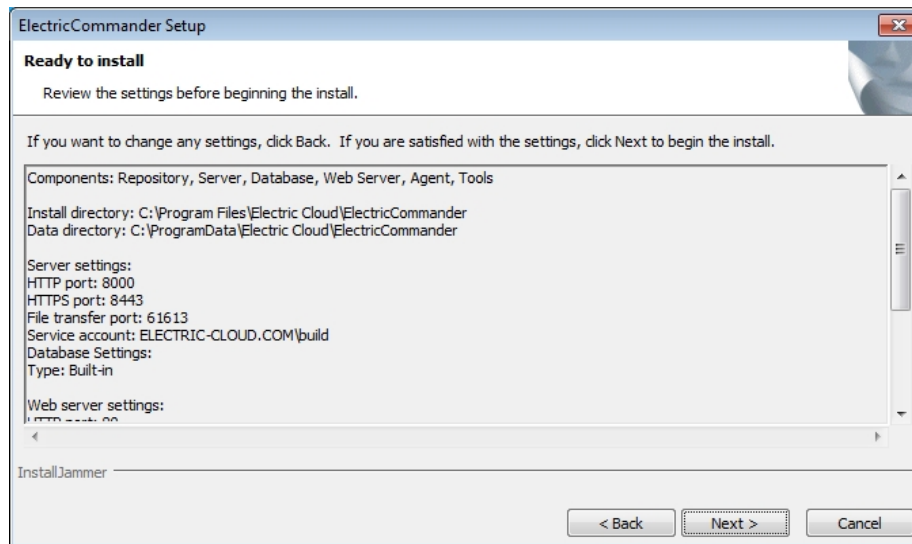


2. Choose one of the following options:
 - Select **Upgrade the existing installation** if you want to upgrade your current Commander installation directory.
 - Select **Perform a clean install** if you want to specify a different installation directory for the new version.

IMPORTANT: During a clean installation, current services remain running until you click **Next** on the Ready to Install screen. This means the new installation cannot use the same ports and directories as the existing installation. To use the same ports and directories, you must manually stop the existing services. This will free the existing ports and directories.

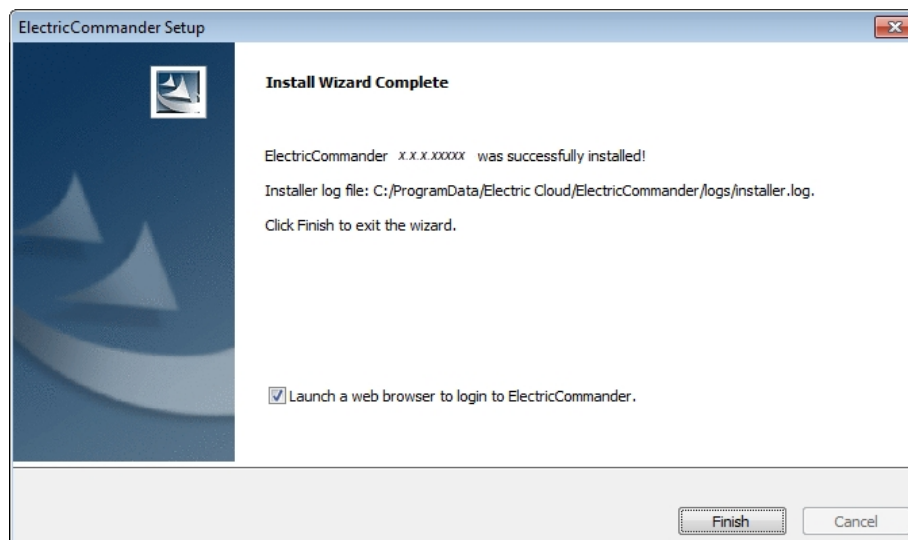
3. Click **Next** to upgrade the existing installation.

The Ready to Install screen appears.



4. Review the upgrade settings. Use the Back button to change your selections if necessary.
5. Click **Next** to continue.

The installer displays a status bar to show the progress of the upgrade process. The time that it takes to complete this process depends on the size of the database. It may take fifteen minutes or longer to complete. Once this process is complete, the new Commander version is installed.

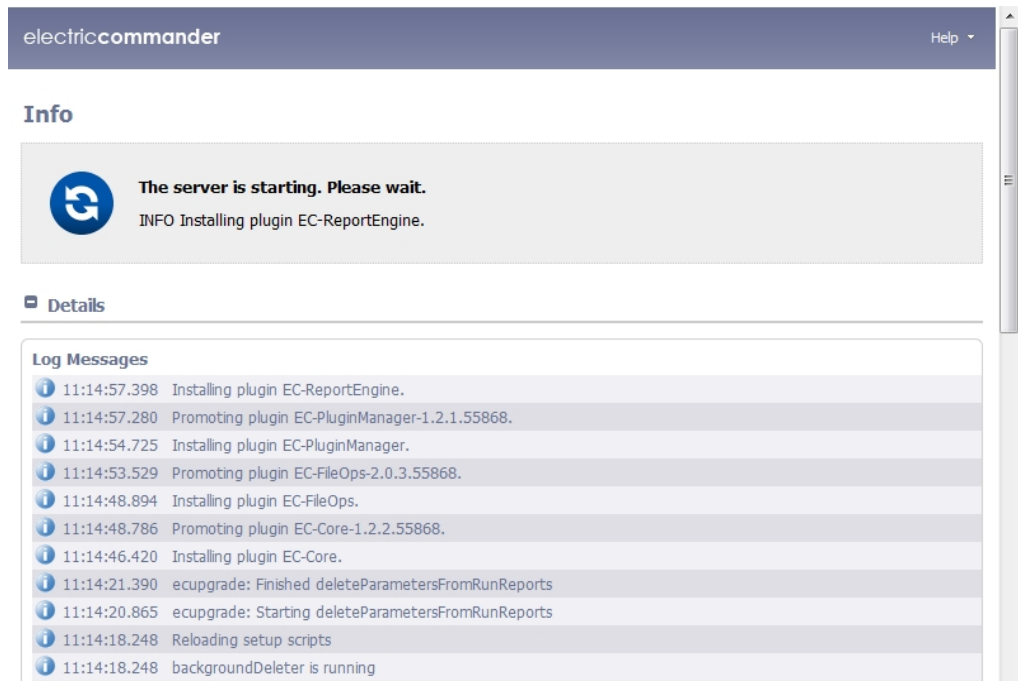


6. Select the **Launch a web browser to login to ElectricCommander** check box if you want Commander to open the login screen now.
7. Click **Finish** to complete the upgrade.

