



Autogen Configuration Merger

Developers Manual

1 Introduction

Microsoft Flight Simulator X, Flight Simulator X Steam Edition and Prepar3D all store the configuration for the autogen in global files¹. This means that all developers who want to customize the autogen as part of their scenery need to alter the same global configuration files. If a developer fails to merge their changes correctly, this might affect other sceneries.

This tool aims to ease this situation, by providing an easy to use way to merge changes to the autogen configuration files that all developers can use. Besides that this tool is intended to be used not only at install time, but also when just using the addon scenery. In that way it can be verified that the autogen customizations of a scenery remain correct over time as well.

The basic approach is that each developer stores their own customizations in a folder that is part of his addon scenery. The tool will scan all active scenery areas and when autogen customizations are found they will be merged into global autogen configurations. This also means that the developer only has to include his own autogen definitions, there is no need to include the default ones or those of other developers, which minimizes conflicts between sceneries.

2 Installation

The Autogen Configuration Merger tool is installed on the computer and afterwards included in the exe.xml file of Flight Simulator, so that it will automatically run when Flight Simulator is started. For Prepar3D v4 the tool is not installed in the exe.xml file, but in an add-on.xml file. However this results in the same behavior. For the users of your scenery your installer should take care of installing the Autogen Configuration Merger tool as well. More details about that can be found in section 4.

As a developer you will have to install the tool on your machine manually. To do this start the tool from the command prompt with the install argument:

AutogenConfigurationMerger.exe install

¹Starting from Prepar3D v4.4 there is native support for merging autogen configuration files in the simulator. So if you develop your addon for Prepar3D v4.4 or above you do not need to use Autogen Configuration Merger. Although it is reported that functionality is buggy, so you might still want to use Autogen Configuration Merger in the end.

This will bring up an installation window (see Figure 1). where you can specify the folder where the tool should be installed and for which Flight Simulator version you want it to be installed. If you want to use it in different versions of Flight Simulator, you need to run the installer multiple times to make sure all exe.xml or add-on.xml files are modified correctly.

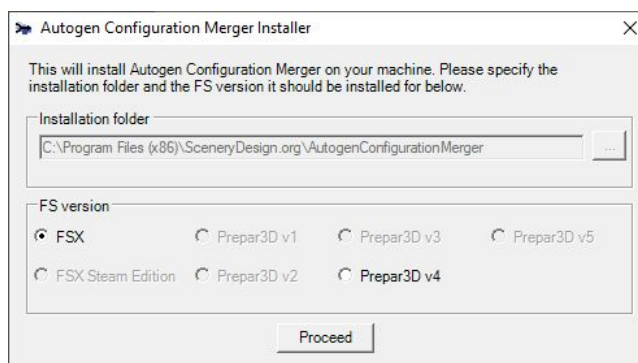


Figure 1: Installation form

The first time you run Flight Simulator after installing you will be asked if you want to run Autogen Configuration Merger at start-up. Confirm this and it is advised to add the tool as a trusted application as well.

3 When the tool runs

Since the tool is installed in the exe.xml or add-on.xml, it will automatically run every time you start Flight Simulator. At each start-up it will be checked if the autogen configuration files are still complete by checking that the entries of the addon sceneries are still there in the global configuration files. If some configuration entries are missing these will be inserted (again).

Since Flight Simulator already loads the autogen configuration files directly at start-up, a restart is needed when changes have been made. Fortunately changes should only be made sparsely, for example when a new scenery was installed or when some entries of an active scenery got lost. To notify you that a restart is needed a GUI is shown, see Figure 2. After 20 seconds this GUI will close automatically.

4 Running the tool from your installer

When installing a new scenery for which you want to use the Autogen Configuration Merger tool to handle the autogen customizations, you need to make sure that the Autogen Configuration Merger tool is installed on the end users computer. Instead of asking the end user to do this, your installer should do it for him. At the end of the installation of the tool an initial merge will be performed as well, to make sure your changes to the autogen configuration files are inserted.

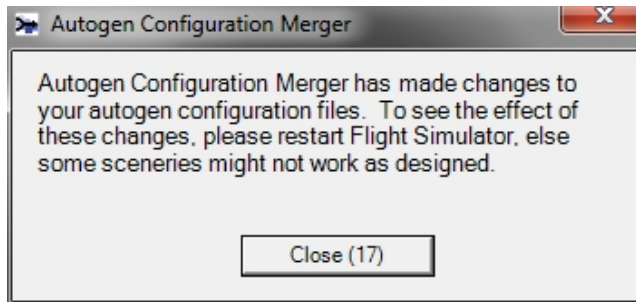


Figure 2: GUI notifying that a restart of Flight Simulator is needed

After you have inserted your addon scenery in the scenery.cfg file, make sure to run Autogen Configuration Merger from your installer in the installer mode. You can include the EXE that you downloaded for Autogen Configuration Merger in your installer. Since the user probably already selected which Flight Simulator version he will be installing to, you can pass that information to the Autogen Configuration Merger installer as well. See the possible values below:

```
AutogenConfigurationMerger.exe install fsx
AutogenConfigurationMerger.exe install fsxse
AutogenConfigurationMerger.exe install p3dv1
AutogenConfigurationMerger.exe install p3dv2
AutogenConfigurationMerger.exe install p3dv3
AutogenConfigurationMerger.exe install p3dv4
AutogenConfigurationMerger.exe install p3dv5
```

