



# Rhino/Grasshopper Plugin User Manual

Esri Aiulfi Loris

October 2020

This document presents how to install and use the PRT Rhino command and Grasshopper components. This plugin allows to use the procedural runtime generation engine authored by Esri within Rhinoceros and Grasshopper.

## Contents

|          |                                     |          |
|----------|-------------------------------------|----------|
| <b>1</b> | <b>Installation Instructions</b>    | <b>2</b> |
| <b>2</b> | <b>Rhino Command Usage</b>          | <b>4</b> |
| <b>3</b> | <b>Grasshopper Components Usage</b> | <b>6</b> |
| 3.1      | Main Component . . . . .            | 6        |
| 3.2      | Report Filter Component . . . . .   | 9        |
| 3.3      | Report Unpack Component . . . . .   | 10       |

# 1 Installation Instructions

1. Rhino 6 has to be installed in order for the plugin to work.
2. In order for the plugin to be correctly loaded, it is needed to tick the "Ask to load disabled plug-ins" box located in Rhino's Tools → Options → Plug-ins.

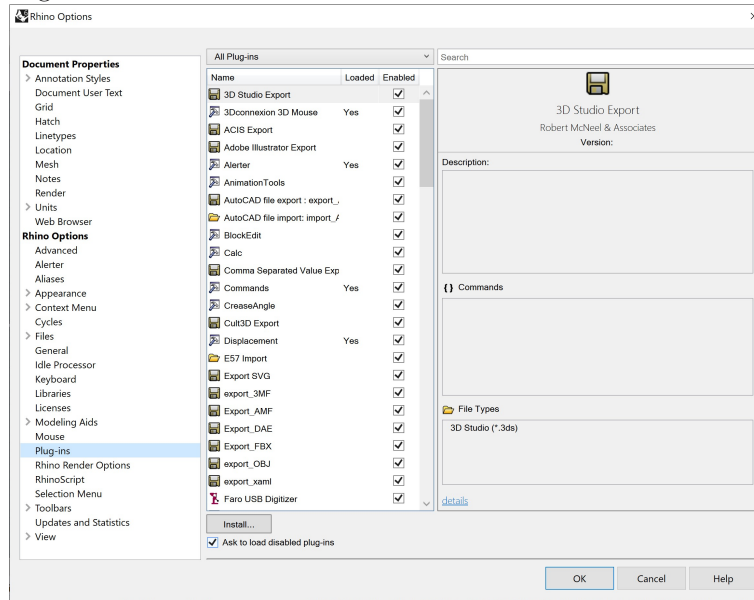


Figure 1: Plug-ins option window.

3. Close Rhino.
4. Run the rhi package by double-clicking it.
5. The package installer will open. Follow the instructions.