If a Commander server is being upgraded, when the installation is complete the server will continue to upgrade the database (if applicable). You will not be able to log in to the Commander server until the database upgrade is complete. You can view the upgrade status by using `ectool` from a command line:


```
ectool getServerStatus.
```

After clicking **Finish**, you may see a web page similar to the following screen if the upgrade is still in progress.



Interactive Command-Line Upgrade Method

Use the following procedure to complete a command-line upgrade of a Linux platform. Review [Preparing for the Upgrade on page 7-13](#) before performing this procedure.

1. Choose one of the following commands to begin the upgrade:
 - If you have a Linux platform, enter `./ElectricCommander-<version> .`
 - If you have a Linux platform with the X Window System, the installer will automatically bring up the graphical user interface. To override this behavior, enter `./ElectricCommander-<version> --mode console.`

The system displays the following message:

```
Copyright (c) 2006-2014, Electric Cloud, Inc. All rights reserved.
This will install ElectricCommander on your computer. Continue? [n/Y]
```

2. Enter: `y`

The System displays the following message:

```
Upgrade the existing <version> installation to version <version>? [n/Y]
```

3. Choose one of the following options:

- If you want to upgrade your current Commander installation directory, enter `y`.
- If you want to specify a different installation directory (clean install) for the new version, perform the following steps:

a. Enter `n`.

The system displays a prompt for a clean installation. During a clean installation, current services remain running until the install process starts. This means the new installation cannot use the same ports and directories as the existing installation. To use the same ports and directories, you must manually stop the existing services. This will free the existing ports and directories.

b. Enter `y` and complete the information for the subsequent prompts.

The system displays the following message:

```
Installing agent...
Installing server...

Copied log file to "/opt/electriccloud/electriccommander/logs"
ElectricCommander <version> was successfully installed!
Installer log file: /opt/electriccloud/electriccommander/logs/installer.log.
```

Silent (Unattended) Upgrade Method

You can run the Commander upgrade in unattended (silent) mode with no user interface for either Windows or Linux.

Enter the following command from a command line to begin a silent upgrade:

```
./ElectricCommander-<version> --mode silent
```

Chapter 10: Uninstalling ElectricCommander

This section contains instructions for uninstalling Commander from various types of platforms.

Uninstalling Commander on Windows

Choose one of the following procedures to completely uninstall Commander from a Windows server, web server, agent, or developer machine.

Uninstalling Windows 2008 or Windows 7

Use this procedure to completely uninstall Commander from a Windows 2008 or Windows 7 machine.

1. Go to **Control Panel > Uninstall a program**.
2. Select **ElectricCommander**.
3. Click **Uninstall**.

The system displays an "uninstall complete" message when the Commander software is removed.

4. Check the `<install_location>\electriccloud\electriccommander` directory and `C:\ProgramData\electriccloud\electriccommander` directory for any files that might remain. The uninstaller does not remove files that have been created or modified by Commander after the initial installation is complete.
5. Remove the files if you will reinstall the Commander software. You might want to move the files to a new location if you need to retain the files for troubleshooting.

Uninstalling Windows XP or Windows 2003

Use this procedure to completely uninstall Commander from a Windows XP or Windows 2003 machine.

1. Go to the **Control Panel**.
2. Double-click **Add or Remove Programs**.
3. Find and select **ElectricCommander**.
4. Click **Remove**.
5. Check the `<install_location>\electriccloud\electriccommander` directory and `C:\ProgramData\electriccloud\electriccommander` directory for any files that might remain. The uninstaller does not remove files that have been created or modified by Commander after the initial

installation is complete.

6. Remove the files if you will reinstall the Commander software. You might want to move the files to a new location if you need to retain the files for troubleshooting.

Uninstalling Commander on UNIX

Use this procedure to completely uninstall Commander from a UNIX server, web server, agent, or developer machine.

1. Log in as **root**.
2. Choose one of the following commands to run:

- For Linux, enter:

```
/opt/electriccloud/electriccommander/uninstall
```

- For any other UNIX platform, enter:

```
/opt/electriccloud/electriccommander/uninstaller/uninstall
```

The system displays the following message.

```
This will completely remove ElectricCommander from your system. Are you sure you  
want to do this? [y/N]
```

3. Enter **y** to uninstall the software.

The system displays the following messages.

```
Uninstalling ElectricCommander...
```

```
Uninstall complete.
```

4. Check the `/opt/electriccloud/electriccommander` directory for any files that might remain. The uninstaller does not remove files that have been created or modified by Commander after the initial installation is complete.
5. Remove the files if you will reinstall the Commander software. You might want to move the files to a new location if you need to retain the files for troubleshooting.

Chapter 11: Maintenance

This section contains common maintenance procedures.

Commander Server Backups

You should back up your existing Commander data frequently. We recommend full regular (nightly) database backups and database backups before an upgrade.

Data Backup Methods

There are two ways to back up your data. You can use a database-specific backup tool to create a database dump, or you can use the Commander (`ectool export`) tool to create a complete XML database backup. This section describes the differences between the two types of backups.

IMPORTANT: In ElectricCommander 4.0 and later, we recommend that you do not use the `ectool export` tool with jobs on an active system to create a trusted database backup.

