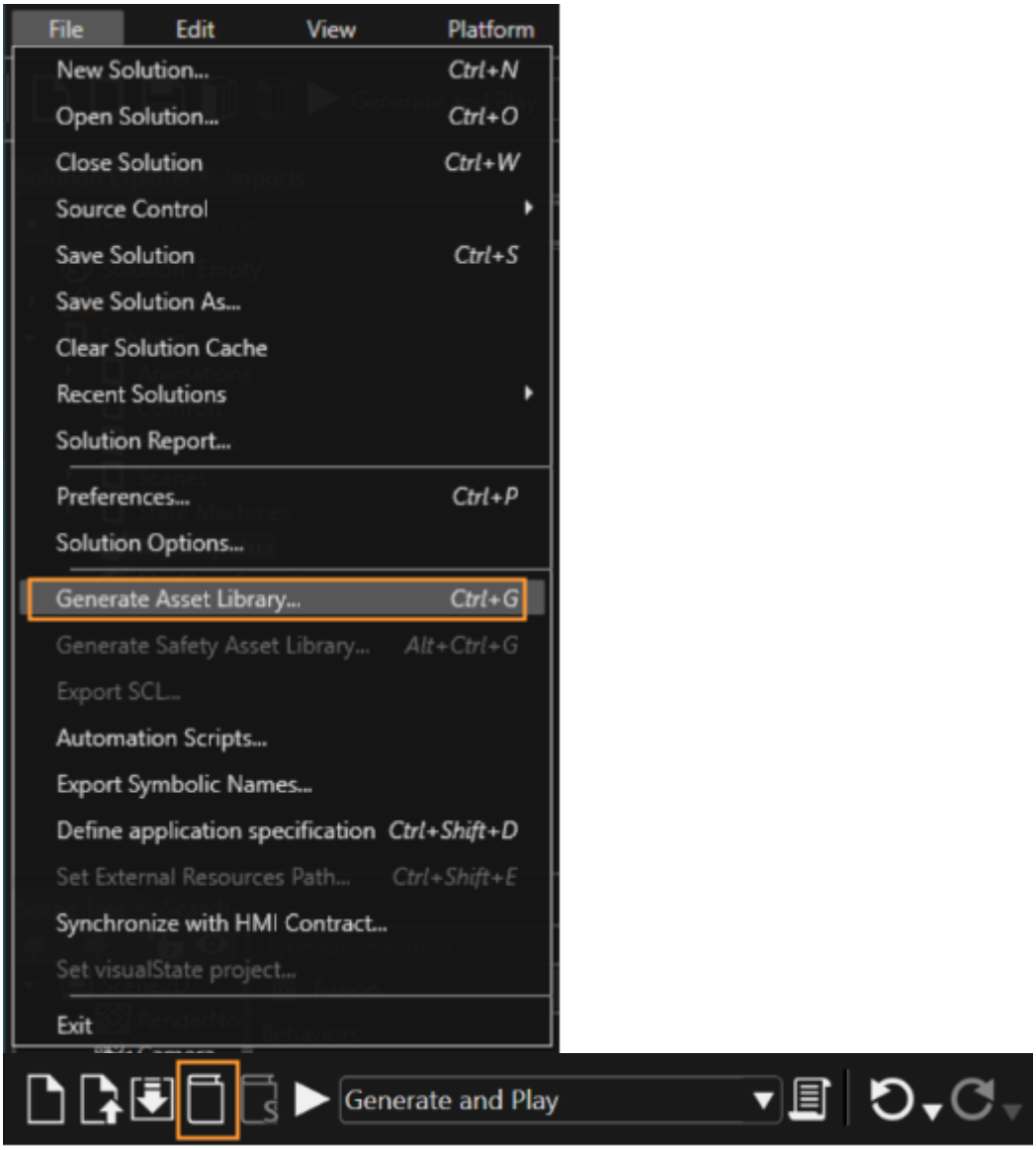
Generate Asset Library

Asset Library

An asset library is a binary compact format containing all user defined artifacts part of a solution.