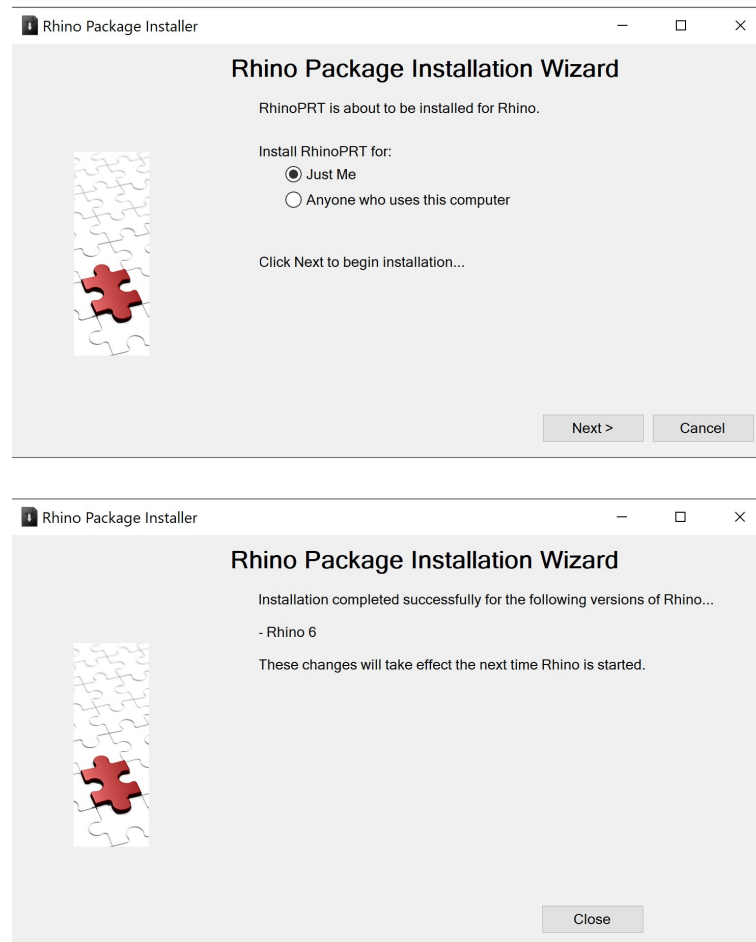


Figure 2: Installation wizard.

6. The plugin will be loaded at the next start of Rhino.

## 2 Rhino Command Usage

1. To start the Rhino plugin, run the command *ApplyRulePackage*.

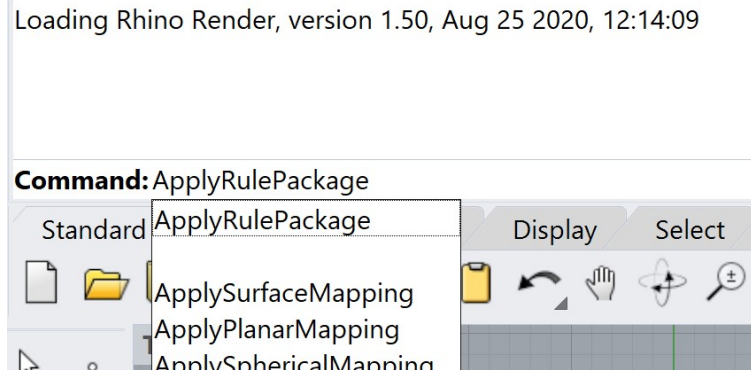


Figure 3: The Rhino Command.

2. It will open the file browser. Select a rule package (.rpk) and click open.

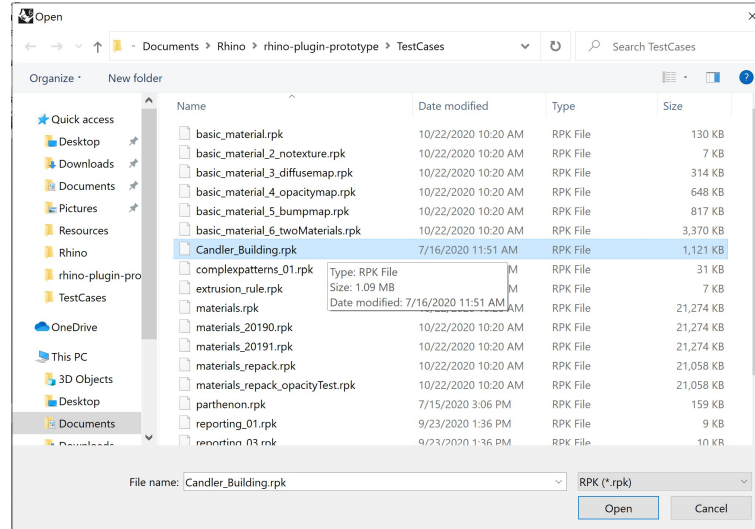


Figure 4: Choose a rule package.

