
SYSTEMA V4.9

User Guide

September 2022

Ref: UM.000041864.AIRB
Edition: 6

Use of the software and of the present software manual is submitted to a license agreement to be accepted before the software installation on a computer.

All suggestion or error concerning the software or this software manual can be sent to:

AIRBUS DEFENCE AND SPACE

To the attention of Mr. M. LEPIILLIEZ

Z.I. du Palays

31 rue des Cosmonautes

31402 TOULOUSE CEDEX 4

FRANCE

This document and the information it contains are property of Airbus Defence and Space. It shall not be used for any purpose other than those for which it was supplied. It shall not be reproduced or disclosed (in whole or in part) to any third party without Airbus Defence and Space prior written consent.

Table of Contents

Table of Contents	3
Introduction	4
General	7
General User Interface	16
Geometrical model management	69
Schematic management.....	146
Trajectory management	169
Kinematics management.....	196
Mission management.....	213
Process management	239
Post-processing management	261
Batch command	277
Modeler technical annex.....	281
Trajectory technical annex	366
Kinematics technical annex.....	374
Post-Processing technical annex	399
File formats annex.....	410
Administration annex	417

Introduction

Introduction

About this document

This User Manual is intended for users of the SYSTEMA application.

This manual is divided into chapters. The table below presents a brief description of each chapter in the manual:

Chapter title	Content
General principles	General principles about the SYSTEMA files and entities How to start and quit the SYSTEMA application
General User Interface	General ergonomical principles and basic procedures for using the SYSTEMA application
Geometrical model management	Working on geometrical models and meshings
Schematic management	Working on schematics
Trajectory management	Working on trajectories
Kinematics management	Working on kinematic models
Mission management	Working on missions
Process management	Working on processes
Post-Processing management	Working on post-processing
Batch command	Description of the batch command to execute a process diagram or a file conversion with no user interaction or to launch the application with a pre-loaded file

About SYSTEMA

The SYSTEMA application can be used to create and test models, trajectories, kinematic models and missions. You can also create processes, run processes in the frame of missions, and consult process results.

This document is based on the V4.8 version of the SYSTEMA application. With this version, you can use files generated with the previous versions of the SYSTEMA application.

Some applications can be interfaced with the SYSTEMA application (THERMICA, DOSRAD...) which are considered as plugins. The system configuration used in the examples of the current document may be different from your system configuration.

Glossary

In this User guide, some terms have a specific meaning:

Term	Specific meaning
arc	Orbital arc used to build SYSTEMA trajectories
attribute	Property of a SYSTEMA entity
"component" or "entity component"	<p>Generic term used to refer to any element of a SYSTEMA entity, within its tree structure. For example, geometrical models ("SYSTEMA entities") are made of "components" which may be objects or shapes.</p>
container	Entity component which contains other components
"entity" or "SYSTEMA entity"	Generic term used to refer to any of the objects managed by the SYSTEMA application, namely geometrical models , trajectories , kinematic models , missions .
kinematic model	Set of kinematic bodies which represent a mechanical structure and its motion
object	Group of shapes
meshing	A meshing divides a model's shapes into meshes
mission	Assembly of a geometrical model, a trajectory and a kinematic model
geometrical model	Set of shapes and objects
SYSTEMA process	Sequence of processes which are specific to an application
SYSTEMA tab	Tab of the SYSTEMA application
trajectory	Sequence of orbital arcs which represents the movement of a spacecraft in relation to celestial bodies
viewer	Viewport, used to display different types of views, such as 3D and others
post-processing	Sequence of boxes to manage the results

Typographic conventions

This manual uses the following typographic conventions:

Convention	Description	Example
Blue bold	Names of: <ul style="list-style-type: none"> • SYSTEMA tabs • SYSTEMA menus • options of SYSTEMA menus • field names in SYSTEMA windows 	Modeler tab File menu Fit – Selected arc option Reverse checkbox
<i>Italics</i>	Names used in this document to indicate options depending on the current entities	Fit – Loaded mission – Trajectory "Loaded mission" is not the option name; it is the name of the mission which has been loaded during the SYSTEMA session
Numbered action list	Procedure containing several steps	<ol style="list-style-type: none"> 1. Create the component. 2. Modify the component by means of the edition window.
Bulleted action list	Procedure containing one step only	<ul style="list-style-type: none"> • In the Edit menu, select the Undo option

Useful additional information is provided using the  symbol, as follows:

	<i>Note:</i> ...
---	------------------

General

General principles

Overview

Introduction

This chapter presents some general principles you have to know before using the SYSTEMA files and entities. It also describes how to start and quit the SYSTEMA application.

In this chapter

This chapter covers the following topics:

- General principles
 - Overview
 - Introduction
 - In this chapter
 - SYSTEMA environment
 - Files organization
 - The SYSTEMA working directory
 - Environment variables
 - SYSTEMA entities
 - The different entities and corresponding files
 - Constraints for entity files
 - Loading entities
 - General rule
 - Meshing load
 - Mission load
 - Processing load
 - Exporting to another SYSTEMA version
 - SYSTEMA versions
 - Using files generated in another version
 - Starting and quitting SYSTEMA
 - How to start SYSTEMA
 - How to quit SYSTEMA
 - SYSTEMA help

SYSTEMA environment

Files organization

Once the SYSTEMA application has been installed in your system, the installation directory contains:

- the main SYSTEMA binary file (with name depending on the operating system):

File name	Operating system
SystemaWIN.exe	Windows
SystemaLNX	Linux

- three subdirectories containing additional binaries and data files:

Directory	Content
applications	For each application interfacing with SYSTEMA, it contains a specific subdirectory with all the necessary binary and data files
bin	All the necessary binary files for SYSTEMA use
data	All the internal data files, such as image files
licenses	All the licenses files for SYSTEMA use, if necessary
doc	Contains SYSTEMA documentation files
python	All the python script files for SYSTEMA use

The SYSTEMA working directory

The SYSTEMA application needs a working directory which will be used by default when creating files and searching files.

The SYSTEMA working directory can be defined via the environment variable "**SYSTEMA_WORKING_DIR**". If not, the working directory is the user home directory ("my documents" for PC system, "Home directory" for UNIX system).

Environment variables

SYSTEMA uses two environment variables which have to be set before running the application:

- "**SYSTEMA_HOME**", defines the directory containing the SYSTEMA executable file and the **applications**, **bin** and **data** directory.
- "**SYSTEMA_WORKING_DIR**", defines the directory that will be used by default to store created files and generated results; this variable is optional.

SYSTEMA also use a set of configurable environment variables:

- To manage SYSTEMA :

Variable	Definition
SYSTEMA_DISABLE_3D_OPTIM = 1	Disable 3D optimizations (useful in case of graphics hardware incompatibility for 4.8.2+ versions).

	<i>Required for OpenGL <= 2.0</i> 3D optimization can also be managed with the settings
SYSTEMA_NOAUTO3D = 1	Deactive the automatic opening of 3D views when opening a file
SYSTEMA_SATRATIO = n	Customized satellite size in the 3D view and access to real size of the celestial body (by default 1000)
SYSTEMA_FONTSIZE = n	Size of the display font This option can also be managed with the settings
SYSTEMA_STEPTAS = 1	Active the import and export of Step-Tas (version < 4.3.3c)
SYSTEMA_BDF_IMPORT = 1	Active the bdf/blk import (version < 4.3.3c)
SYSTEMA_BDF_FIXED_DOUBLE = 1	Export bdf/blk: 'fixe double' format instead of 'libre double'
SYSTEMA_BDF_THERMALID_AS_PID = 1	Export bdf/blk: thermal id exported as property id
SYSTEMA_BDF_GRIDID_START = 1	Export bdf/blk: value of the first grid id
SYSTEMA_BDF_ELEMID_START = 1	Export bdf/blk: value of the first element id
SYSTEMA_ANTI_ALIASING = 1	Enable anti-aliasing
SYSTEMA_SOFTWARE_RENDER = 1	Disable 3D acceleration by graphics card
SYSTEMA_2DGT_RESOLUTION = n	Maximum resolution of arcs in the ground track view (by default 100000)
SYSTEMA_ATT_NOSIGN = 1	Deactivate the icon triangles for performance
SYSTEMA_TEXTURE_HR = 1	Activate the high resolution textures
SYSTEMA_NO3DEFFECT = 1	Disable the planet indicators and the sun effect
SYSTEMA_NOAUTOTIMELINE = 1	Disable the default opening of the timeline in the trajectory and the mission
SYSTEMA_EDITORDOCKED = 1	Automatic lock of edition window
SYSTEMA_STL = 1	Activate the STL export feature
SYSTEMA_STP_CHORD_TOLERANCE = c	Overloads the value of the chord tolerance used for step files tesselation (by default c = 1 mm)

	This option can also be managed with the settings
SYSTEMA_STP_ANGLE_TOLERANCE = a	Overloads the value of the angle tolerance used for step files tesselation (by default a = 60 deg) This option can also be managed with the settings
SYSTEMA_PROFILES_DIR = path	Define the <i>path</i> where the profiles are stored
SYSTEMA_NTHREAD = n	Number of threads for the multithreading computation (only applicable for Thermica and PlumImp)
SYSTEMA_COMPONENTS_PATH = path	Define the <i>path</i> where the components for the schematic are stored
SYSTEMA_DEFAULT_PYTHON_CONSOLE = 1	Use the default Python console instead of the Jupyter console
SYSTEMA_ENABLE_HIGH_RES = 1	Enable the maximum resolution to 8k screen

- To manage Scheduler:

Variable	Definition
MOS_TCMSG_PORT_START = s	Start of the range of ports used by the network (by default 48557)
MOS_TCMSG_PORT_RANGE = r	Area of the range of ports used by the network (by default 17 for version < 4.4.2, then 3 from version 4.4.2WIN)
SYSTEMA_CONNECTING_ATTEMPTS = c	Number of connecting attempts on each port of the list (by default 3)
SYSTEMA_TIMEOUT_CONNECTION = t	Timeout limit for each attempt on each port in ms (by default 3000 from version 4.4.1)

SYSTEMA entities

The different entities and corresponding files

In this User Guide, we will use the "**entity**" or "**SYSTEMA entity**" terms to refer to any of the objects managed by the SYSTEMA application, namely **geometrical models**, **schematics**, **trajectories**, **kinematic models**, **missions**, **processes and post-processing**.

A specific file type is associated with each entity type.

The following table presents the different entities and corresponding files:

Entity	File extension

model	".sysmdl"
meshing	".sysmsh"
schematic	".syschm"
trajectory	".systrj"
kinematic model	".syskyn"
mission	".sysmis"
processing mission	".sysset"
material	".sysmat"
process	".sysprc"
process result	".sysres"
post-processing	".sysppr"
visualisation 2D	".sysvisu"

Constraints for entity files

The geometrical models, schematics, trajectories, kinematic and post-processing models are stored in independent files.

A meshing file depends on the associated model file. It contains information on relative and absolute file location in the SYSTEMA environment.

A mission file depends on the associated model, schematic, trajectory and kinematic model files. It contains information on relative and absolute file location in the SYSTEMA environment. If you have to rename or move files, SYSTEMA may ask you to indicate the path for linked files.

A processing file depends on the associated mission, meshing or schematic model file. It contains information on relative and absolute file location in the SYSTEMA environment.

Loading entities

To work on entities, you have to load them in SYSTEMA.

General rule

As a general rule, if you load an entity involving other entities, the associated entities are loaded too.

In the case of a model, a schematic, a trajectory, a kinematic or a post-processing model, you load them independently.

In the case of a meshing, a mission or a processing, the associated entities are loaded.

Meshering load

When you load a meshing, the related model is automatically loaded.

The following rules apply to this model:

Rule 1:	If you want to close the model related to the meshing, you have first to close the loaded meshing file.
Rule 2:	While the meshing is opened, you cannot use the "save as" function to the related model because it is used by the meshing.

Mission load

When you load a mission, the associated model, trajectory and kinematic models are automatically loaded. As long as the mission is displayed, you cannot save as.

Rule 1:	If you want to close the model, trajectory, kinematic model or meshing related to the mission, you have first to close the loaded mission file.
Rule 2:	While the mission is opened, you cannot use the "save as" function to the related model, trajectory, kinematic model or meshing because they are used by the mission.

Processing load

When you load a processing, the associated meshing, schematic or mission model is automatically loaded. As long as the processing is displayed, you cannot save as.

Rule 1:	If you want to close the meshing, schematic or mission related to the processing, you have first to close the loaded processing file.
Rule 2:	While the processing is opened, you cannot use the "save as" function to the related meshing, schematic or mission because they are used by the processing.

Exporting to another SYSTEMA version

SYSTEMA versions

Two kind of SYSTEMA releases exist :

- STS (Short Term Support)
- LTS (Long Term Support). Here, support means that throughout the lifetime of a release there is a commitment to patch the software if required. For an LTS, there is a longer testing cycle, where more testing and bug fixing takes place to focus on a release's performance and stability.

The current LTS version is Systema-4.9.2.

Using files generated in another version

Any release of Systema can load files generated by older version. For instance, Systema-4.9.2 can open models from Systema-4.8.3. But the reverse operation is not allowed. To do that, Systema allows to export any file to the following releases :

- 4.5.1
- 4.8.3 :
 - Trajectory
 - Time position velocity arcs with ICRF data are removed. This conversion does not take into account the conversion of the external files (for instance, STK ephemerid file with ICRF ref frame)
 - Time position velocity arcs with Planet inertial data and earth centered are replaced by Gamma50/Veis1950 data
 - Time position velocity arcs with Planet inertial data and non-earth centered are removed
 - Kinematics
 - Kinematics bodies using ICRF axis pointing laws are removed
 - Ecliptic north kinematics laws are replaced by Gamma50 Z pointing laws
 - Kinematics bodies with 2 pointing vectors laws based on at least one ICRF axis are removed
 - Processing
 - Option "Processing Directory" used to build the result directory is replaced by "Default Result Directory"
 - Model
 - Conversion of radiative enclosures
 - Grid help item conversion
 - Suppression of transformations on cutters to avoid 483 issues
 - Meshing
 - Removal of solar lamp specific items

In order to export a file, use the "Save As" menu, and choose the type "Export : SYSTEMA ... - version 4.8.3 ...". If the exported entity is associated to other entities, you will be asked to export the associated files also. For instance. if you want to export a meshing, you will be asked to export the associated model and maybe the material. If you export a mission associated to a meshing, you will be asked to export the model two times (once for the mission and once for the meshing) so use the same file name the two times.

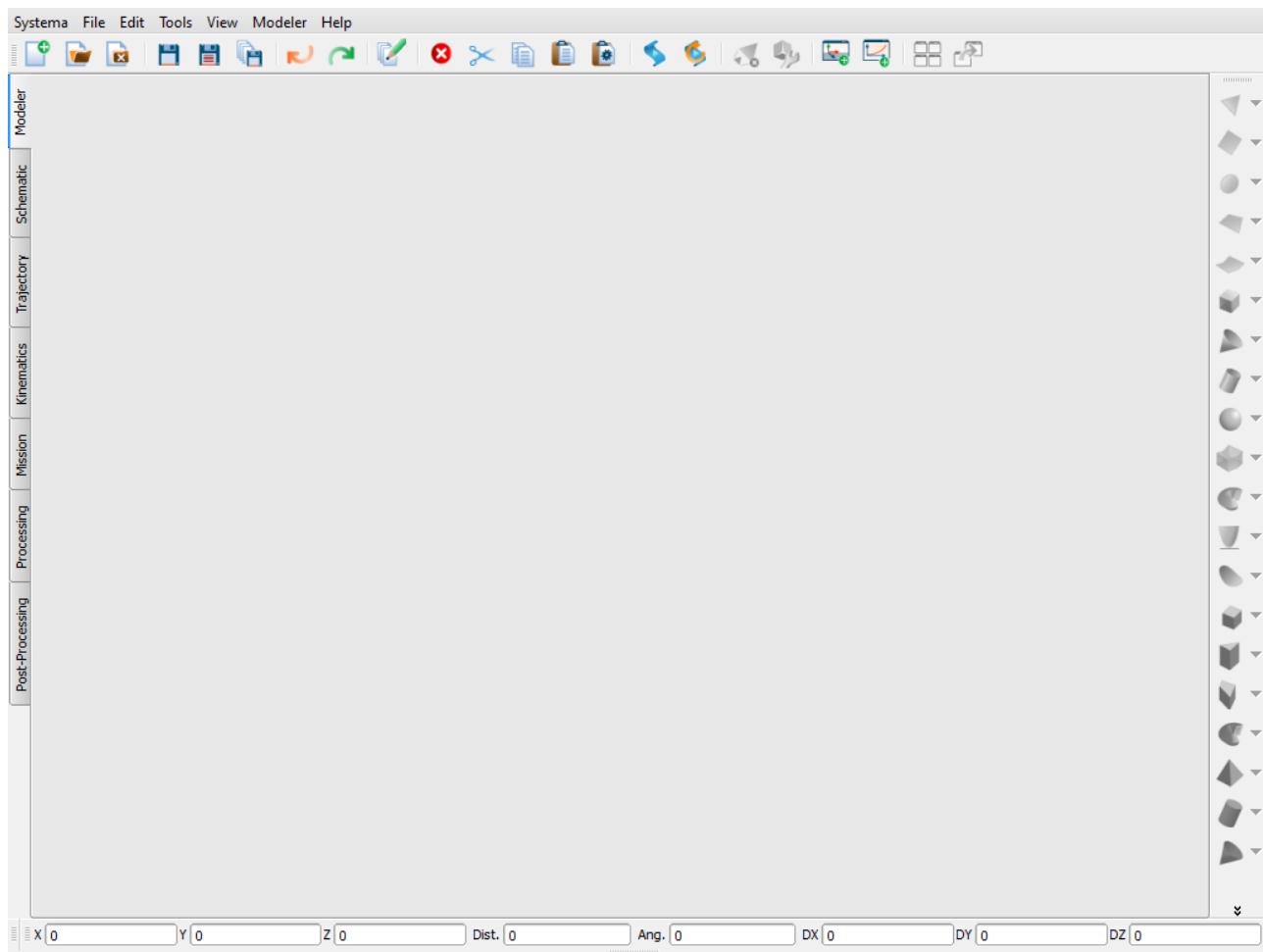
Starting and quitting SYSTEMA

How to start SYSTEMA

To start SYSTEMA:

- Run the main SYSTEMA binary file, available in your "**SYSTEMA_HOME**" directory.

Result: The SYSTEMA main window is displayed as follows:



How to quit SYSTEMA

To quit SYSTEMA:

- In the **Systema** menu, select the **Quit** option.

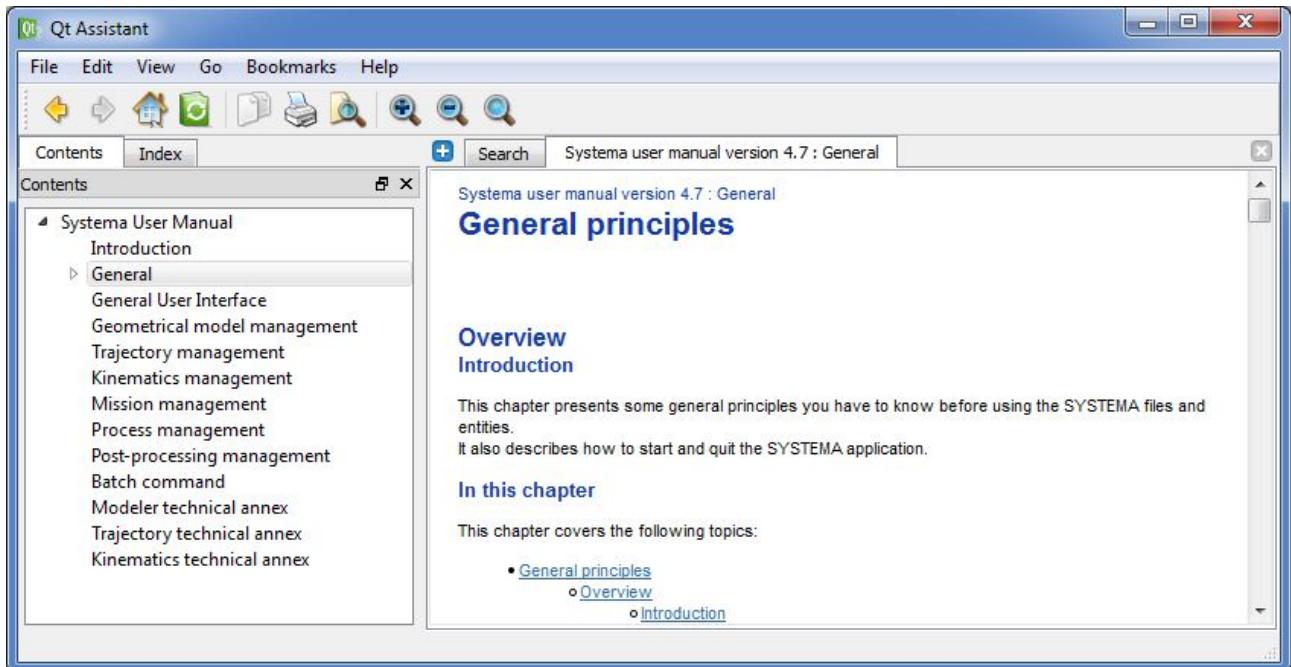
Result: If some entities have been modified during your session, dialog boxes ask you if you wish to save or not the opened entities.

SYSTEMA help

SYSTEMA offers an integrated help tool. It is used to display the user manual pages or search content in the help.

You can call the help tool by clicking:

- In the **Help** menu, select **Systema help**



To search in the help:

- In **Qt Assistant**, open the **Search Tab**
- Type something in the search field and click on **Search**

General User Interface

General User Interface

Overview

Introduction

This chapter describes the general ergonomical principles and basic procedures for using the SYSTEMA application.

In this chapter

This chapter covers the following topics:

- General User Interface
 - Overview
 - Introduction
 - In this chapter
 - Glossary
 - The SYSTEMA main window
 - General description
 - Example
 - Organization of SYSTEMA tabs
 - The menu bar of the main window
 - The icon bar
 - The log bar
 - The location bar
 - Description
 - How to define a point, a distance, an angle or a vector, using the location bar
 - Settings
 - What are the SYSTEMA settings used for?
 - Settings
 - Profiles
 - How to edit and share SYSTEMA settings?
 - Profiles management
 - List of available settings
 - User interface settings
 - File IO settings
 - Modeler settings
 - Scripting settings
 - Working with SYSTEMA tabs
 - What are the SYSTEMA tabs used for?
 - Defining entities through the Modeler, Trajectory, Kinematics and Mission tabs
 - How to open an existing entity
 - How to create a new entity
 - Definition and rules
 - The current entity
 - General principles for working on entities
 - Example:
 - Defining entity components
 - Prerequisite

- General procedure for defining an entity component:
- Select entity components
 - Rules
 - How to select entity components
 - Example
- Hide/show entity components
 - Rules
 - How to hide/show entity components
 - Example
- Working with the browser window
 - What is the browser window for?
 - Diagram
 - Definition: The current entity
 - General principles
 - Appearance of the browser window
 - Fixing/Detaching the browser window
 - Expanding / collapsing components
 - Display options for the entity components in the browser window
 - Managing components from the browser window
 - How to create a component from the browser window?
 - How to modify the organization of the components from the browser window?
 - Editing components from the browser window
 - Multi-Editing components from the browser window (available from Modeler Tab and for Meshing sub Tab only)
 - Search tool in the browser window
- Working with 3D views within the viewer area
 - What is the viewer area for?
 - Diagram
 - Organizing viewers
 - Display options for the entity components in 3D views
 - Handling objects in 3D views
 - Selecting a point in 3D views
 - Loading entities in 3D views and other viewports
 - Rules
 - How to load an entity in a viewport:
 - Using the configuration function
 - Recording a video
- Working with the timeline window
 - What is the timeline window for?
 - The timeline window's lines
 - Diagram
 - Date referencing another event
 - The resolution of the date reference
 - Attributes of an event
 - How to define the Event date
 - Special case : the Generated Event
 - Attributes of an event set
 - How to define the Intervals definition
 - How to define the Events computation method
 - Attributes of an interval
 - How to define the Start date
 - How to define the End date
 - Attributes of an interval set
 - How to define the Intervals computation method
 - Attributes of a date
 - Getting information on the displayed components
 - Time scale
 - Events
 - Intervals

- Managing components from the timeline window
 - How to select a component from the timeline window
 - How to create a component from the timeline window
 - How to delete a component from the timeline window
 - Editing components from the timeline window
- Managing the time scale
- Managing the date in the 3D views
 - How to synchronize 3D views?
 - How to set the date in the 3D view to a special event?
- Working with the edition window
 - What is the edition window for?
 - Component attributes
 - Diagram
 - General principles
 - Modifying component attributes within the edition window
 - How to change the component name?
 - How to add an attribute value?
 - How to remove an attribute value?
 - How to undo/redo editing actions
 - To undo an editing action (from the SYSTEMA main window or the edition window):
 - To redo an editing action (from the SYSTEMA main window or the edition window):
- Working with Python scripts
 - What can be done with Python scripts ?
 - Executing a Python script
 - For more information...
- Working with variables
 - What are variables made for ?
 - Variables edition
 - Variables naming
 - Variables syntax
 - Variables assignment
- Appendices:
 - Shortcuts Summary

Glossary

In this User Guide, some terms have a specific meaning:

Term	Specific meaning
" entity " or " SYSTEMA entity "	Generic term used to refer to any of the objects managed by the SYSTEMA application, namely geometrical models , schematics , trajectories , kinematics , missions .
" component " or " entity component "	Generic term used to refer to any element of a SYSTEMA entity, within its tree structure. For example, geometrical models ("SYSTEMA entities") are made of "components" which may be objects or shapes.

The SYSTEMA main window

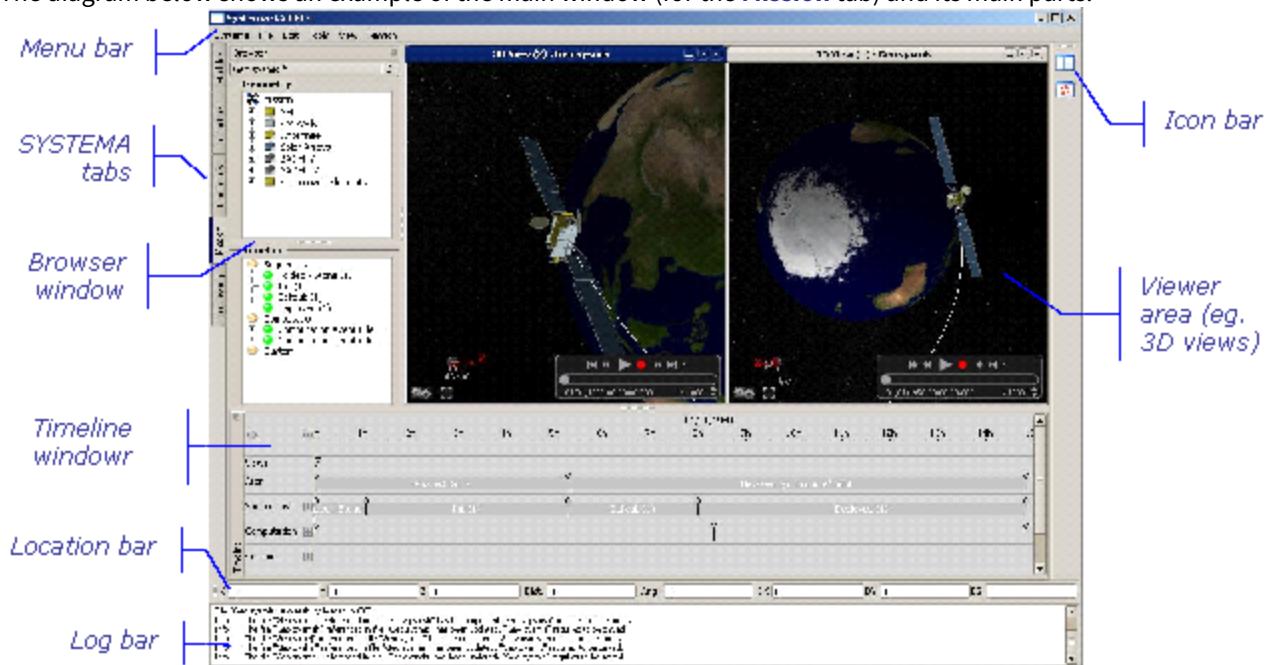
General description

The SYSTEMA main window is organized into tabs, which are used to manage SYSTEMA entities:

- the **Modeler** tab is used to create and display geometrical models and meshings and to display processing results;
- the **Schematic** tab is used to create and display electronic diagrams;
- the **Trajectory** tab is used to create and display trajectories;
- the **Kinematics** tab is used to create and display kinematic models;
- the **Mission** tab is used to define missions (associating a model to a trajectory and a kinematics) in order to run simulations;
- the **Processing** tab is used to define and run processes.
- the **Post-Processing** tab is used to define and run post-processing.