Database Dumps

You must use a database-specific backup tool to create a database dump. Database dumps have the following characteristics:

- The backup process takes much less time to complete than full XML exports
- Database dumps (for example MySQL) must be performed while the database is live, up and running.
- You can quickly restore a database from a database dump.

IMPORTANT: A database dump can only be restored to the same type of database. If you are planning to switch your database type when you restore from the backup, you must create an XML backup.

Complete XML Database Backup

You must use the `ectool export` tool to create a complete XML database backup. Complete XML database backups have the following characteristics:

- The tool must be used while the Commander server is running.
- The database backup process can take considerably longer than simply creating a database dump, but this method is necessary in the following situations:
 - Backing up the database is not an available option.
 - You need to migrate from one type of database to another. For example, MySQL to Oracle.
 - You want a full export in a text form you can search with an editor.

Note: While it might not be feasible to run a full XML export on a regular basis (for example, nightly), it is much faster to export all data except jobs by using the `--excludeJobs` option. We recommend doing job-less exports if the system is active (running jobs). For more information about the `ectool export` command, see the “Using ectool and the Commander API” help topic in the Commander online help system.

Preparing for a Backup

Before you backup your Commander server:

- Make sure you have plenty of free space available because full database dumps and XML export files can be extremely large.
 - Compress database dumps if they are not compressed by default.
- Regularly perform maintenance such as, but not limited to:
 - Pruning job workspaces
 - Deleting or compressing Commander log files
 - Deleting unused projects and/or procedures

Note: You can use Commander to perform backups by creating a procedure that runs the database dump or export command.

Backing Up a Commander Server

Use the following procedure to back up your Commander server data. Review [Preparing for a Backup](#) on page 11-3 before performing this procedure.

1. Choose one of the following methods to back up your data:

- Use a database-specific backup tool to create a database dump.
- Use `ectool export` to create a complete XML database backup.

Note: For more information about database backup methods, see [Data Backup Methods](#) on page 11-1.

2. Save the `passkey` file located in one of these directories:

`/opt/electriccloud/electriccommander/conf/passkey` in Linux or

`C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows. When you restore your server, this passkey must be in place so Commander can decrypt passwords for user impersonation, LDAP, and the database connection.

3. Back up the plugins directory.

- The plugins directory is stored in a server setting property (`/server/settings/pluginsDirectory`).
- If the property does not exist, the server uses the default location, which is the `plugins` subdirectory in the data directory.

4. Back up the files containing your custom configurations and settings to ensure all important settings are saved.

- The default location for Commander server and agent configuration files is the `conf` subdirectory in the data directory.
- The default location for the Apache web server configuration files is the `apache/conf` subdirectory in the data directory.
- The default location for the local MySQL database configuration file, `my.ini`, is the `mysql` subdirectory in the data directory.

5. Verify that your backup contains the following items:

- Database dump and/or XML export
- The `passkey` file
- The contents of the plugins directory
- Configuration files
- Keystores

Commander Server Restores

This section describes common restore related procedures for recovering Commander data.

Preparing for a Restore

Before you attempt to restore a Commander server:

- You *must* have a backup of your source Commander server.
 - If you are restoring your data to the exact same database or the same database type (for example, from one MySQL database to another MySQL database on a different system), a database backup is sufficient.
 - If you are switching to a different database type, you will need an XML export.
- **Note:** Any activity on the source server *after the backup was created* will not exist on the destination server.
- The destination system must have a Commander server already installed and running, and this server must be running the same version or newer version than the source server.

Restoring Your Commander Server

The following section contains various procedures for restoring Commander data. Review [Preparing for a Restore](#) on page 11-4 section before performing any of the following procedures.

Note: All ectool commands used in the following scenarios are fully documented in the Commander online help system. See the “Using ectool and the Commander API” help topic.

Restore the Same Commander Server and Database

Use the following procedure to restore your Commander server due to a catastrophic failure or unsuccessful upgrade.

1. Obtain a backup of the source system.
2. Stop the destination Commander server. For more information, see [Starting and Stopping Servers and Agents Manually](#) on page 11-12 for platform-specific commands.
3. If you are using a database dump (where the source and destination systems must both be using the same type of database), load the backup into the destination database.

This will be done with a command specific to the database you are using.
4. Start the destination Commander server.
5. If you are using an XML export file, use the `ectool import` command to import the data into the destination Commander server.
6. Use the `ectool shutdownServer --restart 1` command to restart the destination server.

Keep the Same Commander Server but Switch the Database

Use the following procedure to restore your Commander server if you are doing one of these tasks:

- Switching from the built-in database installation to an external database.
 - Upgrading to a higher performance system for the database.
1. Obtain a backup of the source system.
 2. Stop the destination Commander server. For more information, see [Starting and Stopping Servers and Agents Manually](#) on page 11-12 for platform-specific commands.
 3. Stop and disable the original database.
 4. If you are using a database backup (where the source and destination systems must both be using the same type of database), load the database dump into the destination database.

This will be done with a command specific to the database you are using.

5. Start the destination Commander server.
6. Set the server database configuration to point to the new database. Point to the new database one of these ways:

See the “Database Configuration” help topic in the Commander web interface.

Use the `ectool setDatabaseConfiguration` command.

7. If you are using an XML export file, use the `ectool import` command to import the data into the destination Commander server.
8. Use the `ectool shutdownServer --restart 1` command to restart the destination server.

Switch the Commander Server but Keep the Same Database

Before switching the server, be aware of the following:

- All files and directories copied to the Destination Commander Server should be owned by user configured to run Commander server daemon.
- Make sure that the host name of local agent is set to `127.0.0.1` using **Cloud > Resources > Local > Resource Details**.
- When you install ElectricCommander without a built-in database, you can configure the database only by using `ectool`.

Use the following procedure to restore ElectricCommander if you are upgrading to a higher performance Commander server system.

1. Make sure you have a backup of the source system.
2. Check the IP Address System property by selecting **Administration > Server > Settings** on the old (source) Commander system.