3. Select one or more starting shapes then press enter.

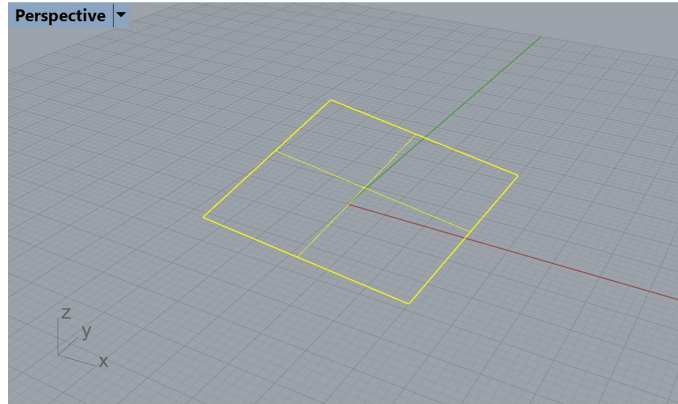


Figure 5: Select one or more starting shapes.

4. The resulting geometry will be generated and appear in the Rhino viewport.

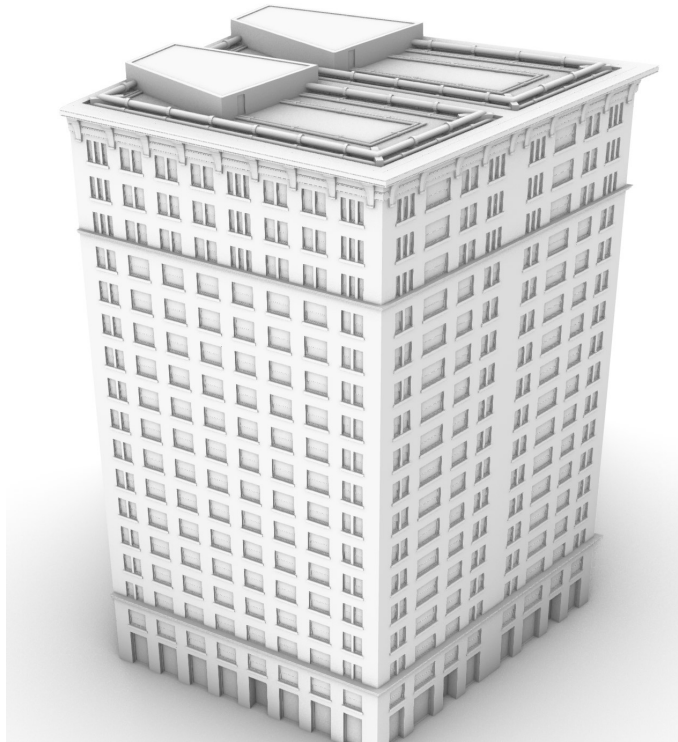


Figure 6: Example of resulting geometry.

### 3 Grasshopper Components Usage

Three components are added in a new tab named *Esri*. The main one, whose purpose is to generate the shapes, materials and cga reports. The second one is used to preview and filter the cga reports. The third one is a simple helper component that unpacks the report into its three attributes. The components and corresponding icons can be seen in figure 7.

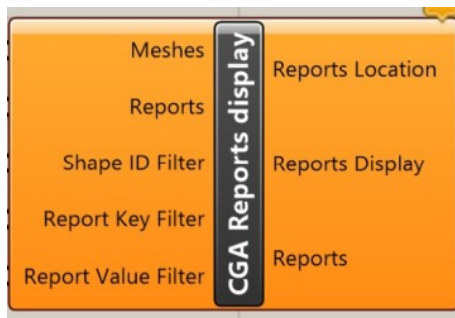
Figure 7: The grasshopper components



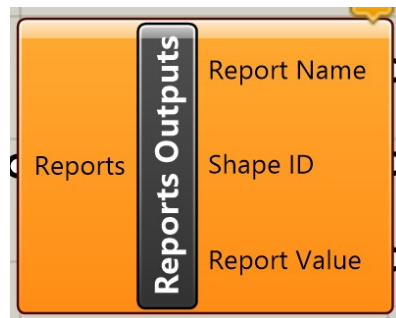
Icons of the components.



Main component.



Report filter and display component.



Report unpack component.

