



GRASSHOPPER PLUGIN FOR AXISVM

Component 4.1

AxisVM X5 R4 OR LATER VERSION
AUGUST 2020

This page is intentionally blank.

Table of contents

1. COMPONENTS	5
11 ATTR	5
12 BASE.....	6
13 LOAD.....	8
14 SEND.....	10
2. KNOWN ISSUES	11

This page is intentionally blank.

1. COMPONENTS

1.1. ATTR



Cross-section

- Usage* This component is used for selecting a Cross-section from the AxisVM profile catalogue. The cross-section can be selected by double-clicking on the *Cross-section component*.
- Input* There is no Grasshopper input for this component.
- Output* Selected cross-section. If cross-section is not selected, then it returns empty parameter.
- Advantage* Graphical user interface for selecting.
- Disadvantage* Loading cross-section tables and cross-sections happens in real-time, and that is why after clicking on elements the loading takes 1-2 seconds.



Materials

- Usage* This component is used for selecting a Material from the AxisVM material catalogue. After loading, the material can be selected by double-clicking on the *Materials component*.
- Input* There is no Grasshopper input for this component.
- Output* Selected material. If material is not selected, then it returns empty parameter.
- Advantage* Graphical user interface for selecting.

12. BASE



Line to AxisVM Line

Usage This component is used for sending lines from GrassHopper to AxisVM. Lines can be defined as members (beam/ribs/truss) if material and cross-section are both defined.

- Input**
- Line (as list)
 - Material (optional)
 - Cross-section (optional)
 - Member type : Integer (0=Beam; 1=Rib; 2=Truss) (optional)

Output AxisVM Line (if lines can be defined as a member, they can be sent as members)

Lines are created independently in the component.



Mesh to AxisVM Mesh

Usage This component is used for sending mesh structures from GrassHopper to AxisVM. Mesh structures can be defined as

- domains
- surfaces (shell)
- lines or lines as members (from mesh edges)

- Input**
- Mesh (as list)
 - Thickness : Number (optional)
 - Material (optional)
 - Cross-section (will be used only if the output is connected to edges) (optional)
 - Member type : Integer (0=Beam; 1=Rib; 2=Truss) (optional)

Output AxisVM Meshes as a list. If necessary inputs are missing then basic values will be used in another's component input (material=S235 and thickness=10cm for surfaces and domains).



NodeSupports

Usage This component is used for creating node supports.

Input (Rx,Ry,Rz,Rxx,Ryy,Rzz) :6 double-precision floating numbers for stiffnesses of node support.

Output AxisVM NodeSupport



Point to AxisVM Point

Usage This component is used for creating nodes from GrassHopper to AxisVM

- Input**
- Point (as a list)
 - AxisVM NodeSupport (optional)

Output AxisVM Point



**PolyLine to
AxisVM
Domain**

Usage This component is used for creating domains from a closed polyline, which can be represented in one plane.

- Input*
- Polyline (define the edges of domain - need to be closed, and must fit in one plane)
 - Thickness : Number (*optional*)
 - Material (*optional*)
 - Cross-section (unused in the component) (*optional*)
 - Member type : Integer (0=Beam; 1=Rib; 2=Truss) (unused in the component) (*optional*)
 - Holes : List of closed polylines which will represent the holes in the domain. Must fit into the domain's area. (*optional*)

Output AxisVM Domain. If necessary inputs are missing then basic values will be used in another's component input (material=S235 and thickness=10cm for domains).

13. LOAD



AxisVM Distributed Domain Load

Usage This component is used for creating distributed domain loads for domains, which are already sent into AxisVM.

Input	• <u>AxisVM Mesh</u>	AxisVM Mesh which can get the load, if it is sent as domain (<i>optional</i>)
	• <u>AxisVM Domain</u>	AxisVM Domain which can get the load, if it is sent (<i>optional</i>)
	• <u>Integer</u>	0=Global load; 1=Local Load (<i>optional</i>)
	• <u>Text</u>	Direction (X/Y/Z) (<i>optional</i>)
	• <u>Number</u>	Intensity of load force (default:0) (<i>optional</i>)

Output AxisVM Distributed Domain Load



AxisVM Distributed Line Load

Usage This component is used for defining distributed line loads for lines which are already sent into AxisVM.

Input	• <u>AxisVM Line</u>	AxisVM Line which will get the load if it is sent. (<i>optional</i>)
	• <u>AxisVM Mesh</u>	AxisVM Mesh which can get the load, if it is sent as edges (<i>optional</i>)
	• <u>Integer</u>	Global=0; Local=1 (<i>optional</i>)
	• <u>Text</u>	Direction (X/Y/Z) (<i>optional</i>)
	• <u>Number</u>	Start position of load (<i>optional</i>)
	• <u>Number</u>	End position of load (<i>optional</i>)
	• <u>Number</u>	Start point intensity (<i>optional</i>)
	• <u>Number</u>	End point intensity (<i>optional</i>)

Output AxisVM Distributed Line Load



AxisVM Distributed Surface Load

Usage This component is used for defining distributed surface loads for AxisVM mesh which are already sent as surfaces into AxisVM.

