This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Composer Help

Upgrading Projects and Diagrams

4/3/2025

Upgrading Projects and Diagrams

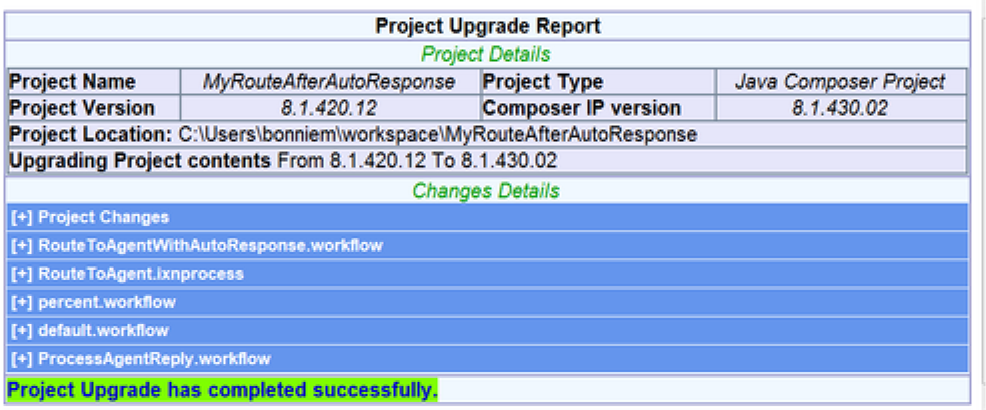
Contents

When deciding whether to upgrade, consult the Composer 8.1.x Release Note for a summary of new features and other updates.

> **Important**
>
> Composer does not support upgrading diagram files from 8.0.4 versions to 8.1.2 or higher versions. If a callflow/workflow diagram upgrade is required, first upgrade the Projects to 8.1.1 versions and then upgrade to 8.1.2 or higher versions.

Project Upgrade Report

Introduced in 8.1.400.33. Whenever a Project is imported into the workspace as part of the process of upgrading to a newer version, you must perform a Project Upgrade. Composer applications will not work or work unpredictably unless the Project is successfully upgraded. Right-click the Project and select **Upgrade Composer Project.** After the upgrade completes, a Project Upgrade report appears in the design area. An example is shown below.



Project Upgrade Report

Project Details

Project Name	*MyRouteAfterAutoResponse*	**Project Type**	*Java Composer Project*
Project Version	*8.1.420.12*	**Composer IP version**	*8.1.430.02*

Project Location: C:\Users\bonniem\workspace\MyRouteAfterAutoResponse

Upgrading Project contents From 8.1.420.12 To 8.1.430.02

Changes Details

[+] Project Changes

[+] RouteToAgentWithAutoResponse.workflow

[+] RouteToAgent.ixnprocess

[+] percent.workflow

[+] default.workflow

[+] ProcessAgentReply.workflow

Project Upgrade has completed successfully.

> **Important**
>
> Other than when pooling reusable subflows, accessing system resources (include/jsp) across Projects is not supported.

Java Composer Projects

Java Composer Projects were referred to as Java Voice projects in earlier versions of Composer, such as Composer Voice. While working with the current version of Composer, an upgrade is required for a previously-created Composer Project and Project diagrams. If you simply copy diagrams into a new Composer Project instead of upgrading the Project itself, then you must use the diagram upgrade procedure as described below. Genesys recommends that you create a dedicated workspace for 8.1 Projects and do not reuse the previously created workspace. This will provide a clean separation between the two versions as well as ensure that a backup copy is preserved for later reference or rollback.

Upgrade Summary

A summary of the Composer diagram upgrade process is as follows:

1. Obtain Composer 8.1 through Genesys Technical Support.
2. Uninstall the older version of Composer. Before uninstalling the older version of Composer:
    - Make a copy of your Composer workspace folder (which contains all your Project files), as your workspace may be deleted if it is located under the installation directory (`C:\Program Files\GCTI\Composer 8.1\workspace`).
    - Uninstall the older version of Composer.
3. Install Composer 8.1.
4. Upgrade at a Project level or at the Diagram level as described below.

Routing Upgrade Limitations

See IRD To Composer Migration Guide.