This field is empty by default to enable dynamic connections between the Commander server and agents.

If the field is not empty, you must enter the IP address for the Destination Commander Server.

3. Stop the destination Commander server.

For more information, see [Starting and Stopping Servers and Agents Manually](#) on page 11-12 for platform-specific commands.

4. Stop and disable the source Commander server.
5. Copy the `passkey` and `keystore` files from the source Commander backup to the destination system in one of these directories: `/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows.
6. Copy the backed-up plugins to the destination system.

You may encounter one of these scenarios:

- If the `/server/settings/pluginsDirectory` property does not exist, the server uses the default location (the `plugins` subdirectory in the data directory).

Copy the backed-up plugins to that directory on the destination system.

- The plugins are stored in a local directory valid on both systems.

Copy the backed-up plugins to the same directory on the destination system.

- The plugins are stored in a shared directory valid on both systems.

You do not need to do anything.

- The plugins are stored in a directory not accessible on the destination system.

This can happen

- If the source and destination systems have different operating systems (such as Windows to Linux).
- If the plugins directory on the source system is on a drive that does not exist on the destination system.

Copy the backed-up plugins to a new directory accessible to the destination system. When the server starts, set the `/server/settings/pluginsDirectory` property to the new directory and restart the Commander server.

7. If you use MySQL database, do these steps on destination system:
 1. Install MySQL JDBC driver: http://docs.electric-cloud.com/commander_doc/4_2_6/HTML5/Install/Content/ec_install_guide/installing_ec/10MySQL%20Driver.htm.
 2. Configure access to the Commander database user from the IP address or FQDN on the destination system.
8. Start the destination Commander server.

9. Because the Commander host changed, connect the ElectricCommander database to the new host:

On the command-line:

1. Use `ectool setDatabaseConfiguration` to specify a database configuration and set the `--ignoreServerMismatch` option.
2. Use the following command to restart the destination server: `ectool shutdownServer --restart 1`.

In the web interface, you should automatically be redirected to the **Database Configuration** page.

1. Enter the appropriate database configuration.
 2. Select the **Ignore server hostname mismatch** check box.
 3. Select **Same instance on a new host**.
 4. Click **Save and Restart**.
10. If you copied the plugins directory to a directory that does not match the plugins directory on the source system:
 1. Set the `/server/settings/pluginsDirectory` property to this new directory.
You can use the `ectool setProperty` command to set this value.
 2. Restart the Commander server.

Switch Both the Commander Server and Database

Use the following procedure to restore Commander if you are upgrading to higher performance systems for both the Commander server and the database.

1. Make sure you have a backup of the source system.
2. Check the IP Address System property by selecting **Administration > Server > Settings** on the old Commander system.

This field is empty by default to enable dynamic connections between the Commander server and agents.

If the field is not empty, you must enter the IP address for the new Commander server.
3. Stop the destination Commander server.

For more information, see [Starting and Stopping Servers and Agents Manually](#) on page 11-12 for platform-specific commands.
4. Stop and disable the source Commander server.
5. Stop and disable the original database.
6. Copy the `passkey` file from the backup to the destination system in one of these directories:
`/opt/electriccloud/electriccommander/conf/passkey` in Linux or
`C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows.

7. Copy the backed-up plugins to the destination system.

You may encounter one of these scenarios:

- If the `/server/settings/pluginsDirectory` property does not exist, the server uses the default location (the `plugins` subdirectory in the data directory).

Copy the backed-up plugins to that directory on the destination system.

- The plugins are stored in a local directory valid on both systems.
Copy the backed-up plugins to the same directory on the destination system.

- The plugins are stored in a shared directory valid on both systems.

You do not need to do anything.

- The plugins are stored in a directory not accessible on the destination system.

This can happen:

- If the source and destination systems have different operating systems (such as Windows to Linux).
- If the plugins directory on the source system is on a drive that does not exist on the destination system.

Copy the backed-up plugins to a new directory accessible to the destination system. When the server starts, set the `/server/settings/pluginsDirectory` property to the new directory and restart the Commander server.

8. If you are using a database backup (where the source and destination systems must both be using the same type of database), load the database dump into the destination database.

This operation is completed with a command specific to the database you are using.

9. Start the destination Commander server.

10. Because we have replaced the `passkey`, the database password is no longer valid. You need to reset the database password (default: `commander`) and ignore the `passkey` mismatch either from the command-line or the web interface.

- On the command-line, use `ectool setDatabaseConfiguration` to specify the password and set the `--ignoreServerMismatch` and `--ignorePasskeyMismatch` options.
- In the web interface, you should automatically be redirected to the **Database Configuration** page. Enter the database password and select the **ignore invalid passkey** check box.

11. If you are using an XML export file, use the `ectool import` command to import the data into the destination Commander server.

12. Use the `ectool shutdownServer --restart 1` to restart the destination server.

13. If you copied the plugins directory to a directory that does not match the plugins directory on the source system, set the `/server/settings/pluginsDirectory` property to this new directory and restart the Commander server.

You can use the `ectool setProperty` command to set this value.

Create a Clone of the Commander Server and the Database

Use the following procedure to restore your Commander server if you are setting up a production-like environment for testing.

1. Make sure you have a backup of the source system.
2. Check the IP Address System property by selecting **Administration > Server > Settings** on the old Commander system.

This field is empty by default to enable dynamic connections between the Commander server and agents.

If the field is not empty, you must enter the IP address for the new Commander server.

3. Stop the destination Commander server.

For more information, see [Starting and Stopping Servers and Agents Manually](#) on page 11-12 for platform-specific commands.

4. Copy the `passkey` file from the backup to the destination system in one of these directories:
`/opt/electriccloud/electriccommander/conf/passkey` in Linux or
`C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows.

5. Copy the backed-up plugins to the destination system.

You may encounter one of these scenarios:

- If the `/server/settings/pluginsDirectory` property does not exist, the server uses the default location (the `plugins` subdirectory in the data directory).