You can use the `silent` command line argument to tell Autogen Configuration Merger to not report visual feedback. In that case you will not get a message when a restart is required and the installation will run without a form as well. This allows you to integrate seamless with your installer if you want. In this case you need to check if the exit code returned by the tool is zero to be sure the installation went correctly.

With the command line option `install_folder` you can specify the folder where Autogen Configuration Merger is installed. This allows you to set a different location than the default install folder when running in silent mode. The argument `install_folder` should be followed by a second argument that is the path where the install should be performed.

Also make sure that you install the End User manual that is part of the package you downloaded on the machine of the end user. That way he can find information about how Autogen Configuration Merger works. You can also include this manual in your own documentation if you want.

5 Prepare your scenery

With the Autogen Configuration Merger tool installed, it's time to prepare your scenery to use the tool. You need to follow the steps below:

- Make sure your scenery is active in the scenery library
- In the root folder of your addon scenery, add a folder called **autogen**. So this folder will be at the same level as your **scenery** and **texture** folders.
- Put your custom autogen configuration files in the folder you just created. See Figure 3 for an example.

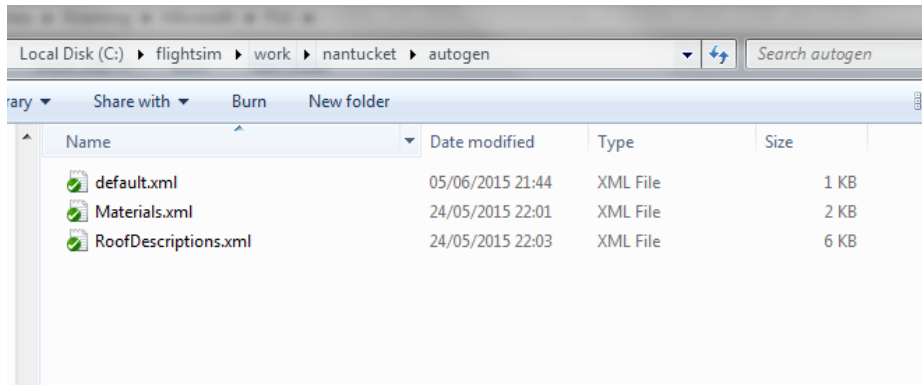


Figure 3: Custom autogen folder

The files you can put in your local configuration folder are the same as you would find in the global autogen folder of FS. So you can have:

- default.xml
- AutogenDescriptions.spb or AutogenDescriptions.xml
- Extrusions.spb or Extrusions.xml
- Materials.spb or Materials.xml
- RoofDescriptions.spb or RoofDescriptions.xml
- TerrainAutogenClassDescriptions.spb or TerrainAutogenClassDescriptions.xml

Each of these files should only include the new definitions you would like to add for your addon scenery. It is not needed to insert the default definitions as well. Also don't include other definitions that your addon scenery doesn't use, as that only increases the risks of conflicts between different sceneries.

5.1 Example: custom default.xml

This section gives an example of a custom default.xml file that adds a new class of autogen library objects. The GUID of that class is used in your autogen files. The GUID as mentioned in the LIBRARYOBJECT entry refers to the GUID of the MDL file that should be used. Each class should be placed in the appropriate autogen region.

```

<?xml version="1.0" encoding="utf-16"?>
<AUTOGEN>
  <REGION>
    <CODE>B</CODE>
    <CLASS>
      <NAME>_iBY_Nantucket_Gas_Station</NAME>
      <GUID>117278df47655ce4dd026983db083b36</GUID>
      <WIDTH>18</WIDTH>
      <DEPTH>2</DEPTH>
      <LIBRARYOBJECT>
        <NAME>Gas_Station</NAME>
        <GUID>efe5e90f47db72463c9e749a3ae183ae</GUID>
      </LIBRARYOBJECT>
    </CLASS>
  </REGION>
</AUTOGEN>

