



# ElectricFlow 6.0.5

## Release Notes

July 18, 2016

These Release Notes contain supplemental information about ElectricFlow™, Version 6.0.5.

Topics include:

ElectricFlow 6.0.5 .....	2
Product Description .....	2
What's New or Modified .....	2
Release Notes for Previous ElectricFlow 6.0 Versions .....	3
Installation and Upgrade Notes .....	3
Configuration Notes .....	7
Limitations .....	7
Known Issues .....	7
Documentation .....	9

## ElectricFlow 6.0.5

ElectricFlow 6.0.5 is a maintenance release (MR). For more information about ElectricFlow software release strategy, go to the [ElectricFlow/ElectricCommander Release Strategy Update](#).

## Product Description

ElectricFlow™ is an enterprise-grade DevOps Release Automation platform that simplifies provisioning, build and release of multi-tiered applications. Our model-driven approach to managing environments and applications allows teams to coordinate multiple pipelines and releases across hybrid infrastructure in an efficient, predictable and auditable way.

## What's New or Modified

### Resolved Issues

- `evalDsl` now works when a project is created using a DSL script and the resource name is not specified (and is represented by an empty string). (CEV-11027)
- Generating a workflow definition using `generateDs` now includes formal parameters. (CEV-10917)
- Branch conditions between process steps can now be longer than 255 characters. (CEV-10916)
- A resource pool in an environment can now be changed or removed if the resource pool was renamed while it was still attached to the environment. (CEV-10840)
- Pipeline tasks in DSL scripts now support credential arguments. (CEV-9858)
- The Diagnostics subtab of the Job Details page now correctly displays the diagnostic information when the local workspace is used on the resource to run the job step. (NMB-23814)
- The artifact retrieval operation now works for large artifacts. (NMB-23775)
- After an agent or server is installed, passwords that were used to run the Windows service will be removed from the temporary installation files. (NMB-23734)
- The logback groovy settings now work when messages are logged. (NMB-23700)
- An ElectricFlow agent can be installed using `sudo` with the EC-AgentManagement plugin. (NMB-23639)
- The appropriate error message appears when two projects have the same name. (NMB-23605)
- In the Electric Flow UI and in the platform Home page, the **Change History** links and other clickable objects are now underlined or change color when clicking or mousing over them. (NMB-23386 and NMB-23384)
- Artifacts can have names with double-byte character set (DBCS) or German UTF-8 characters. (NMB-23281)
- `generateDsl` generates valid DSL code for a group. (NMB-23076)
- The libraries have been upgraded to Apache 2.2.31 and PHP 5.6.21 to fix security vulnerabilities. (NMB-22815)
- During a database upgrade, a custom URL (`COMMANDER_CUSTOM_DB_URL` in `database.properties`) can be specified for a specific Oracle database, such as a URL for a `tnsnames.ora` file through the JDBC API. (NMB-22798)

- If a step is inserted into an existing procedure in a DSL file, the step is now added in the specified order. (NMB-22761)
- When the server starts and the `getServerStatus --diagnostics 1` command is executed, the proper output with diagnostic information is now displayed. (NMB-22423)
- An agent will not time out when executing a long running `evalDsl` or any other server API call. (NMB-21832)
- References to MySQL were removed from the `.profile` scripts in UNIX installations. (NMB-18770)
- After canceling the Search results on the Jobs page, the default Jobs page is now displayed instead of the previous Search results or an error message. (NMB-5922)

## Release Notes for Previous ElectricFlow 6.0 Versions

- [ElectricFlow 6.0 Release Notes](#)
- [ElectricFlow 6.0.1 Release Notes](#)
- [ElectricFlow 6.0.2 Release Notes](#)
- [ElectricFlow 6.0.3 Release Notes](#)
- [ElectricFlow 6.0.4 Release Notes](#)

All ElectricFlow documentation is available on the [Electric Cloud Documentation website](#).

## Installation and Upgrade Notes

### IMPORTANT: Product Name Change and Deprecation Notice

To bring a singular focus to the ElectricFlow brand moving forward, the name “ElectricCommander” is being changed to “ElectricFlow”. All of the capabilities you are familiar within ElectricCommander are still available and intact. All changes (involving the new name) are being introduced in a way that ensures backward compatibility. The scripts you’ve written and URL/shortcuts you currently have should work without any changes required. We’ve posted an [FAQ](#) to help answer questions you may have concerning this name change.

The installation documentation refers to the installer using the new ElectricFlow product name. Anywhere the installer is referenced you can also use the ElectricCommander named installer. However, be aware that we intend to stop providing the installer with “ElectricCommander” in the filename after this version.