#### 3.1 Main Component

The main component takes two inputs by default (i.e. Fig 8). The path to a rule package (.rpk), can be provided by connecting a *file path* component. The second input accepts various starting shapes. The list of compatible starting object is as follows:

- *Mesh*
- *Rectangle*
- *Brep*

- *Surface*

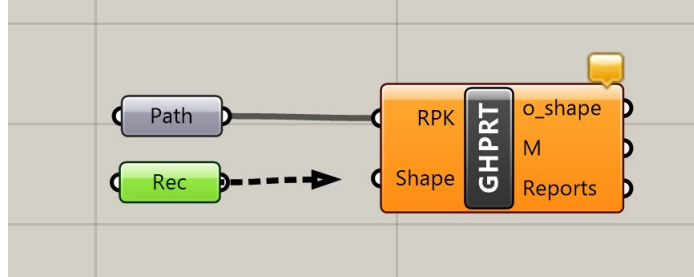


Figure 8: Main component input parameters.

Support for lines and points will be added in the future. When both default inputs are connected, the component is updated with rule attributes defined in the cga rules (Fig 9). These rule attributes are added to the main component as input parameters. These parameters can then be connected normally with other components, or a value can be set directly by right-clicking on each parameter. Here is a list of currently supported input parameters and the corresponding component that can be connected to them:

- Number (and array of numbers): *number slider*
- Boolean (and array of booleans): *Boolean, Boolean Toggle*
- Text (and array of text): *Panel, Text*
- Colour (and array of colours): *Colour, Colour Picker* (Fig. 11)

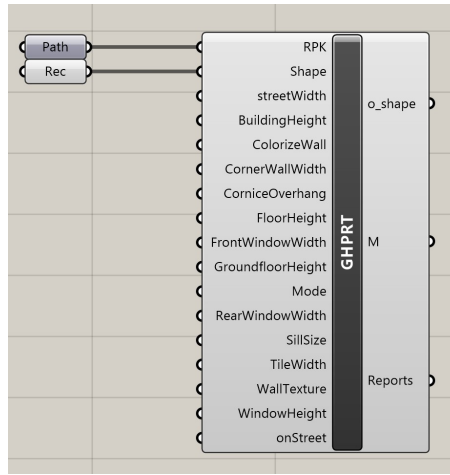
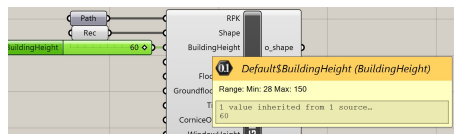


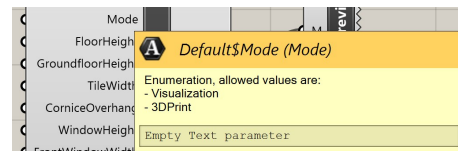
Figure 9: An example of component containing several rule attributes

In order to gain more information on each rule attribute parameters, the user can hover with the mouse over each of them. A tool-tip is displayed, containing information on the expected data. Figure 10 shows two examples of such tool-tips. The first one is a number parameter that expects a number in the range from 28 to 150. The second example accepts a text parameter. The text can be chosen from a list of accepted choices listed in the tool-tip. Figure 11 shows the tool-tip of a colour attribute with a *colour picker* component connected to it.

Figure 10: An example of rule attribute tool-tips.



Tool-tip of a number parameter.



Tool-tip of a text parameter accepting list of defined strings.

**WARNING:** There currently is an issue that prevents the tool-tips to be displayed before at least one of the input parameter is connected.



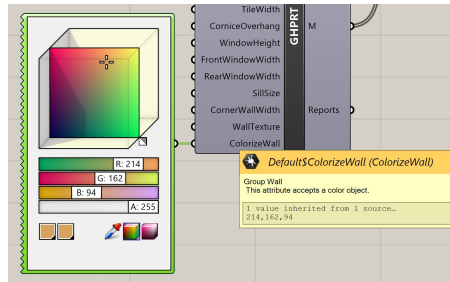


Figure 11: The color picker component connected to a compatible parameter.

