

CATIA Embedded Code Generator

The CATIA Embedded Code Generator allows you to generate and build code from a SysML model. You can generate code from the SysML State Machine or SysML Internal Block Definition diagrams. The code generation and building are executed via the 3DEXPERIENCE platform.

You can switch projects (or even shut down your computer saving your work first) while waiting for the generation result as it is running on the cloud and as the widget keeps the ids of the cloud assets in the project options (you have access to it if you need it for debugging purpose for example). When you will come back, another dialog will tell you that operations were running while you were away and will propose you inspect the result and download it.

Information

A workspace is automatically created on our cloud servers for each of your projects using the widget. This workspace gathers the assets needed for the generation. A cloud workspace is automatically deleted after 30 days of inactivity. You can also delete it manually in the **Environment Options** dialog. If you delete the workspace, all the components will have to be regenerated on the cloud on your next session because you will not be able to push any files to a workspace through the plugin.

Prerequisites

- The CATIA Software Producer plugin is installed in your modeling tool. Learn more about how to [download the installation files](#).
- The 3DEXPERIENCE platform is on the cloud.
- You have the Software Production Engineer Role. It will give you permission to use the service.


Working with the code generator


Before starting to work on code generation, first, you have to **connect to 3DEXPERIENCE** by entering the URL of the Software Producer cloud server. After that, you will be able to log in with your regular identifiers.

For the information about how to authenticate with the 3DEXPERIENCE platform, see [Authentication with 3DEXPERIENCE platform](#).


To generate the code in the SysML State Machine diagram

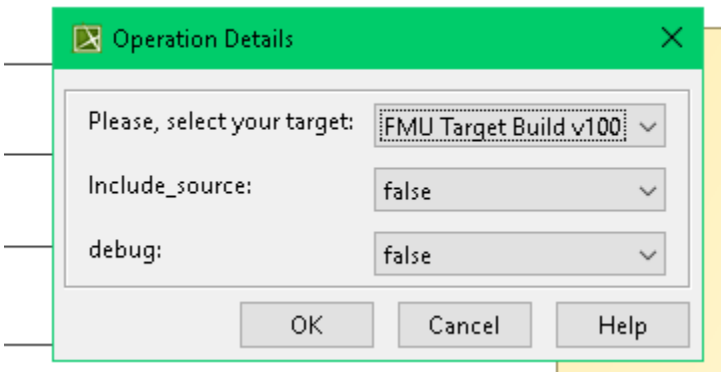
1. In the open SysML State Machine diagram, select the element you want to generate.

 If you do not select any particular element in the diagram, the entire diagram view is selected as the generation input.

2. In the diagram toolbar, click the Code Generation  button.
3. The State Machine Software Production Operation (SPO) on the cloud is automatically called and the code generation is initiated.
4. After code generation is completed, the results file is saved on your local disk drive.

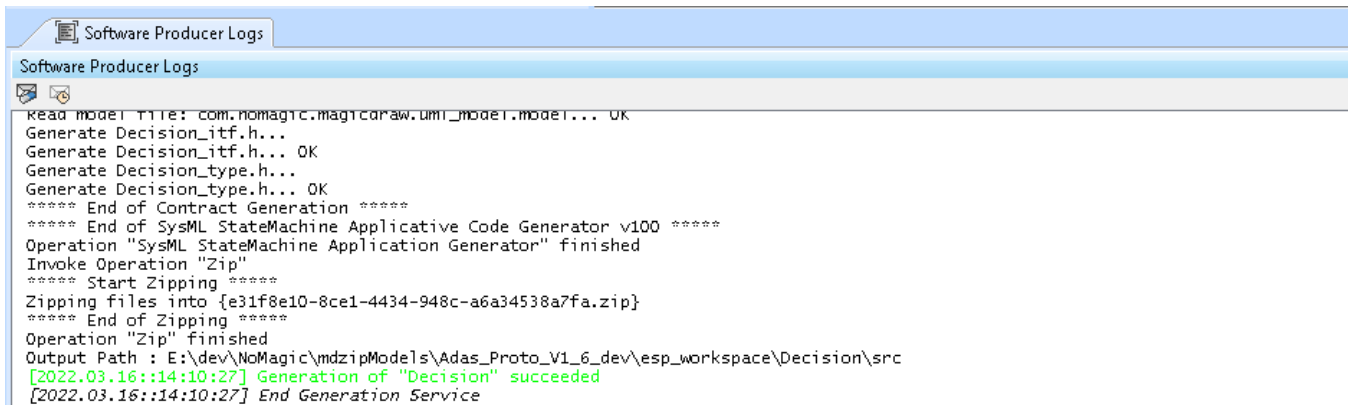
To generate the code in the SysML Internal Block Definition diagram