- For complete installation and upgrade information, see the *ElectricFlow Installation Guide*: [http://docs.electric-cloud.com/eflow\\_doc/FlowIndex.html](http://docs.electric-cloud.com/eflow_doc/FlowIndex.html).
- In `DATA_DIR/conf/wrapper.conf`, change the default prefix for StatsD as follows (NMB-22835):  
 from  

```
wrapper.java.additional.802=-DCOMMANDER_STATSD_PREFIX=commander
```

 to  

```
wrapper.java.additional.802=-DCOMMANDER_STATSD_PREFIX=flow
```
- In the Oracle database, set the `OPEN_CURSORS` parameter to at least 1000 to prevent ElectricFlow from running out of open cursors.

**IMPORTANT:** Before beginning the upgrade process, make sure you have backed up your existing ElectricFlow data.

## Upgrading Your Existing ElectricFlow Environment

- Unlike most ElectricFlow patch-level upgrades, the upgrade from Version 6.0.3 to Version 6.0.4 upgrade includes a minor change to the database schema (specifically, a new column called `non_project_change_tracking_en` in the table `ec_configuration_history`). For all upgrades affecting the database schema, once you have started up an ElectricFlow 6.0.4 server or server node and the database schema has been upgraded, an ElectricFlow 6.0.3 server or server node will no longer run against the revised database schema. (NMB-23346)
- You can only upgrade ElectricFlow 6.0.4 and later to ElectricFlow 6.2 or later.(NMB-23268)
- Upgrades to ElectricFlow 6.x are supported only from ElectricCommander 4.2.x or from ElectricFlow 5.x. Any ElectricCommander systems and servers that are pre-Commander 4.2 must be upgraded to an ElectricCommander 4.2.x release. For upgrade instructions, see the *ElectricFlow Installation Guide*.
- Upgrading to ElectricFlow 6.x from ElectricCommander 4.2.x requires a database upgrade.
- You cannot upgrade the ElectricCommander 4.2.x built-in database to an ElectricFlow 6.x database. If you want to continue using the built-in database in ElectricFlow 6.x, follow the database upgrade procedures described in the *ElectricFlow Installation Guide*.

## Behavior Changes

- Use the `reducedDetailChangeHistory` argument in the `export` API call when exporting a project with Change Tracking enabled but not allow changes to be tracked when ElectricFlow exports part or all server data to an XML file. Follow these usage guidelines (CEV-7785):
  - The argument is a Boolean argument: `<Boolean flag - 0|1|true|false>`.
  - Use this argument for large projects containing over 20,000 audited objects with Change Tracking enabled.
  - When this argument is set to `true` or `1`, ElectricFlow automatically decreases the amount of Change History indexing information that it saves in a large project, reducing the level of detail for Change Tracking-intensive operations in the Change History. This can make it harder to revert an object to a specific state and to find information in the Change History when you are troubleshooting or debugging an issue.
  - Set this argument to `false` or `0` to suppress to this behavior so that ElectricFlow does not change the amount of indexing information for a large project. This will cause the operation to take longer and put more load on the database, but the Change History will have the full details of the entities owned by objects in the project.

- Use the following API calls to attach a parameter to a procedure step, application process step, or workflow state (CEV-8408):

To attach a parameter to a procedure step, use an API call such as:

```
ectool attachParameter testParam paramProcedure myStep credparam1
```

To attach a parameter to an application process step, use an API call such as:

```
ectool attachParameter --projectName default --formalParameterName credparam1
--applicationName myApp --processName echoHello --processStepName myStep
```

To attach a workflow state to a workflow state, use an API call such as:

```
ectool attachParameter --projectName default --formalParameterName credparam1
--workflowDefinitionName myworkflow --stateDefinitionName mystate
```

- Added the `smartDeploy` argument to the `runProcess` command to enable Smart Deploy when the application process is deployed. (CEV-9871)
- Added a security fix for Cross-Site Request Forgery (CSRF) Protection. (NMB-20928)

CSRF is an attack that forces end users to execute unwanted actions on a web application in which they are currently authenticated. For more information about CSRF, go to **Automation Platform Basics > Setup Tasks > Security Issues** in the [ElectricFlow 6.0.5 User Guide](#). CSRF protection is disabled by default.

You can enable and disable CSRF protection as follows:

- To enable CSRF protection, enter `ecconfigure --webCsrProtection=true`.
- To disable CSRF protection, enter `ecconfigure --webCsrProtection=false`.
- ElectricFlow 6.0 no longer supports these server and agent platforms (NMB-21606):
  - All 32-bit platforms for servers only
  - Windows XP (32-bit and 64-bit) for servers
  - Microsoft Windows 2000 for agents
- Generating a DSL Script for an Existing ElectricFlow Object (NMB-22760)

To generate a DSL script for an existing ElectricFlow object, which was created through a Perl API, RESTful API, or the UI, enter `ectool generateDsl <path>` where `<path>` is the path to the ElectricFlow object for which you want to generate the DSL script.