This component has three outputs:

1. o\_shape: The generated meshes.
2. M: The generated materials
3. Reports: The generated cga reports.

If materials are generated, they can be applied to the mesh by connecting a *Custom Preview* Grasshopper component (Fig 12). The reports are outputted as a custom type. They are therefore not process-able by built-in Grasshopper components. To address this issue, two helper components are available. They are presented in the next two sections.

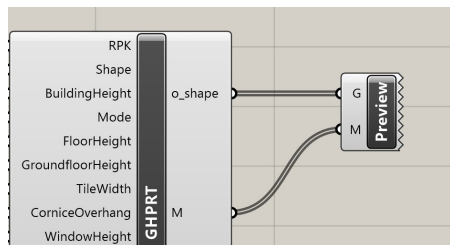


Figure 12: Use the custom preview component to apply materials.

### 3.2 Report Filter Component

This component has 5 inputs. The first two can be connected to the main component's generated meshes and reports as shown in figure 13. The next three inputs are optional filters:

1. Shape ID Filter: Used to filter the reports by initial shape ID. Accepts a *Domain* component.

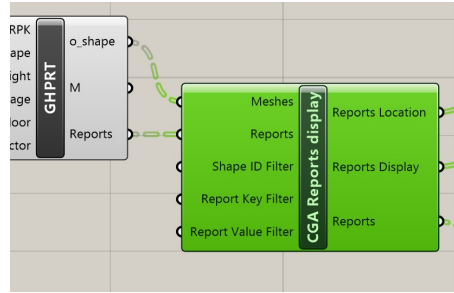


Figure 13: The report preview/filter component.

2. Report Key Filter: Filters the reports by name. Accepts a *Panel* or *Text* component, or a list of them. Report keys can be written on multiple lines.

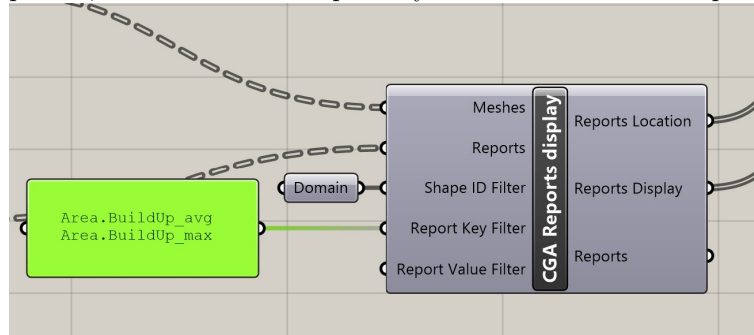


Figure 14: Filter value usage.

3. Report Value Filter: This input allows to select specific values for each keys selected in the second input.

This component has three outputs, the first two can be connected to a *3D Text Tag* component to display the selected reports in the Rhino view. The first one is simply a located plane used to correctly position and align the reports above each generated geometry. The second one provides the formatted reports, so they can be displayed. The third output simply provides the selected reports. These can then be unpacked by the component presented below.

### 3.3 Report Unpack Component

This helper component is very simple. Its task is to unpack the provided cga reports objects into primitive types that can be further processed by built-in Grasshopper components. It takes as input a list of report objects and outputs, for each report, its name (or key), the initial shape ID it belongs to, and the report value itself. Figure 15 shows an example of such component.

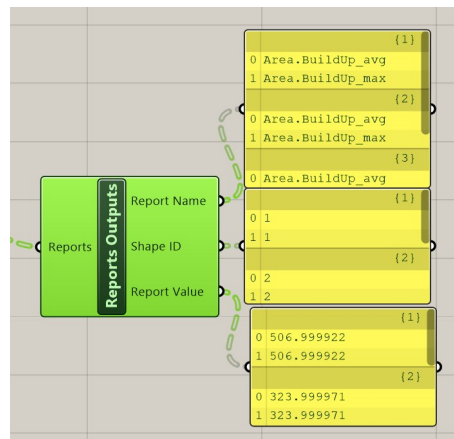


Figure 15: Report unpack component.