Input	• <u>AxisVM Mesh</u>	AxisVM Mesh which will get the load if it is sent as surface.
	• <u>Integer</u>	Global=0; Local=1 (<i>optional</i>)
	• <u>Text</u>	Direction (X/Y/Z) (<i>optional</i>)
	• <u>Number</u>	Intensity (<i>optional</i>)

Output AxisVM Distributed Surface Load



AxisVM Domain Area Load

Usage This component is used for defining domain area loads for domains which are sent to AxisVM

Input	• <u>AxisVM Mesh</u>	AxisVM Mesh which can get the load if it is sent as domain. (<i>optional</i>)
	• <u>AxisVM Domain</u>	AxisVM Domain which can get the load if it sent (<i>optional</i>)
	• <u>Integer</u>	Global=0; Local=1 (<i>optional</i>)
	• <u>Text</u>	Direction (X/Y/Z) (<i>optional</i>)
	• <u>Number</u>	Intensity (<i>optional</i>)
	• <u>PolyLine</u>	Among this PolyLine the load force is distributed

Output AxisVM Domain Area Load



AxisVM LoadCase

Usage This component is used for connecting the loads to the GrassHopperToAxisVM component.

Input (* - as a list)

- AxisVM NL* AxisVM Nodal Loads (*optional*)
- AxisVM DLL* AxisVM Distributed Surface Loads (*optional*)
- AxisVM SW* AxisVM Self Weights (*optional*)
- AxisVM DSL* AxisVM Distributed Surface Loads (*optional*)
- AxisVM DDL* AxisVM Distributed Domain Loads (*optional*)
- AxisVM DAL* AxisVM Domain Area Loads (*optional*)

Output AxisVM LoadCase



AxisVM Nodal Load

Usage This component is used for defining Nodal Loads for points that are already sent to AxisVM.

Input

- AxisVM Point Point which will get the load if it is already sent.
- Numbers Six numbers defining the nodal loads numeric parameters

Output AxisVM Nodal Load



AxisVM SelfWeight

Usage This component is used for defining SelfWeights to AxisVM Members.

Input (* - as a list)

- AxisVM Line* AxisVM Line which can get selfweight (*optional*)
- AxisVM Mesh* AxisVM Mesh which can get selfweight if it is sent as edges (*optional*)
- AxisVM Mesh* AxisVM Mesh which can get selfweight if it is sent as surface (*optional*)
- AxisVM Mesh* AxisVM Mesh which can get selfweight if it is sent as domain (*optional*)
- AxisVM Domain* AxisVM Domain which can get selfweight (*optional*)

Output AxisVM SelfWeight

14. SEND



GrasshopperToAxisVM

- This is the main component of the plugin. Currently, every communication between Grasshopper and AxisVM is done by this component.
- If any of the input parameters are changed the component will refresh the newly created AxisVM model's status.

Usage This component is used for sending Objects to AxisVM.

Input (* - as a list)

- AxisVM Point* (optional)
- AxisVM Line* (optional)
- AxisVM Mesh* Surfaces (optional)
- AxisVM Mesh* Domains (optional)
- AxisVM Mesh* Edges (optional)
- AxisVM Domain* (optional)
- AxisVM LoadCase* (optional)
- AxisVM Settings This parameter sets the parameters of sending

Output There is no Grasshopper output for this component.

 **Run of this component for first-time blocks any Grasshopper - user interactions while an AxisModel is loaded / connected.**



AxisVM Settings

Usage This component is used for setting the AxisVM Model and the transmission between AxisVM and Grasshopper

Input

- Bool (New)
TRUE: new AxisVM model is opened and connected to Grasshopper
FALSE: then an existing and already loaded AxisVM model will be connected to Grasshopper.
- Text (FilePath)
If the first parameter is false, then the opened AxisVM model's filepath is necessary.
- Bool (JustPoints)
TRUE: The sender component will only change the coordinate of all points in the model
Set true only if the number and connections of points is constant, and only their coordinates are changed!
- Bool (SendOnlyChanges)
TRUE: The sender resends only changed objects.
FALSE: The sender resends every object generated in Grasshopper.
Use with care, disjoint and joint objects can cause errors like multiply sent or unsent objects.
- Bool (AxAnalysis)
TRUE: AxisVM runs linear analysis on the model after the model is changed.
- Bool (Double-Check)
TRUE: AxisVM checks and deletes double nodes and objects.

Output AxisVM Settings

2. KNOWN ISSUES

- Loading a Grasshopper project which contains *GrasshopperToAxisVM* components can cause multiple AxisVM applications opened at the same time.