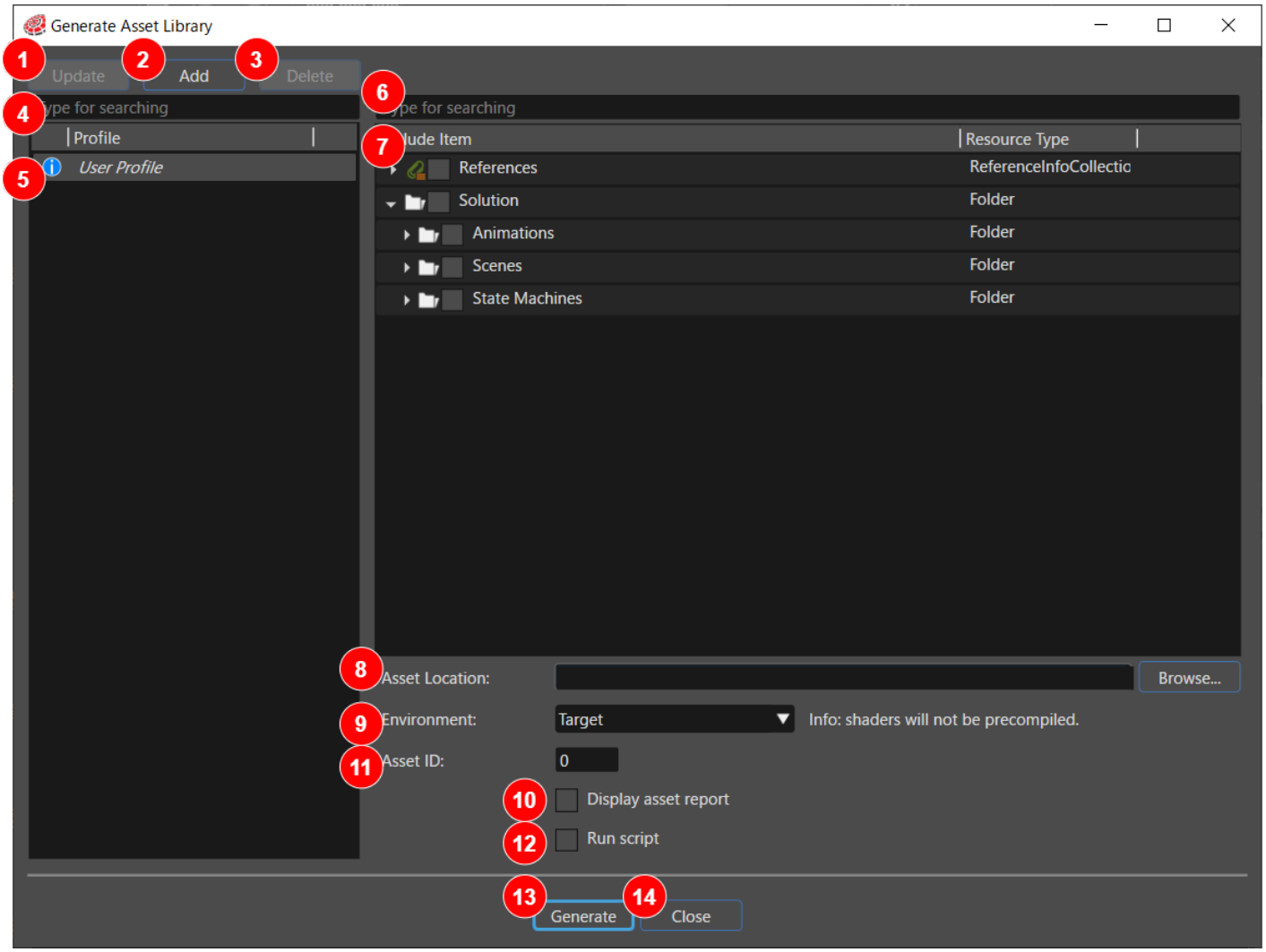
Open "Generate Asset Library" dialog via menu or toolbar item.



The "Generate Asset Library" dialog lists following items part of the solution for asset generation:

- All items having the property "Export Always" enabled.
- All items which are referred by exported items.