Example

The diagram below shows an example of the main window (for the **Mission** tab) and its main parts.



Organization of SYSTEMA tabs

The **Modeler**, **Trajectory**, **Kinematics** and **Mission** tabs are all organized in the same way (you can see in the example above):

- A browser window, used to display hierarchical views of loaded entities (see the detailed description in the following pages);
- A viewer area made of viewport(s), used to display different types of views, such as 3D and others (specific to each tab).

The **Trajectory** and **Mission** tabs have an additional area :

- A *timeline window*, used to display all the components linked to the time such as orbital arcs, eclipse and penumbra intervals.

The second browser window is used to manage the timeline components in the Mission tab but is not usefull in the Trajectory tab.

The **Schematic** tab has a specific content; it is described in the "Schematic" chapter.

The **Processing** and **Post-Processing** tabs have a specific content; it is described in the "Running processes" chapter.

The menu bar of the main window

For each tab, the menu bar consists of both common and specific menus. The common menus are as follows:

- The **File** menu (contains options used to open, create and close SYSTEMA entities):

Option	Use
New	Creates a new entity (according to the current tab)
Open	Opens a previously saved entity
Close	Closes the current entity (dialog boxes are displayed to enable you to save modifications and confirm closing)
Save / Save as...	Saves the current entity in a file
Save all	Save all the opened entities in files
Archive	Archive the current file and its referenced files in a new directory
Model and sub options	<i>not available for trajectories, kinematics and missions</i> Creates, opens, closes, saves a model.
Meshing and sub options	<i>not available for trajectories, kinematics and missions</i> Creates, opens, closes, saves a meshing associated with a model
Material and sub options	<i>not available for trajectories, kinematics and missions</i> Creates, opens, closes, saves a material database
Mission log ...	<i>Available for missions</i> , generates a log file containing all the mission description
Quit	Closes the application (if modifications have been made, SYSTEMA asks if you want to save them)

- The **Edit** menu (contains options applicable to entity components):

Option	Use
Undo	<i>not available for schematic</i> Cancels the last editing action
Redo	<i>not available for schematic</i> Cancels the last Undo action
Edit selection	Opens the edition window of the selected component
Multi Edition	Opens the multi-edition window of the selected components. This option is active in the Modeler tab and for Meshing sub-tab only.
Delete	Removes definitively the selected component(s) <i>For missions, only act on timeline components</i>

Cut	<i>not available for missions and schematic</i> Removes the selected component(s) and places it in a buffer in order to paste it if necessary
Copy	<i>not available for missions and schematic</i> Copies the selected component(s)
Paste	<i>not available for missions and schematic</i> Pastes the last copied/cut component(s)
Unselect	Cancels all component selections in all entities

- The **Tools** menu (contains an option to launch a text editor):

Option	Use
Launch python script	Launch the selection and the execution of a python script. See <i>Working with Python scripts</i> in <i>Geometrical Model management</i> chapter for more information
Relaunch python script	Relaunch the execution of the last selected python script.
Open python console	Open the python interactive console.

- The **View** menu (contains options used to organize the different parts of the main window):

Option	Use
Add	Adds a viewer in the viewer area
Close	Closes the current viewer
Arrange	Displays the different views in cascade mode or in tiled mode
Location	<i>checked by default</i> Check/Uncheck it to display/hide the location bar Not defined for Trajectory , Mission and Processing tabs
Icons	<i>checked by default</i> Check/Uncheck it to display/hide the icons bar
Log	<i>unchecked by default</i> Check/Uncheck it to display/hide the log bar
Timeline	<i>checked by default</i> Check/Uncheck it to display/hide the timeline window Reserve to the Trajectory and Mission tabs
Meshing browser	<i>unchecked by default</i> Check/Uncheck it to display/hide the meshing browser Reserved to the Mission tab

- The **Help** menu:

Option	Use
Systema help	Open the Systema help in a new Window. The page opened is linked to the actual tab.

Option	Use
Systema Python API Help	Open the Python API Help at the main page in a web browser. A default web browser needs to be set to use this option.
About Systema	Gives information about the version of SYSTEMA
	<i>Note: The use of specific menus (Modeler/Schematic/Trajectory/Kinematics/Mission/Processing/Post-Processing) is described in the chapter relating to the concerned module.</i>

The icon bar

The icon bar contains different icons, depending on the display context:

Icon	Use
	Tiles the views using all the available space in the screen
	Shows the next view
	<i>Note: More icons may be provided in SYSTEMA tabs for creating basic elements interactively.</i>

The log bar

The log bar contains all messages provided by SYSTEMA since the application was launched.

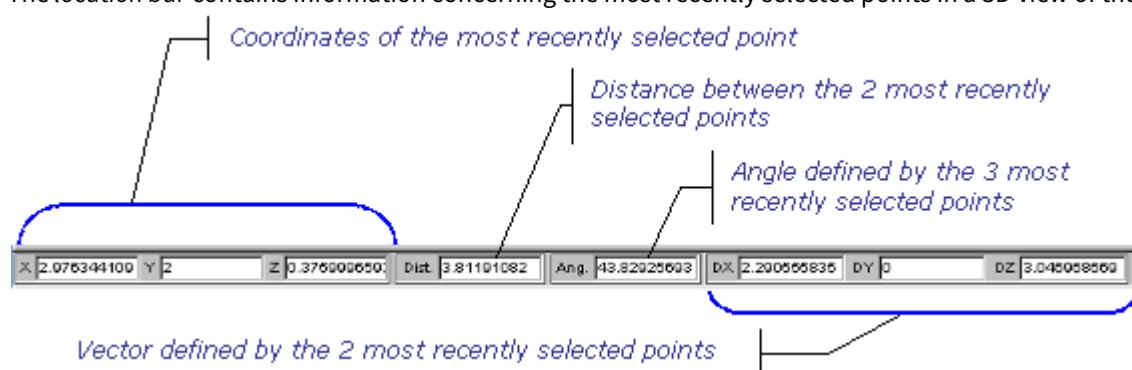
The location bar

The *location bar* is displayed within the **Modeler**, **Kinematics** and **Mission** tabs, above the log bar.

The *location bar* contains information concerning points selected in the **Modeler** tab and that can be used to enter point coordinates via the edition window of an entity.

Description

The *location bar* contains information concerning the most recently selected points in a 3D view of the **Modeler** tab.



How to define a point, a distance, an angle or a vector, using the location bar

The values in the *location bar* can be used when entering the attributes of entities:

- the **X**, **Y** and **Z** values, to enter coordinates,
- the **Dist.** value, to enter a distance,
- the **Ang.** values, to enter an angle,
- the **DX**, **DY** and **DZ** values, to enter a vector coordinates.

Refer to "How to add an attribute value?", page in the "Working with the edition window"

Settings

What are the SYSTEMA settings used for?

Settings

SYSTEMA settings are some global user parameters that are defined once and are persisted even after Systema restarts.

They apply to various aspects of the software (GUI displays, file import/export, etc) but can never depend on a specific file.

Profiles

A given set of settings values is associated to a profile.

Profiles contain two sets of settings values:

- Current values: these values can be edited by the user
- Default values: these values are the one that Systema assign to the current settings when the "Reset" function is called

Profiles are stored as .ini files in the directory where persistent application data can be stored. For instance:

- \$HOME/.local/share/data/Airbus Defence And Space/Systema/profiles on Linux
- %HOMEPATH%\AppData\Local\Airbus Defence and Space\Systema\profiles on Windows

This location can be defined with SYSTEMA_PROFILES_DIR environment variable.

 To use profiles available in SYSTEMA_INSTALL_DIR/data/profiles, please import them (see § Profiles management)

How to edit and share SYSTEMA settings?

Settings can be edited from the Settings window which is accessible from the **File>Settings menu**.

A "default" profile is selected at the first Systema run. This specific profile cannot be deleted.

The default values of this profiles are defined by Systema itself and cannot be modified.

Nevertheless, the current values of the default profile can be modified.

Users switch from one profile to the user using the profiles combo box.

Profiles management

Profiles can be edited using the "**Manage profiles**" window, accessible from the Settings window by clicking on "**Manage profiles**".

Option	Description
New profile	Create a new profile. At first, it will contain the current values of the default profile.
Import profile	Import an existing profile from an .ini file. If a profile with the same name already exists, a confirmation message will be displayed.
Export profile	Export the current values of the selected profile in a .ini file The default values of the generated profile will correspond to the current values of the selected profile.
Delete profile	Delete the selected profile after confirmation
Duplicate profile	Duplicate the selected profile

To rename a profile, just double-click on it and enter a new name (not an empty or existing one).

 Profiles are no longer shared by all users.

List of available settings

The following list of existing settings is subject to evolutions along the Systema versions.

Nevertheless, it is possible to reuse a profile file from an older version.

In this case, the settings that were added will take the default values defined by Systema.

User interface settings

- Tabs can be hidden/shown using the "Tabs display" setting
- The 3D optimization algorithm can be enabled/disabled using the "Enable optimization" setting.

 This setting requires the creation of a new 3D view to be effective (enabled: 

 It is highly recommended to close and reopen any 3D views to take into account this modification

- The font size (10 by default).
 - When opening 3D views (mission, trajectory, kinematics, meshing with results), the player can be displayed in 3D views in elapsed time (sec). By default, this option is turned off.

Ref: UM.000041864.AIRB

© Airbus Defence and Space SAS 2022 - All rights reserved

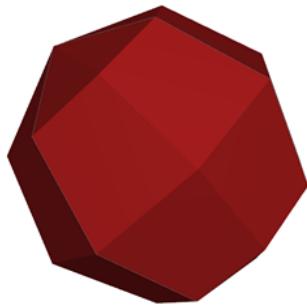
24

⚠ It is highly recommended to close and reopen any 3D views to take into account this modification

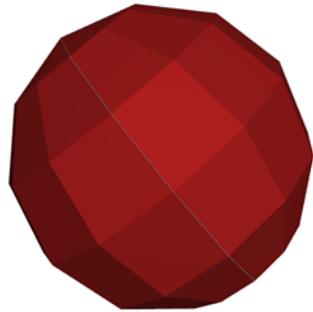
File IO settings

- The chord (c) and angle (a) tolerances used for CAD model tessellation can be tuned using the Step import chord/angle tolerances settings
Depending on the values you choose for these parameters, the resulting 3D display of step models will change as shown in the images below (same model with different tolerances).
The results computations are not affected.

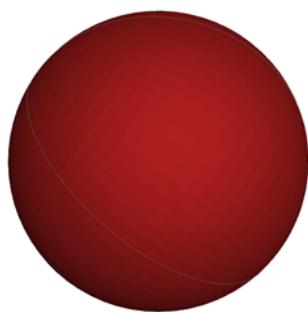
$c = 4 \text{ mm}, a = 60^\circ$



$c = 1 \text{ mm}, a = 60^\circ$



$c = 1 \text{ mm}, a = 5^\circ$



- Each time a STEP-TAS file is opened or saved, its validity regarding the STEP-TAS format is checked. This validation can be turned off/on with the settings STEP-TAS / "Disable Step-TAS validation"
- STEP-TAS is very strict when defining quadrangles, an option can be enabled/disabled to correct the coplanarity automatically before exporting the model in STEP-TAS format. It will apply the same correction as the script "correctQuadrangleDNVersion.py" available in the installation directory of Systema "python\publicScripts\".

❗ This correction is performed on the model

- An optional warning can be activated to warn you when you override the version of a file during the Save/SaveAs operation
"Be warned when file version is updated during Save"
- The saving date is usually stored in Systema files (except sysset files). It can be deactivated through the "Store the saving date in Systema files" option. The main purpose of this option is to ease the configuration management of Systema files.
- An optional warning can be raised when a Systema file (model, meshing, material, trajectory, schematic, kinematic, mission, processing, post-processing) is modified outside Systema.
Three options are available to manage this case :
 - Do not watch (default value)
 - Watch and raise a message in the log area
 - Watch and raise a popup message in the log area

A warning can be raised when a file has been modified, renamed or removed from disk. Monitoring files is stopped once files have been renamed or removed from disk, and directories once they have been removed from disk. By default, modifications of files are not watched at all. Activating this monitoring is not taken into account by already opened files.

Watching modifications is highly dependant to the operating system. For example, on Linux it requires that the kernel supports inotify and that files are not stored on remote mount path (such as NFS, samba, ..).

- The number of recent files displayed in "File" menus can be set. Default is 5.

⚠ It is highly recommended to close and reopen Systema to take into account this modification

Modeler settings

- Shapes and Cutters creation menu can be hidden/show using the "Menu visibility" setting
- The interactive mode to create shape can be set/unset as a continuous mode
- The default values of the visibility and selection options of the modeler 3D view can be set

Scripting settings

- The max buffer size for the getLastSelectedPoints(N) function (default is 20). This buffer size must be between 1 and 100 otherwise it will not be taken into account. Changing this setting will reset the buffer.
- Favorite scripts can be defined and displayed in the python toolbar in all tabs or in dedicated ones. Only python scripts are supported.

How to define a favorite script

- In the scripting settings tab, select the tab where you want to install the favorite script
- Click on **Add script**
- A new script is created. In the formular, you can define :
 - the script file name. You can select the file by clicking on "..." button next to the script field or by entering directly the filename. The filename shall be an absolute one. You can use the \$(INSTALLDIR) keyword to reference the Systema installation directory. Please note that if Systema is launched with a shortcut, \$(INSTALLDIR) will reference the path where the shortcut is located. The favorite script will be displayed whatever the validity of the script filename.
 - the icon filename. You can select the file by clicking on "..." button next to the script field or by entering directly the filename. The filename shall be an absolute one. Only PNG files are supported. By default, the python standard icon is used.
 - the description. If this information is filled, it will be used as a tooltip of the icon otherwise the name of the favorite is used.

Favorite script created in "All tabs" section will always be displayed in Systema whereas the other favorite scripts are only displayed when the associated tab is selected. You can easily move a favorite script from a tab to another one by drag and drop.

How to remove a favorite script

- In the scripting settings tab, select the script you want to remove
- Click on **Remove script**
- Confirm the deletion.

Please note that removing a favorite script removes only the item in the python toolbar. There is no impact on the associated files.

Working with SYSTEMA tabs

What are the SYSTEMA tabs used for?

Each SYSTEMA tab is used to define entities (model, schematic, trajectory, kinematics, mission, processing and post-processing), and components of entities.

In each SYSTEMA tab, you can:

- open an existing entity (previously saved in a file),
- create a new entity,
- define an entity's components.

Defining entities through the Modeler, Trajectory, Kinematics and Mission tabs

How to open an existing entity

1. Display the tab corresponding to the entity to be opened.
 2. In the **File** menu, select the **Open** option and select the file containing the required entity.
- The selected entity is then displayed in two areas:

- In the browser window, the hierarchical organization of the selected entity is displayed.
- In the viewer area, a new 3D view is opened with the selected entity displayed in it.

3. You can also drag and drop directly the files with the extension defined in [The different entities and corresponding files](#).

How to create a new entity

1. Display the tab corresponding to the entity to be created.
 2. In the **File** menu, select the **New** option and select the file containing the required entity.
- Result:

- In the browser window, the folder root of the new entity is displayed.
- In the viewer area, a new 3D view is opened in which the entity components will be visualized.

Definition and rules

The current entity

At any moment, the entity displayed in the browser window is called the current entity.
This entity is the one which will receive component creations performed within the current tab.

General principles for working on entities

To work with an entity, you have first to open it (or create it).



Note: In a given tab, several entities can be opened at the same time.

You can work on an entity, either from the browser window or from a viewport.

Reflected actions in the browser window and viewports

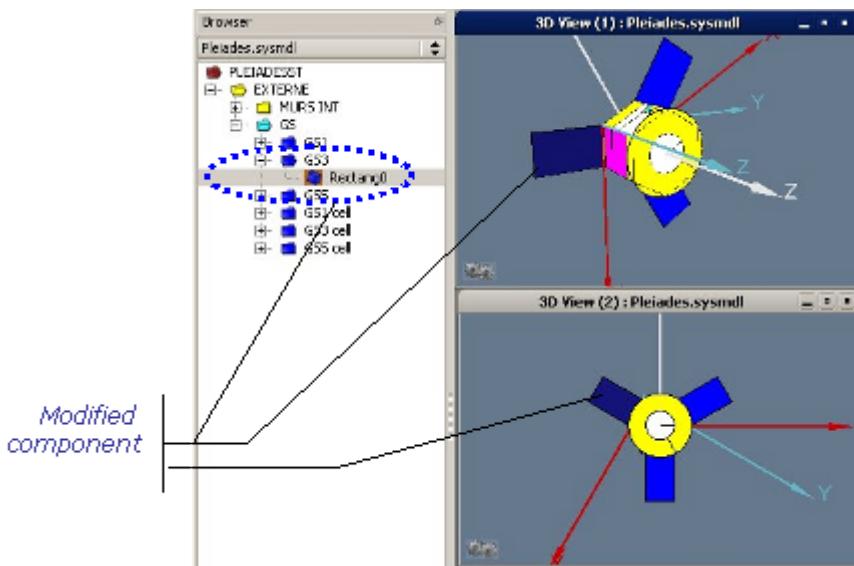
In a given tab:

- *modifications* made to an entity are reflected anywhere the entity is displayed;
- *component selections* are also reflected anywhere the entity is displayed;
- *hidden/shown components* are reflected anywhere the entity is displayed.

Example:

In this example, the color of a model component has been changed. This modification is visible:

- in the browser window,
- in a first 3D view where the model has been loaded,
- in another 3D view where the model has also been loaded together with another model.



Defining entity components

Prerequisite

The concerned entity must be opened in the related tab.

General procedure for defining an entity component:

1. Create the component.

You can process it either through the browser window (see "Working with the browser window" in the next section), the timeline window (see "Working with the timeline window") when it is defined, the menu bar (via specific menu), or the icon bar.



Note: Components created through the menu bar or the icon bar are added to the current entity.

2. Modify the component by means of the edition window.

Select entity components

This action may be applied either from the browser window, from the timeline window when it is defined or from a 3D view containing the concerned entity.

Rules

When entity components have been selected, they appear selected wherever they are displayed (browser window, timeline window or 3D view).

The options in the Edit menu of the menu bar apply to selected elements.

How to select entity components

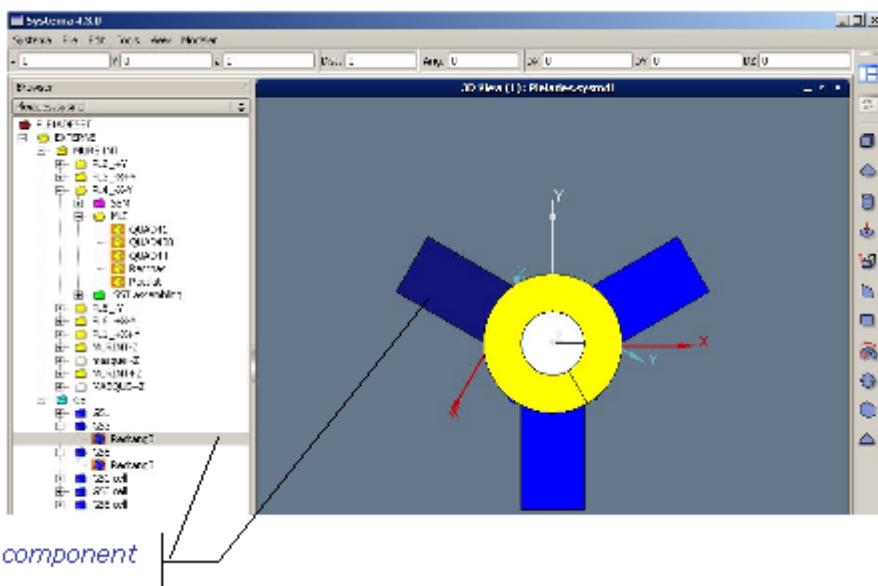
From the browser window, the timeline window or from a 3D view:

- Select the required component by clicking it.



Note: You can select more than one components (simultaneously hold down the Ctrl key and left click with the mouse) from any displayed view.

Example



Hide/show entity components

This action may be applied either from the browser window or from a 3D view containing the concerned entity.



Note: Hidden components are displayed in italics in the browser window.

Rules

When an entity component has been hidden/shown, it appears with the same characteristics (ie, from an hidden/shown point of vue) wherever it is displayed (browser window or 3D view).

When the hide/show action is applied to a container (from the browser window), it is also applied to all its components.

How to hide/show entity components

From the browser window or from a 3D view:

1. Select the component to be hidden/shown.

You can select more than one components (simultaneously hold down the *Ctrl* key and left click with the mouse).

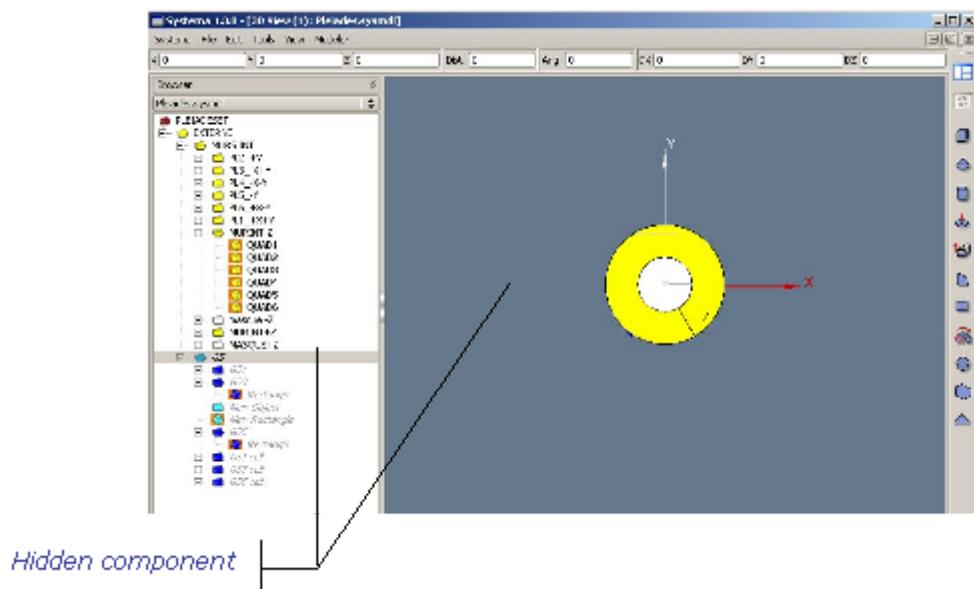
2. In the right click menu, select the appropriate option.

The hide/show option may be one of the following:

Option	Use

Hide	Hides the selected component, and all components it contains
Hide others	Hides all components excepted the selected components and components it contains
Hide all	Hides all components of the current entity
Show	Shows the selected component Only available from a component in the browser window
Show all	Shows all components of the current entity

Example



Working with the browser window

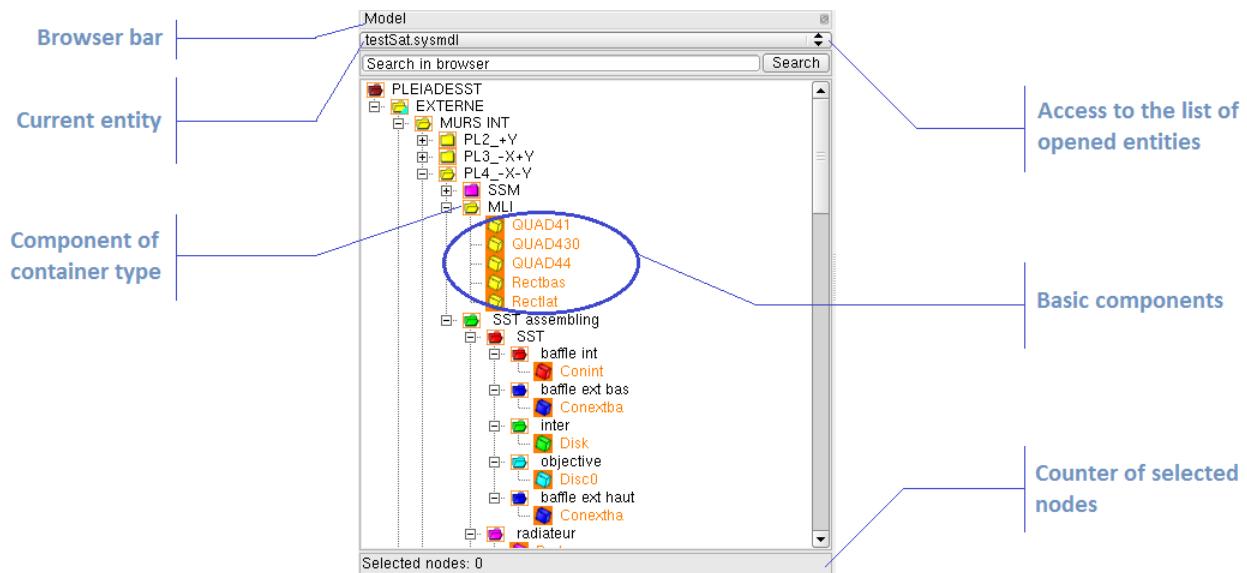
What is the browser window for?

The browser window is associated with the **Modeler**, **Trajectory**, **Kinematics** and **Mission** tabs.

The browser window is used to show an entity (the current entity) in the form of a tree structure. From this window, you can consult, create or modify the entity components.

The browser window is displayed in a tab as soon as an entity has been loaded in the corresponding tab.

Diagram



Definition: The current entity

The entity which is displayed in the browser window is called the **current entity**.

All actions that are executed from the browser and from anywhere else concern the current entity.

To change the current entity:

- In the browser window, select an entity from the list of values below the menu bar (e.g., in the example above).

General principles

All entities that have been opened (newly created or existing ones) can be accessed via the browser window. The browser is navigable through both keyboard and mouse inputs.

Appearance of the browser window

Fixing/Detaching the browser window

By default, the browser window is a fixed window.

You can detach the browser window from the main window. The browser window is a floating window that can be moved anywhere in the screen. This leaves the entire main window available for the viewports.

To detach the browser window from the main window:

There are different ways:

- In the browser window, click in the browser bar or double-click the browser bar.
or
- Click the browser bar and drag the browser window into the main window.

To fix the browser window:

There are two different ways:

- In the browser window, double-click the browser bar.

or

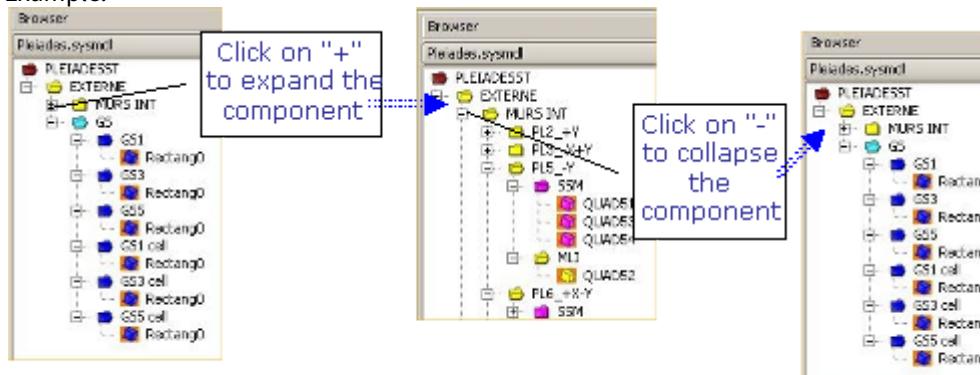
- Click the browser bar, drag and drop the browser window to the left or to the right of the main window.

Expanding / collapsing components

In order to facilitate consultation of the tree structure, you can expand or collapse components:

- expand ***all components of every level*** of the selected component, using the **Expand all** option from the right click menu;
- collapse ***all components of every level*** of the selected component, using the **Collapse all** option from the right click menu;
- expand the selected component, by clicking "+" or pressing the Right key;
- collapse the selected component, by clicking "-" or pressing the Left key.

