Working with Model Checker Automation

Version 1.2 – 07/25/2020

AUTODESK.

Getting Started

The Autodesk Model Checker for Revit recognizes some simple configuration files that allow you to automate running checks on your models. To get started with automation you will need a basic familiarity with XML. Simply create the XML configuration files and place them in the correct location and start Revit. Then, Model Checker will run the checks as soon as Revit starts. You can also automate this process by creating Windows tasks to copy the files into the correct locations and start Revit at specific times.

Configuration Files

An XML configuration file can be created as a simply text file saved with an *.xml file extension. A full xml configuration file will look something like this:

```
<?xml version="1.0" encoding="utf-8" ?>
<AutomatedRun Cleanup="delete">
    <Model Path="C:\Some\Path.rvt">
        <CheckSet Path="C:\Some\CheckSet.xml" ExportHtml="true" ExportExcel="true"
        HtmlFolder="C:\Some\Folder" ExcelPath="C:\Some\Path.xlsx" CheckLinks="true" />
    </Model>
</AutomatedRun>
```

The individual nodes are outlined below.

Root Node

```
<AutomatedRun Cleanup="delete">
```

The root node (named 'AutomatedRun') is the container for an entire run. It has a single attribute called 'Cleanup'. The cleanup attribute is required and determines what model checker will do with the XML configuration file once complete. The following are valid **Cleanup** attribute values (values are case insensitive):

- **None** – This will do nothing with the XML file; it will be left in the same location. This is advised if you are running this on a server machine where Revit startup is uncommon. However, if you are running this on a standard machine leaving the file in place means that Model Checker will run the check again the next time Revit starts which may not be desired as this can delay the user from being able to acces Revit.
- **Delete** – This will delete the XML file. You can keep a set of configuration files elsewhere and then have a task that copies them to the run location and starts Revit nightly for instance. The delete cleanup attribute will then simply delete the configuration when it completes so checks are not run again when Revit starts next.

- **Move** – This is similar to Delete except that the completed configuration files will be moved into a sub-directory named 'Complete'.
- **Rename** – Similar to Move, except that files will be renamed with the '.complete' extension instead of moved to a new directory.

Model Node

<Model Path="C:\Some\Path.rvt">

The model nodes designate models on which you would like to run checks. There can be multiple model nodes, and each should be a direct child of the root node. The only attribute on model nodes is the 'Path' attribute. This attribute is required and should be set to the full path of the model file.

CheckSet Nodes

<CheckSet Path="C:\Some\CheckSet.xml" ExportHtml="true" ExportExcel="true" HtmlFolder="C:\Some\Folder" ExcelPath="C:\Some\Path.xlsx" CheckLinks="true" />

CheckSet nodes should be direct children of a model node and determine what check set files to run on the given model. If you do not specify any check set nodes for a model the check set that is currently set for the model (if any) will be run. However, if you specify CheckSet nodes the attached check set will NOT be run unless you specifically specify it.

CheckSet nodes have several attributes as they are the main configurators of the run:

- **Path** – This attribute is required. It may be either the full path to the check set file (including web based public check set files) or it can be the text *Attached*. If set to attached, the check set that is currently set for the model will be run rather than a specific check set file.
- **ExportHtml** – This attribute is optional and allows you to specify if the result of the check is to be exported to html. If this attribute is not present no HTML will be exported.
- **ExportExcel** - This attribute is optional and allows you to specify if the result of the check is to be exported to Excel. If this attribute is not present no Excel file will be exported.
- **HtmlFolder** – This attribute is required if HTML export is on and is the base folder location where html will be exported. Note that Model Checker will create a sub-folder here with the date of the run to avoid overwriting old reports.
- **ExcelPath** – This attribute is required if Excel export is on and is the location of the Excel file that will be created. Note that Model Checker will append the date to the file name to avoid overwriting old reports. So, in the example above, the ExcelPath is set to *C:\Some\Path.xlsx*, the actual filename would be *C:\Some\Path 2020-06-17 9_36_48.xlsx*.

- **CheckLinks** – This attribute is optional and determines if links are to be checked as well as the main model. If this attribute is not present links will not be checked.

Starting Check Runs

The Model Checker will look for configuration files in the directory %programdata%\Autodesk\BIT\Model Checker**REVIT YEAR**\Automated Runs. The next time Revit starts, Model Checker will immediately open models and run checks, export if specified, etc. based on the settings in the XML configuration file.

Additional Information

Logging

Model Checker creates log files during an automation run to assist diagnosing what happened if the output is not what was expected. These can be found in %programdata%\Autodesk\BIT\Model Checker\Logs.

There will be one log for each automation run and it will report what happened during the run as well as any error encountered.

True/False Values

```
<CheckSet Path="C:\Some\CheckSet.xml" ExportHtml="true" ExportExcel="true"
HtmlFolder="C:\Some\Folder" ExcelPath="C:\Some\Path.xlsx" CheckLinks="true" />
```

XML attributes that are true/false type attributes recognize several methods of input. They will recognize any the following as valid data (case insensitive):

- True/False
- T/F
- Yes/No
- Y/N
- 1/0

Notes

The Model Checker Automation will not open the Revit home page when it is done.

The Model Checker Automation is not able to close Revit when complete. This functionality is not supported by the Revit API.

The Model Checker Automation cannot respond to **Unsigned Add-in** security prompts that may show during Revit startup.