`generateDsl` will now create a DSL script in a more concise form and follow a more idiomatic groovy coding style that a DSL script author might use.

For an example, go to **Using the ElectricFlow DSL > Getting Started with DSL > Generating a DSL Script for an Existing ElectricFlow Object** in the *ElectricFlow User Guide*.

- Passing Parameters or Arguments to Your DSL Script (NMB-22870)

ElectricFlow DSL allows you to create a DSL template script using script parameters instead of hard-coding all the values in the script. You can then invoke the same script with different parameter values each time to create different instances of ElectricFlow objects. For example, you have the following script to create a resource that uses SSL in the secure zone:

```
zone 'secure'
resource() {
    resourceName = args.resourceName
    hostName = args.resourceIP
    hostType = 'CONCURRENT'
    resourceDisabled = '0'
    trusted = '1'
    useSSL = '1'
    zoneName = 'secure'
}
```

The script has the values `args.resourceName` and `args.resourceIP` for the `resourceName` and `hostName` resource attributes, respectively. These argument or parameter values can be passed from the command line to the DSL script in JSON form using the following commands:

- For Linux:

```
ectool evalDsl --dslFile myScript.dsl --parameters '{"resourceName": "MyFirstResource", "resourceIP": "192.168.10.12"}'
```

- For Windows:

```
ectool evalDsl --dslFile myScript.dsl --parameters "{\"resourceName\": \"MyFirstResource\", \"resourceIP\": \"192.168.10.12\"}"
```

Note the special handling required on Windows for passing command-line arguments that contain double-quotes and spaces. To allow spaces and other special characters in a command-line argument, Windows requires wrapping the value in quotes. In addition, if the value itself contains quotes, you need to escape those quotes using a backslash `\`. An alternate method for passing parameter values for DSL is to use a file that contains the parameters in JSON format with the following command:

```
ectool evalDsl --dslFile myScript.dsl --parametersFile myParams.json
```

Where the file `myParams.json` may contain:

```
{
  "resourceName" : "MyFirstResource",
  "resourceIP" : "192.168.10.12"
}
```

- The `service.log` file produced by the Java Service Wrapper has been renamed to start with a prefix for the name of the component that the service is running, such as `commander-service.log`, `repository-service.log`, or `jagent-service.log`. (NMB-23201)
- Ensure that a swap is added to the Linux instance where the repository server is running. (NMB-23463)
- The default built-in database for ElectricFlow is now HyperSQL Database (HSQLDB). In releases earlier than ElectricFlow 5.0, the built-in database was H2.

## Configuration Notes

- When you export your project data before upgrading from ElectricCommander 4.2.x to ElectricFlow 6.x, you must replace the component plugin versions, including EC-Artifact, in the export file before importing the project data to ElectricFlow 6.x (CEV-6679).
- You must register your plugin to display it as an option in the following situations (CEV-3649).
  - When a user uses the plugin to configure a step in a component or application process.
  - In a procedure in the automation platform.

For details, see the “Register your procedure for the step creation dialog” section in the “Examples and Tutorials” chapter of the *ElectricFlow Plugin Developer Guide* at [http://docs.electric-cloud.com/eflow\\_doc/FlowIndex.html](http://docs.electric-cloud.com/eflow_doc/FlowIndex.html).

## Limitations

These are the session management limitations:

- When a user logs out, the user is logged out only on the node where the logout occurred.
- When a user is deleted from the system, the user’s session is active until it expires.
- When a job ends, the user’s session is active until it expires.

## Known Issues

- When you are importing a previously exported application from ElectricFlow 5.4 to ElectricFlow 6.x and the application has parameters with options, the application process parameters that you defined in ElectricFlow 5.4 need to be recreated. (CEV-7788)
- An error occurs in the following scenario (CEV-7890):
  1. In ElectricFlow 5.4, attach credentials to a component process or a component process step.
  2. Export the application that contains the component process.
  3. Upgrade to ElectricFlow 6.x.
  4. Import the application to ElectricFlow 6.x.

When you deploy the application process that contains the component process, the error occurs.