Example:



Display options for the entity components in the browser window

The browser window can be used to modify the display of the current entity.

You can:

- select component(s),
- hide/show entity components.

Managing components from the browser window

The browser window is used to manage entity components, i.e.:

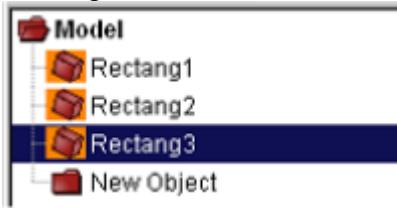
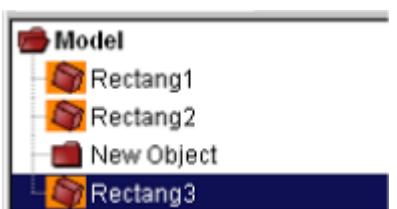
- creating components,
- modifying the organisation of components (cut, copy and paste).

How to create a component from the browser window?

1. In the browser window, select a component (from which the new component will be created).
2. In the right click menu, select the **Create object** option.
3. In the case of a model component, select the component type.
4. If necessary, select the hierarchical position of the new component.

The position may be one of the following:

Option	Use	Example of result

In	<i>Available if the selected component is the entity root or a container</i> Use it to create a component as the child of the selected one	"New Object" is created in "Object1": 
After	<i>Available if the selected component is any component other than the entity root</i> Select it to create a component as the brother of the selected component, just after the selected one	"New Object" is created just after "Rectang3": 
Before	<i>Available if the selected component is any component other than the entity root</i> Select it to create a component as the brother of the selected component, just before the selected one	"New Object" is created just before "Rectang3": 
	<i>Note: The position options are only available when there is more than one possible position for the new component</i>	

The new component is created with default attributes. It can be modified via the edition window.

How to modify the organization of the components from the browser window?

To move a component within the entity structure:

1. In the browser window, select the component to be moved.
 2. In the right click menu, select the **Cut** option.
 3. Select the component in which you want to place the *cut component*.
 4. In the right click menu, select the **Paste** option to insert the *cut component*.
- The previously *cut component* appears, in the entity structure, at the indicated place.

- if you select a container, the *cut component* is inserted after the last child;
- if not, the *cut component* is inserted in sequence after the selected component.

To copy/cut a component within the entity structure:

1. In the browser window, select the component to be moved.
2. In the right click menu, select the **Copy/Cut** option.

You can then put the copied/cut component within an entity structure using the **Paste** action.

Editing components from the browser window

The browser window can be used to edit a component in the edition window. You can then modify the component attributes.

To display the edition window for a component:

- Use one of the following:

Double click the component, in the browser window or in one of the viewports.

or

Select the component, then select the **Edit selection** option either in the **Edit** menu of the main window or in the browser window.

or

Right click the concerned component and select the **Edit** option.

Multi-Editing components from the browser window (available from Modeler Tab and for Meshing sub Tab only)

The browser window can be used to edit meshing attributes for a selection of more than one component. This action is called Multi edition.

To display the Multi edition window for a list of components:

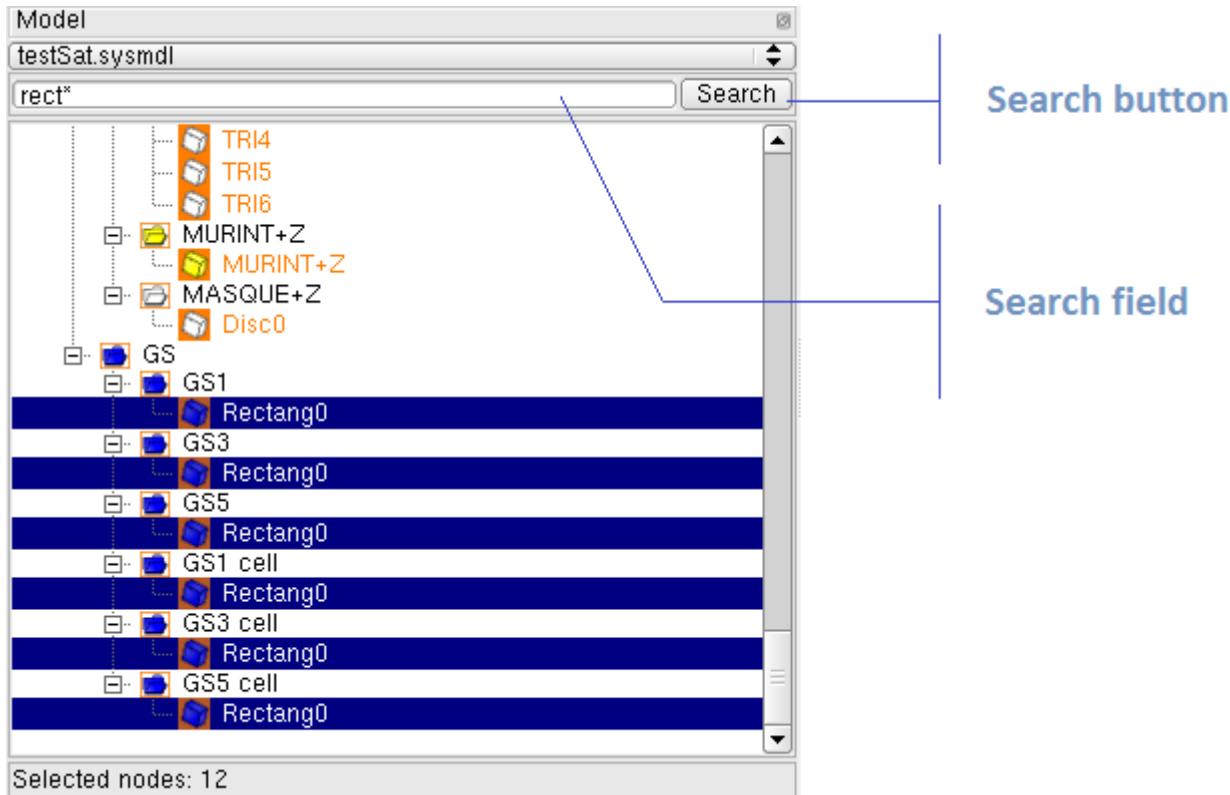
- Select components, in the browser window or in one of the viewports by using the CTRL key.
- Open the multi edition window by selecting the  **Multi Edition** option either in the **Edit** menu of the main window or in the browser window, or use CTRL+SHIFT+E shortcut.



Note : Multi Edition is restricted to objects and shapes with the same type (Shell, volumic or faces). The Log view will show an error message if this restriction is not satisfied.

Search tool in the browser window

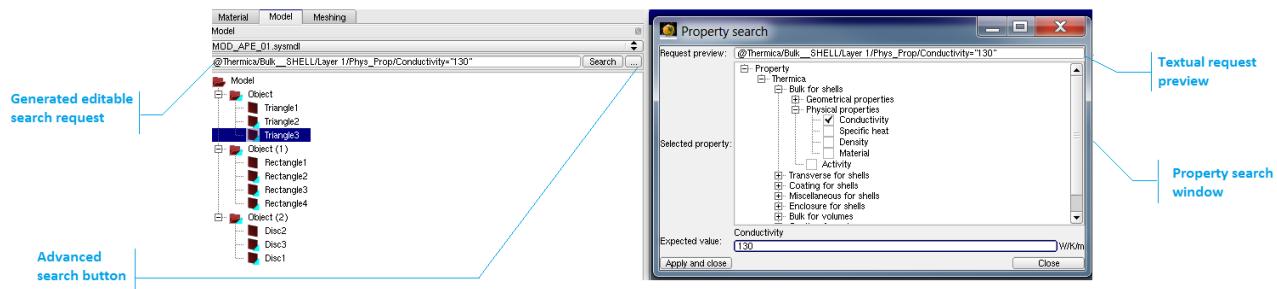
The browser window can be used to find a component matching a given criteria in the opened tree. The search tool is accessible in all Systema browsers. The components that matches the given criteria is **selected** in the browser.



- The search can apply on nodes names in a case insensitive way. Name can be searched with regular expressions including the following operators: *, ?, \[-\] and \.

Here are some search examples using regular expressions:

- find all components which name starts by "MLI": **MLI*** returns: "MLI_1", "MLI12"
- find all components which name contains "enn": ***enn*** returns: "Antenna", "Antenna123"
- find all components which name is exactly "Antenna1", "Antenna2", "Antenna3", "Antenna4", "Antenna5": **Antenna[1-5]**
- find all components which name is "ANtennaA", "AnteNna5": **Antenna?**
- The assignment of Systema variables in nodes can be checked through the search tool by typing "~" + variable_name
Example: **~myVariable** will look for every nodes where "myVariable" is assigned
- In the modeler, it is possible to search for applicative characteristics (property, activity or material). These complex search request does not require to be manually typed. They can be easily built through the **Property search window** (accessible through the **Advanced search button**). Once selected in the applicative search window, an editable search request is automatically generated as follows:
 - find all components with a thickness property equal to 0.001: **@Thermica/Bulk__SHELL/Layer 1/Geom_Prop/Thickness="0.001"**
Rmk: Note that it is possible to change the "=" operator to ">", "<", "<=" or ">=" to work with ranges: **@Thermica/Bulk__SHELL/Layer 1/Geom_Prop/Thickness>="0.001"**
 - find all components with a "Both Sides" activity: **@Thermica/Bulk__SHELL/Activity="BOTHFACE"**
 - find all components with "Ones" material as physical property: **@Thermica/Bulk__SHELL/Layer 1/Phys_Prop/Material="Ones"**



- In the meshing, it is possible to search for a numerical range using the following syntax: #<min value>-<max value> (ex: #100-150).
This will select the meshes which are well numbered.

Finally, the user may perform logical combinations of search criterion using the AND ("&&") and the OR ("||") operators, as shown in the following examples:

- #1000-2000&&*0 will search for every numerical nodes between 1000 and 2000 which finish with a zero decimal digit
- Triangle*||Rectangle* will look for every node which have a name starting by either "Triangle" or "Rectangle".

Working with 3D views within the viewer area

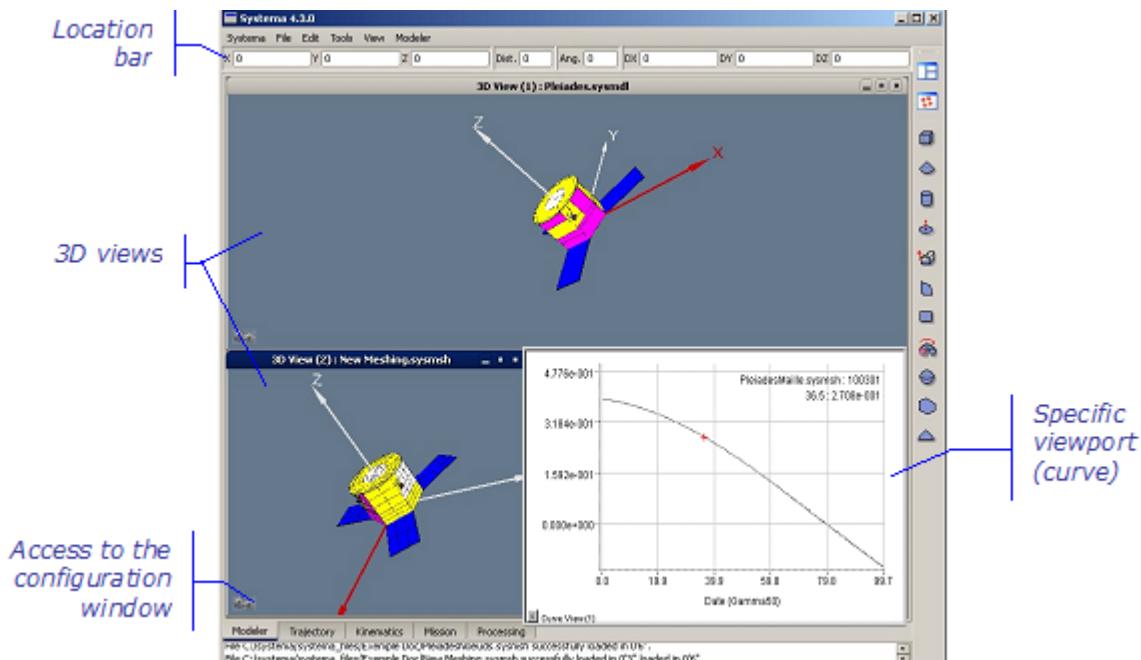
What is the viewer area for?

The viewer area is used to show entities in 3D views or some other specific viewport. It is associated with the **Modeler**, **Trajectory**, **Kinematics** and **Mission** tabs.

A new 3D view is displayed in a tab as soon as an existing entity has been opened or a new entity has been created in the corresponding tab.

Empty 3D views can be created, into which entities can then be loaded.

Diagram



Organizing viewers

Several viewers can be displayed.

The **View** menu of the menu bar can be used to:

- add/close viewers (3D views or specific views),
- display/hide any viewer.

Display options for the entity components in 3D views

In a 3D view, you can modify how the entity is displayed.

- Using options from the right click menu, you can:
- hide/show entity components.
- fit the content to the window.
- adjust the camera parameters (**Camera** option).
- create geometrical objects to help you to create points and new surfaces (**Help items** option).
- configure the content of viewports (**Configuration Window** option).
- copy to the clipboard the displayed 3D view.
- Using the mouse (see "Handling objects in 3D views" below)

Handling objects in 3D views

If you want to ...	Perform the following action....	
Select a basic component	<ul style="list-style-type: none"> • <i>Left</i> click the component To select several non-consecutive components, click the first one, then hold down the <i>Ctrl</i> key while clicking each additional element 	
Open the edition window for a component	<ul style="list-style-type: none"> • Double <i>left-click</i> the component 	
Rotate an entity in 3D	<ul style="list-style-type: none"> • In the 3D view, hold down the <i>left</i> mouse button while moving the mouse according to the required entity rotation 	
Spin the view	<ul style="list-style-type: none"> • In the 3D view, simultaneously hold down the <i>Ctrl</i> key and the <i>left</i> mouse button while moving the mouse 	
Disable rotation / spinning of the view	<ul style="list-style-type: none"> • In the 3D view, click on the locker icon 	
Display the contextual menu	<ul style="list-style-type: none"> • <i>Right</i> click in the 3D view 	
Zoom in	<ul style="list-style-type: none"> • In the 3D view, hold down the <i>right</i> mouse button while moving the mouse forward 	

	<i>Ctrl+Shift+left mouse button can be used instead of the right button</i>	
Zoom out	<ul style="list-style-type: none"> In the 3D view, hold down the <i>right</i> mouse button while moving the mouse backward <i>Ctrl+Shift+left mouse button can be used instead of the right button</i> 	
Move the entity without rotating it	<ul style="list-style-type: none"> In the 3D view, hold down the <i>middle</i> button while moving the mouse <i>Shift+left mouse button can be used instead of the middle button</i> 	
Move the camera using keyboard	<ul style="list-style-type: none"> Up directional arrow in order to go forward in the direction of the focal point Down directional arrow in order to go backward in the opposite direction of the focal point Right directional arrow in order to turn right, around the focal point Left directional arrow in order to turn left, around the focal point Page up in order to go up Page down in order to go down 	

Selecting a point in 3D views

To select a point in 3D views, simultaneously press the *Alt* key and *left* click the point.

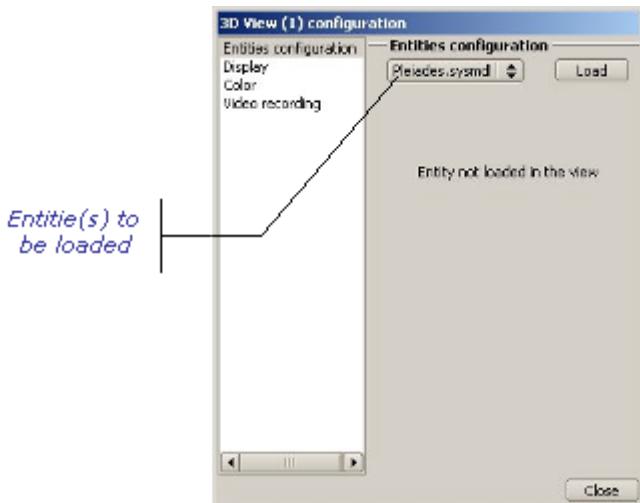
Loading entities in 3D views and other viewports

Rules

In a viewport of a tab, you can load entities which have been opened.
Several entities may be loaded at the same time in the same 3D view.

How to load an entity in a viewport:

1. If necessary, create a new viewport.
2. Use one of the following:
 - In the right click menu of the viewport, select **Configuration window**



- select the entitie(s) to be loaded
- Click **Load**

The selected entity is then displayed in the 3D view.

or

- use the configuration window (see "Using the configuration function" below).

Using the configuration function

The configuration window is used to configure the content of viewports. It allows to:

- specify an entity to be loaded
- select visualizable information to be displayed for the loaded entity
- set recording parameters
- modify the background and text colors

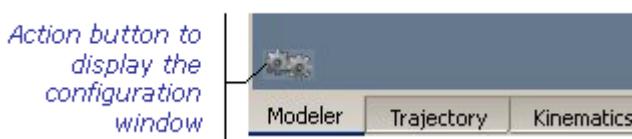


Note: In a configuration window, you can only load entities which have been opened.

To apply a specific configuration to a viewport:

1. In the concerned viewport:

- Click at the bottom of the viewer



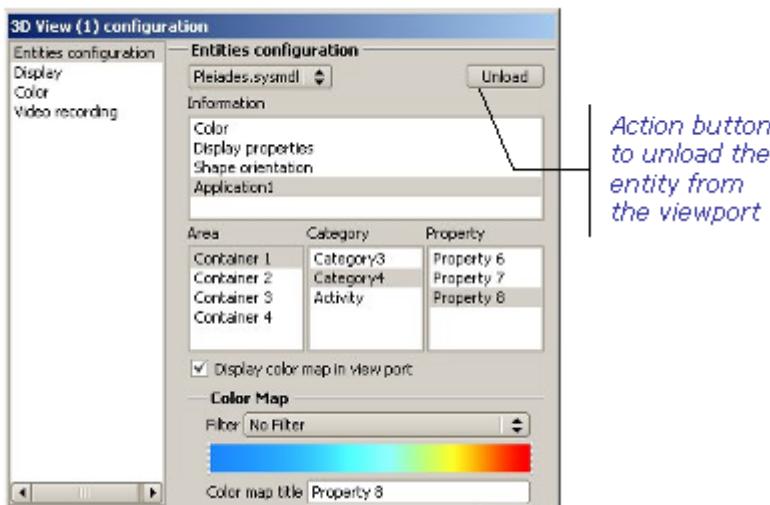
or

- In the right click menu of the viewport, select **Configuration window**

Result: The configuration window is displayed. It contains the loaded entity and the available attributes.

2. Select the information you want to display in the viewport.

Example of configuration window after selecting a property related to an application:



Result: The selected attribute is immediately applied in the viewport.

3. Click **Close to close the configuration window.**

Result: In the viewport, you can see the configuration result.



Note: The configuration window can be closed only when a valid configuration has been chosen.

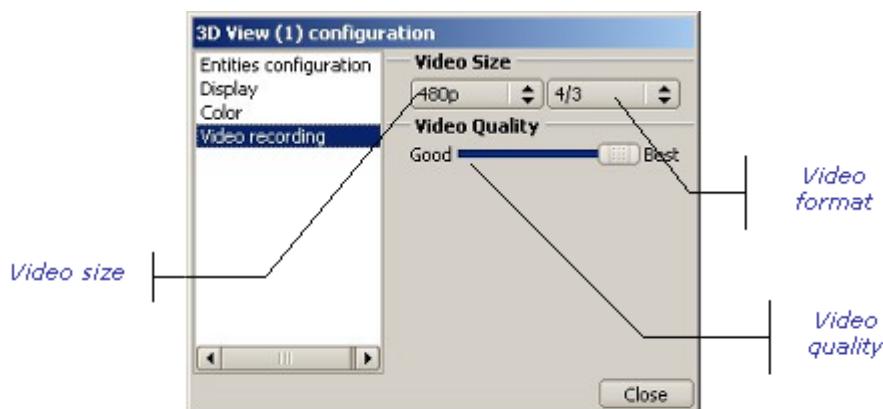
Recording a video

To configure the video recording



Note: The video recording is relevant for entities such as model meshing, trajectory, kinematics, mission.

1. In the concerned viewport, display the configuration window (see above "Using the configuration window").
2. Select **Video recording**.

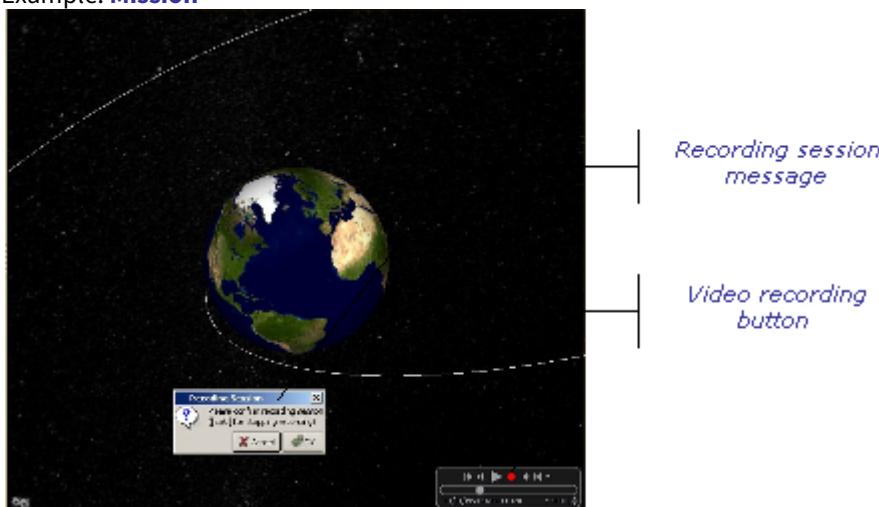


1. Select the video size (480p, 720p, 1080p or custom), the video format (4/3 or 16/9) and the different levels of the video quality (from Good to Best).
2. Click **Close** to close the configuration window of video recording.

To run the video recording

1. In the concerned viewport, at the bottom of the 3D view, click to start the video recording.

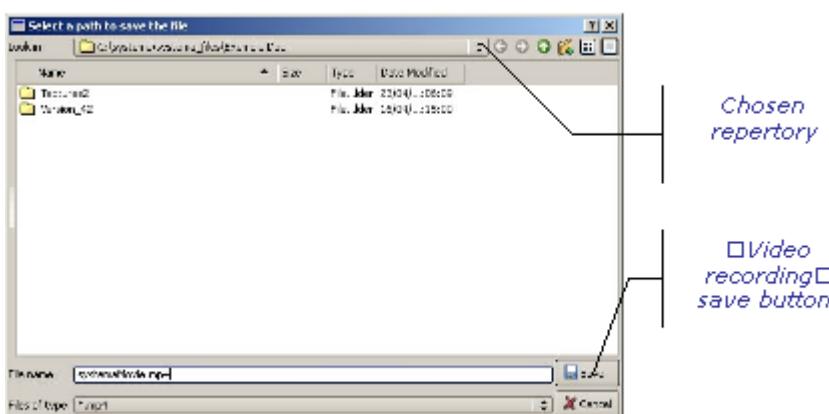
Example: Mission



1. In the displayed "Recording session" message, click **OK** to confirm the recording session*.*

The video is recording until the end of scenario.

The "saving" screen is displayed:



1. Click **Save** to save the video recording.

	<p><i>Note: The SYSTEMA movie can be recorded in the following video formats :</i></p> <ul style="list-style-type: none"> • mpeg • m1v • mp4 • avi
	<p><i>Note: If you want to stop the recording video and save the recording session before the end of scenario, click Escape. The "saving" screen is displayed.</i></p>

Working with the timeline window

What is the timeline window for?

The *timeline window* is displayed within the **Trajectory** and **Mission** tabs, below the browser window and the viewer area.

The *timeline window* provides graphical information concerning all the components linked to the time and can be used to create new components and to manage the date in the viewer area.

The managed components are events characterized by a date and time intervals characterized by start and end dates.

The *timeline window* is split into lines. The first line is the *time scale*, the second line contains icons to control the date in 3D views and the following lines contain events or time intervals grouped by categories.

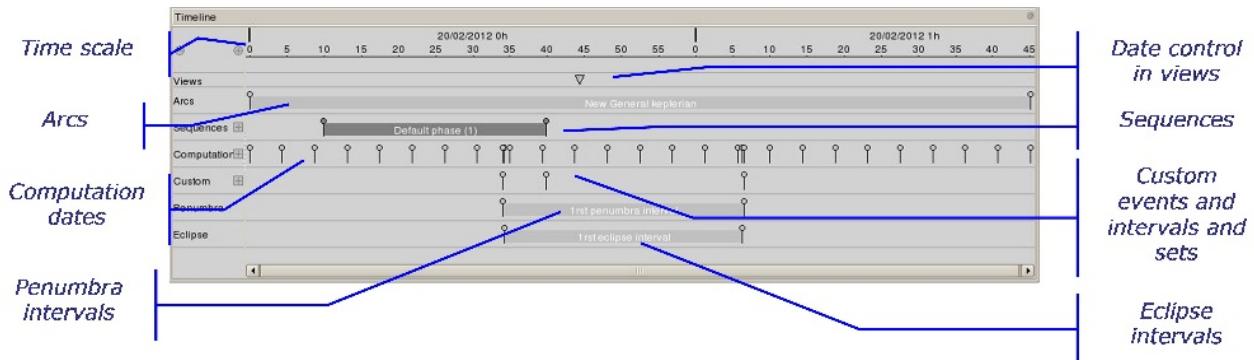
The *timeline window* is displayed in a tab as soon as an entity has been loaded in the corresponding tab. The information displayed in the *timeline window* corresponds to the **current entity** loaded in the *browser window*.

The timeline window's lines

The timeline window contains:

Line	Components	Tab	
		Trajectory	Mission
Time scale	Time scale for the current entity	X	X
Views	Icons  to manage the date in 3D views	X	X
Arcs	Orbital arcs defined for the current entity	X	X
Sequences	Intervals corresponding to the sequences		X
Computation	Events corresponding to the computation dates		X
Custom	Custom events, intervals, events sets and intervals sets		X
Penumbra	Intervals corresponding to the penumbras	X	X
Eclipse	Intervals corresponding to the eclipses	X	X
	<i>Note: for more information about Sequences and Computation dates, refer to "What is a mission" and "Mission attributes" in the "Mission" chapter.</i>		

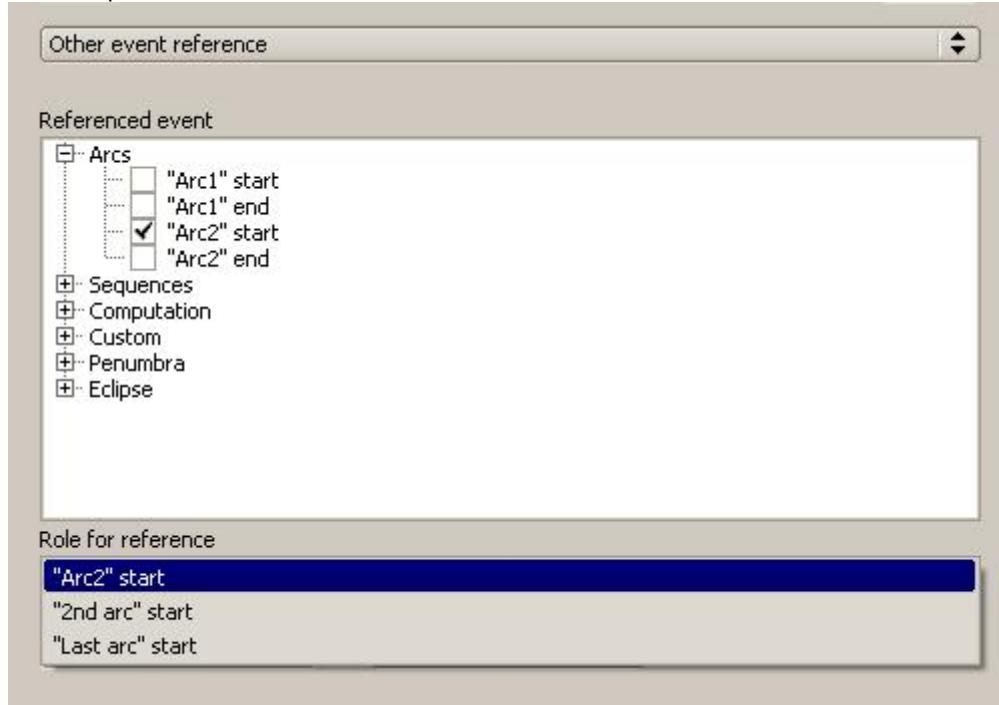
Diagram



Date referencing another event

A date may reference an other event through the **Other event reference** option and the way it refers to it can be configured through a **Role for reference**. The other event can be an event or the start or end of an interval. The role is a "logical role" automatically computed such as "*Arc2*" start or "*7th Computation*" event of "*Computation event rule(1)*". For a given reference, many logical roles can be available.