Project Upgrades

A Project-level upgrade will automatically apply diagram-level upgrades for all the diagram files directly residing within the diagram (Callflows or Workflows) folder. As part of the upgrade process, Composer makes a back-up of the Project and its files, which are saved under the archived folder; for example: `./JavaComposerProject/archive/JavaComposerProject20100809184446388.zip`

Note: If you are using your previous Workspace, importing Projects is not required. For a new Workspace folder, Projects have to be imported.

> **Important**
>
> Composer does not support importing projects directly from network mapped drives. Instead, you can use shared drives to load and deploy Composer projects into the Tomcat server.

To upgrade a Project when using a new Workspace folder:

1. Import an old Composer Project into Composer's Project Explorer view. From the menu, select **File** > **Import.**
2. In the Import dialog, navigate to **General** and double-click **Existing Projects into Workspace**.
3. Browse to the Composer Project location and select the Project(s).
4. Mark the checkbox **Copy projects into workspace**.
5. Click **Finish**.
6. In the Project Explorer, select the imported project and type **F5** to refresh.
7. Right-click the imported Project and select **Upgrade Composer Project** from the context menu.
8. If the Project is upgraded, a message appears indicating that it is the current version. Otherwise, a prompt appears asking if you would like to upgrade this Project. Click the **Yes** button to start the upgrade process.
9. View the upgrade report. Once the upgrade process is complete, Composer displays a report. The report is located in the Reports folder of the Project; for example:`C:\Work\Temp1\Gate3IPTest\Reports\UpgradeReport_Gate3IPTest20090513155840979.html`

Project Version Validation

Starting with Release 8.1.410.14, Composer validates the Project version during the **Generate All** and **Upgrade Project** operations. Projects of a version lower than installed version are validated to upgrade before continuing with these operations. Projects of a version higher than the installed version are not allowed to proceed with these operations.

Example Scenarios

Scenario #1

- Create A JAVA/.NET Project in any older version of Composer.
- Uninstall the older version and install Composer 8.1.410.xx.
- Open Eclipse and import the Project.
- Right click the Project and select **Generate All**. The **Finish** button is disabled state. The Generate All dialog shows `Project is not up-to-date, upgrade the Project and then Generate Code`. The error message is `Project version is not up-to-date. Project version should match the Composer IP version`.

Scenario #2

- Create A JAVA/.NET Project in any older version of Composer.
- Uninstall the older version and install Composer 8.1.410.xx.
- Open Eclipse, select the **Project** menu from the menu bar and select **Build Automatically**.
- Import the Project and open the **Problems** view. A warning message appears: `The diagram version is not up-to-date. Consider upgrading the diagram or the enclosing Project.`

Scenario #3

- Create a JAVA/.NET Project in Composer 8.1.410.xx.
- Uninstall 8.1.410.xx and install an earlier Composer 8.1.410.xx version.
- Open Eclipse and perform a **Generate All** or **Upgrade Project**. A warning message appears: `Mismatched project version or Project version is higher than IP version.`

Workbench Project builds will validate Project versions and show any error in the Problems View.

Intra Version Upgrades

Starting with Release 8.1.400.33, Composer Project and Diagram upgrades between minor versions are now supported (for example, 8.1.400.26 to 8.1.400.32). This enables upgrading diagrams to the IP version from both the major and minor versions. An intra-version upgrade for an IPD diagram to the 8.1.400.33 version works as follows:

1. IPD events are categorized to media-specific sets to improve the interaction handling. Please check the Events property for more details on this and the items below.

1. The pre-defined sets are non-editable: **Voice**, **Multimedia**, and **Ixn-less** processing.

1. A **Custom** category can be used for customizing the events.

1. If Events were customized, an upgrade from 8.1.301.01 versions and later would use the Custom category. Older versions will be chosen to one of the pre-defined sets based on the media-specific blocks used in the IPD diagram.

1. If Events are not customized, one of the predefined sets will be selected upon an upgrade.

1. Using the pre-defined sets provides improved future upgrades.

1. In the case where a Composer upgrade overwrites any custom changes, use the **Load Last Revision** option in the Events dialog to reload the selected event to the last revision. This is applicable only for the **Custom** category.