Workaround:

1. View the details of the component process step where you previously set the credentials in the Edit Step dialog box.
  2. Click **Next**.
  3. Click **OK** to close the dialog box.
  4. Redeploy the application process.
- The following entry in the wrapper.conf file might cause performance slowdowns (a gradual slowdown of everything over time), and should therefore be deleted (NMB-19735):

```
wrapper.java.additional.105=-XX:+TieredCompilation
```

- When you are adding a resource to a remote ElectricFlow server during an agent installation, the server does not discover the host name of the agent machine through DNS, and an error message about the “Name or service not known” appears. (NMB-20605)

The workaround is to do one of the following so that the resource is available after the agent installation:

- Add the host name of the agent machine to the host file of the remote server.
  - In the Resource Details panel, edit the Agent Host Name of your resource and use the IP address of the agent machine instead of the fully qualified domain name (FQDN).
- If this sequence of events occurs (NMB-21278):
    1. Changes are made to the list of credentials that are attached to a procedure, component, process, process step, or a schedule while change tracking is disabled at either the project level or the server level.
    2. Change tracking is enabled.
    3. The procedure, component, process, process step, or a schedule is reverted to a point after change tracking was enabled.

Then the changes that were made while change tracking was disabled may be lost.

- The `evalDsl` API fails if a DSL script contains strings that have more than 65,535 characters. The `generateDsl` API can generate such a DSL script if step commands have more than 65,535 characters. In this case, you need to update the DSL script and use a file to load the required content.(NMB-23816)
- If you delete the default project and do not recreate it, ElectricFlow will no longer be available. To use it again, you must reinstall ElectricFlow.
- Before importing an export file, you must change the plugin name, including the plugin version, in the file.
- Change tracking

**IMPORTANT:** It can take a while to restart the ElectricFlow server, because new records are being created for all the tracked objects. This may take at least as long as it would take to export or import all the projects (a large project can take long as 10 to 40 minutes).

- You can revert changes only for high-level design objects such as applications, procedures, procedure steps, workflow definitions, and state definitions.
- When you disable change tracking and then later re-enable it, the system performance may be reduced during this sequence of events:
  1. Change tracking is disabled at the server level.
  2. Change tracking is re-enabled at the server level.

The change history for all objects, including those not in projects, is now tracked.

It can take a while to restart the ElectricFlow server, because new records are being created for all the tracked objects. This may take at least as long as it would take to export or import all the projects (a large project can take long as 10 to 40 minutes).

- Pages in the ElectricFlow UI may be slow to render if the application or environment has too many tiers.

## Performance and Scalability Issues

- For hundreds of parallel job steps, you may experience Job Scheduler performance issues at job startup (NMB-16185).
- The amount of time needed to add a property to a job increases as the number of properties increases (NMB-16120).
- The amount of time needed to add a step to a procedure increases as the number of steps increases (NMB-16118).

## Documentation

ElectricFlow documentation is available at [http://docs.electric-cloud.com/eflow\\_doc/FlowIndex.html](http://docs.electric-cloud.com/eflow_doc/FlowIndex.html). Updated documentation will be available on that page when any documents are updated post-release.

ElectricFlow has the following product documentation:

- *ElectricFlow User Guide*
- *ElectricFlow Installation Guide*
- *ElectricFlow API Guide*
- A complete, robust online help system. Click any Help link in the upper-right corner of each web page in the platform UI.
- A standalone Help system is available for viewing outside the product at [http://docs.electric-cloud.com/eflow\\_doc/FlowIndex.html](http://docs.electric-cloud.com/eflow_doc/FlowIndex.html).
- *ElectricFlow Help Guide*
- *ElectricFlow Release Notes*
- Additional ElectricFlow documentation that accompanies the ElectricFlow release, but not necessarily updated with each ElectricFlow release
  - *Plugin Developers Guide*, which is used with the ElectricFlow SDK, is updated on its own release cycle
  - *ElectricFlow Plugin Developer Release Notes*



# Troubleshooting and Getting Help

## Technical Support

Contact Electric Cloud technical support:

- 408.419.4300, option 2. Hours are 9 A.M.–5 P.M. PT Monday–Friday (except holidays)
- [support@electric-cloud.com](mailto:support@electric-cloud.com)

You will be asked to provide the following information:

- Your name, title, company name, phone number, and email address
- Operating system and version number
- Product name and release version
- Problem description

## Electric Cloud "Ask" Website

Go to <http://ask.electric-cloud.com>:

- Ask questions or read answers to questions from other users
- Get help with installation and configuration
- Submit feedback

Copyright © 2002–2016 Electric Cloud, Inc. All rights reserved.

Electric Cloud® believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." ELECTRIC CLOUD, INC. 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.

### Trademarks

Electric Cloud, ElectricAccelerator, ElectricCommander, ElectricFlow, 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.

All other trademarks used herein are the property of their respective owners.