For example :



The computed roles for reference are :

- "Arc2" start
- or "2nd arc" start
- or "Last arc" start

The resolution of the date reference

If the role for reference corresponds to the a specific event and if this event is removed, the date is computed and becomes an absolute date.

If the role for reference corresponds to a logical event such as the "Last arc" start , the date automatically references the new last arc if arcs are added or deleted. The date can be in error if no more component corresponds to the logical role.

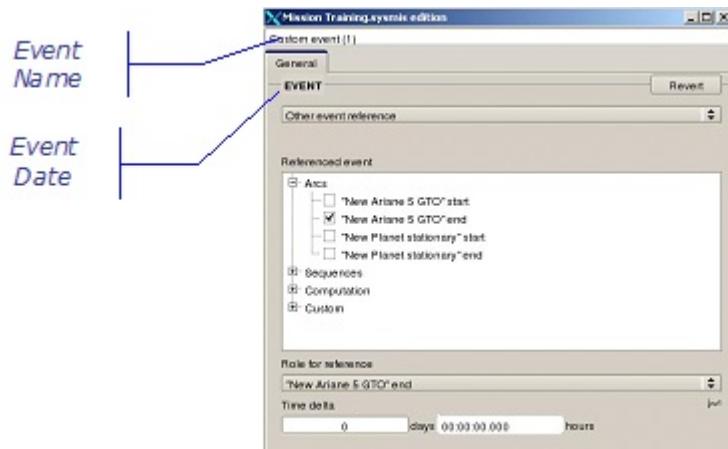


Note: Refer to the Attributes of a date in this chapter to get the definition of the date attributes

Attributes of an event

The **event** has two attributes

- the event name,
- the event date



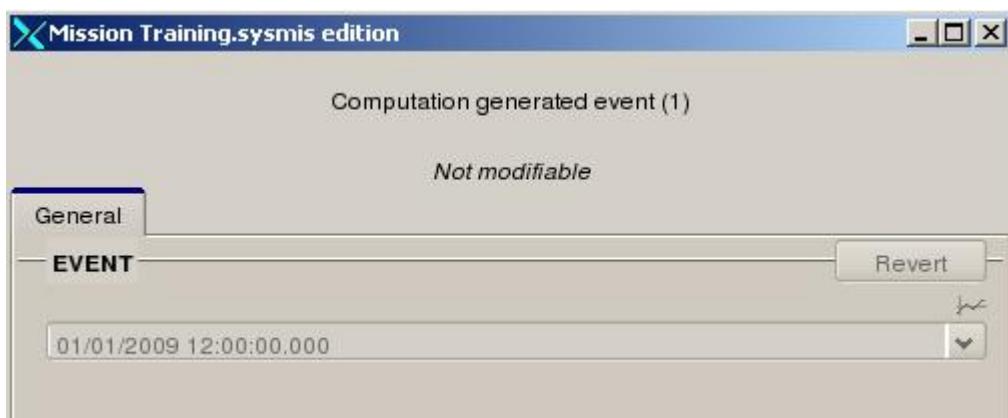
The name of the event is automatically computed as **Computation event (index)** : for a single event and **Computation generated event (index)** : for an event belonging to an events set if the event belongs to the **Computation** category and as **Custom event (index)** : or **Custom generated event (index)** : if it belongs to the **Custom** category. It can be changed for single events.

How to define the Event date

Refer to the Attributes of a date in this chapter to get the definition of the date attributes.

Special case : the Generated Event

In the case of a **generated event** belonging to an events set through a **event rule** , the attributes cannot be modified



Attributes of an event set

The **event set** has three attributes

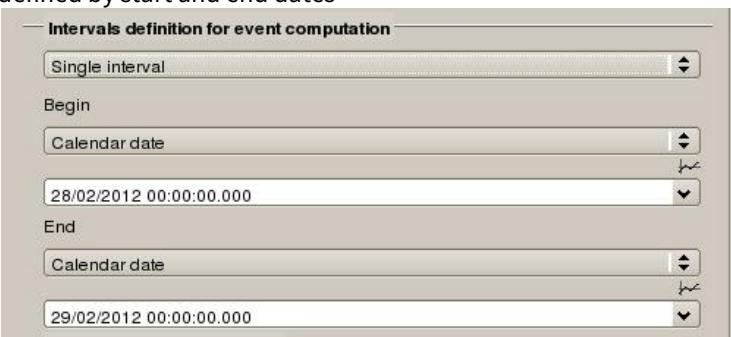
- the event set name,
- the intervals definition for event computation
- the events computation method

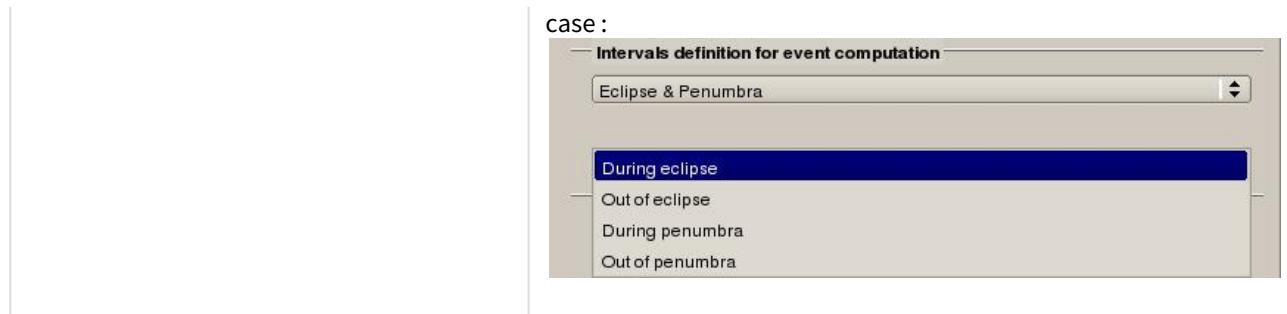


The name of the event set is automatically computed as **Computation event rule (index)** : if the events set belongs to the **Computation** category and as **Custom event rule(index)** : if it belongs to the **Custom** category. It can be changed

How to define the Intervals definition

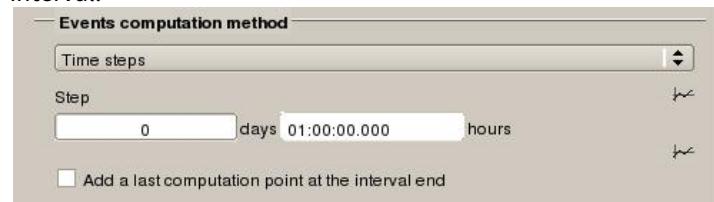
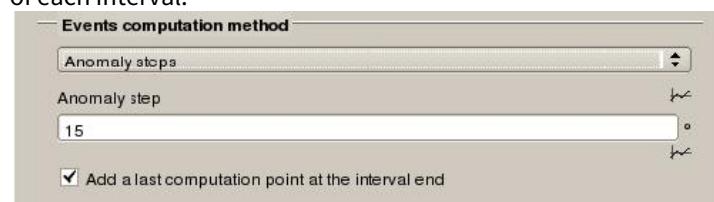
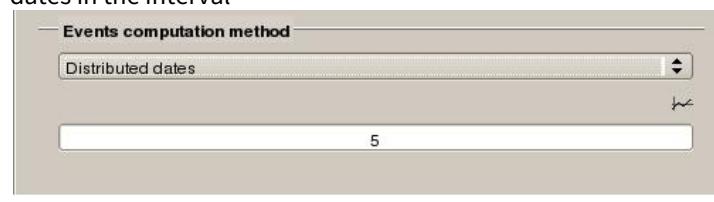
The available options for the intervals definition are the following ones:

Intervals definition value	Description
Whole trajectory	The Events computation method is applied on the whole trajectory
Single interval	The Events computation method is applied on a single interval defined by start and end dates 
Eclipse & Penumbra	The Events computation method is applied on intervals related to eclipse and penumbra intervals. Four options are available in that



How to define the Events computation method

The available options for the Events computation method are the following ones:

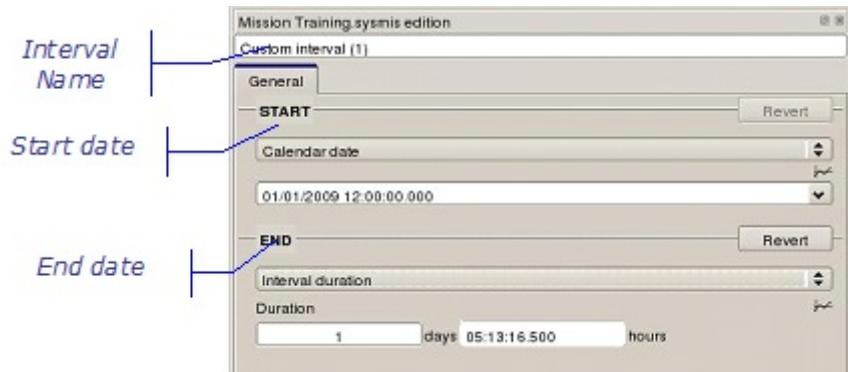
Method value	Description
Time steps	<p>The time steps is expressed in number of days and hours. The user may add a last computation point at the end of each interval.</p> 
Anomaly steps	<p>The anomaly steps are defined in degrees related to the anomaly. The user may add a last computation point at the end of each interval.</p> 
Distributed dates	<p>The number of dates is defined to create regulary distributed dates in the interval</p> 
Angular steps on kinematics linear attitude	<p>The angular steps related to the selected kinematics body are defined in degrees with choice of the start and end angles. The user may add a last computation point at the end of each</p>

Method value	Description
	<p>interval.</p> 
Eclipses & penumbra dates	<p>The events are linked to the Eclipse and penumbra dates with the following options :</p> <p>For the considered events</p> <ul style="list-style-type: none"> • Eclipse & penumbra • Eclipse • Penumbra <p>For the considered dates</p> <ul style="list-style-type: none"> • Start & end • Start • End 

Attributes of an interval

The **interval** has three attributes

- the interval name,
- the interval start date



- the interval end date

The name of the interval is automatically computed as **Custom interval (index)** : for a single interval and **Custom generated interval (index)** : if it belongs to a custom interval rule. The intervals are all defined in the **Custom** category. It can be changed.

How to define the Start date

Refer to the Attributes of a date in this chapter to get the definition of the date attributes.

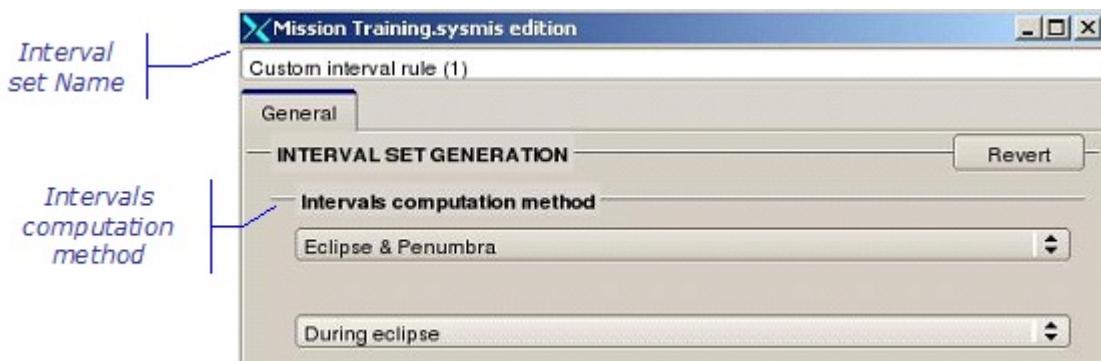
How to define the End date

Refer to the Attributes of a date in this chapter to get the definition of the date attributes.

Attributes of an interval set

The **interval set** has two attributes

- the interval set name,
- the intervals computation method

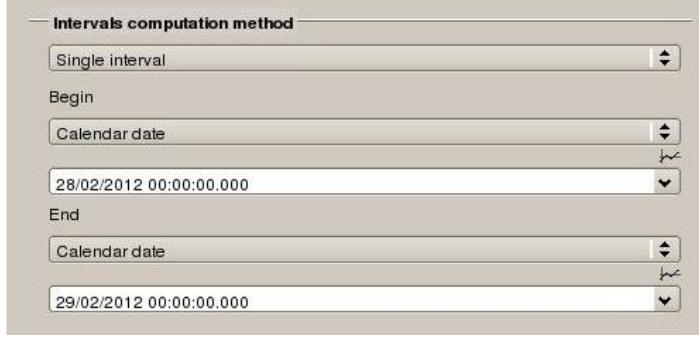
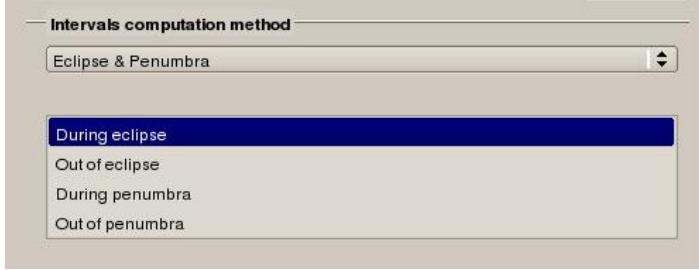


The name of the interval set is automatically computed as **Custom interval rule (index)**. It can be changed. All the interval sets belong to the **Custom** category.

How to define the Intervals computation method

The available options for the intervals computation method are the following ones:

Intervals computation method value	Description
Whole trajectory	The interval set contains a single interval defined as the whole trajectory

Single interval	<p>The interval set contains a single interval defined by start and end dates</p> 
Eclipse & Penumbra	<p>The interval set contains intervals related to eclipse and penumbra intervals. Four options are available in that case :</p> 

Attributes of a date

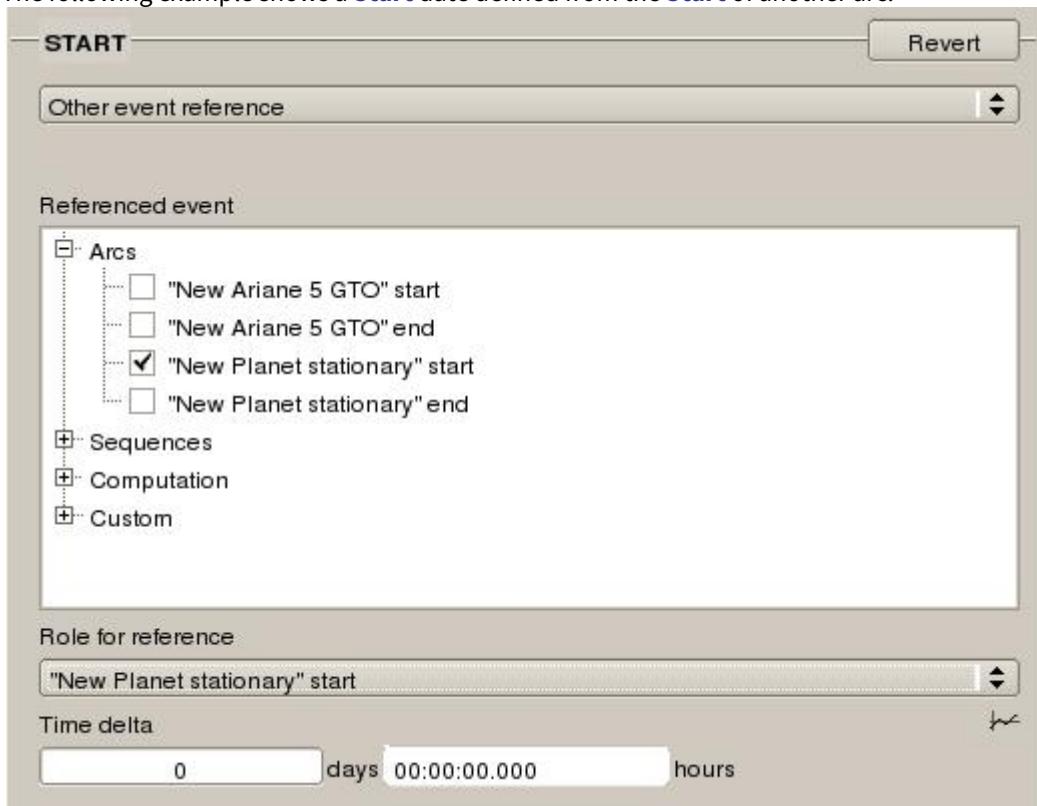
The date is used as attribute for the events and as start and end dates for the intervals
The date may be entered or **calculated** in a number of ways:

- enter the absolute value ([Gamma 50](#) , [J2000](#) , [Calendar date](#)),
- or enter an [Other event reference](#)
- enter the value relative to the [Start](#) or the [End](#) date of a trajectory arc
- or enter the value relative to the [Start](#) or the [End](#) of a previous sequence.
- or enter the value relative to a computation event
- or enter the value relative to a custom event, the [Start](#) or the [End](#) of a custom interval, a custom generated event or the [Start](#) or the [End](#) of a custom generated interval
- or, **in case of end date**, enter the [Interval duration](#) in number of days.

In case of reference to another event :

- a time delta can be defined. The delta is expressed in number of days and hours.
- a list of [roles for reference](#) is automatically computed with selection of the first choice in the list.

The following example shows a **Start** date defined from the **Start** of another arc:



In that case, the computed list of **Roles for reference** will be the following one :

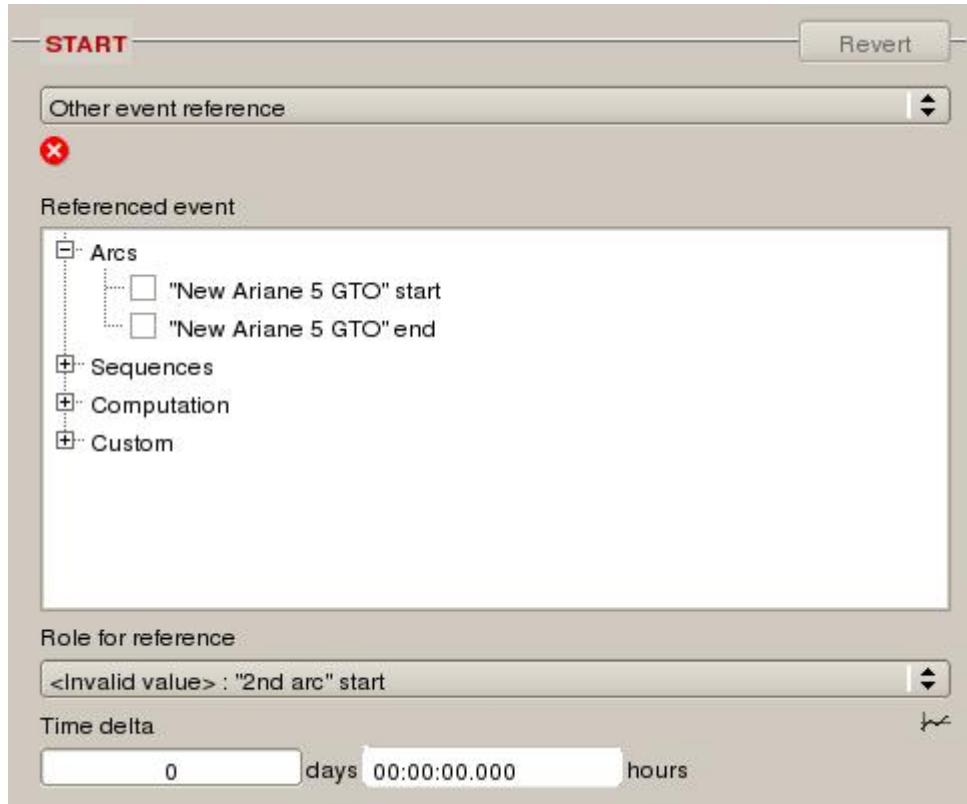


In fact, all the available roles have been computed for the user and he can choose between :

- **"New Planet stationary" start** : the start of a particular arc is referenced and if the arc is deleted, the date will be resolved as an absolute date



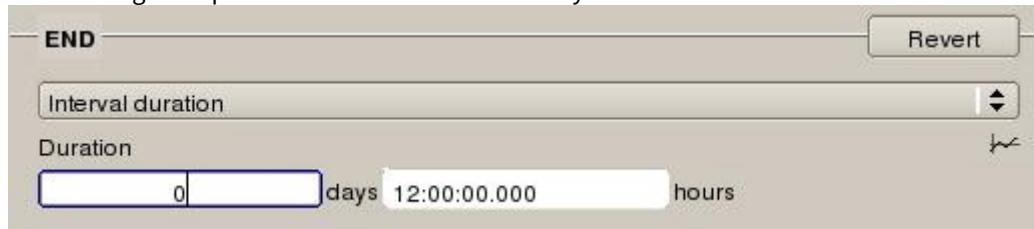
- **"2nd arc" start** : the start of the 2nd arc will always be referenced and if the 2nd arc is no more defined, the role is conserved but the start date is in error



- **"Last arc" start**: the start of the last arc will always be referenced and the start date cannot be in error until no arc is defined in the trajectory

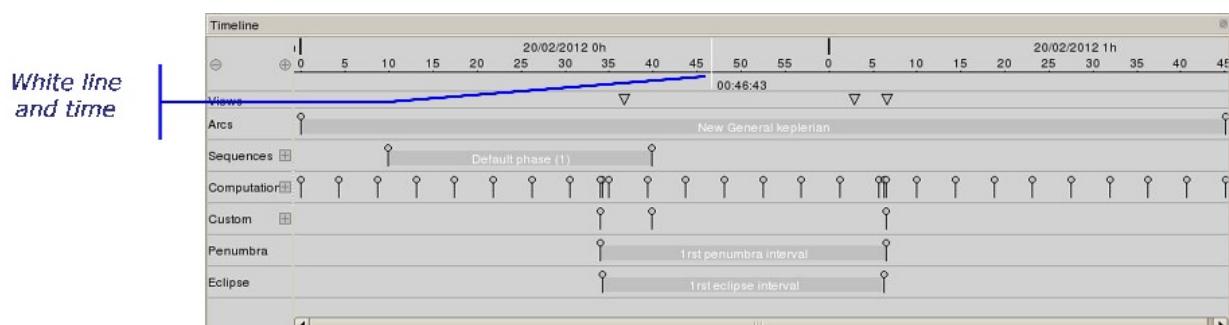


The following example shows an **End** date defined by an **Interval duration**



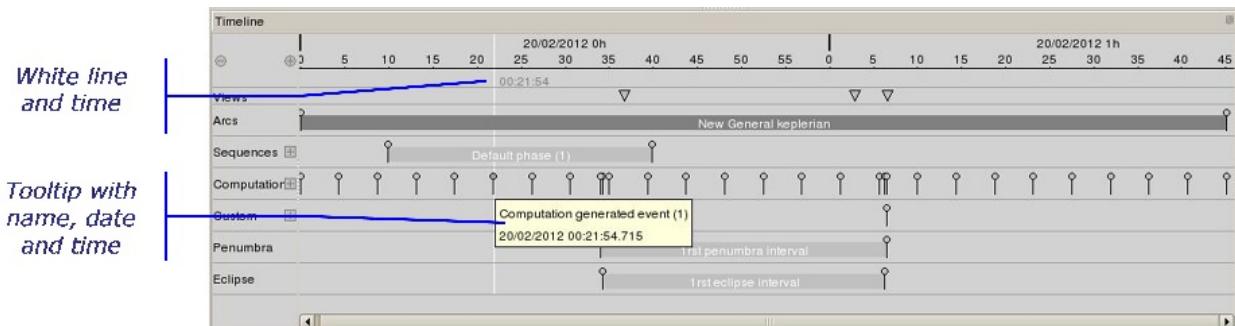
Getting information on the displayed components

Time scale



- Put the mouse in the *time scale* area. A little white line appears corresponding to the date with the display of the elapsed time since the beginning of the current day just below the *time scale*.

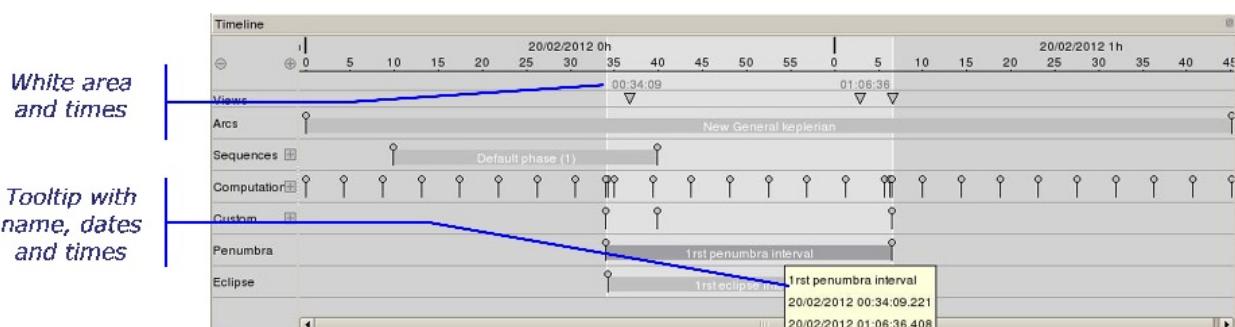
Events



- Put the mouse on an event and leave it on it. A *white line* appears corresponding to the event's date with the display of the elapsed time since the beginning of the concerned day below the *time scale*. A *tooltip* is displayed showing the event's name, date and time.

The events can be computation dates, custom events, start or end of intervals (arcs, sequences, custom intervals, penumbra and eclipse intervals) and icons linked to the 3D displays. For the icons, only the *white line* with the time is displayed. Computation dates, custom events, sequences and custom intervals are only available in **Mission** tab.

Intervals



- Put the mouse on an interval and leave it on it. A *white area* appears corresponding to the interval with the display of the start and end times related to the concerned day below the *time scale*. A *tooltip* is displayed showing the interval's name, date and time.

The intervals can be arcs, sequences, custom intervals, penumbra and eclipse intervals. Sequences and custom intervals are only available in **Mission** tab.

Managing components from the timeline window

How to select a component from the timeline window

- Any component displayed in the timeline window can be selected by *left click* on it. If a browser window displayed by the current tab contains the component, the selection is updated in this browser: The selection is also updated in concerned 3D viewports if necessary.

If many components share the same position in the timeline window, a tooltip menu appears to let the user choose the required component.



How to create a component from the timeline window

In the **Mission** tab, new components can be created from the timeline window.

There are different ways:

1. Click on the  icon available for some components (events or intervals) managed in the current tab. A tooltip menu appears.
2. In the tooltip menu, select the appropriate option. A new component is created with default values and an edition window is automatically opened. The new component is displayed and selected in the timeline and in the browser windows.

The option may be one of the following:

Option	Use
Event	Creates an event defined by an absolute date or the reference to another event
Interval	Creates an interval defined by start and end dates. Each date can be defined by an absolute date or the reference to another event
Events set	Creates a set of events defined by the intervals definition and the events computation method
Intervals set	Creates a set of intervals defined by an intervals computation method

The available options depend of the component.