Upgrade Error Message

After a Composer Project upgrade, the Project Upgrade Report may display the following error message: `error while updating the .studio_config.properties file` or `error while` `updating the .composer file`. In this case, permissions for the `.studio_config.properties` and/ or the `.composer` file may be read-only or hidden. To resolve this issue, go to the file system and check for the `studio_config.properties` file and the `.composer` file located under the Composer Project directory. Set the file permissions so that the read-only and hidden file attributes are disabled/ unchecked. Hint: To find where the current Project directory is located, do the following:

1. Go to Composer's Project Explorer view.
2. Right-click the Composer Project.
3. From the shortcut menu, select **Properties** > **Resource** and look for **Location**., for example, Location: `C:\Program Files\GCTI\Composer 8.1\workspace\JavaComposerProject`.

Tip

While upgrading Composer 812 projects to higher versions, if you encounter the error, *The project is corrupted or version higher than the IP version. Project cannot be upgraded.*, check the **.studio_config.properties** file for unwanted timestamp texts and remove them. The **.studio_config.properties** file must not have multiple lines and any extra characters and must only contain the following: `createdVersion=8.1.2xx.xx`.

Diagram File Upgrade

Important

Composer does not support upgrading diagram files from 8.0.4 versions to 8.1.2 or higher versions. If a diagram upgrade is required, first upgrade the Projects to 8.1.1 versions and then upgrade to 8.1.2 or higher versions.

In Composer 8.0.2 and later, diagrams for voice applications are called callflow diagrams whereas in earlier versions of Composer they were called *studio_diagrams*. Follow the steps below if you have only copied older diagram files to a current version of Composer Project (or to an already upgraded Composer Project).

1. In Composer's Project Explorer, select the Project destination folder to where you want the files to be imported, such as the `Callflows` or `Workflows` Project folder.
2. Right-click and select **Import**.
3. In the Import wizard, select the diagram files to import.
4. After the import operation completes, right-click the imported diagram file and select the upgrade option: **Upgrade Callflow Diagram** or **Upgrade Workflow Diagram**.

Changes as a Result of Upgrading

It is important to note the following:

- When upgrading to 8.1.1, references to internal variable names may have to be edited manually. See Variables Project and Workflow, Internal Variables Naming for details and examples. It is recommended that internal variables such as DB Data block database result variables not be used; instead, create User variables to store these results.
- A Project upgrade does not upgrade any custom blocks. When Composer is launched, it checks if any custom blocks need upgrading and upgrades them. There are no manual steps involved.
- When upgrading to 8.0.4/8.1, Project upgrading creates the folder `./WEB-INF/lib`, copies files from `./lib` to `./WEB-INF/lib`, then removes the `./lib` folder from the Java Composer Project.
- When upgrading from 8.0.2, the Entry block variable `_COMPOSER_WSSTUBBING` is renamed`COMPOSER_WSSTUBBING`.
- When upgrading from 8.0.1 to 8.0.2, the Studio Diagram file extension changes from `.studio_diagram` to `.callflow`. For example: `MyDiagram.studio_diagram` changes to `MyDiagram.callflow`.
- To avoid any resulting file name conflict, the diagram upgrade will append a timestamp to the file name only if a .callflow file with the same file name already exists in the same folder; for example: `Main_2010_02_19_123010.callflow`. The Timestamp is of the following format: yyyy_MM_dd_HHmmss
- Starting with 8.0.2, the following callflow blocks contain a mandatory Output Result property: Menu, DB Input, Grammar Menu, Input, Get Access Number, Transfer, Statistics and Record. You supply this property by selecting a variable. Since this property is mandatory; if not supplied, an error occurs in the Problems View when validating the callflow.
- Upgrading to 8.0.2 or higher automatically populates this variable. For example, if the block is a Menu block and the block's name is Main_Menu, upgrading will add a Main_Menu variable to the Entry block (as if you added it manually) and will set the Output Results property to this variable.
- The GVP Next Generation Interpreter does not support the error.badfetch.badxmlpage event. If upgrading a callflow application from an earlier version that listed this event under Supported in its Entry block Exceptions dialog box, you will need to modify that Entry block by removing that event under Supported in the Exceptions dialog box.
- Composer workflow and callflow diagrams do not directly store diagram grid information. This preference is workspace-specific. If you are using a new workspace, you can set this value prior to upgrading Projects and diagrams so that the grid information does not change during the upgrade process.

Note: Workspace preferences can be exported and imported from **File** > **Export** or **Import** > **General** - **Preferences**.