1. Open the SysML Internal Block Definition diagram.
2. In the diagram toolbar, click the Code Generation  button.
3. In the Operation Details dialog, specify the target for which you want to generate the code and other options if needed.

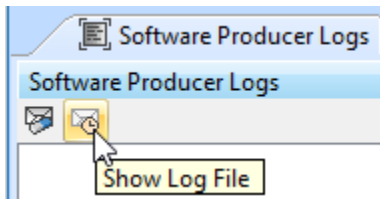


4. Click **OK** when you are done.
5. The Software Production Operation (SPO) on the cloud is automatically called and the code generation is initiated.
6. After code generation is completed, the results file is saved on your local disk drive.

In the Software Producer Log panel, you can see the messages of the running SPO.




This information is saved in the log file next to your project files. To open the log file location quickly, in the **Software Producer Logs** window, click the **Show Log File** button. Each project has its own log file.




A log file for general Software Producer messages not related to a specific project is stored next to the modeling tool log file.

At the right bottom corner of the modeling tool, in the progress bar, you can see whether the SPO is running. After the SPO is completed, the notification is displayed showing the status of the generation.

You can also view the results stored in your file system. To quickly open the location of the results, in the diagram toolbar, click the  **Show in File Explorer** button.


The full set of code you need to generate to cover your project is composed at least of a component and a target-specific part. Component part is relative to the behavior of your application (such as State Machine) whereas the target-specific part is relative to the architecture of your application (such as IBD). In other words, target specific is the binder and the link between the component and the target. As one does not go without the other, this is why at the end of the IBD generation, a dialog will ask you if you want the applicative code to be automatically generated if you have not yet done it (or if it has changed).

A dialog asking to save your project might appear sometime because our generators use for the moment the *.mdzip* file as input and if you have made any changes in your model, it needs to be saved to be reachable from this file.

Once the full set of code has been generated, you can launch a **build** SPO by clicking on the **Build**  button.

When you click a Build button, a dialog opens, wherein you need to specify the target and the available parameters (if there are any).

The code is downloaded to your file system following a specific folder hierarchy:





 The target name is present only if the related folder is the result of target-specific generation or build.

- esp_workspace
 - <componentA_name>[_<targetX_name>]¹
 - src
 - bin²
 - <componentB_name>[_<targetX_name>]
 - src
 - bin
- etc

- backup³
- logs
 - softwareProducer.log⁴

1. You can change the root of the local workspace path in the project options.
2. A bin folder is created only if the related component is the main component for the build.
3. By default a backup of your downloaded files is saved in this folder so that in case something wrong happens with the download of new files, you can still reach your previous files. You can disable this option in the **Project Options** dialog.
4. Log files are rotated by size and date so you might see multiple log files here.

The **roundtrip** action is a *beta* feature that enables you to put back in a State Machine Diagram any modification you would have written in the associated generated usr.c code and to synchronize the file on the cloud.

Button	Description
 Code Generation	<p>Generates the code for the selected elements in the State Machine or Internal Block Definition diagrams. If the entire diagram is selected, the code is generated for the whole content.</p> <p>The generation result is downloaded to your machine.</p>
 Build	<p>After the code is generated, launch the SPO build.</p>
 Show in File Explorer	<p>Open a location in your file system wherein the Software Production Operation results file is stored.</p>
 Roundtrip	<p>The changes made in the generated code are applied in the State Machine diagram.</p>