Copy the backed-up plugins to that directory on the destination system.

- The plugins are stored in a local directory valid on both systems.
Copy the backed-up plugins to the same directory on the destination system.
- The plugins are stored in a shared directory valid on both systems.

You do not need to do anything.

- The plugins are stored in a directory not accessible on the destination system.

This can happen:

- If the source and destination systems have different operating systems (such as Windows to Linux).
- If the plugins directory on the source system is on a drive that does not exist on the destination system.

Copy the backed-up plugins to a new directory accessible to the destination system. When the server starts, set the `/server/settings/pluginsDirectory` property to the new directory and restart the Commander server.

6. If you are using a database backup (the source and destination systems must both be using the same type of database), load the database dump into the destination database.

This operation is completed with a command specific to the database you are using.

7. If you are using a database backup, disable schedules, resources, or both on both servers.
 - Two servers should never *talk* to the same agent. The two servers share the same identity because they share exact copies of the database.
 - Disabling schedules prevents jobs from launching unexpectedly.
 - Disabling resources prevents scheduled or manually launched jobs from running on production agents. This operation is completed with a command specific to the database you are using.
8. Start the destination Commander server.
9. Because we have replaced the `passkey`, the database password is no longer valid. You need to reset the database password (default: `commander`) and ignore the `passkey` mismatch either from the command-line or the web interface.
 - On the command-line, use `ectool setDatabaseConfiguration` to specify the password and set the `--ignoreServerMismatch` and `--ignorePasskeyMismatch` options.
 - In the web interface, you should automatically be redirected to the **Database Configuration** page. Enter the database password and select the **ignore invalid passkey** check box.
11. If you are using an XML export file, disable schedules, resources, or both on both servers.
 - Two servers should never “talk” to the same agent. The two servers share the same identity because they share exact copies of the database.
 - Disabling schedules prevents jobs from being launched unexpectedly.
 - Disabling resources prevents scheduled or manually launched from running on production agents.

Disable the schedules and resources one of these ways:

- Modify the import file by replacing `<resourceDisabled>0</resourceDisabled>` with `<resourceDisabled>1</resourceDisabled>`.
 - Use the `ectool import` command with the `--disableSchedules` flag turned on to disable schedules.
12. Use the `ectool shutdownServer --restart 1` command to restart the destination server.
 13. If you copied the `plugins` directory to a directory that does not match the `plugins` directory from the source system, set the `/server/settings/pluginsDirectory` property to the new directory and restart the Commander server.

You can use the `ectool setProperty` command to set this value.

Apache Web Server or Agent Certificates

By default, ElectricCommander generates a temporary self-signed certificate during web server installation. This certificate is used whenever a browser makes an HTTPS connection to the Apache server. Because the certificate is self-signed, browsers will generate untrusted certificate messages. To prevent these types of warnings, you must generate a new Apache web server or agent certificate signed by a recognized certificate authority (CA).

Generating a CA Request

Use the following procedure to generate a CA request.

1. Locate the `$DATA_DIR` directory for your platform. The default directory locations are:
 - Linux - `/opt/electriccloud/electriccommander/conf`
 - Windows 2008 or Windows 7 - `C:\ProgramData\Electric Cloud\ElectricCommander\conf`
 - Windows XP or 2003 - `C:\Documents and Settings\All Users\Application Data\Electric Cloud\ElectricCommander\conf`
2. Locate the appropriate certificate signing request file generated during installation. The request file is located in one of the following directories.
 - Agent - `$DATA_DIR/conf/agent.csr`
 - Web Server - `$DATA_DIR/apache/conf/server.csr`

Sending the CA Request

- Send the `server.csr` (or `'agent.csr'`) file to a certificate authority to sign the certificate.

The CA verifies the information inside and sends you a signed certificate in response. The signed certificate includes the original certificate and the CA signature.

Installing the Signed Certificate

Use the following procedure to install a signed certificate:

1. Replace any existing certificates in the `$DATA_DIR` directory with the new signed certificate you received from the CA. The signed certificate file should be placed in one of the following directories.
 - Agent - `$DATA_DIR/conf/agent.crt`
 - Web Server - `$DATA_DIR/apache/conf/server.crt`
2. Restart the service(s).

Using chkconfig

`chkconfig` is a simple command-line tool for maintaining the `/etc/rc[0-6].d` directory hierarchy. This tool relieves system administrators from the task of directly manipulating numerous symbolic links in those directories. The Linux `chkconfig` command can be used to manipulate Commander services running on UNIX platforms.

`chkconfig` - updates and queries runlevel information for system services

```
chkconfig --list [name]
chkconfig --add name
chkconfig --del name
chkconfig [--level levels] name <on|off|reset>
chkconfig [--level levels] name
```

Examples

```
(list current settings for the local Commander repository service)
/sbin/chkconfig commanderRepository --list
commanderRepository 0:off 1:off 2:off 3:on 4:off 5:on 6:off
```

```
(disable autostart on reboot)
/sbin/chkconfig commanderRepository off
/sbin/chkconfig commanderRepository --list
commanderRepository 0:off 1:off 2:off 3:off 4:off 5:off 6:off
```

Note: For every service, each runlevel has either a “start” script or a “stop” script. When switching runlevels, `init` will not restart an already-started service, and will not re-stop a service that is not running.

Starting and Stopping Servers and Agents Manually

Commander servers and agents need to be manually stopped and started during administrative maintenance, upgrades, third party software installs, or system maintenance.