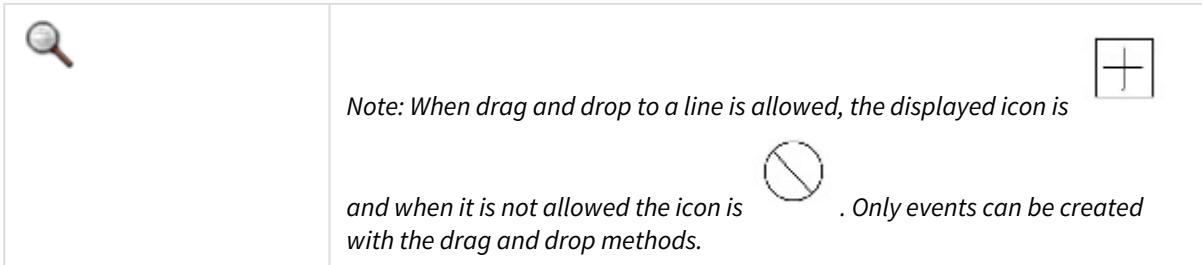
Component	Available options
Computation	<ul style="list-style-type: none"> • Event • Events set
Sequences	<ul style="list-style-type: none"> • Interval • Intervals set
Custom	<ul style="list-style-type: none"> • Event • Interval • Events set • Intervals set

or

- *Left click on an event and drag and drop this event to the Computation dates or Custom events lines in the timeline window. A new component is created with a **date of type other event reference related to the source component**. If the date of the source component is modified, the date of the new component is impacted. The new component is displayed in the timeline and in the browser windows.*

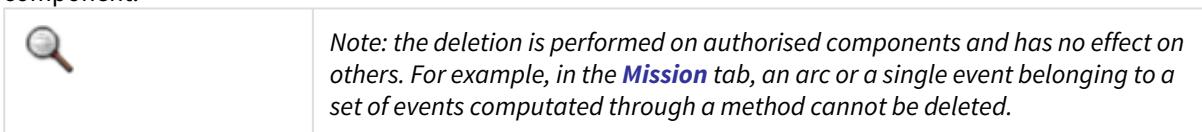
or

- *Simultaneously hold down the **Ctrl** key and *left* click on an event and drag and drop this event to the Computation dates or Custom events lines in the timeline window. A new component is created with a **computed date of type calendar date from the date of the source component**. If the date of the source component is modified, the date of the new component is not impacted. The new component is displayed in the timeline and in the browser windows.*



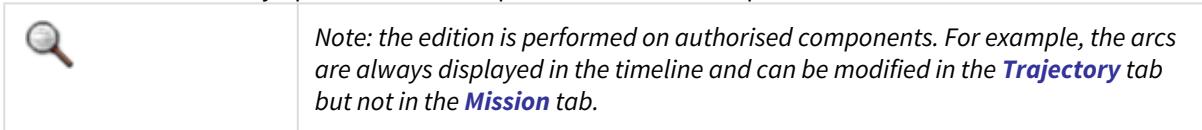
How to delete a component from the timeline window

- Once the component is selected in the timeline window, it can be deleted by clicking the [Delete] button. In fact, this button is a shortcut to the **Delete** option of the **Edit** menu and acts on the current selected component.



Editing components from the timeline window

- By double-click on a component (event or interval) which can be modified in the current tab, the edition window is automatically opened for the component and it can be updated.



Managing the time scale

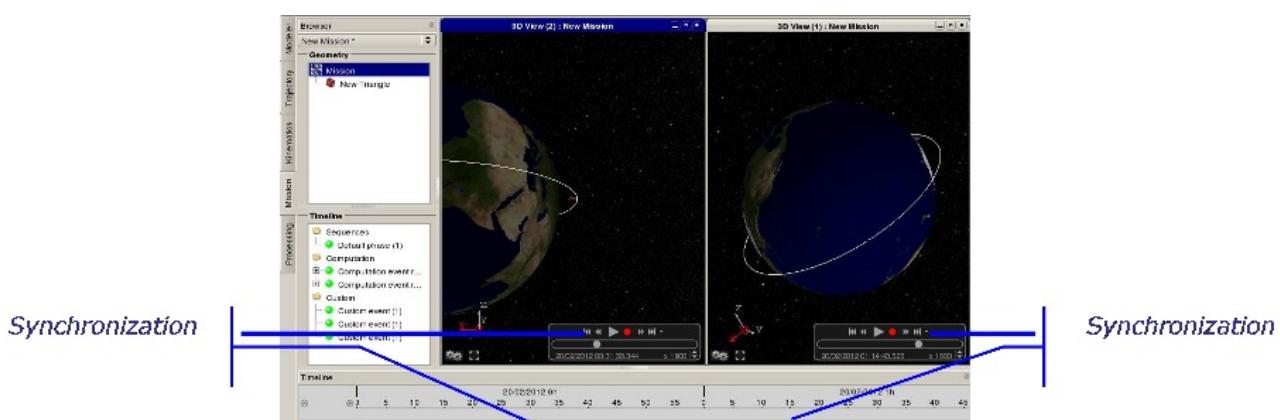
The time scale's size can be modified and the time interval is automatically computed according to this size.

- When the cursor is positioned in the timeline window, the *middle* mouse button can be used to increase and decrease the time scale's size and interval up to 5 seconds and down to 5 minutes.

or

- The and icons can be used instead of the *middle* mouse button.

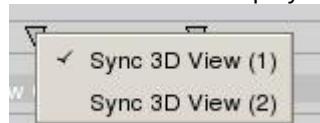
Managing the date in the 3D views



- The displayed dates in the viewer area can be managed through the  icons. As many icons as 3D views are automatically created. When an icon is moved holding down the *left* mouse button on it while moving the mouse along the time scale, the displayed date in the corresponding 3D view is automatically synchronized.

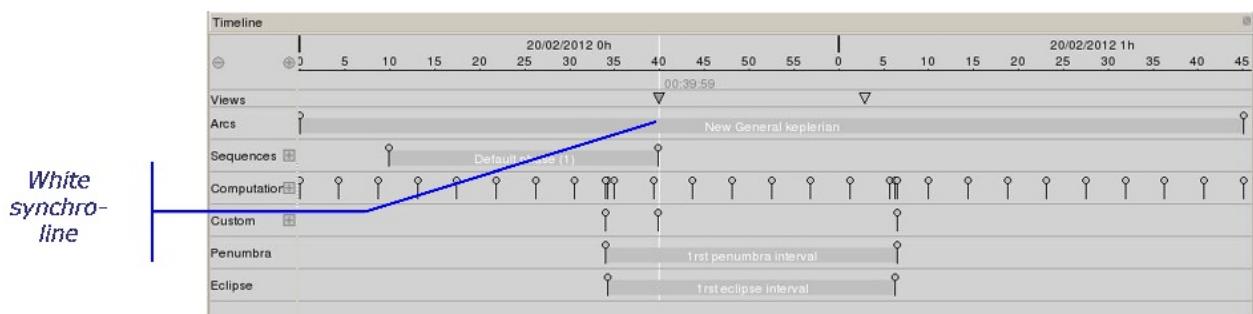
How to synchronize 3D views?

- Right* click on the icon of a 3D view to display the list of 3D views related to the current trajectory or mission



in a tooltip menu . The user may synchronize many 3D views by checking them. Only one icon is displayed for all the synchronized 3D views. A view can be desynchronized by unchecking it in the list. An icon is automatically created for it at the same place and can be moved.

How to set the date in the 3D view to a special event?



- When an icon is moved, holding down the *left* mouse button on it, the move of the mouse down leads to the display of a white line when flying over an event. It is very easy to match the cursor and the white line

Working with the edition window

What is the edition window for?

The edition window is used to define the attributes of entity components.

For some entity components, the edition window is automatically displayed as soon as the component is created.

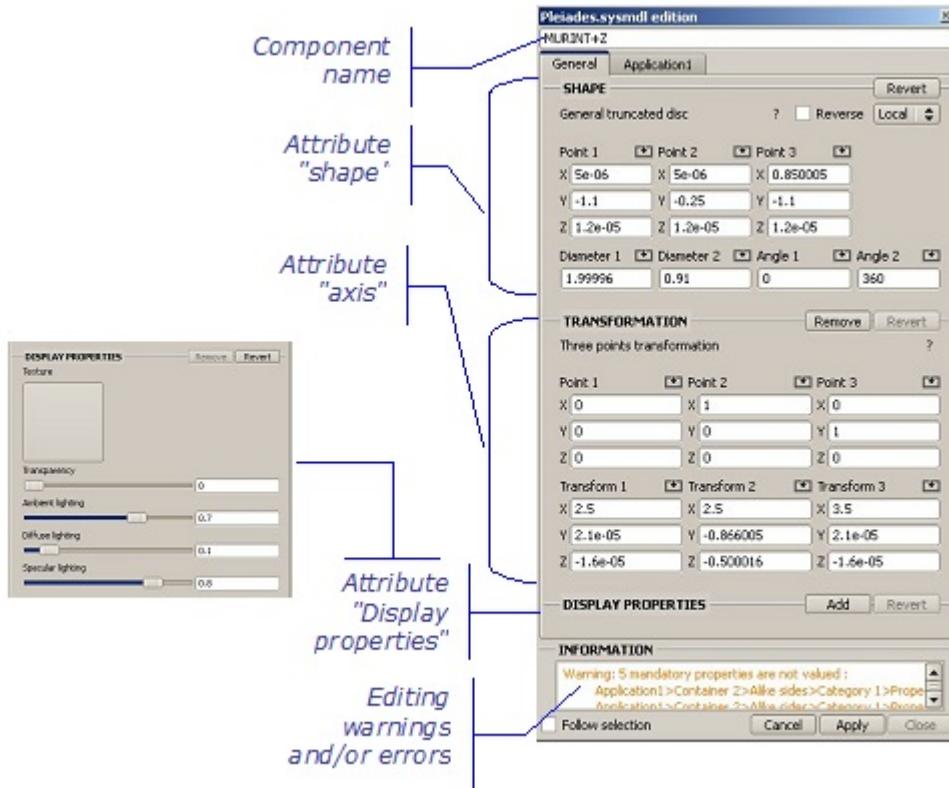
Component attributes

Attributes are properties defining a component, such as the name, color, point coordinates, etc.

The **Information** field contains comments relative to the entity component's state within the entity. This field may contain:

- warnings (orange text), which inform of a not invalidating problem (for example, a mandatory property which is not valued);
- errors (red text), which inform of an invalidating problem (for example, successive arcs with the same start time).

Diagram



General principles

All component attributes can be modified via the edition window.
The edition window is navigable through both mouse and TAB key inputs.

Modifying component attributes within the edition window

How to change the component name?

1. Modify the name in the input zone.

Example:	
----------	--

2. Click **Apply** to validate the modification.

The component name is updated in the browser window.

Example:	
----------	--

How to add an attribute value?

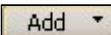
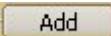
If an attribute has no value, the Add button is displayed in the associated field of the edition window.

Example:	TRANSFORMATION 
----------	---

To add an attribute value:

1. Click Add :

This button may be one of the following ones:

- the  button is displayed when an additional option has to be selected in a list of values;
- the  button is displayed when you have just to enter the attribute values.

2. Enter the attribute values.

You may find different types of fields helping you to enter values:

Field	Use																																
<i>Simple field</i> 	Enter the appropriate value																																
<i>Field with the  symbol</i>	Click  to fill in the field with the related value in the location bar which contains the most recently value of a point coordinates, an angle, a length, a vector...																																
	Click  to get interactive help for the concerned attribute																																
Example:	Transformation attribute TRANSFORMATION    <p>Three points transformation</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%;">Point 1</td> <td style="width: 25%;"> Point 2</td> <td style="width: 25%;"> Point 3</td> <td style="width: 25%;"></td> </tr> <tr> <td>X 1</td> <td>X 0</td> <td>X 0</td> <td></td> </tr> <tr> <td>Y 0</td> <td>Y 1</td> <td>Y 0</td> <td></td> </tr> <tr> <td>Z 0</td> <td>Z 0</td> <td>Z 1</td> <td></td> </tr> <tr> <td>Transform 1</td> <td> Transform 2</td> <td> Transform 3</td> <td></td> </tr> <tr> <td>X 1</td> <td>X 0</td> <td>X 0</td> <td></td> </tr> <tr> <td>Y 0</td> <td>Y 1</td> <td>Y 0</td> <td></td> </tr> <tr> <td>Z 0</td> <td>Z 0</td> <td>Z 1</td> <td></td> </tr> </table>	Point 1	 Point 2	 Point 3		X 1	X 0	X 0		Y 0	Y 1	Y 0		Z 0	Z 0	Z 1		Transform 1	 Transform 2	 Transform 3		X 1	X 0	X 0		Y 0	Y 1	Y 0		Z 0	Z 0	Z 1	
Point 1	 Point 2	 Point 3																															
X 1	X 0	X 0																															
Y 0	Y 1	Y 0																															
Z 0	Z 0	Z 1																															
Transform 1	 Transform 2	 Transform 3																															
X 1	X 0	X 0																															
Y 0	Y 1	Y 0																															
Z 0	Z 0	Z 1																															
	<i>Note: If any errors are made while entering values, the corresponding attribute appears in red and the values must be modified</i>																																

If necessary, click **Revert** to cancel the modification.

3. Click **Apply to validate the modification.**

How to remove an attribute value?

When an attribute can be removed, the **Remove** button is displayed in the associated field of the edition window.

Example:

The screenshot shows the 'TRANSFORMATION' dialog box with the title 'Three points transformation'. It includes a 'Remove' button and a 'Revert' button. Below the title, there are two sets of input fields for 'Point 1', 'Point 2', and 'Point 3', each with X, Y, and Z coordinates. There are also sections for 'Transform 1', 'Transform 2', and 'Transform 3' with their respective X, Y, and Z coordinates.

	Point 1	Point 2	Point 3
X	1	0	0
Y	0	1	0
Z	0	0	1

	Transform 1	Transform 2	Transform 3
X	1	0	0
Y	0	1	0
Z	0	0	1

To remove an attribute value:

1. In the area defining the attribute to be removed, click **Remove**.

All values defining the attribute are removed.

If necessary, click **Revert** to cancel the removal.

Example:

The screenshot shows the 'TRANSFORMATION' dialog box with the 'Remove' button highlighted in blue. The 'Revert' button is also visible.

2. Click **Apply** to validate the removal.

The **Add** button replaces the **Revert** button.

Example:

The screenshot shows the 'TRANSFORMATION' dialog box with the 'Add' button highlighted in blue. The 'Remove' button is also visible.

How to undo/redo editing actions

In the edition window, once a modification has been made and validated (using the **Apply** button), you can undo it before validating a new modification.

An undo action may be cancelled using the redo action, before it is validated.

To undo an editing action (from the SYSTEMA main window or the edition window):

- In the **Edit** menu, select the **Undo** option

or

- press the [Ctrl]+ Z keys of the keyboard

To redo an editing action (from the SYSTEMA main window or the edition window):

- In the **Edit** menu, select the **Redo** option

or

- press the [Ctrl]+ Y keys of the keyboard

Working with Python scripts

What can be done with Python scripts ?

Creation of a new model or meshing associated to the current model, modification of an existing model or meshing can be performed by the execution of a Python script.

We provide a Python interface so that the user may act on models, objects, shapes, geometries, transformations, application properties, meshing and numbering parameters.

The scripting is particularly indicated to interface Systema with external tools, to perform repetitive tasks (activity inversion, meshing reduction ...) or easy-to-automate tasks (homotheties, symmetries ...).

A library is provided with Systema, composed with common and high level methods.

Executing a Python script

There are three ways to execute python code in systema.

1. Launch a .py script with the graphical user interface



- Click on the icon or on the **Launch Python script** option of the **Tools** menu. A new file manager window appears to select the text file to open
- Choose a .py file and open it.
- Result: The Python script contained in the text file is executed.

The results of the script are displayed in the browser window and in the viewer area : a new model or meshing may have been loaded or modifications may have been done to the current model or meshing.

- All errors in the script file, and outputs, are reported in the log bar.



- Click on the icon and on the **Relaunch Python script** option of the **Tools** menu to relaunch the execution of the script.
- In the **Tools** menu, you will find a list of the recent scripts you launched. To relaunch it, simply click on the name of the script.

2. Launch a .py script in command line

- Launch systema in command line, with the -script option, followed by the path of your script.
- Launching systema in script mode with the command line allow better performances, and reduced execution time, since there is no 3D scene and graphical user interface to update.
- It also means that you can launch your systema script on computers which don't have any X server.
- If you have a big script that does not require any direct interaction with the graphical user interface, you may want to run it this way to have the result faster.
- All errors in the script file, and outputs, are reported directly in your command line tool standard output.

3. Use the interactive console



- Click on the icon or on the **Open Python Console** option of the **Tools** menu. A new window appears with two main areas.
- You can enter your python code directly in the input zone. If you want to type several lines, just use "Alt + Return" to line jump, then "Return" to execute it.
- You can copy/past the console history with common shortcuts (CTRL+C/V) or using contextual menu on right click, the copy will automatically remove > symbols.
- The interactive console allows you to easily execute some simple python code, in order to test things or to run throwable code to do specific tasks.

It also allows to have a mixed workflow, using both the graphical user interface and python code at the same time.

- You can navigate in the history of commands using up and down arrows key. The navigation is done

between blocks. You can also call back a previous code block by using up and down arrows + ctrl; in this case, text entered in the command area is searched in the history.

→ All errors in the script file, and outputs, are reported in the console output.

→ If you close the console and re-open it, all your python context and history will be kept.

⚠ Be carefull when mixing interaction between the graphical user interface and interactive console, you could corrupt your data :

Let's say you have a variable containing list of systema object in python. If you delete some objects in the graphical user interface, you may still try to access to it in python.

In that case you will have a "ValueError : Incorrect object" in your ouput, which means that you try to access a data which no longer exists. Example :

```
model = getCurrentModelFile() # enter this in the console, then close your
model file.
print model.getName() # will output "ValueError : incorrect file" because the
model is no longer accessible.
```

How to import your own modules

You can easily import your own modules in the systema python interpreter. To do so, you have to specify the path to the module and all its dependencies :

```
import sys
sys.append("/path/to/myModule")
sys.append("/path/to/myModule_dependencies") # if there are dependencies elsewhere
from myModule import * # now python is able to locate and import your module
```

For more information...

All documentation and examples are included in the Systema installation directory, in the "python" folder.

Working with variables

What are variables made for ?

It can be sometimes difficult to define with certainty the values of some specific parameters because:

- they might be received with certainty later in the process, as an input for the analyst
- they might be dependent on other parameters that will be defined later in the process
- their determination might be the objective of the analysis (ex: sensitivity analysis)

Systema provides a mechanism that enables a temporary definition of those parameters values while guaranteeing an easy update of their value later in the process: variables.

Variables can be defined within every Systema files. They are defined by:

- a name that is unique
- a type (vector, value, date, etc)
- a value
- an expression (mathematical formula that defines how the value shall be computed)

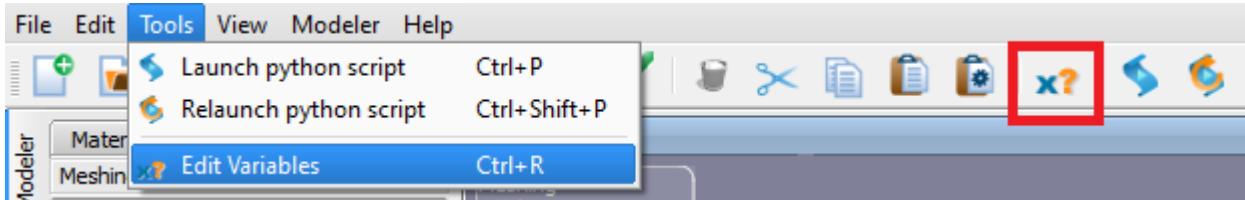
- an index (integer defining the location in the list of variables)
- a state (valid or in error)

Variables can be defined with inter-dependencies within a file but not between two different files.

Their values can be changed easily, either permanently or temporarily (through overloading).

Variables edition

Variables can be defined through a table that is accessible from the Tools menu, by keyboard shortcut (Ctrl + R) or from the top icon bar.



The table window is not blocking (possibility to work with multi-windowing).

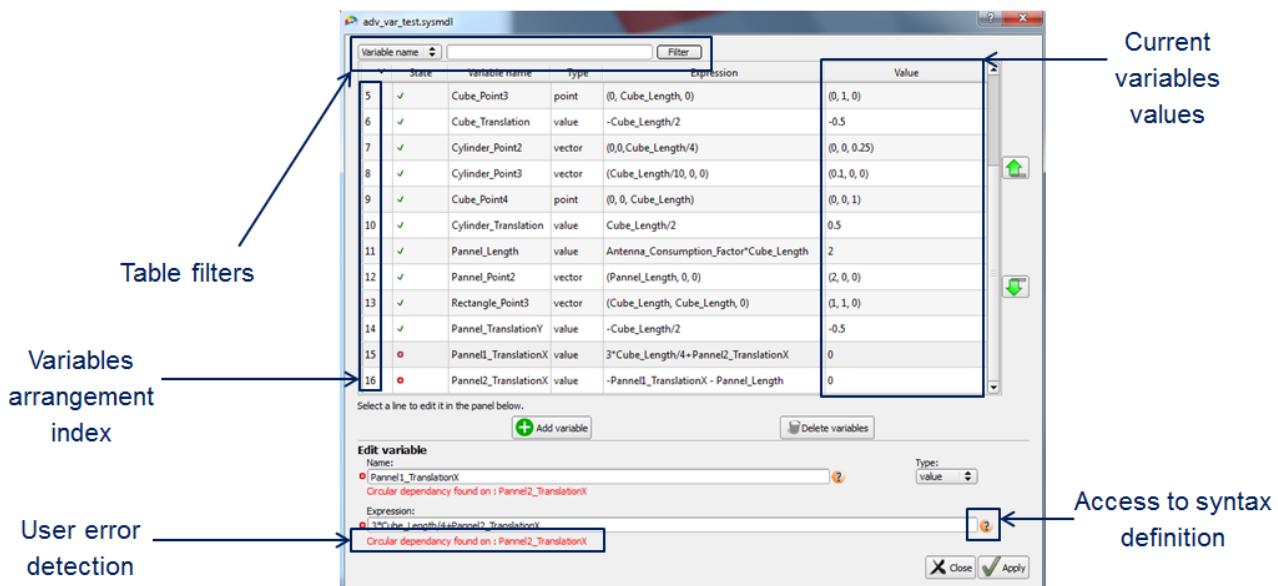
Only one table window per tab can be opened at a time.

It contains all the variables attributes (index, state, type, expression and value).

The table lines can be sorted by every columns by clicking on columns headers.

Warning: lines can be moved up and down only if the current sort criteria is the index.

Moreover, it is possible to filter the list of variable using the top filter field upon variables state, name, type or value.



Variables can be created/deleted either by clicking on the buttons at the bottom of the table ("Add variable" and "Delete variables") or by right clicking in the table ("Insert new variable" or "Delete variables").

When a variable is selected in the table, it becomes editable in the bottom edition panel.

Variable names can be changed (as long as they remain unique), as well as their type and expression (as long as it respects the right syntax).

To validate the edition of a variable, it is necessary to click on "Apply" or to press the "ENTER" key.

The complete list of variables values is computed after each variables edition.

In case of syntax error or circular dependency, the variable state turns to error and a detailed error message is displayed below the variable expression field.

Variables naming

Variables names are always unique.

Names can only start with a letter (lower or upper case) or with an underscore ("_"). They are composed of letters, underscores and numbers.

Variables syntax

Variables can be defined with inter-dependencies using a mathematical syntax as long as they belong to one of the following categories:

- vectorial types: vector or point
- scalar types: value, integer, angle or length

Other types (boolean, date, duration, and string) cannot be defined with inter-dependencies.

Remark: integer variables values must be defined within integers limits (depends on the machine, typically [-2147483648, 2147483647]).

For smaller/bigger values, use the "value" type instead. It provides up to 16 significative numbers.

The available operations are:

- vectorial operations: scalar product (*), cross product (^), term to term product (.*), term to term addition (+), term to term subtraction (-) and accessors (.x, .y, .z, .X, .Y, .Z)
- scalar operations: addition (+), subtraction (-), multiplication (*), division (/), powers (**), scientific notation (E, e)

Remark: The power operator ** works in Systema with left precedence. It means for example that $2^{**}3^{**}2 = (2^{**}3)^{**}2 = 64$ (not $2^{**}(3^{**}2) = 512$).

If you want to use a right precedence instead, please use parenthesis.

Some basic mathematical functions are available:

- Square root (sqrt(x)) with x between]0; +Inf[
- Vector norm (norm(x))
- Exponential (exp(x))
- Natural logarithm (log(x)) with x between]0; +Inf[
- Trigonometric basics (pi, cos(x), sin(x) and tan(x) with x different from Pi/2 modulo Pi)

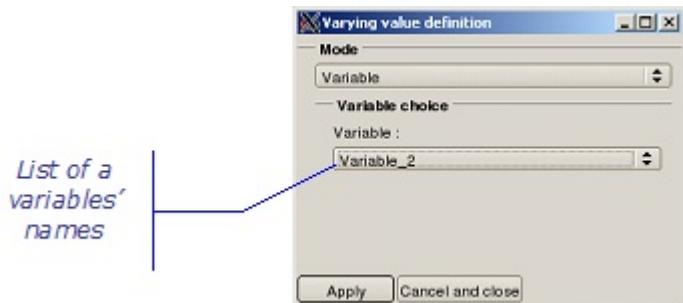
Examples:

- myFloat = 4.423
- myVector = (1.5, 2.568, 0)
- myScalar = (-variable_1 + 5*variable_2) / variable_3
- myScalar = variable_1**2 + variable_1E-02
- myVector = vector_1^vector_2
- myScalar = vector_1*vector_2 + vector_1.*vector_2 + vector_1.X
- myScalar = norm(myVector) + sqrt(2) + log(exp(10))
- myValue = sin(x)*cos(y+x) + cos(x+y)*sin(y)

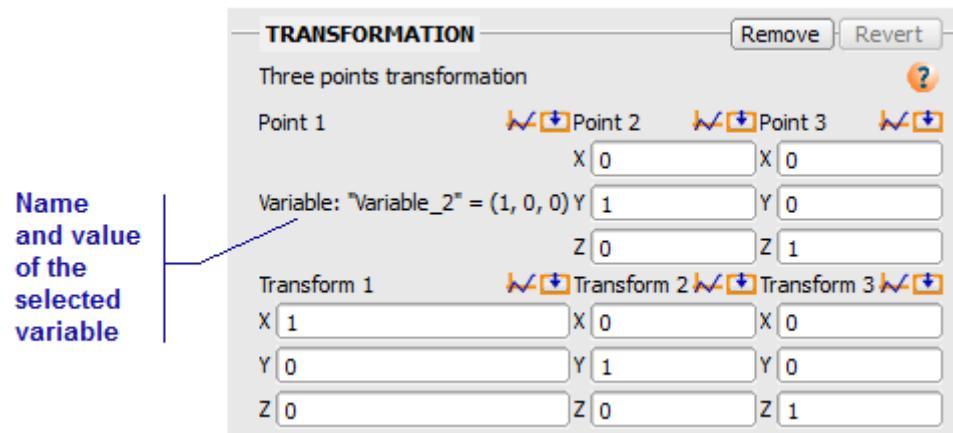
Variables assignment

These variables are made to be assigned to real parameters.

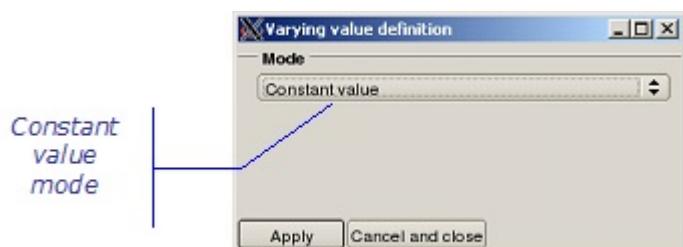
Any value in any attribute can be replaced by the reference to a variable by clicking the  icon. A new window named **Varying value definition** appears with the mode set to **Variable** and the user can select a variable name in the list of compatible variables and click **Apply**



The name of the variable and its value are displayed in the attribute.



Suppress the reference to the variable by clicking the icon. A new window named **Varying value definition** appears with the mode set to **Variable** and the chosen variable. The user can select the **Constant value mode** and click **Apply** to cancel the variable choice



1. If a variable is renamed in the Variables attribute, it is automatically renamed everywhere it is referenced.
2. If a variable is removed from the Variables attribute, it is replaced everywhere it is referenced by its value

The variables assignment in a file can be automatically verified with the search tool.