```

5.2 Example: custom AutogenDescriptions.xml

This section shows a sample of a custom AutogenDescriptions.xml file. In the SeasonalModels section you can define custom models to use for the vegetation. Each model is referenced by a GUID in the groupings. The SeasonEntry record below specify with GUID from the MDL file should be used for the vegetation for a specific season. In the Grouping you can refer to one or more models that will become part of this group, weights can be assigned to each model.

```

<SimBase.Document Type="AceXML" version="1,0" id="AutogenXMLMappings">
  <Descr>AceXML Document</Descr>
  <Filename>AutogenDescriptions.xml</Filename>
  <Autogen.SeasonalModels>
    <ModelEntry id="{361570e5-77ac-419f-af27-abee7bcf67d8}">
      <FriendlyName>Custom tree 1</FriendlyName>
      <MinScale>0.900</MinScale>
      <MaxScale>1.100</MaxScale>
    </ModelEntry>
    <SeasonEntry>
      <ModelGuid>{16a1586f-5b63-4af1-abb0-030bbd0f54e9}</ModelGuid>
    </SeasonEntry>
    <SeasonEntry>
      <Season>FALL</Season>
      <ModelGuid>{8f4230c0-8e35-4e96-92e1-c65b17ec6499}</ModelGuid>
    </SeasonEntry>
    <SeasonEntry>
      <Season>MILD_WINTER</Season>
      <ModelGuid>{41b0e15e-cf70-4540-a626-28efadb52cb7}</ModelGuid>
    </SeasonEntry>
    <SeasonEntry>
      <Season>HARD_WINTER</Season>
      <ModelGuid>{3a001ba5-933e-468d-9074-619a9da3219f}</ModelGuid>
    </SeasonEntry>
    <SeasonEntry>

```

```

        <Season>SPRING</Season>
        <ModelGuid>{597b9bc9-0661-44c4-9307-1e12562067cb}</ModelGuid>
    </SeasonEntry>
</ModelEntry>
<ModelEntry id="{f0b3106d-f60b-43ee-8f78-aceebbca731c}">
    <FriendlyName>Custom tree 2</FriendlyName>
    <MinScale>0.900</MinScale>
    <MaxScale>1.100</MaxScale>
    <SeasonEntry>
        <ModelGuid>{8dec1f0c-6c1c-438c-a211-e704eb42da9c}</ModelGuid>
    </SeasonEntry>
    <SeasonEntry>
        <Season>FALL</Season>
        <ModelGuid>{60fa991d-5f6e-41ff-af25-1355b23c049b}</ModelGuid>
    </SeasonEntry>
    <SeasonEntry>
        <Season>MILD_WINTER</Season>
        <ModelGuid>{7380181d-d776-4e03-96e5-533ab858efd7}</ModelGuid>
    </SeasonEntry>
    <SeasonEntry>
        <Season>HARD_WINTER</Season>
        <ModelGuid>{cb6b7bd5-b55f-416c-b6b0-510344117abc}</ModelGuid>
    </SeasonEntry>
    <SeasonEntry>
        <Season>SPRING</Season>
        <ModelGuid>{a8582d73-c8c6-4b9b-b22b-55b968fc5395}</ModelGuid>
    </SeasonEntry>
</ModelEntry>
</Autogen.SeasonalModels>
<Autogen.AutogenGroupings>
<Groups.Grouping id="{53ee720a-8cd6-4802-acd2-b5c525f299f0}">
<FriendlyName>Vegetation Group1</FriendlyName>
<ReferenceEntry>
<TargetGuid>{361570e5-77ac-419f-af27-abee7bcf67d8}</TargetGuid>
<RelativeWeight>40.0000</RelativeWeight>
</ReferenceEntry>
<ReferenceEntry>
<TargetGuid>{f0b3106d-f60b-43ee-8f78-aceebbca731c}</TargetGuid>
<RelativeWeight>20.0000</RelativeWeight>
</ReferenceEntry>
</Groups.Grouping>
</Autogen.AutogenGroupings>
</SimBase.Document>

```

5.3 Example: custom RoofDescription.xml

This section shows an example of a custom RoofDescription.xml file, that can be used to modify the roofs of your autogen buildings. The BuildingRoof entry specifies a specific roof, where the ModelGuid is the GUID of the MDL file that should be used and the MaterialOverride specifies the GUID to the entry in the