Dialog UI



1	Uddate button	When you change the settings of a profile selected in the profile list, press this button to save the changes. If you do not press this button, the changes will not be saved to the target profile.
2	Add button	Creates a new asset profile. Clicking the button displays the [Add New Asset Profile] dialog. The created profile will be displayed in the profile list in the [Generate Asset Library] dialog. It will also be added to the Solution Explorer.
3	Delete button	Select a profile from the profile list and click this button to delete it.
4	Profile search field	Enter a string in this field to search for the corresponding profile in the profile list.
5	Profile list	Asset profiles are listed.

6	Item search field	Enter a string in this field to search for the corresponding profile in the item list.
7	Item List	The items to be output as assets are listed in a tree structure. Items with a check mark in the checkbox will be excluded from the asset library generation. However, if other items in the asset require them, the checked items will not be excluded. Also, if a folder is checked, all the items in that folder that can be excluded will also be checked. If you right-click while a folder is selected, you can use the [Select/Expand All/ Collapse All] options.
8	Asset Location	Specify the location where the generated asset files will be saved. You can specify the location by clicking the [Browse] button.
9	Environment	Select either Simulation or Target. Note that all shaders used in the solution and any additional shaders with the “Export Always” property enabled will be added to the asset file.
10	Asett ID	Specify the asset ID.
11	Display asset report	Display a report containing the serialized items after asset generation. For details, see here.
12	Run script	Executes the specified script after the asset is generated. For details, see here.
13	Generate button	Pressing this button executes the generation of the asset library.
14	Close button	Close Button Pressing this button closes the dialog.

While Scene Composer is open, the saved asset profile is not automatically selected. To use it, select it manually.

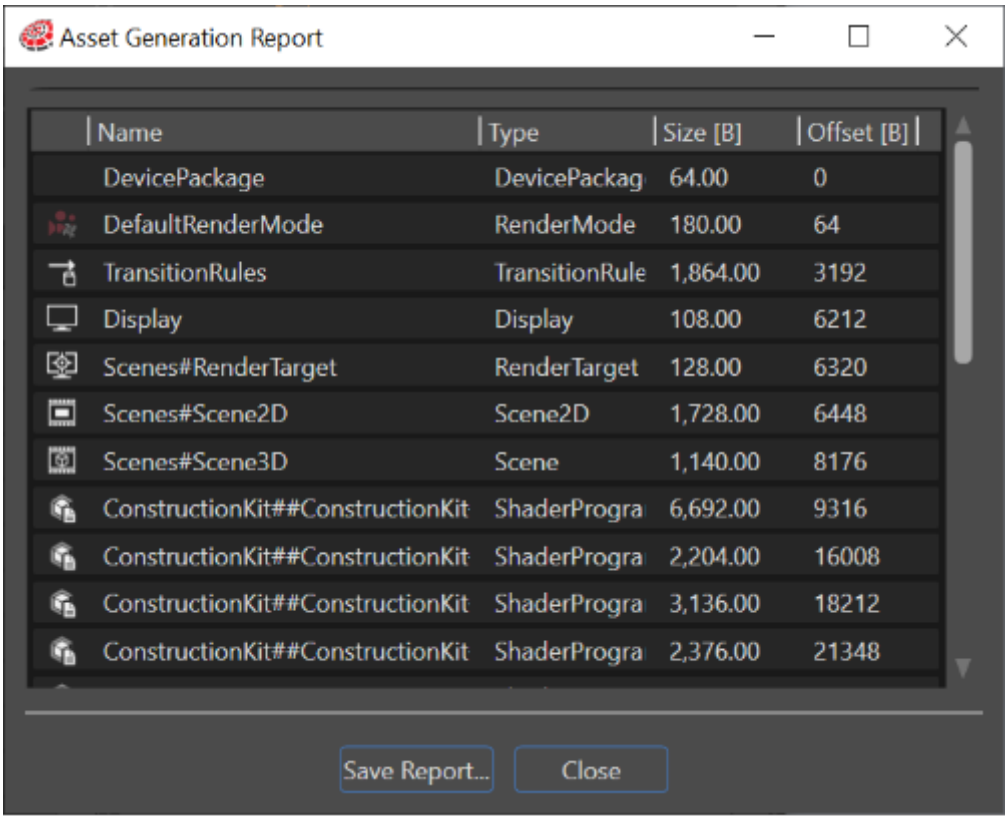
Generate the Asset

Generate the asset file by pressing the "Generate" button. If the generation operation will end successfully, in the location where the asset is generated three different files will become available.