Cf [Variables search tool](#)

Appendices:

Shortcuts Summary

Actions	Shortcut
Systema	
Quit systema	Ctrl+Q
Files	
Open File	Ctrl+O
Close File	Ctrl+W
Save File	Ctrl+S
Save	
Save File	Ctrl+S
Save File As	Ctrl+Alt+S
Save All Files	Ctrl+Shift+S
Archive File	Ctrl+Shift+A
Edit	
Undo	Ctrl+Z
Redo	Ctrl+Y
Delete	Del
Cut	Ctrl+X
Copy	Ctrl+C
Paste	Ctrl+V
UnSelect	Ctrl+D

Actions	Shortcut
Edit Selection	Ctrl+E
Multi Edition Meshing	Ctrl+Shift+E
Multi Selection	Ctrl+LeftClick or Ctrl+Space
Contiguous Selection	Shift+LeftClick or Shift+ArrowKey
Display the contextual menu	RightClick
Tools	
Launch Python script	Ctrl+P
Relaunch Python	Ctrl+Shift+P
Edit variables	Ctrl+R
Open Python console	Ctrl+Alt+T
Copy to clipboard	
3D view	Ctrl+Alt+C-V
3D view with controls (time controller and param manager)	Ctrl+Alt+C-C
Edition Panel	
Apply	Return
Close	Esc
Apply and Close	Ctrl+Return
3D Viewports	
Show all 3D views as tiles	F9
Maximize 3D view	F10
Show next 3D view	F11

Actions	Shortcut
View	
Zoom in	Ctrl+Shift+LeftClick-Move-Forward
Zoom out	Ctrl+Shift+LeftClick-Move-Backward
3D view	
Select a basic component and helps items	LeftClick
Select several basics component	Ctrl+LeftClick
Rectangular selection	Shift+LeftClick
Open the edition window for a component	Double LeftClick
Rotate around focal point in 3D	LeftClick-Move or ArrowKey+PageUp/PageDown
Rotate in 2D (spin)	Ctrl+ LeftClick-Move
Pan camera	MiddleClick -Move
Modeler	
New meshing	Ctrl+Alt+N
Open meshing ...	Ctrl+Alt+O
Close meshing	Ctrl+Alt+W
Interactive create last shape	Ctrl+H
Create last shape	Ctrl+G
Create last help item	Ctrl+I
Create object	Ctrl+T
Create material	Ctrl+M
Bounding box	Ctrl+B
Select point	Alt+LeftClick

Actions	Shortcut
Computation point	Ctrl+Shift+B
Cancel last point selection	Ctrl+U
Abort Interactive creation	Esc
Edit the gizmo transformations	Ctrl+Shift+X
Schematic	
Created point on line	Shift

Geometrical model management

Geometrical model management

Overview

Introduction

This chapter describes how to use SYSTEMA in order to create and display geometrical models and meshings. It also describes how to display graphs (curves and tables) representing the result of a process applied to a given meshed geometrical model.

With SYSTEMA, you use the Modeler tab to work on material files, models and meshings. Everything described in this chapter takes place within the Modeler tab.

In this chapter

This chapter covers the following topics:

- Geometrical model management
 - Overview
 - Introduction
 - In this chapter
 - The Modeler tab
 - What is the Modeler tab for?
 - The different parts of the Modeler tab
 - Example
 - The menu bar
 - Options from the Modeler menu
 - Options from the Tools menu
 - The Modeler tabs
 - The viewer area
 - Loading models in 3D views
 - Graphs
 - The icon bar
 - The browser window
 - The right click menu
 - The Advanced edition sub menu
 - Geometrical models in SYSTEMA
 - What is a geometrical model ?
 - The geometrical model in the browser window
 - The geometrical model in the 3D views
 - Axis systems
 - Axis system display
 - Example
 - Quick and easy 3D manipulation
 - How to use the gizmo
 - How to change the position of the gizmo
 - How to use the gizmo with magnetism
 - The SYSTEMA shapes
 - The Systema cutters
 - Object attributes

- The Color attribute
- The Transformation attribute
- The Display Properties attributes
- Variables
- Shape attributes
 - The Shape attribute
 - The Transformation attribute
 - The Display properties attribute
- Cutter attribute
 - The cutter boolean operation attribute
 - The cutter shape attribute
 - The cutter transformation attribute
- Application-dependent properties
 - Application-dependent properties at root level
 - Visualization of the defined properties in the browser
 - Application-dependent properties at object and shape level
 - The container configuration
 - The Side management parameter
 - The Activity parameter
 - The properties
 - The parametric mode
 - The variable mode
 - The constant mode
 - For more information...
- Display options in the 3D views of the Modeler tab
 - Right click menu in the 3D views
 - Specific display options for viewing models
 - Zoom facilities
 - Help items
 - Help items creation
 - Help items modification and suppression
 - Configuration window
 - Display facilities
 - Orientation Information: used to create different frames to display orientation information
 - Color
 - Video recording
 - Copy to the clipboard
- Working with material files
 - General
 - Creating a new material database
 - Opening an existing material database
 - Modifying a material database
 - Creating materials
 - Editing materials
 - Deleting materials
- Working with geometrical models
 - General
 - Creating a new model
 - Opening an existing model
 - Loading a model in a 3D view
 - Modifying a model
- Building a geometrical model
 - How to build a model
 - Creating objects
 - Creating shapes
 - Creating cutters
 - Adding shapes statically
 - How to "statically" add a shape with default coordinates

- To add a shape from the browser window:
- To add a shape from the Modeler menu:
- **Adding shapes interactively**
 - **How to add a shape interactively**
 - To create a shape interactively using the Modeler menu:
 - **Result in the 3D views**
 - **Result in the hierarchical model organization**
- **Creating cutters**
 - **How to create a cutter of an object/shape in the model**
 - To create a cutter from the browser window:
 - To create a cutter from the Modeler menu:
 - **Result in the 3D views**
- **Example1: Simplifying a model using help items**
 - Principles
 - Supported file formats for models
 - Opening the CAD model and creating a new model
 - Opening the CAD model
 - Simplification of the CAD model
 - Convert a CAD shape into Systema Triangles
 - Using help items (lines) to create a copy of a CAD object
 - Introduction
 - Context
 - Procedure
- **Working with meshings**
 - What are meshings for?
 - Displaying a meshing
 - Defining a meshing when working with the geometrical model shapes
 - How to create a meshing
 - How to open an existing meshing
 - Defining the meshing attributes
 - The meshing attributes
 - Principles
 - How to modify the meshing for an object or a shape
 - How to modify the meshing for a list of objects or shapes
 - Displaying the result of a process in tables and curves
 - How to display a result in tables and in graphs
 - Reading points on the curve
 - Selection in the graph and the table
 - Minimum, maximum and average in the table
 - Data export
 - Navigation in curve view
 - Selected meshes management
 - Displaying the computation rays used in the application
 - How to display the application's computation rays

The Modeler tab

What is the Modeler tab for?

You can use the **Modeler** tab of the SYSTEMA application to:

- open materiel files and, if necessary, modify them,
- open existing geometrical models and, if necessary, modify them,
- create new models by combining basic shapes,
- create new models based on CAD models,

- define a meshing for a model, in order to apply certain processes to the meshing,
- display graphs (curves and tables) representing the results of processes applied to the meshing of a model.

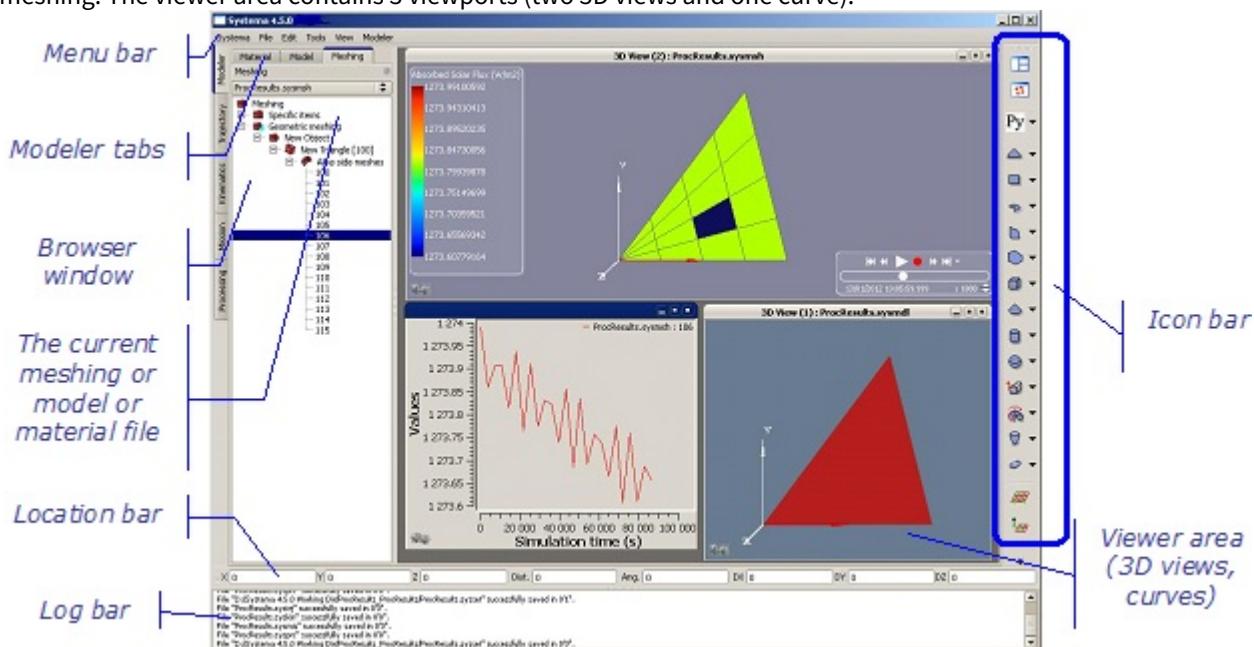
The different parts of the Modeler tab

When models have been opened in the **Modeler** tab, information about them is provided in different areas:

- the browser window, containing the model's hierarchical description,
- the viewer area, containing viewports,
- the location bar, containing information about the most recently selected point(s).

Example

The following diagram shows an example of the Modeler tab, in which the current model is associated with a meshing. The viewer area contains 3 viewports (two 3D views and one curve):



The menu bar

The menu bar contains:

- The menus which are common to all SYSTEMA tabs: the **File**, **Edit**, **Tools**, **View** and **Help** menus;

For more information, refer to the "General User Interface" chapter

- One specific menu:
- The **Modeler** menu, which can be used to add shapes to the current geometrical model;

Options from the Modeler menu

The following table presents the options available from the **Modeler** menu:

Option	Use
Create object	Creates objects in the current model

Create default Shape <i>(for each shape type)</i>	Creates shapes inside the current model with predefined positions and default attribute values <i>Shapes available here are the same as those available from the right click menu in the browser window</i>
Interactive Create Shape <i>(only for some shapes)</i>	Creates shapes in the current model by interactively selecting points in 3D views
Create cutter	Creates a cutter of an existing object/shape in the current model
Create help item	Create help items by interactively selecting points in 3D view.
Cancel last point selection	When interactively creating a shape, cancels the most recent point selection <i>Option only available if the shape creation has not been terminated</i>
Abort interactive creation	Cancels interactive shape creation <i>Option only available if the shape creation has not been terminated</i>
Create material	Creates material in the current material database.
Create specific meshing items	Creates specific meshing items in the current meshing. Choices available depend on the application you use. See "Working with meshings" for more information.
Suppress model transformations	Removes all transformations and transforms all shapes in the main system
Correct accuracy errors	In cases of imported models, you may have accuracy errors, and have invalid shapes. This functionality will automatically correct them.
Select small volumes	On CAD model, selects the shapes with a volume inferior to a threshold
Remove chamfers	Removes the chamfers on all CAD shapes of the current model
Remove holes	Removes the holes on all CAD shapes of the current model
Remove fillets	Removes the fillets on all CAD shapes of the current model
Tessellate CAD shapes	Tesselates all the CAD shapes of the current model. This is the first step for converting CAD shapes into Systema shapes
Facet shape	Facets all the shapes previously tessellated on the current model. This is the second step for converting CAD shapes into Systema shapes

Transform shape in triangles	Transforms all the faceted shapes on triangles on the current model. This is the third step for converting CAD shapes into Systema shapes
-------------------------------------	---

Options from the Tools menu

The following table presents the options available from the **Tools** menu:

Option	Use
Launch python script	Launch the selection and the execution of a python script. See Working with Python scripts for more information
Relaunch python script	Relaunch the execution of the last selected python script.

The Modeler tabs

The Modeler tabs do not appear until a first model is opened. The Modeler tabs contain: three tabs which manage the choice of entity displayed in the browser area : the **Material**, **Model** and **Meshing** tabs; the **File** menu manages the entity's type according to the current selected tab.

Option	Use
Material	Set the choice of entity to "Material file". The content of the selected material file is displayed in the browser area
Model	Set the choice of entity to "Model file". The content of the selected model file is displayed in the browser area
Meshing	Set the choice of entity to "Meshing file". The content of the selected meshing file is displayed in the browser area

The viewer area

The *viewer area* may contain two types of viewports, 3D views and graphs. One or more viewports can be opened at the same time.

Loading models in 3D views

The following rules (which are general to all tabs) are applicable to 3D views of the **Modeler** tab:

- In a 3D view, you can load any geometrical model which has been opened.
- Several models may be loaded at the same time in the same 3D view.

Graphs

"Graph" viewports are used to display results of processes applied to meshings of geometrical models.

The icon bar

The icon bar contains icons that can be used to create basic shapes interactively (where points are selected directly in the viewer, or by entering the coordinates in the location bar) in the current model. It also contains an icon to execute Python scripts.

The browser window

In the browser window of the **Modeler tab**, geometrical objects, shell shapes or cutters can be moved or copied using drag and drop. Press CTRL key after starting to drag to copy rather than move model items.

- Any modifications done with the drag and drop can be cancelled with **Undo(Ctrl+Z)**.

Restrictions

Faceted shapes or shapes containing CAD cannot be moved using drag and drop

The right click menu

In the browser window of the **Modeler** tab it contains some quiet generic and straightforward actions plus the **Special paste** option:

With the simple **Paste** option, a pasted object / shapes / cutter keeps its characteristics in the *Local axis system* of the object it is pasted in: **local coordinates of shapes are unchanged**.

With the **Special paste** option, a pasted object / shapes / cutter keeps its characteristics in the *Main axis system*: **local coordinates of shapes are changed but coordinates in the Main axis system are re-calculated**.

For more information about Local and Main axis system, refer to "Axis systems" topic in the following pages.

- Pasting object / shapes / cutter in the different model will force the replacement of the copied variables by their values if the source model is still opened (otherwise default values are used).

The Advanced edition sub menu

When working with a complex model this menu will give you some extra features:

Action	Description
Reverse shape orientation in children	Changes positive faces into negative ones and vice versa
Convert and extend infinite cutters in children	Convert cutters from "Infinite difference" boolean operation to "Difference" and extend the extrusion to simulate an infinite cut.
Remove color property in children	Removes the color attribute of all objects in the selected sub-tree
Remove Thermica property in children	Wipe clean the thermal properties of the shapes in the selected sub-tree
Convert extruded polygon into 3 points base extruded polygons	Takes all 4 and more points base extruded polygons and cut them into several 3 points base extruded polygons. This is necessary for exporting into STEP-TAS for example

Action	Description
Convert polygon to triangles	Converts all 4 and more points flat polygons into triangles. This applies only on the simple polygons (not extruded) and is necessary for exporting into STEP-TAS as well
Select small volumes in children	Opens a pop-up asking you for the maximum volume of shapes to select
Remove holes in children	Applies only on CAD shapes : remove the holes into these shapes
Remove chamfers in children	Removes chamfers of CAD shapes
Remove fillets in children	Removes fillets of CAD shapes
Tessellate CAD shapes in children	Converts CAD shapes into real geometric data. This is the first step for converting CAD shapes into Systema shapes
Facet shape in children	Converts Tessellated shapes into Faceted shapes. This is the second step for converting CAD shapes into Systema shapes
Transform shape in triangles in children	Converts Faceted shapes into Systema triangles. This is the third and last step for converting CAD shapes into Systema shapes

All these actions are accessible from the bottom bar and from the "Modeler" menu of the top menu bar.

Warning: from the menu bar the actions perform on the whole model.

Geometrical models in SYSTEMA

What is a geometrical model ?

A geometrical model is a set of *shapes* which can be grouped into geometrical *objects*. Shapes can be of two types : *shell shapes* or *volumic shapes*.

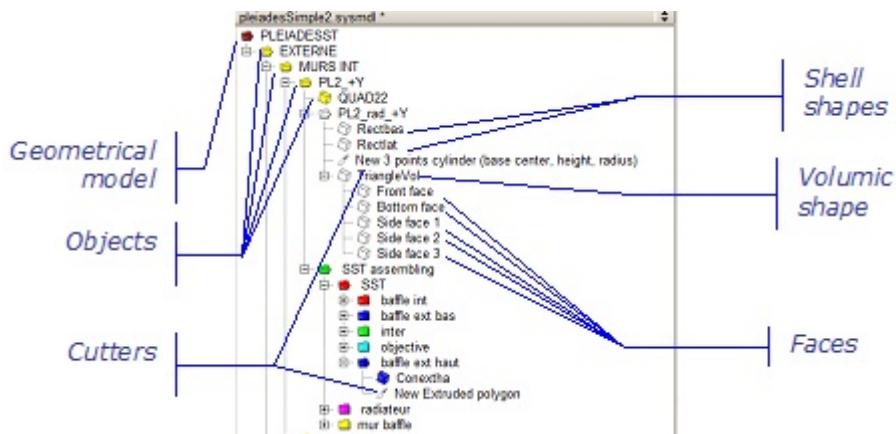
A *volumic shape* contains the list of its *faces* that delimit its frontiers.

A geometrical model may contain *cutters* which allow to cut shapes and/or objects using cylinder or extrusion (extruded polygon).

A geometrical *object* may contain *shapes* and/or other geometrical *objects*. In this case, it is called a *container*. Shapes, objects and cutters have properties (called *attributes* in SYSTEMA) that can be consulted or modified using the edition window.

The geometrical model in the browser window

The browser window displays the hierarchical organization of models organized into objects, shapes (shell shapes, volumic shapes and their faces)and cutters, as shown in the example below:



Objects are symbolized by a "folder" icon followed by the name (example MURS INT).

Shapes and faces are symbolized by a "shape" icon followed by the name (example: QUAD22).

Cutters are symbolized by a "shape" icon followed by the name (example: New Extruded polygon).	<i>Note: The color of an object/shape is determined by its properties (attributes). The blue triangle means that properties are defined. Further information is given below.</i>
	<i>Note: The color of the shapes icon's background indicates if there are any warnings (orange) or errors (red) associated with it.</i>

The geometrical model in the 3D views

The 3D views show the geometrical model in a three-dimensional form. They are used to manipulate and modify the model interactively.

Axis systems

Shape coordinates are expressed by means of Axis systems.

The *Main axis system* is the default axis system.

A *Local axis system* is defined for an object or for a shape, as soon as a transformation has been applied to it, via the **Transformation** area in the edition window (refer to the "Object attributes" and "Shape attributes" topics).

Axis system display

The *Main axis system* is **always** displayed when a model has been loaded in a 3D view.

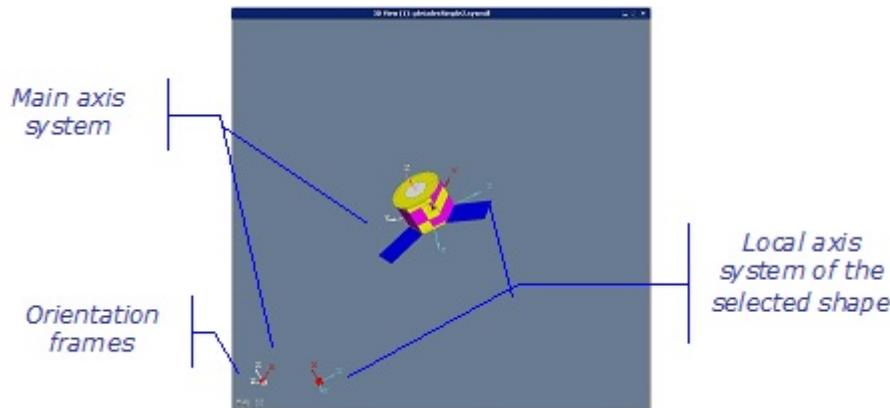
When you select an object or a shape, its *Local axis system* is displayed **if it exists**, else, the *Local axis system* of the first parent associated to a *Local axis system*.

The display of the the axis system at model's level can be desactivated from the configuration window but the orientation frames are always displayed.

	<i>Note: If no Local axis has been defined, only the Main axis system is displayed.</i>
--	---

Example

The following example shows a model in a 3D view where a shape with a *Local axis system* has been selected:



Quick and easy 3D manipulation

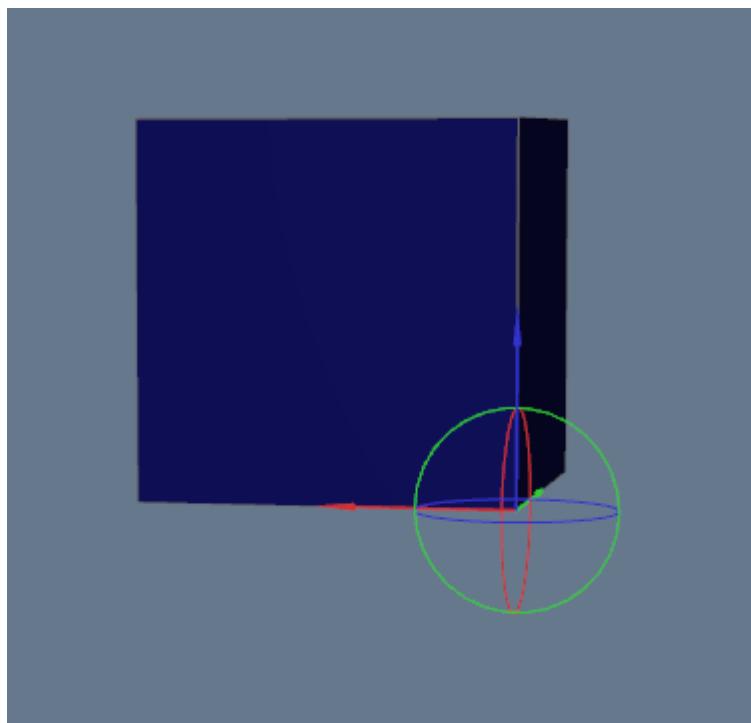
The gizmo manipulator provides a new way to move and rotate objects and shapes of a model directly in the 3D view. The gizmo is an alternative to the transformations available in the modeler. Depending on the actions you are looking for, you can choose between the definition of transformations and the Gizmo to manipulate models.

The gizmo :

- Can quickly be repositioned and reoriented for better control of the transformation.
- Let's you transform your current selection (shapes and objects)
- Provides a quick way to define the exact value of the angle of a rotation or the distance of translation
- Can take into account magnetism to easily align or snap model elements.

How to use the gizmo

- Open a model and display it in the 3d view. The gizmo icon  is available on the left bottom corner of the 3D view
- Click on the gizmo icon to active it in the 3d view:





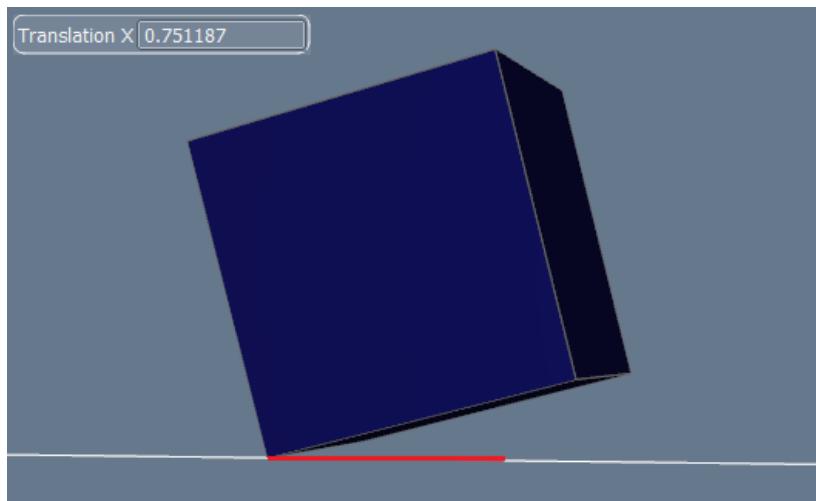
Moreover the gizmo icon is now and a new icon is displayed to activate/deactivate the magnetism when using the gizmo. Axis of the gizmo are always displayed according to the *Main axis system*. The colors used are :

Red : transformations on X

Blue : transformations on Y

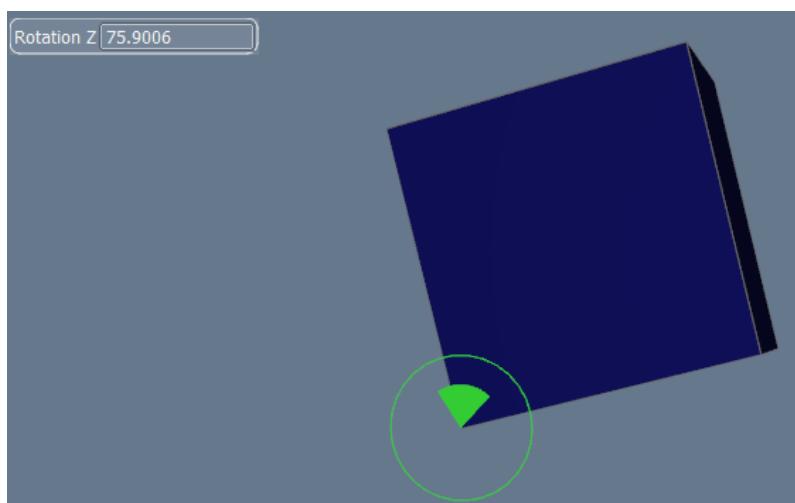
Green : transformations on Z

- To move the selected elements in a direction, select the relevant arrow and drag it :



The value of the translation is displayed in the 3D view. You can edit directly the value with **Ctrl+Shift+X** shortcut.

- To rotate the selected elements around one axis of the gizmo, select the relevant circle and drag it :



The value of the angle is displayed in the 3D view (in degrees). You can edit directly the value with **Ctrl+Shift+X** shortcut.



Any modifications done with the gizmo can be cancelled with **Undo(Ctrl+Z)**.

⚠ Transformations done with the gizmo are stored as a single cosines matrix. So any transformations defined previously are taken into account in this unique transformation. Please refer to [Modeler technical annex](#) for more details.

How to change the position of the gizmo

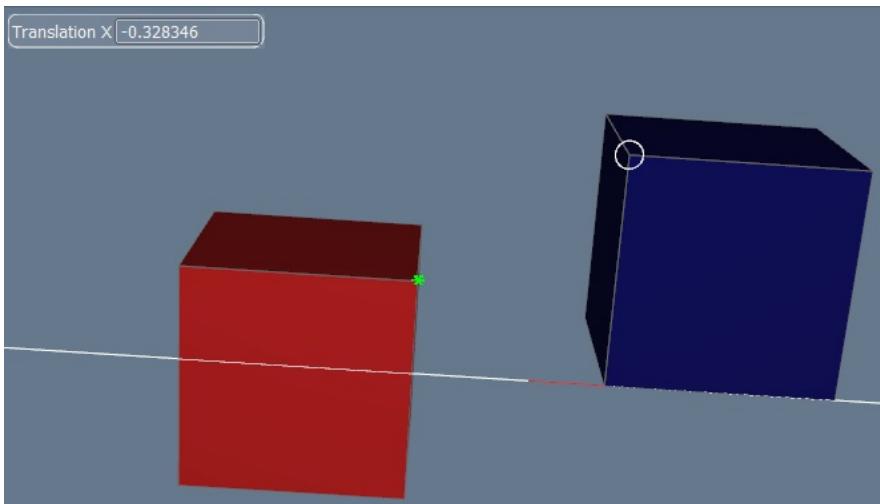
- In the contextual menu of the 3D view of the model, select **Gizmo - Select center of Gizmo**
- With **Alt+click**, pick a point in the current 3D view to define the new center of the gizmo
- If you want to get back the default position of the gizmo, select **Gizmo - Reset center**

⚠ The position of the gizmo is not saved in the model file.