Command Line Code Generation

A command line option is available in Composer to generate code for all diagrams for all Projects in a Workspace.

```
eclipse.exe -application com.genesyslab.composer.voice.generator.commandline.app
-nosplash -console -consoleLog -data .\workspace -options
```

Where:

- You open the command prompt as Administrator.
- You execute the command line application under where `eclipse.exe` is located.
- `.\workspace` is the relative path to the workspace from where the command line is run. You can specify an absolute path.
- Code will be generated for all the diagram files irrespective of the options specified.
- At least one of the below options is specified.

Options

Option	Description	Comment
`-p`	Upgrade the Projects in the workspace including common and diagram files. Mutually exclusive with -u option (recommended option).	**Note**: The `-p` option does not work for Composer Maven projects.
`—u`	Upgrade diagram files only. Mutually exclusive with `-p` option.	
`-d`	Deploys the .NET Project to the IIS Web Server. Port number must be specified, for example, `-d 80`. For .NET Projects, open the Command prompt as Administrator.	
`-j`	Deploys the Java Project to the Tomcat web server. Port number must be specified, for example, `-j 8090`. For Tomcat admin role access login should be admin/admin.	
`-i`	Update the IPD diagrams Events Property to Voice default set.	
`-c`	Publish IPD to Configuration Database. Publishes to default Tenant. Parameters: `$CMEApplicationName$ $HostIP$ $CmePort$ $UserName$ $Password$`	Release 8.1.410.14 adds support publishing IPDs at the command line:

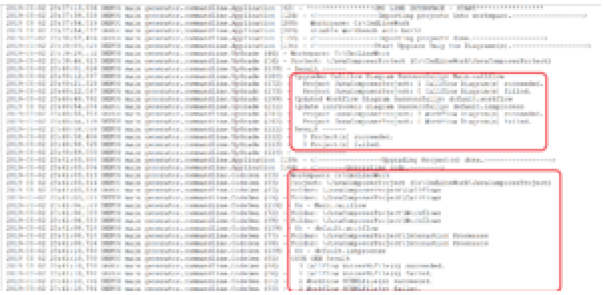
Operation Status

Starting with release 8.1.530.17, a report in text format is displayed at the console, indicating the status of the operation being performed (success/failure), and Composer returns an exit code of 0 or 1.

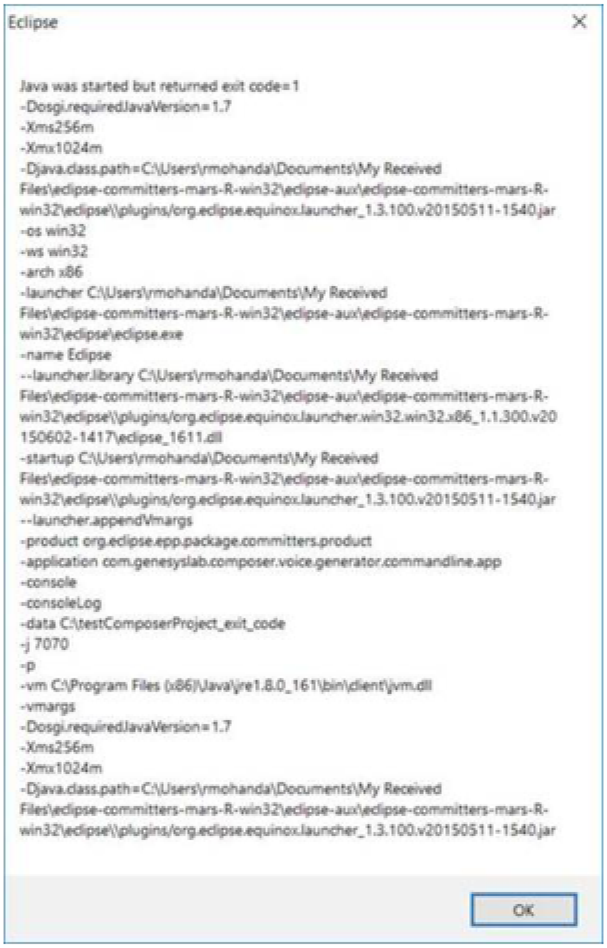
- 0 indicates success and 1 indicates failure.