Stop All Commander Agent Services

To stop all ElectricCommander agent services, choose one of the following options:

- For Windows:
 - a. Go to **Control Panel > Administrative Tools > Services**
 - b. Stop “ElectricCommander Agent”
- From a command line, enter `sc stop CommanderAgent`
- For UNIX / Mac: using a shell, logged in as `root`, enter one of the following commands:
 - Linux - `/etc/init.d/commanderAgent stop`
 - Solaris - `/etc/init.d/ecmdrAgent stop`
 - HP-UX - `/sbin/init.d/ecmdrAgent stop`
 - Mac - `launchctl unload /Library/LaunchDaemons/ecmdrAgent.plist`

Stop All Commander Server Services

To stop all Commander server services, choose one of the following options:

- For Windows:
 - a. Go to **Control Panel > Administrative Tools > Services**
 - b. Stop “ElectricCommander Server”
 - c. Stop “ElectricCommander Web Server”
 - d. Stop “ElectricCommander Database” (if it exists)
 - e. Stop “ElectricCommander Repository Server”
- From a command line, enter:
 - a. `sc stop CommanderServer`
 - b. `sc stop CommanderApache`

- c. `sc stop CommanderMySQL (if it exists)`
- d. `sc stop CommanderRepository`
- For Linux: using a shell, logged in as `root`, enter:
 - a. `/etc/init.d/commanderServer stop`
 - b. `/etc/init.d/commanderApache stop`
 - c. `/etc/init.d/commanderMySQL stop (if it exists)`
 - d. `/etc/init.d/commanderRepository stop`

Start All Commander Agent Services

To start all Commander agent services, choose one of the following options:

- For Windows:
 - a. Go to **Control Panel > Administrative Tools > Services**
 - b. Start "ElectricCommander Agent"
- From a command line, enter `sc start CommanderAgent`
- For UNIX / Mac: using a shell, logged in as `root`, enter one of the following commands:
 - Linux - `/etc/init.d/commanderAgent start`
 - Solaris - `/etc/init.d/ecmdrAgent start`
 - HP-UX - `/sbin/init.d/ecmdrAgent start"`
 - Mac - `launchctl load /Library/LaunchDaemons/ecmdrAgent.plist`

Start All Commander Server Services

To start all Commander server services, choose one of the following options:

- For Windows:
 - a. Go to **Control Panel > Administrative Tools > Services**
 - b. Start "ElectricCommander Database" (if it exists)
 - c. Start "ElectricCommander Server"
 - d. Start "ElectricCommander Web Server"
 - e. Start "ElectricCommander Repository Server"
- From a command line, enter:
 - a. `sc start CommanderMySQL (if it exists)`
 - b. `sc start CommanderServer`
 - c. `sc start CommanderApache`
 - d. `sc start CommanderRepository`
- For Linux: using a shell, logged on as `root`, enter:
 - a. `/etc/init.d/commanderMySQL start (if it exists)`
 - b. `/etc/init.d/commanderServer start`

- c. `/etc/init.d/commanderApache start`
- d. `/etc/init.d/commanderRepository start`

Log File Locations

The following information is for default “run time” log locations.

Agent Logs

Platform	Default Path
Windows XP	C:\Documents and Settings\All Users\Application Data\Electric Cloud\ElectricCommander\logs\agent
Windows 2003	
Windows 2008	C:\ProgramData\Electric Cloud\ElectricCommander\logs\agent
Windows 7	
UNIX	/opt/electriccloud/electriccommander/logs

Note: Agent logs “roll over” periodically so individual logs do not grow too large and older logs are deleted. Roll-over parameters are configurable in `conf/logback.xml` and `conf/agent.conf`.

Server Logs

Platform	Default Path
Windows XP	C:\Documents and Settings\All Users\Application Data\Electric Cloud\ElectricCommander\logs
Windows 2003	
Windows 2008	C:\ProgramData\Electric Cloud\ElectricCommander\logs
Windows 7	
UNIX	/opt/electriccloud/electriccommander/logs

Note: Server logs “roll over” periodically so individual logs do not grow too large and older logs are deleted. Roll-over parameters are configurable in `conf/logback.xml` and `conf/agent.conf`.

Web Server Logs

Platform	Default Path
Windows XP	C:\Documents and Settings\All Users\Application Data\Electric Cloud\ElectricCommander\apache\logs
Windows 2003	

Platform	Default Path
Windows 2008	C:\ProgramData\Electric Cloud\ElectricCommander\apache\logs
Windows 7	
UNIX	/opt/electriccloud/electriccommander/apache/logs

Repository Server Logs

Platform	Default Path
Windows XP	C:\Documents and Settings\All Users\Application Data\Electric Cloud\ElectricCommander\logs\repository
Windows 2003	
Windows 2008	C:\ProgramData\Electric Cloud\ElectricCommander\logs\repository
Windows 7	
UNIX	/opt/electriccloud/electriccommander/logs/repository

Installer Logs

Platform	Default Path
Windows XP	C:\Documents and Settings\All Users\Application Data\Electric Cloud\ElectricCommander\logs
Windows 2003	
Windows 2008	C:\ProgramData\Electric Cloud\ElectricCommander\logs
Windows 7	
UNIX	/opt/electriccloud/electriccommander/logs

Switching to an Alternate Database

If you did not deselect the “database” check box during installation, you can switch to another database at any time. You can use this procedure to switch from the built-in database or to switch from the current alternate database to a different alternate database.

IMPORTANT: If you are using two different Commander servers, they cannot point to the same database.

Use this procedure to configure a new database and migrate the existing data.

1. Export your data by entering the following command:

```
ectool export <filename> --compress 1
```

Note: If running the export on an active system, do not include jobs in the export.

2. Set the database configuration using the web interface or ectool. For more information, see [Configuring Commander to Use an Alternate Database](#) on page 5-2.
3. Restart the Commander server by entering the following command:

```
ectool shutdownServer --restart 1
```

4. Import your data by entering the following command:

```
ectool import <filename> --force 1
```

Web Interface Online Help System

Open the ElectricCommander online help system for more information. Click the **Help** link in the top-right corner of any product web page to see a help topic for that page.

When the help system opens, We recommend reviewing the Help table of contents. All Help folders above the Web Interface Help folder are user-guide style help topics that provide more detailed information on each of their subjects.