How to use the gizmo with magnetism

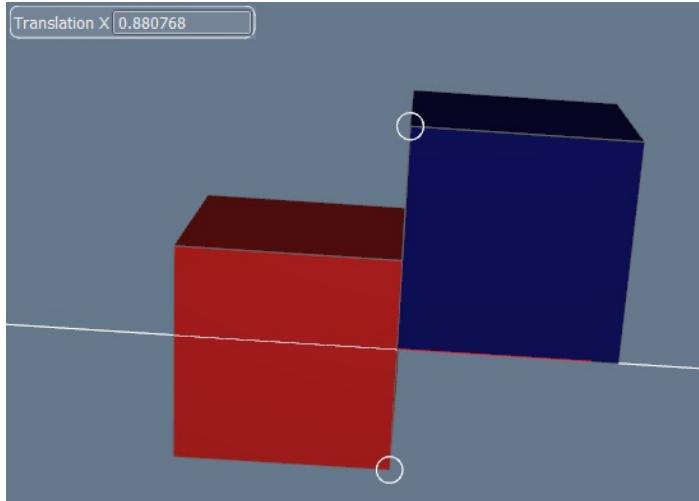
The magnetism helps you to move elements of the model according to other elements. For example, you can easily stick or align shapes with it. Both translations and rotations done with gizmo can use magnetism.

- Once the gizmo is displayed, you can activate the magnetism by clicking on  . The icon becomes 
- Select an axis of the gizmo and drag it :



The default magnetic point of the translated element is displayed as a white circle.

- Move the cursor on the point you want to align with (in the example on the red cube).

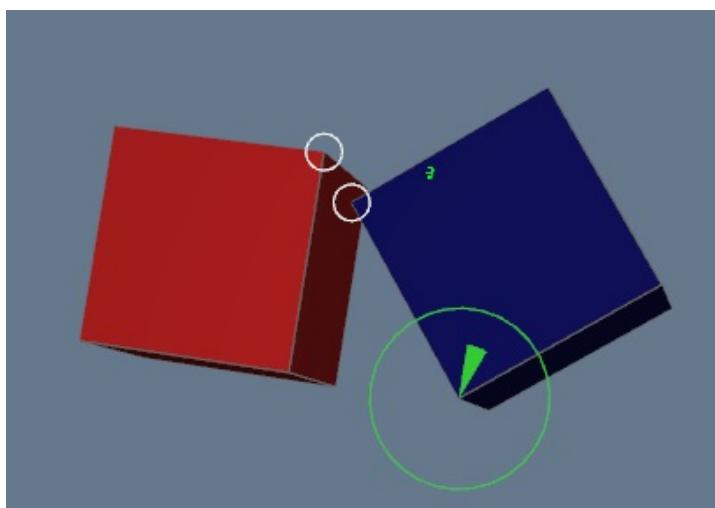


In this example, both magnetic points have now automatically the same X value.

You can change the magnetic point of the selected elements. To do this :

- Select **Gizmo - Select magnetic point on a shape** in the contextual menu
- Pick a point in the current 3d view with **Alt+click**
- You can reset the definition of the magnetic point with **Gizmo - Reset point** option

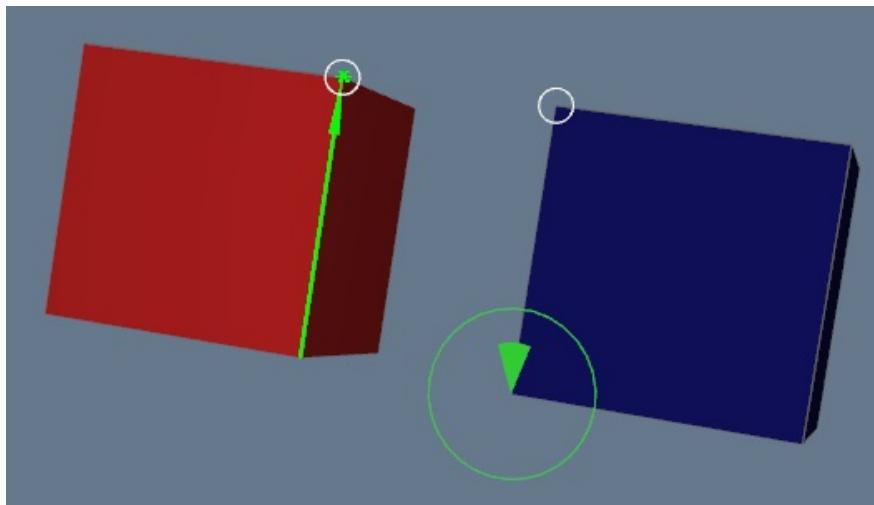
By default, rotating an element with magnetism will try to align the projection in the rotation plane of the magnetic points and the center of the gizmo :



You can define a magnetic segment instead of magnetic point. to do this :

- Select **Gizmo - Select magnetic segment to reach** in the contextual menu of the 3D view
- Pick 2 points in the current 3d view with **Alt+click**

When a magnetic segment is defined, magnetism will try to make the magnetic segment and the segment defined by the magnetic point and the center of the gizmo colinear:



You can reset the magnetic segment to reach with **Gizmo - Reset magnetic segment** option

The SYSTEMA shapes

The available shapes are divided into families. Each family is characterized by a basic geometric shape (box, polygon, etc.) and may itself be divided into subcategories (for example, the box family comprises 4-sided boxes, 5-sided boxes and 6-sided boxes).

The shapes may be drawn statically, interactively and/or by using an icon in the icon bar.

Shapes can be either *volumic* or *shell*. A *volumic shape* is composed of *faces*. When not stated explicitly in the name, shapes are of type *shell*.

Contrary to volumic and shell shapes: *faces* cannot be renamed, copied, or pasted. They are part of the definition of a volume.



Note: By default, volumic shapes are not available. Go to File->Settings->Modeler to activate them.

The following table presents the list of available SYSTEMA shape families, subcategories and drawing means (static, interactive, icon):

Shape family	Subcategories	Drawing means	
		Static	Interactive
Antenna	Antenna with focal	X	
	Antenna with focus	X	
Box	4 faces box	X	
	5 faces box	X	
	6 faces box	X	X

Shape family	Subcategories	Drawing means	
		Static	Interactive
Cone	3 points cone (base center, height, radius)	X	X
	4 points cone (3 points on base, height)	X	X
	4 points truncated cone (3 points on base, 1 point for other base)	X	X
	4 points truncated closed cone (3 points on base, 1 point for other base)	X	X
	General truncated cone (3 characteristic points, cone base diameters, angles)	X	
Cylinder	3 points cylinder (base center, height, radius)	X	
	4 points cylinder (3 points on base, height)	X	X
	4 points truncated cylinder (3 points on base, height)	X	X
	General truncated cylinder	X	
Disc	3 points disc (center, disc axis, disc radius)	X	
	Disc	X	X
	4 points truncated disc (3 points on outer border, 1 point for inner border)	X	X
	4 points truncated closed disc (3 points on outer border, 1 point for inner border)	X	X
	General truncated disc	X	
Polygon	Polygon	X	X

Shape family	Subcategories	Drawing means	
		Static	Interactive
Polygon with holes	Polygon with holes (external border and holes)	X	
Parabola	Paraboloid (3 points)	X	
	Default truncated paraboloid	X	
Extrusion	Extruded polygon	X	X
	Extruded polygon with holes	X	
Quadrangle	Quadrangle (4 points)	X	X
Rectangle	Rectangle (3 points)	X	X
Revolution	Revolved polygon	X	X
	Revolved polygon with holes	X	
Sphere	3 points sphere (center, radius, plane of X direction)	X	X
	4 points sphere (4 points on sphere)	X	X
	4 points truncated sphere (3 points on upper border, 1 in downer)	X	X
	General truncated sphere		
Triangle	Triangle	X	X
Volumic box	Volumic Box	X	X
Volumic extruded triangle	Volumic extruded triangle	X	X
Volumic extruded quadrangle	Volumic extruded quadrangle	X	X
Volumic tetrahedron	Volumic tetrahedron	X	X
Volumic cylinder	Volumic cylinder	X	

Shape family	Subcategories	Drawing means	
		Static	Interactive
Volumic cone	Volumic cone	X	
Volumic sphere	Volumic sphere	X	
Volumic revolved quadrangle	Volumic revolved quadrangle	X	X

The Systema cutters

The available cutters are divided into families. Each family is characterized by a basic geometric shape (cylinder, extrusion) and may itself be divided into subcategories (for example, the cylinder family comprises 3 points cylinder (base center, height, radius), 4 points cylinder (3 points on base, height), 4 points truncated cylinder (3 points on base, height) and truncated cylinder).

They can define different boolean operations such as differences (inside cut), intersections (outside cut).

The infinite difference type is also available to ensure retro-compatibility with old models (prior to 4.8.3) but we highly recommend not to use them for stability and format compatibility concerns.

The following table presents the list of available SYSTEMA cutters families and subcategories.

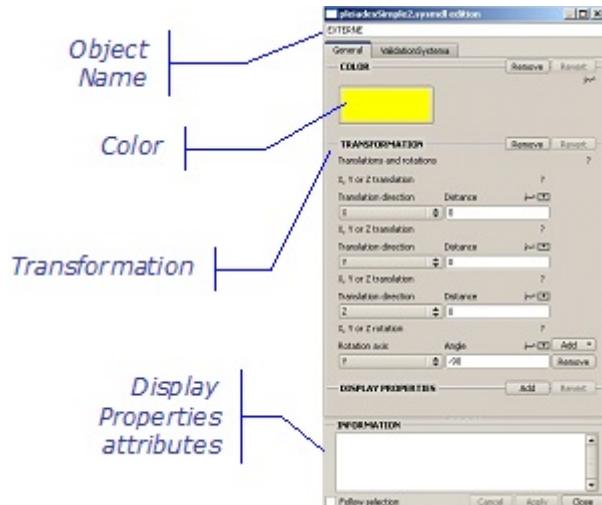
Cutter family	Subcategories
Cylinder	3 points cylinder (base center, height, radius)
	4 points cylinder (3 points on base, height)
	4 points truncated cylinder (3 points on base, height)
	General truncated cylinder
Extrusion	Extruded polygon

Limitations :

- Cutters cannot be applied on volumic shapes. They only apply to shell shapes.

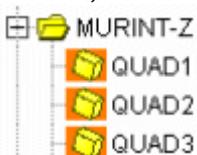
Object attributes

The root object has four attributes : the root object name, Color, Display Properties and Variables attribute and in some cases applicative attributes (accessible via a dedicated tab). Other objects have four geometrical attributes: the object name, Color, Transformation, Display Properties and in some cases applicative attributes (accessible via a dedicated tab). You modify them via the object edition window:



The Color attribute

This corresponds to the color of the object folder in the browser. It then becomes the default color of every shape it contains, as shown in the example below:



The default color for first level objects is brown-red. The color of these objects can then be changed. Color is an inherited attribute that can be modified:

- | | |
|--|--|
| <ul style="list-style-type: none"> each object has a default color inherited from its parent | |
| <ul style="list-style-type: none"> each object may have its own color (different from its parent), defined via the edition window | |

The Transformation attribute

For an object, the Transformation attribute defines a transformation applicable to this object and consequently to all shapes and objects contained in the object.



Note: The Transformation attribute is not inherited between objects.

When a Transformation attribute has been defined for an object, the associated transformation defines a *Local axis system* that is attached to this object (see "Axis systems" in the previous page). Then, coordinates for shapes contained in the object may be expressed either in the *Main axis system* or in the *Local axis system* (local coordinates).



Note: The local coordinates of a shape are those related to its first parent associated to a transformation.

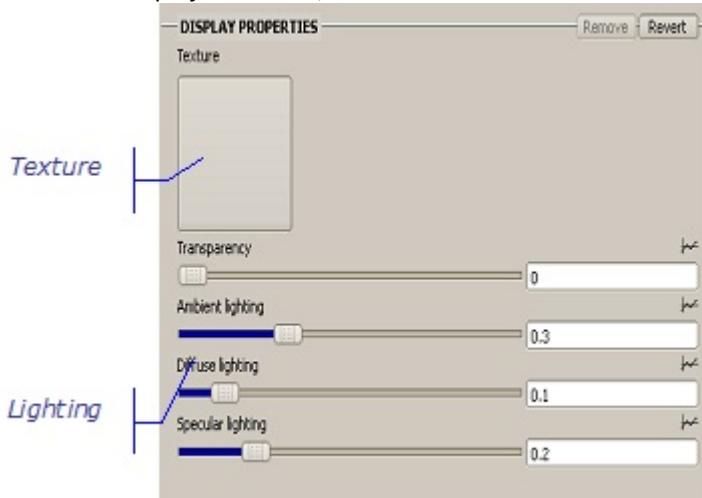
The Display Properties attributes

For an object, the "Display Properties" attributes define the texture and the lighting applicable to this object and consequently to all shapes and objects contained in the object.



For the selected object:

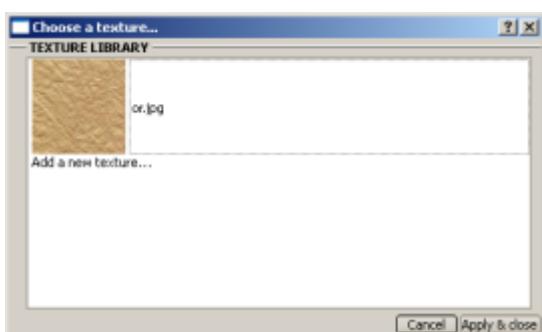
1. To add new display attributes, click **Add**.



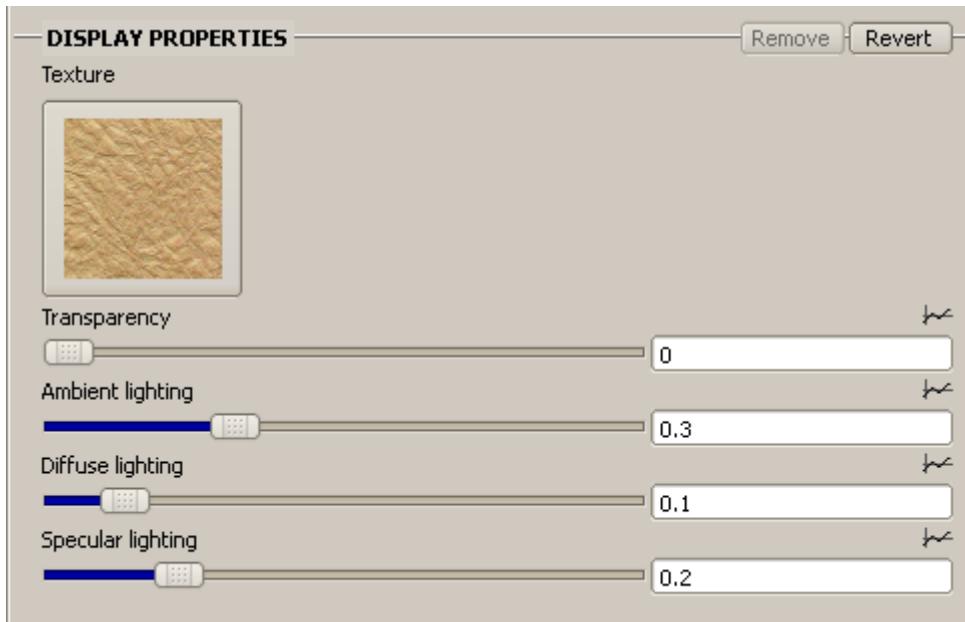
2. Click **Texture**.



1. Click « Add a new texture » to select the "texture" file.
2. Select the type of texture and click **Open**.



1. Click **Apply & close**.



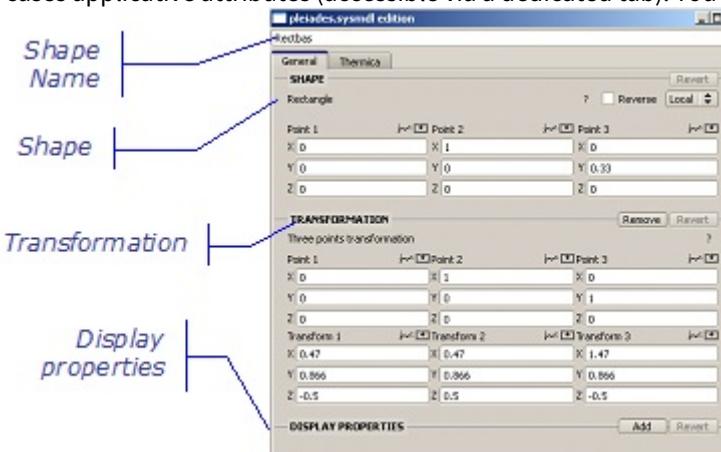
1. Possibly, adjust the lighting cursor.
2. To validate the new display properties attributes, click **Apply**.

Variables

Cf [Working with variables](#)

Shape attributes

Shapes have four geometrical attributes: the shape name, Shape, Transformation, and Display Properties and in some cases applicative attributes (accessible via a dedicated tab). You modify them via the object edition window:



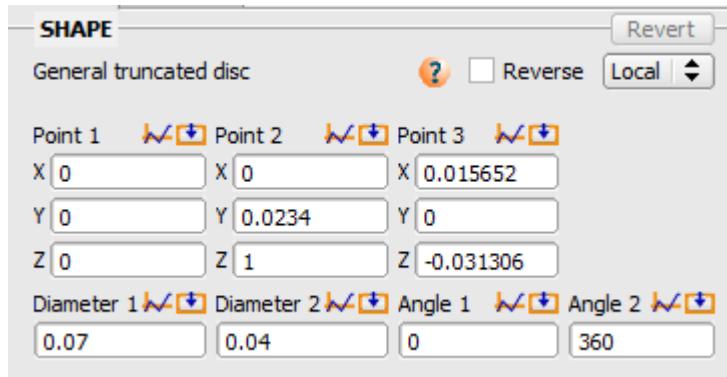
Note: The color of a shape is an implicit attribute inherited from its container.

The Shape attribute

The Shape attribute is a set of points and values that exactly define the shape. These values can be displayed either in the *Main axis system* or in the *Local axis system* of the object containing it, by selecting the required value in the **SHAPE** area:

- select **Main** to display coordinates related to the *Main axis system*,
- select **Local** to display coordinates related to the *Local axis system* of its parent (or of the first parent associated to a transformation).

The example below shows the Shape attribute expressed with local coordinates for a truncated disc, in the edition window:

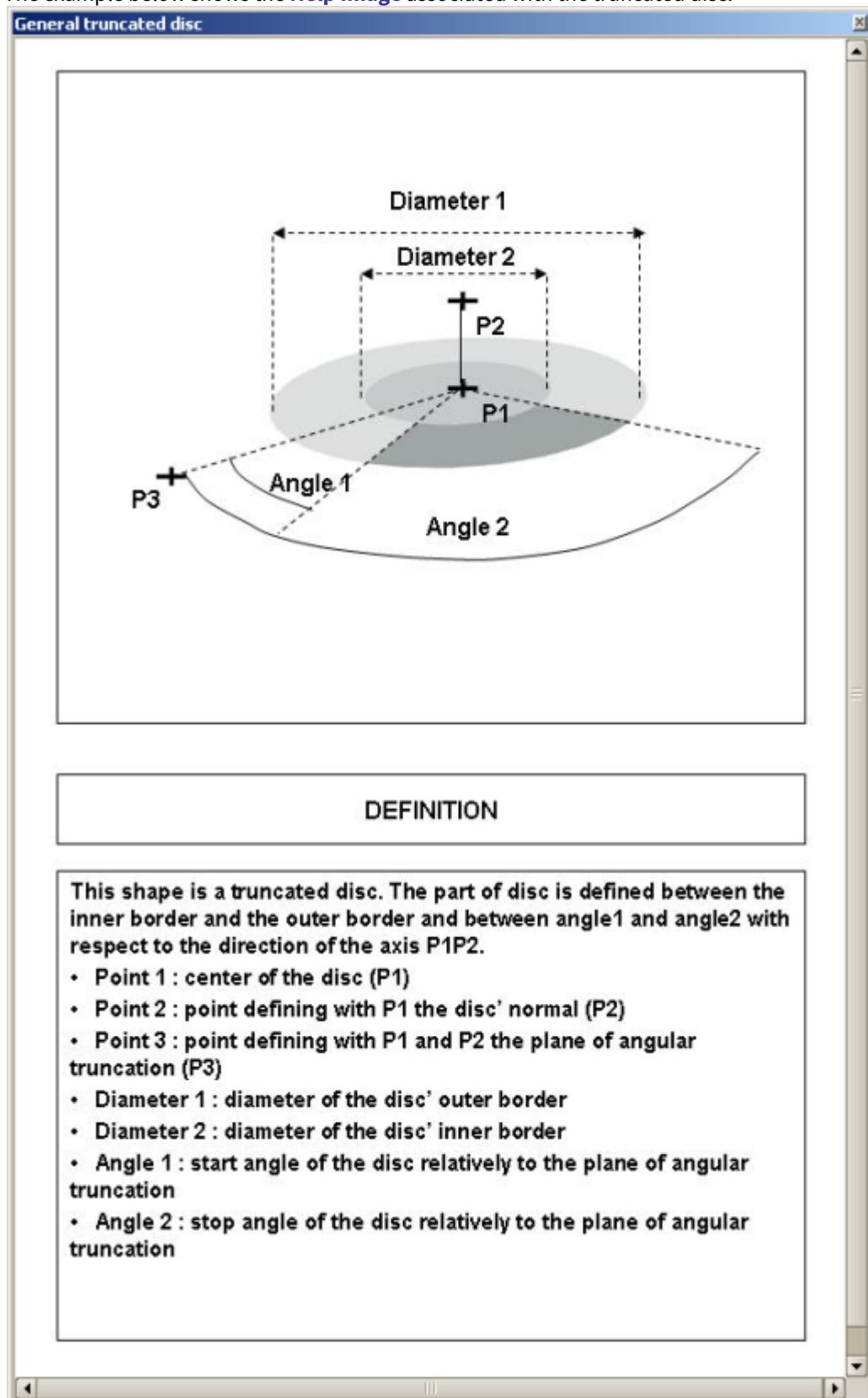


The **Reverse** checkbox can be checked to reverse the normal axis (the normal axis orientation defines the positive side of a shape).



Note: In the edition window, shapes are defined using points (Point 1, Point 2, etc) and geometric values such as diameter and angle. The meaning of this information is explained in the Help Image, which can be accessed by clicking ? associated with the shape identifier.

The example below shows the **Help Image** associated with the truncated disc:



DEFINITION

This shape is a truncated disc. The part of disc is defined between the inner border and the outer border and between angle1 and angle2 with respect to the direction of the axis P1P2.

- Point 1 : center of the disc (P1)
- Point 2 : point defining with P1 the disc' normal (P2)
- Point 3 : point defining with P1 and P2 the plane of angular truncation (P3)
- Diameter 1 : diameter of the disc' outer border
- Diameter 2 : diameter of the disc' inner border
- Angle 1 : start angle of the disc relatively to the plane of angular truncation
- Angle 2 : stop angle of the disc relatively to the plane of angular truncation

The Transformation attribute

For a shape, the Transformation attribute defines a transformation applicable to the shape in its container's *local axis system*.

When a Transformation attribute has been applied to a shape, the associated transformation defines a *local axis system* that is attached to the shape.

When a shape is selected, *its local axis system* is visible in the 3D views in which the corresponding model has been loaded.

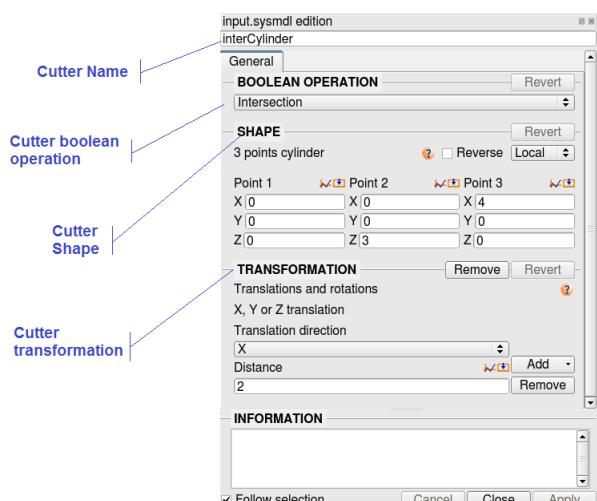
The Display properties attribute

For a shape, the "Display Properties" attributes define the texture and the lighting applicable to the shape in its container's *local axis system*.

The process is the same as the one used for an object.

Refer to the paragraph above "Object attributes - Display properties attributes".

Cutter attribute



Cutters have one attribute defining its boolean operation as well as two geometrical attributes: the Cutter shape attribute, and the transformation attribute.

The cutter boolean operation attribute

The boolean operation attribute defines the way the cutter influence the geometries to which it is linked.

The "Difference" operation make an inside cut of the related shape while the "Intersection" operation makes an outside cut.

An "Infinite Difference" option is also available for retro-compatibility with old models (prior to 4.8.3) but we highly recommend not to use it for new models (stability and format compatibility issues).

The boolean operation attribute is removed during .sysmdlits export (4.5.3) because only the "Infinite Difference" operation was available by then.

The cutter shape attribute

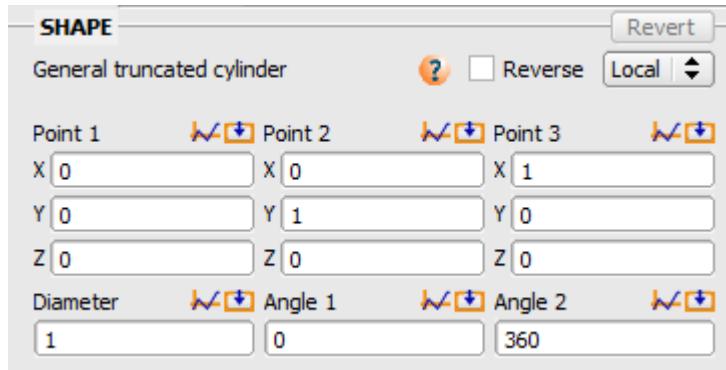
The example below shows the cutter shape attribute for an extruded polygon.

The cutter shape attribute is a set of points and values that exactly define the cutter shape.

These values can be displayed either in the *Main axis system* or in the *Local axis system* of the object/shape containing it, by selecting the required value in the **SHAPE** area:

- select **Main** to display coordinates related to the *Main axis system*,
- select **Local** to display coordinates related to the *Local axis system* of its parent (or of the first parent associated to a transformation).

The example below shows the cutter shape attribute expressed with local coordinates for a truncated cylinder, in the edition window:

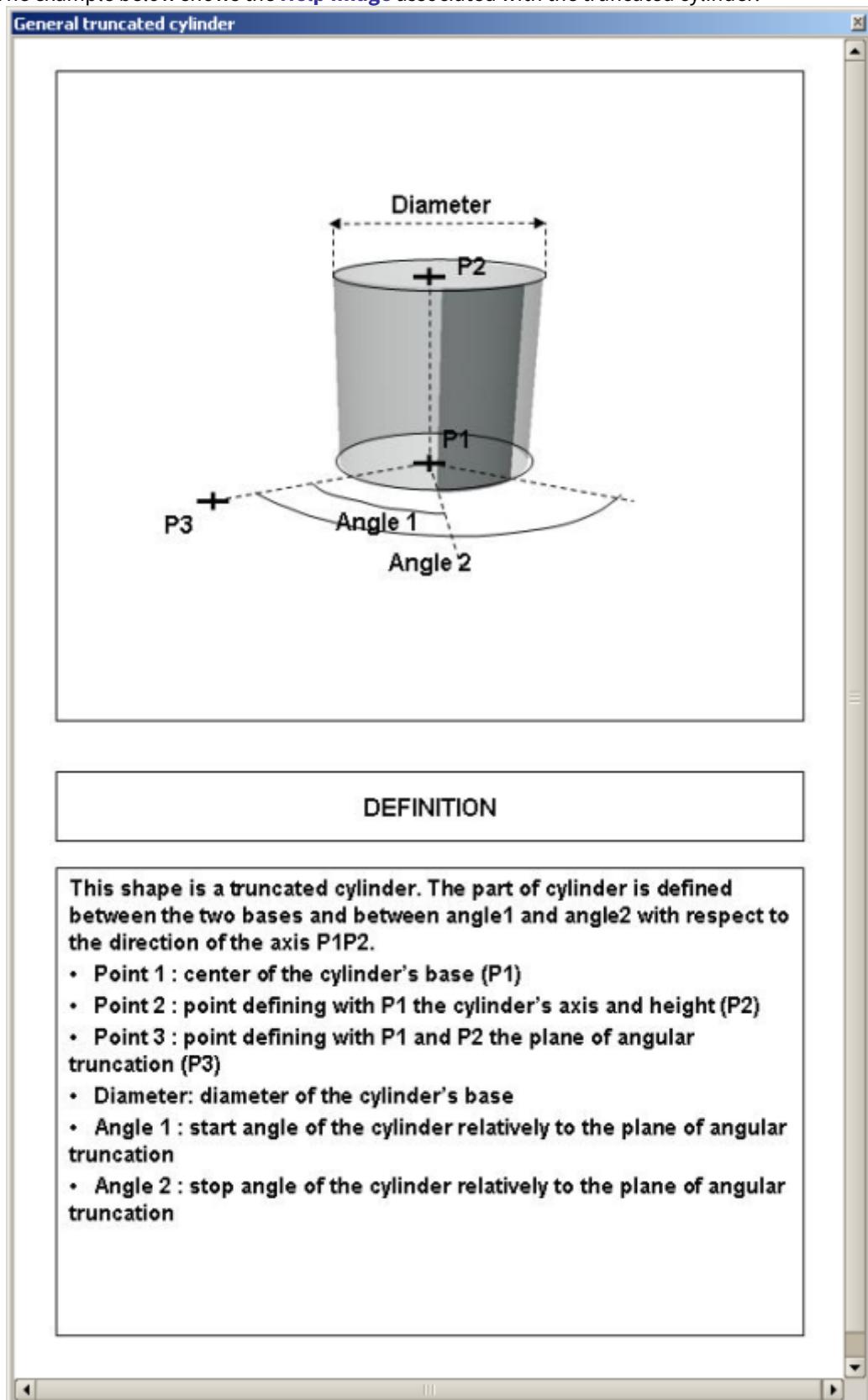


The **Reverse** checkbox can be checked to reverse the normal axis (the normal axis orientation defines the positive side of a cutter shape).