Materials.xml that specifies the texture that shall be used. Next a Grouping is defined that can contain different models or other groups. Finally all your new groups should be part of a regionalization else they don't seem to work.

```
<?xml version="1.0" encoding="UTF-8"?>
<SimBase.Document Type="AceXML" version="1,0" id="FenceXMLMappings">
  <Descr>AceXML Document</Descr>
  <Filename>RoofDescriptions.xml</Filename>
  <Autogen.BuildingRoofs>
    <RoofEntry id="{c0bd2dbc-2c1b-44a6-b7c9-befde31541c7}">
      <FriendlyName>custom_roof_model_1</FriendlyName>
      <ModelGuid>{3fd4caa1-c3e8-4521-b3be-0ee97ce9e43e}</ModelGuid>
      <MaterialOverride>{a501104e-3002-4b5d-b0b2-b4112b9c429d}</MaterialOverride>
    </RoofEntry>
    <RoofEntry id="{33c3f9f4-60ad-4989-a889-818ea048a0c5}">
      <FriendlyName>custom_roof_model_2</FriendlyName>
      <ModelGuid>{1071b407-a343-4864-97a5-65aa2cf6b692}</ModelGuid>
      <MaterialOverride>{a501104e-3002-4b5d-b0b2-b4112b9c429d}</MaterialOverride>
    </RoofEntry>
  </Autogen.BuildingRoofs>
  <Autogen.BuildingRoofGroups>
    <Grouping id="{3a4f3f62-3dfc-4fa4-888e-c6d856527a6e}">
      <FriendlyName>Custom roof group</FriendlyName>
      <ReferenceEntry>
        <TargetGuid>{c0bd2dbc-2c1b-44a6-b7c9-befde31541c7}</TargetGuid>
      </ReferenceEntry>
      <ReferenceEntry>
        <TargetGuid>{33c3f9f4-60ad-4989-a889-818ea048a0c5}</TargetGuid>
      </ReferenceEntry>
    </Grouping>
  </Autogen.BuildingRoofGroups>
  <Regionalization id="{6086d8d4-847a-4fea-80bd-93d2c9b6c9ad}">
    <FriendlyName>Custom roofs regionalization</FriendlyName>
    <RegionDescription>
      <RegionId>-1</RegionId>
      <ReferenceEntry>
        <TargetGuid>{3a4f3f62-3dfc-4fa4-888e-c6d856527a6e}</TargetGuid>
      </ReferenceEntry>
    </RegionDescription>
  </Regionalization>
</Autogen.BuildingRoofGroups>
<Autogen.ExternalDependencies>
  <ExternalDependency>
    <Filename>Materials.xml</Filename>
  </ExternalDependency>
</Autogen.ExternalDependencies>
</SimBase.Document>
```

6 Default and 3rd parties definitions

As written in the section above this tool works best when you only include your own custom autogen definitions. Do not include the default definitions (again) and also do not include definitions from other developers. Doing these two things increases the risks of getting conflicts when merging the autogen definitions and it can also make it harder in the future to update your own definitions.

In general addon sceneries should not modify the default autogen configurations, since this would affect the default scenery of the entire earth. So since you won't change them, there is no need to include these definitions in the definitions for your addon scenery. In the rare case that you have updated some of the default configurations you can include them in your custom definitions and the tool will merge them for you.

But if another developer would include the original default definitions again they could be overwritten. The tool can only see they are different, not which one is better. When a definition is found in multiple autogen configurations the last one is taken (the one of the scenery at the end of the scenery.cfg file). So therefore don't include definitions you haven't made or changed.

For definitions of other scenery developers the same applies. To prevent merge issues some developers now also ship the definitions of other developers with their sceneries. Although this might sound useful to the end user, it is a bad practice. Allowing other developers to distribute your autogen configurations, makes it much harder for you as a developer to update them in the future. There could still be old definitions floating around in other products in that case and these could overwrite updated versions you have distributed with your own product.

So to keep things simple: only distribute autogen definitions that you have made yourself for your own scenery!

7 Troubleshooting

In case there are issues with merging autogen configuration files you can perform some first troubleshooting. A log file with the merging progress is saved in the following folder:

```
C:\Users\{username}\AppData\Roaming\SceneryDesign.org\AutogenConfigurationMerger
```

In the same folder you will also find sub-folders for each FS version you have on your system. In these sub-folders you will find backups of the old autogen configurations that Autogen Configuration Merger has saved before making changes. A maximum of 20 backups is kept.

8 Support

If you have any problems while using the tool or if you have suggestions and other feedback to improve the tool, please let me know. You can either contact me directly or visit the Autogen Configuration Merger subforum at FSDeveloper.com.

For the end users of your scenery you are supposed to be the first point of contact of support, since you are the one that has made the autogen customizations. If there are specific issues with the Autogen Configuration Merger tool you can forward these to me of course.

9 Update notification

I would like to ask you to drop me an email if you are using Autogen Configuration Merger in your scenery product. Then I can notify you when the tool gets updated or when serious issues are found (hopefully this doesn't happen too often).

10 Thanks to

- stonelance for the SimProp library
- All users on FSDeveloper.com whole helped to discuss the idea of this tool and who tested the initial beta releases

11 User license

(c) 2015-2020 SceneryDesign.org / Arno Gerretsen

This software program is distributed without charge to other addon developers. Redistribution of the original installer as part of an addon product is allowed, for both freeware and commercial addons. You are NOT allowed to sell this software program itself or ask money for its distribution.

The copyright and any intellectual property relating to this program remain the property of the author.

The software distributed in this way may represent work in progress, and bears no warranty, either expressed or implied.