If you generally prefer to use a command-line tool rather than the Commander web interface, you will find complete ectool (the Commander command-line tool) and API (perl script) commands and options within the online help system too.

Chapter 12: Troubleshooting

This chapter contains troubleshooting procedures for some of the more common issues you might experience during the Commander installation process. More troubleshooting information can be found in Commander Knowledge Base articles located at <https://electriccloud.zendesk.com/forums/>.

Commander Server is Unresponsive and Displays an OutOfMemory Error

Description:

The Commander server becomes unresponsive and displays an `OutOfMemory` error message that states that the server is out of `PermGen` space. This message occurs when the server is not configured properly for a 64-bit JVM.

Workaround:

Change a setting in `wrapper.conf`.

1. Locate the `wrapper.conf` file.

- The file to edit on Linux is:

```
/opt/electriccloud/electriccommander/conf/wrapper.conf
```

- The file to edit on Windows is:

```
<ElectricCommander data dir>/conf/wrapper.conf
```

- On Windows 7 or 2008, the `data-dir` is typically:

```
C:\ProgramData\Electric Cloud\ElectricCommander
```

- On Windows 2003 or XP, the `data-dir` is typically:

```
C:\Documents and Settings\All Users\Application Data\Electric  
Cloud\ElectricCommander
```

2. Add the following line to the end of the existing "`wrapper.java.additional`" section, being careful to use the next consecutive number.

In the following example, 7 was the last pre-existing sequence number in this section.

```
wrapper.java.additional.8=-XX:MaxPermSize=128m
```

Note: We recommend setting the Java `MaxPermSize` to a number greater than the default (which is 84m) on 64-bit commander server (java) installation.

3. After the file has been edited, restart the server to utilize the new value:
 - On Linux: `/etc/init.d/commanderServer restart`
 - On Windows: use “Services”

Commander Fails to Install on a Virtual Machine (VM)

Description:

A local Commander Server and database installation fails to install on a VM. If the VM Manager already has a high disk or CPU load, the Windows installers for Commander might have problems installing software on a virtual machine (VM). One symptom of this problem is the error message, “Failed to start service CommanderMySQL”.

Workaround:

Try to improve the performance of the VM environment, and then delete the folders where the install was attempted. Rerun the installation.

Note: You might experience performance issues if you attempt to run both the Commander server and the database on a VM.

Windows PHP Does not Handle Time Zones Correctly

Description

PHP does not handle certain operating system time zones correctly on a Windows system. If the web server is running on a machine set for one of these time zones, users connected to that web server will see all times displayed as UTC times, instead of the web server time zone.

Workaround:

In the `config.php` file, you must explicitly set the PHP "timezone_identifier".

To set the timezone:

1. Edit the following file.

```
C:\Program Files\Electric
Cloud\ElectricCommander\apache\htdocs\commander\config.php
```

2. Add the following line anywhere between the opening and closing PHP tags:

```
date_default_timezone_set("<timezone_identifier>");
```

For example:

To set the timezone for Taipei, you would add: `date_default_timezone_set("Asia/Taipei");`

Note: For a complete "List of Supported Timezones", see <http://us2.php.net/manual/en/timezones.php>

Commander Server Certificate Fails Security Scan

Description:

You might need to replace the Commander server certificate if it fails the security scan. There are three relevant configuration entries in the `server/conf/commander.properties` file:

```
COMMANDER_HTTPS_PORT=8443
COMMANDER_KEYSTORE=file:conf/keystore
COMMANDER_KEYSTORE_PASSWORD=abcdef
```

Where:

- `COMMANDER_HTTPS_PORT` configures the SSL port
- `COMMANDER_KEYSTORE` is the location of the java keystore where the Commander HTTP server finds its host certificate
- `COMMANDER_KEYSTORE_PASSWORD` is the password to the keystore

Workaround:

Follow these steps to generate and inject a self-signed certificate for 1 year.

1. Back up the keystore file.

2. Delete the original key.

```
user@USER /cygdrive/c/ProgramData/Electric Cloud/ElectricCommander/conf
$ "c:/Program Files/Electric Cloud/ElectricCommander/jre/bin/keytool" -delete -a
lias jetty -keystore keystore -keypass passkey
Enter keystore password: abcdef
```

3. Generate and inject a new certificate.

```
user@USER /cygdrive/c/ProgramData/Electric Cloud/ElectricCommander/conf
$ "c:/Program Files/Electric Cloud/ElectricCommander/jre/bin/keytool" -keystore
keystore -alias jetty -genkey -keyalg RSA -sigalg MD5withRSA -validity 365
Enter keystore password: abcdef
What is your first and last name?
[Unknown]: localhost
What is the name of your organizational unit?
[Unknown]: <Enter>
What is the name of your organization?
[Unknown]: <Enter>
What is the name of your City or Locality?
[Unknown]: <Enter>
What is the name of your State or Province?
[Unknown]: <Enter>
What is the two-letter country code for this unit?
[Unknown]: <Enter>
Is CN=localhost, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown correc
t?
[no]: yes
Enter key password for <jetty>
(RETURN if same as keystore password): <Enter>
```

4. Restart the server.

Your new certificate will look similar to this:

```
user@USER /cygdrive/c/ProgramData/Electric Cloud/ElectricCommander/conf
$ "c:/Program Files/Electric Cloud/ElectricCommander/jre/bin/keytool" -list -v -
keystore keystore_orig -keypass passkey
Enter keystore password: abcdef
Keystore type: JKS
Keystore provider: SUN
Your keystore contains 1 entry
Alias name: jetty
Creation date: Jan 31, 2012
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=localhost, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown
Issuer: CN=localhost, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown
```



```
Serial number: 4f28603f
Valid from: Tue Jan 31 13:42:23 PST 2012 until: Wed Jan 30 13:42:23 PST 2013
Certificate fingerprints:
MD5: 38:50:CD:29:8C:16:3A:78:29:0F:45:56:E0:CA:42:D9
SHA1: 9B:A3:E4:EA:A7:C0:3A:ED:BF:63:24:18:F0:08:78:22:59:85:BC:8A
Signature algorithm name: MD5withRSA
Version: 3
*****
*****
```

Linux Upgrade Breaks Symbolic Links

Description:

When using the Linux installer to perform an upgrade, you might encounter problems moving broken symbolic links. You might see errors that begin with a line similar to, "could not read "/opt/electriccloud/electriccommander/workspace/FileOperationsLinux-LocalMove-7689/fileSymLink": no such file or directory".