- A binary (.bin) file containing all user defined artefacts that are part of a solution
- A C++ header (.h) with all the symbolic names from the solution
- An Extensible Markup Language (.xml) document that contains information about the asset items which accessed strings from the name table (offset, value, item name, etc.)

Asset Generation Report

If the generating was done successfully and the checkbox "Display asset report" is checked, the "Asset Generation Report" dialog will appear. The report contains a list of all serialized items and the output of the script if the "Run script" checkbox was enabled.



> The total sum of the items displayed in the "Asset Generation Report" is different than all the items summed up. This is the consequence of the fact that the "Total" size should match the size of the asset file which contains more data than the list of "regular" items. The asset header, the name table, AutoUniforms, ShaderAttributes and the index table of each sub lib (scenes, animations, etc.) contain additional information.

The content of this dialog can be saved in CSV (Comma Separated Values) format by clicking the "Save..." button.

Run Script

If enabled this option allows to run a script after the asset is generated. The syntax of the command is the following:

> Script_path [arg1] [arg2] [arg3] ...

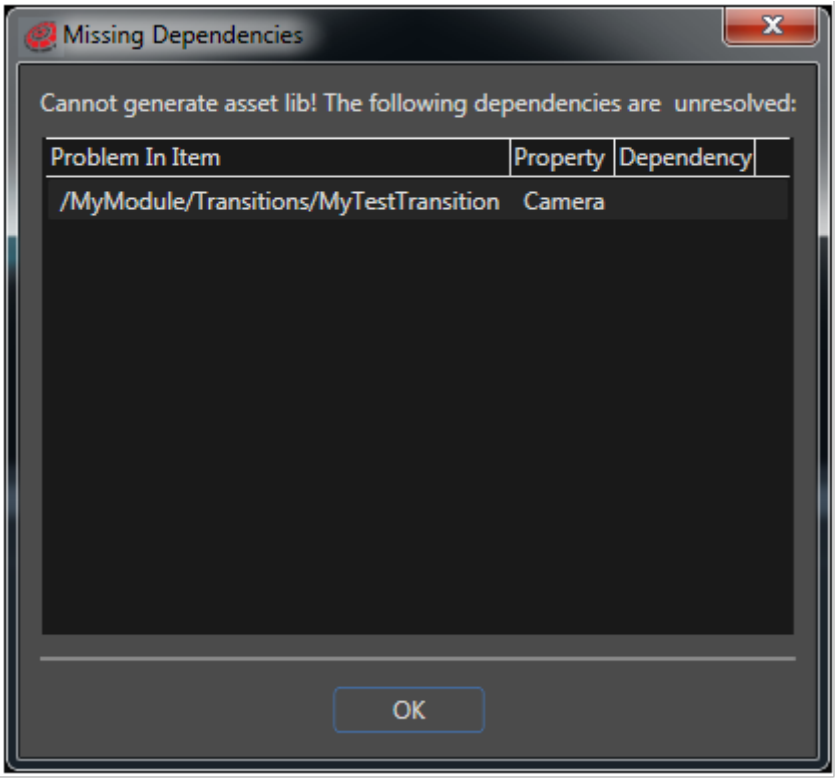
- Script_path is absolute or relative to the solution
- Arguments are optional
- Arguments containing white spaces should be put between double quotes (e.g. "this is one argument")
- The following macro variables can be used as arguments:
  - $AssetFileName: the path to the asset file
  - $AssetProfileName: the name of the selected asset profile
  - $CustomId: The Asset Id of the selected asset profile

Example:

> c:\scripts\my_script.cmd $AssetProfileName "another param"

Missing Dependencies

In case of missing dependencies, no asset generation is possible. In this case, the "Missing Dependencies" dialog lists all items with missing dependencies:



Revision #11
Created 2023-02-14 06:26:09 UTC by Kanai Tomoaki
Updated 2025-04-23 03:07:28 UTC by Tsuyoshi.Kato