For example, if you are performing a bulk deployment of projects and, some projects are deployed successfully and some have failed, Composer will now display the associated status (success/failure) at the console and return an exit status of 1 indicating that the intended operation has not been completed successfully.

Sample:



> ## Important
> The exit code is returned for all supported command line operations such as, -p, -u, -d, -j, -i, and -c.



```
Java was started but returned exit code=1
-Dosgi.requiredJavaVersion=1.7
-Xms256m
-Xmx1024m
-Djava.class.path=C:\Users\rmohanda\Documents\My Received
Files\eclipse-committers-mars-R-win32\eclipse-aux\eclipse-committers-mars-R-
win32\eclipse\\plugins/org.eclipse.equinox.launcher_1.3.100.v20150511-1540.jar
-os win32
-ws win32
-arch x86
-launcher C:\Users\rmohanda\Documents\My Received
Files\eclipse-committers-mars-R-win32\eclipse-aux\eclipse-committers-mars-R-
win32\eclipse\eclipse.exe
-name Eclipse
--launcher.library C:\Users\rmohanda\Documents\My Received
Files\eclipse-committers-mars-R-win32\eclipse-aux\eclipse-committers-mars-R-
win32\eclipse\\plugins/org.eclipse.equinox.launcher.win32.win32.x86_1.1.300.v20
150602-1417\eclipse_1611.dll
-startup C:\Users\rmohanda\Documents\My Received
Files\eclipse-committers-mars-R-win32\eclipse-aux\eclipse-committers-mars-R-
win32\eclipse\\plugins/org.eclipse.equinox.launcher_1.3.100.v20150511-1540.jar
--launcher.appendVmargs
-product org.eclipse.epp.package.committers.product
-application com.genesyslab.composer.voice.generator.commandline.app
-console
-consoleLog
-data C:\testComposerProject_exit_code
-j 7070
-p
-vm C:\Program Files (x86)\Java\jre1.8.0_161\bin\client\jvm.dll
-vmargs
-Dosgi.requiredJavaVersion=1.7
-Xms256m
-Xmx1024m
-Djava.class.path=C:\Users\rmohanda\Documents\My Received
Files\eclipse-committers-mars-R-win32\eclipse-aux\eclipse-committers-mars-R-
win32\eclipse\\plugins/org.eclipse.equinox.launcher_1.3.100.v20150511-1540.jar
```

In addition to displaying an exit code, when an exit code of 1 is returned, Eclipse will display a pop-up message (sample shown below). To suppress this message prompt, you can add the `--launcher.suppressErrors` switch to the command.

Beginning with release 8.1.540.07, different exit codes are returned (see table below) to help identify which operation has failed:

Operation	Exit Code
Diagram related errors (code gen)	1
Tomcat/IIS Deployment (-j & -d)	2
Project upgrade (-p & -u)	3
IPD Publish (-c)	4
IPD Event update (-i)	5

Important

> If multiple issues are present in the workspace (say for instance that both deployment and code generation has failed), then the most recently executed operation's exit code is returned. So, if project deployment and code generation fails in the Composer workspace, the exit code 1 is returned to indicate that code generation has an issue, and once that is fixed, issue with project deployment - that is, the exit code 2 is returned. In short, exit codes are returned in the following order: Code Generation > IPD Publish > Project Deployment > IPD Diagram update > Diagram upgrade > Project upgrade.

Notes

- Eclipse should not be running. This command line will launch a headless instance of Eclipse that will exit once code generation is complete.
- `Eclipse.exe` should be executed from its installed location.
- `.\workspace` is the relative path to the Workspace that contains your Projects for which code should be generated. This will generate code for all supported types of diagrams:
    - callflow : VXML
    - sub-callfow : VXML
    - workflow : SCXML
    - sub-workflow : SCXML
    - interaction process diagram : SCXML

Examples

Project Upgrade Report

```
eclipse.exe –application com.genesyslab.composer.voice.generator.commandline.app
-nosplash -console -consoleLog -data .\workspace –p )
```

Publishing IPDs

```
eclipse.exe -refresh -application
com.genesyslab.composer.voice.generator.commandline.app -nosplash -console
-consoleLog -data “WorkspacePath” -c $CMEApplicationName$ $HostIP$ $CmePort$
$UserName$ $Password$
```