Workaround:

Manually remove the file and rerun the installer if you encounter these types of errors.

Index

A

- access to plugins directory 5-7
- adjust
 - Linux swappiness 5-12
 - server memory use 2-10
- agent
 - certificates 11-11
 - platforms 2-4
- agent services, starting and stopping 11-12
- agents on non-server platforms 3-34
- alternate database 11-15
- apply enterprise license 5-1
- architecture 1-3

B

- backup
 - Commander server 11-3
 - complete backup 11-1
 - database dump 11-1
 - methods 11-1
 - preparation for 11-3
- browser requirements 2-7

C

- certificates 11-11
- challenges solved 1-3
- change to enterprise license 5-1
- changing Commander database 11-15
- checksum utility 2-11
- chkconfig 11-11

- choosing the correct
 - install process 3-1
 - upgrade process 7-11, 8-2, 9-1
- clustering 4-1
 - access Commander 4-22
 - additional availability 4-22
 - architecture 4-1
 - benefits 4-1
 - configuration 4-5
 - considerations 4-2
 - dependencies 4-4
 - failure points 4-22
 - load balancers 4-7
 - single-server mode 4-5, 4-16
 - software 4-4
 - strategies 4-2
- command-line
 - database configuration 5-3
 - upgrade 7-34, 8-8, 9-6
- Commander
 - agent disk space 2-9
 - backing up 11-3
 - change database 11-15
 - database user 5-2
 - login 3-37
 - memory use 2-10
 - ports 2-7

- restoring 11-4
- server backups 11-1
- server disk space 2-9
- server restores 11-4

configure

- clustering 4-5

configuring

- alternate database 5-2
- proxy agents 5-11
- proxy servers 5-10
- proxy settings 5-10

D

- data backup methods 11-1

database

- dumps 11-1
- interactions 5-2
- requirements 2-8
- sizing 2-8
- user 5-2

database configuration

- command line 5-3
- web 5-2

- database.properties file 5-2, 5-6

default

- installation directories 3-3
- ports 2-7

- disk space use 2-9

E

ecconfigure

- set agents 5-11
- set servers 5-10

- ectool, set database 5-3

- enterprise license 5-1

- environment proxy servers 5-10

- external database configuration 5-1

F

- file descriptors 5-12

G

graphical user interface

- advanced install 3-9
- express agent install 3-16
- express server install 3-4
- installation methods 3-4

H

- hardware requirements 2-5

horizontal scalability

- agent separation 4-6, 4-21
- ZooKeeper import 4-17

I

installation

- directories 3-3
- non-server platform agents 3-34
- options 3-1
- prerequisites 3-3
- processes 3-1
- silent unattended 3-27
- with graphical user interface 3-4
- with interactive command-line 3-21
- ZooKeeper 4-7

interactive command-line

- advanced install 3-22
- express agent install 3-25
- express server install 3-21
- installation methods 3-21

L

- licensing 2-11

- Linux swappiness 5-12

- log files 11-14

- log into Commander 3-37

M

- memory use 2-10

MySQL

- characteristics 7-14

- JDBC driver 3-37

N**non-server platform**

- agent installation methods 3-34

- silent install 3-36

- UNIX install 3-34

O

- online help 11-16

- options for installation 3-1

overview

- upgrade process 8-1

P

- platforms for agents 2-4

- plugins directory 5-7

- port use 2-7

- prepare for database backup 11-3

prerequisites

- installation 3-3

- upgrade 7-13, 8-4, 9-3

- proxy agents 5-11

- proxy server 5-10

- configuration 5-10

- test settings 5-11

R

- recommended disk space 2-9

- replicating plugins directory 5-10

requirements

- browser 2-7

- database 2-8

- hardware 2-5

- restoring Commander 11-4

- run time log files 11-14

- running silent install 3-27

S**server**

- backups 11-1

- local configuration 1-4

- platforms 2-1

- remote configuration 1-4

- restore methods 11-4

- server services, starting and stopping 11-12

set database

- with ectool 5-3

setDatabaseConfiguration

- command options 5-4

- procedure 5-3

- settings for memory use 2-10

silent install

- command arguments 3-28

- examples Linux 3-31

- examples Windows 3-33

- methods 3-27

- running 3-27

- silent unattended upgrade 7-35, 8-9, 9-7

- sizing Commander database 2-8

software

- compatibility 2-5

- functionality 1-1

- licenses 2-11

- SQL server authentication 5-3

start and stop

- agent services 11-12

- server services 11-12

supported

- agent platforms 2-4

- databases 2-8

- server platforms 2-1

- swappiness kernel parameter 5-12

- switch to alternate database 11-15

T

test proxy server 5-11

troubleshooting 12-1

types of installations 3-1

U

uninstall Commander

 UNIX 10-2

 Windows 10-1

universal access to plugins 5-7

UNIX installer for agents 3-34

untrusted certificates 11-11

upgrade

 methods 7-11, 8-1, 8-2, 9-1

 prerequisites 7-13, 8-4, 9-3

 silent unattended 7-35, 8-9, 9-7

 with command-line 7-34, 8-8, 9-6

 with user interface 7-31, 8-5, 9-4

user interface upgrade 7-31, 8-5, 9-4

W

web interface

 database configuration 5-2

Z

ZooKeeper 4-11

 install 4-7