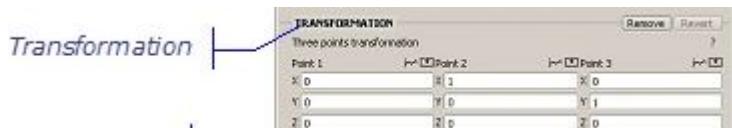
Note: In the edition window, cutters shapes are defined using points (Point 1, Point 2, etc) and geometric values such as diameter and angle. The meaning of this information is explained in the Help Image, which can be accessed by clicking ? associated with the cutter shape identifier.

The example below shows the **Help Image** associated with the truncated cylinder:



The cutter transformation attribute

For a cutter, the Transformation attribute defines a transformation applicable to the cutter in its container's *local axis system*. When a Transformation attribute has been applied to a cutter, the associated transformation defines a *local axis system* that is attached to the cutter. In fact, cutters transformations are exactly the same as shapes transformations.

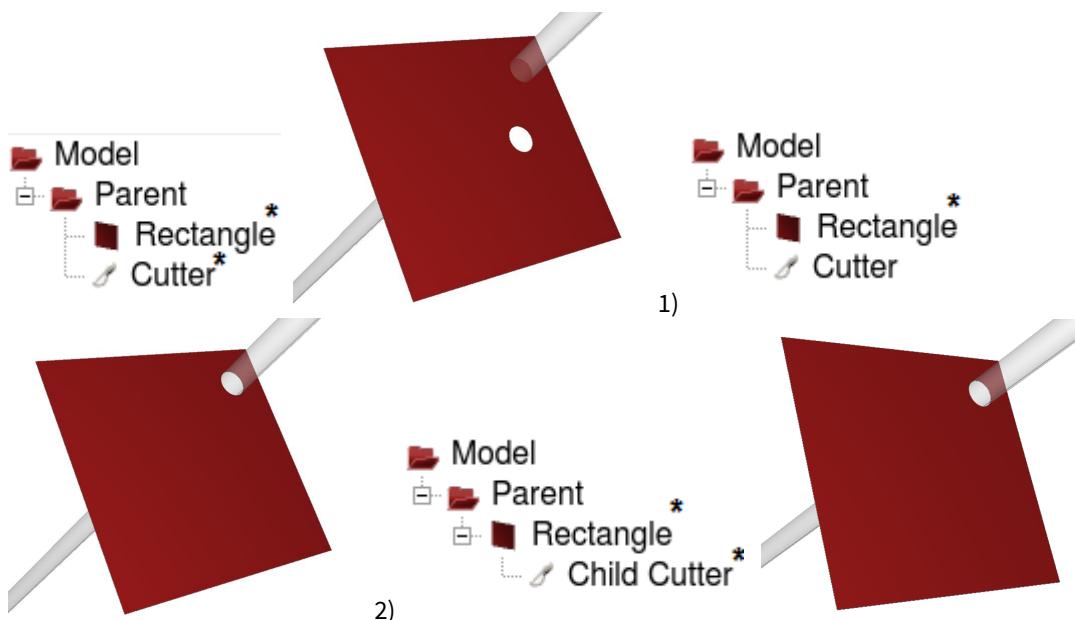


Limitation (4.8.1+):

In some particular cases where shapes and cutters are at the same level in the tree and both hold different transformations, the cut can be glitched in the 3D.

Two workarounds exist:

- 1) Resolving one of the transformations (example: the one of the cutter)
- 2) Inserting the cutter as a child of the shape to cut and adapt its transformation



*: holds a transformation

Application-dependent properties

According to your SYSTEMA configuration, some applications can be interfaced with the SYSTEMA application. In this case, the geometrical model elements have application-dependent properties, in addition to the geometric attributes.

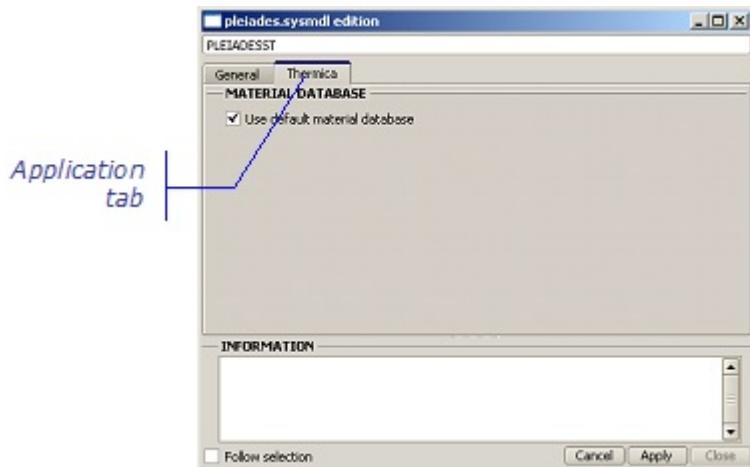
These properties can be consulted or modified using the edition window, via a specific tab (which is called the *application* tab in this document). Each application has its own tab.

Application-dependent properties at root level

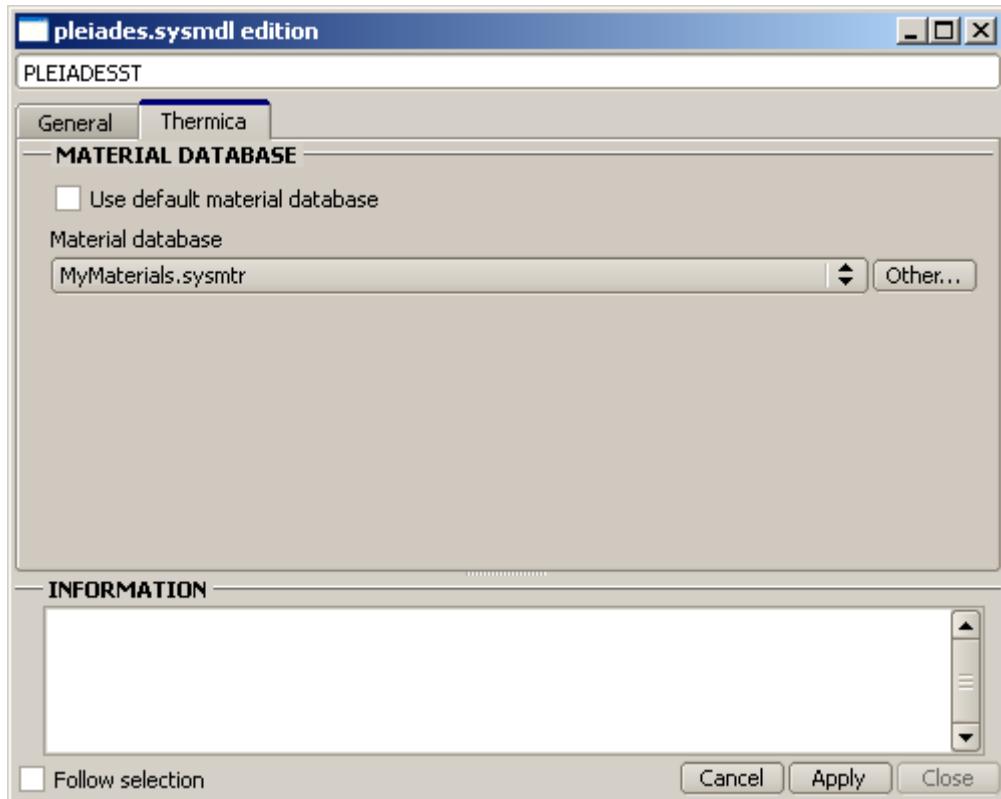
For a given application, a material file may be needed to define the application-dependent properties. The used file is declared at the root level of the geometric model.

The application generally has a default material file which is used by the model until the user choose another material file.

The choice of the material file can be done via the *application* tab of the edition window:

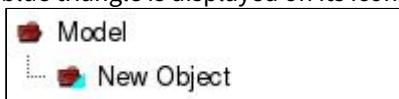


The user unchecks the **Use default material database** option and selects the material database among the material databases loaded in memory or by choice through a file manager opened with **Other** button. The selection is recorded via **Apply**. Once a material file not already loaded in memory is chosen as material file, it becomes accessible in the **Materiel** tab of the Modeler.



Visualization of the defined properties in the browser

You can see in the browser that a node (shape or object) has an application-dependent property defined when a blue triangle is displayed on its icon.

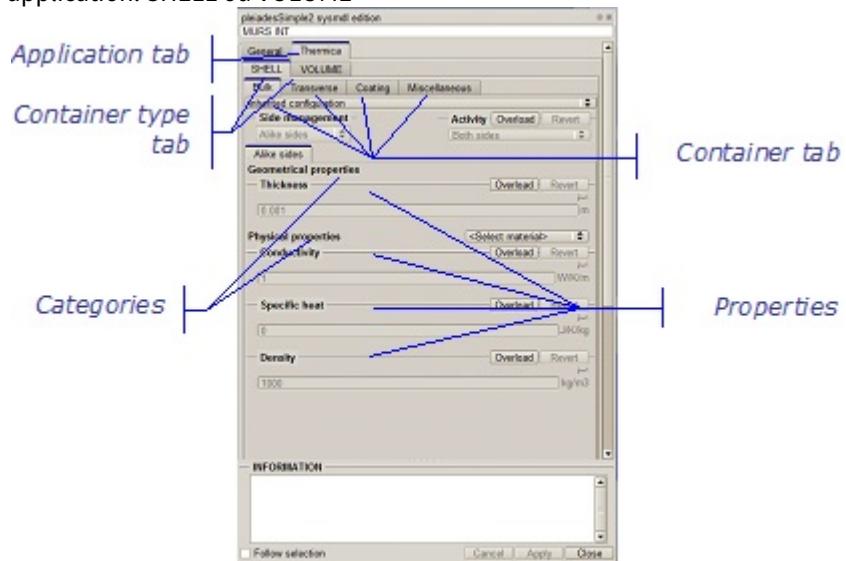


Here, "New Object" has an application-dependent property defined.

Application-dependent properties at object and shape level

For a given application, objects and shapes may have application-dependent properties. These properties are grouped into sets of properties accessible via specific tabs (which are called *container tabs* in this document) within the *application tab* of the edition window.

Container tabs can themselves be grouped in two container types if volumes management is enabled for the application: SHELL ou VOLUME



Note: The names of the application and container tabs are defined by means of the application parameterization. Each *container tab* corresponds to a set of application-dependent properties which are defined by parameters: a configuration, the **Side management** parameter, the **Activity parameter** and application-dependent properties (grouped into categories).

The container configuration

For the set of properties in a container, the configuration indicates whether or not they are inherited from the parent level:

Configuration value	Description
Inherited configuration	<p>The container parameters are inherited from the parent. The properties (grouped into categories) can be modified using the available options (Overload, Remove, Add, etc). The new values become the default values for the child levels.</p> <p>Side management is not modifiable. Activity may be overloaded.</p>
Local configuration	<p>Side management is modifiable in order to define a new configuration.</p> <p>Properties are not inherited and must be defined, or are set to their default values.</p> <p>Activity is inherited from the parent and may be overloaded.</p>

The Side management parameter

This parameter is used to indicate how to apply the properties to the element sides:

Side management	Description
Positive side	The properties only concern each element's positive side.
Negative side	The properties only concern each element's negative side.
Distinct sides	The positive side and the negative side do not have the same properties; they are defined via two different tabs.
Alike sides	The properties concern both sides.
	<p><i>Note: The positive side of a shape is indicated by the direction of the normal axis.</i></p>
	<p><i>Note: In the case of an Inherited configuration, this parameter is not modifiable.</i></p>

The Activity parameter

When there is no Side management, this parameter indicates whether the properties are active or not, independently of the element sides. Note that Side management is never enabled on volumes.

When associated with Side management, this parameter indicates on which element sides the properties are active:

Activity parameter	Description
Only positive side	Only the properties of the positive sides are applied.
Only negative side	Only the properties of the negative sides are applied.
Both sides	The properties are applied to both sides.
No activity	No property is applied.

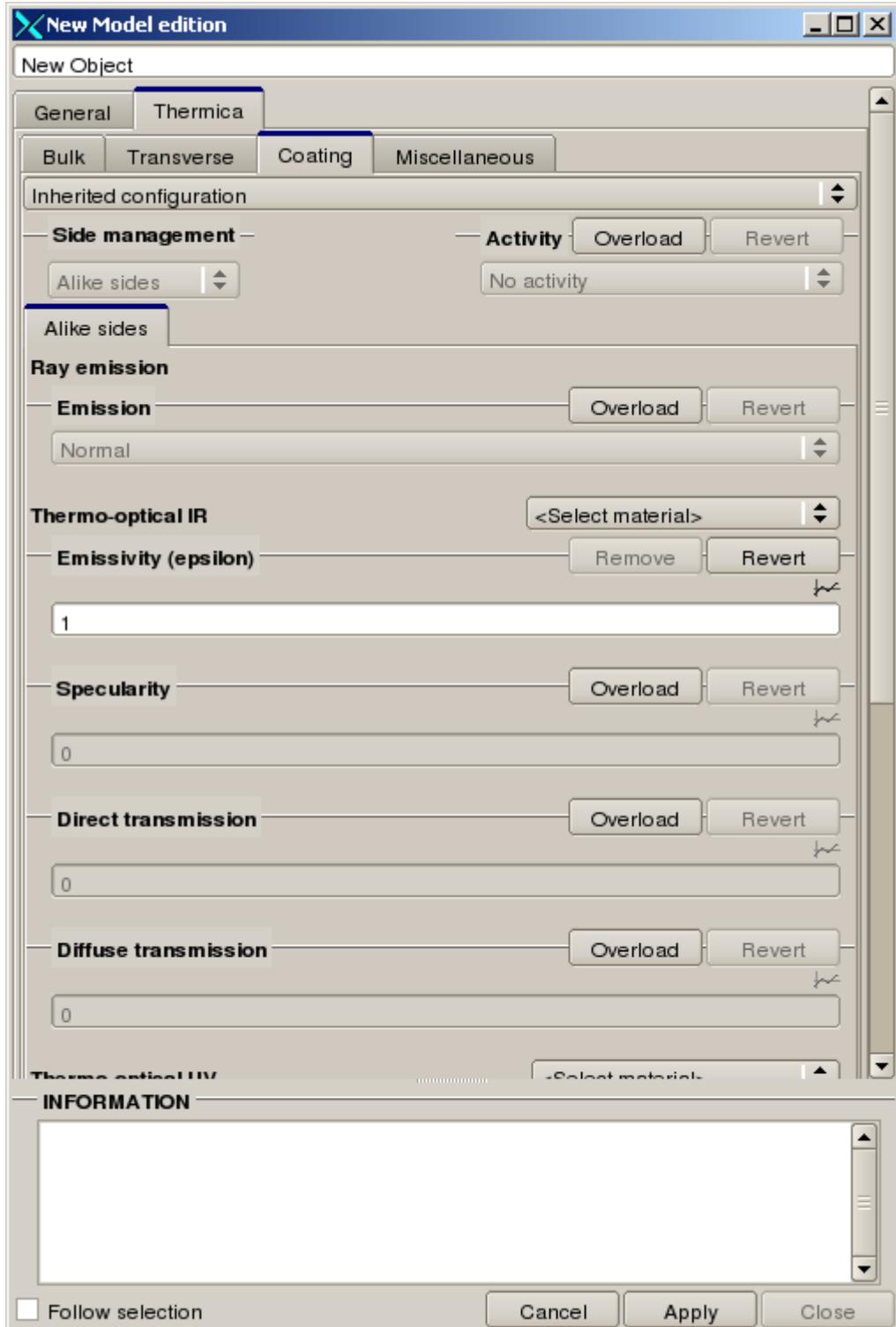
The properties

The properties are grouped into categories. Grouping properties by categories makes it possible to apply the same material to the properties. When some material is associated with a category, the category properties are inherited from the material.

If a property has no default value, you have to use the Add button to define it.

If a property has a value which has been inherited from the parent level or from some material, you can use the Overload button to modify it.

The value of a property can be a textual value which can be chosen in a list, a real value, a variable defined in the root object (cf. page) or a curve. The access to the variable choice or to the curve definition is done through the  icon.



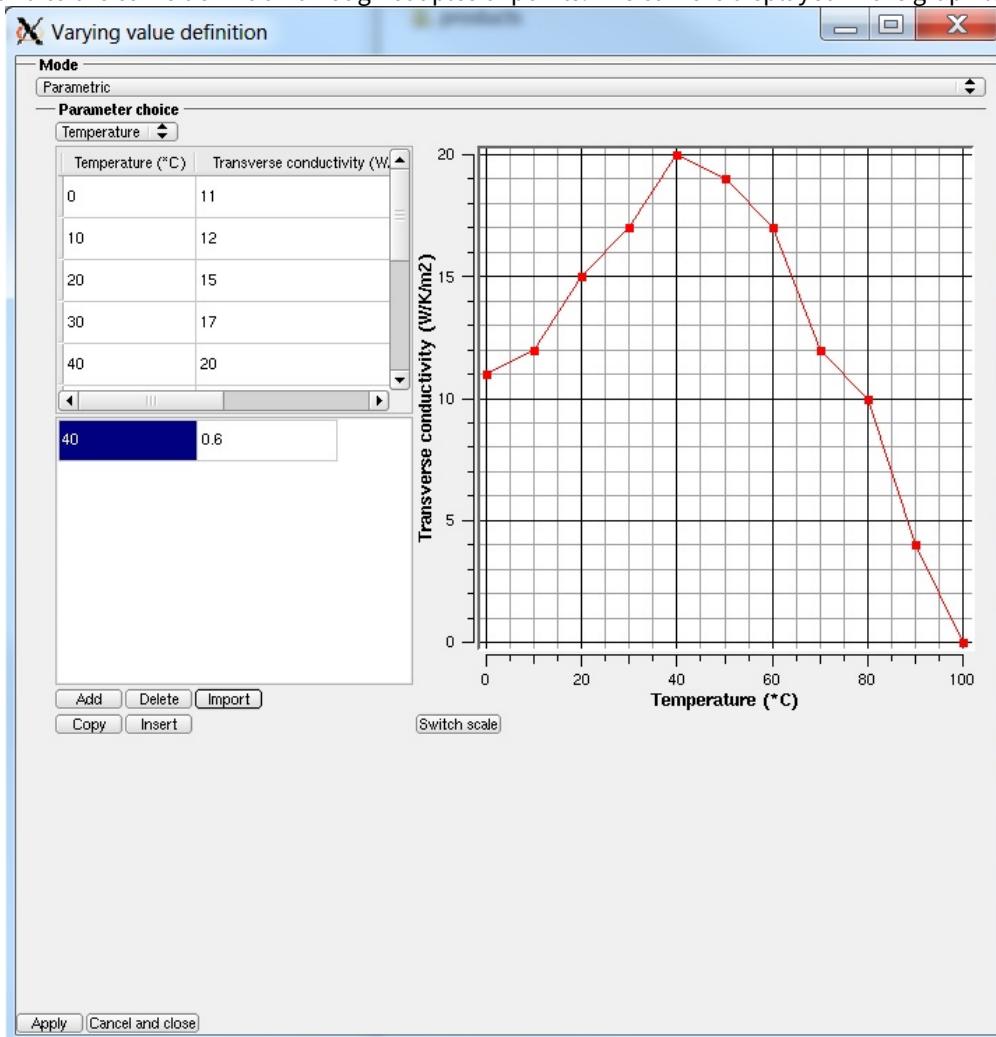
Click on the  icon opens a new window named "Variying value definition". The first option in the window is the choice of the mode for the attribute.

Mode value	Description
Parametric	The property's value is defined by a curve relatively to the abscissa which is a parameter defined in a list specified by the application for the property (Thermica in our example).

Mode value	Description
Variable	The attribut takes the value of a variable defined in the root object.
Constant	The attribut has a constant value defined by the user*. This choice authorises to leave the Variable or Parametric mode.

The parametric mode

The parametric mode gives access to the parameter's choice in the list defined by the application for the attribute and to the curve definition through couples of points. The curve is displayed in the graph located in the window.



Points can be defined either in the secondary table and then added to the main table using the Add button or directly imported from a .txt file structured with two columns (value1 <tab> value2) by using the Import button. The CIC files are also supported provided that the time is described as <day> <seconds> and not with a calendar date.

Once a couple of points exists in the first table, it can be modified by direct edition in the cells. An interval can be specified by the application for abscissa and ordinate. The cell of an incorrect value takes the pink color and the authorised interval is given as tooltip. The couple of points can be deleted by selection of the two cells and use of the Delete button.

Many cells can be selected in the first table and then copied to the clipboard using the Copy button.

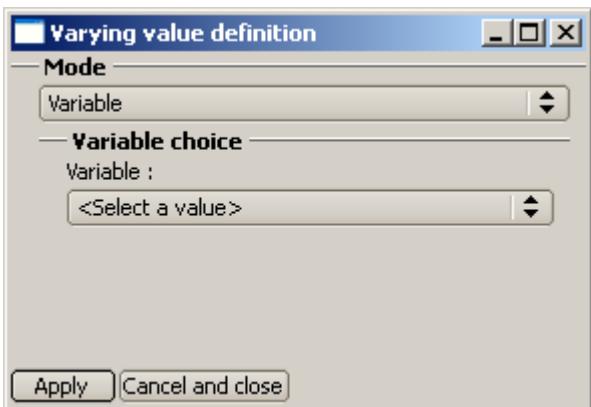
The current content of the clipboard (copy from an Excel file or from a text file editor as values separated by a

tabulation) can be added to the first table using the Insert button.

The user can switch from normal scale to logarithm scale for the abscissa using the Switch scale button.

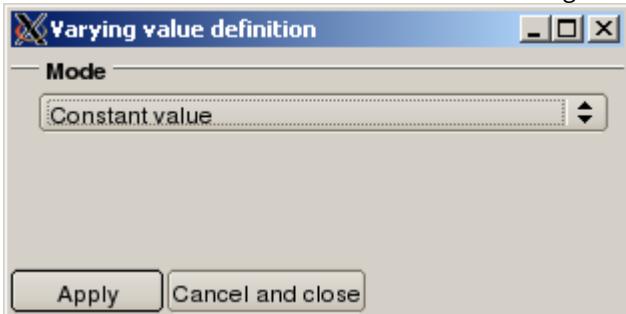
The variable mode

The variable mode authorises the choice of the variable in the list of compatible variables defined in the root object.



The constant mode

The constant variable mode authorises the user to give a constant value to the attribute.



For more information...

Refer to the application documentation to obtain more information about the application-dependent properties within SYSTEMA.

Display options in the 3D views of the Modeler tab

Right click menu in the 3D views

Within the 3D views, the right click menu contains various display options.

Some of them are options common to all 3D views of every SYSTEMA tab:

Hide , **Hide others** , **Hide all** and **Show all** options concern either the selected components or all components;

	<i>Note: Hide / show actions are reflected in all 3D views in which the model is loaded, and also in the browser window.</i>
--	--

the **Configuration window** option.

	<i>Note: This action is local to the viewport (several models may be loaded in a viewport).</i>
--	---

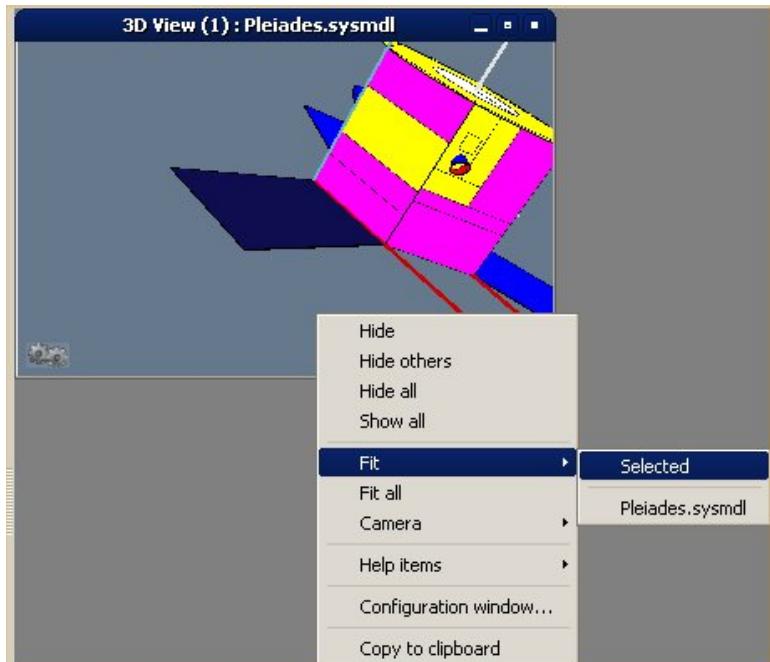
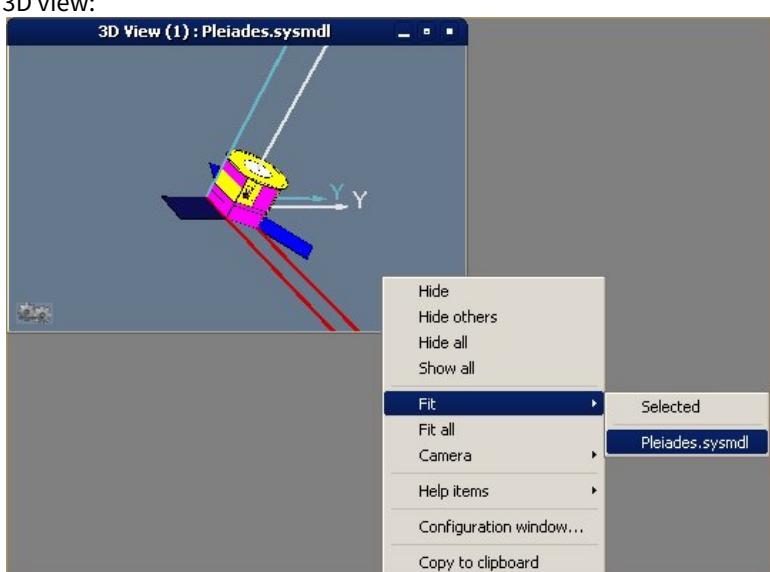
For more information about these options, refer to the "General User Interface" chapter.

Specific display options for viewing models



Note: All these actions are local to the 3D view.

Zoom facilities

Option	Use
Fit - Selected	<p>Adjusts the zoom in order to fit entirely the selected shape(s) in the 3D view:</p> 
Fit - Imported loaded model	<p>Adjusts the zoom in order to fit entirely the designated model in the 3D view:</p> 

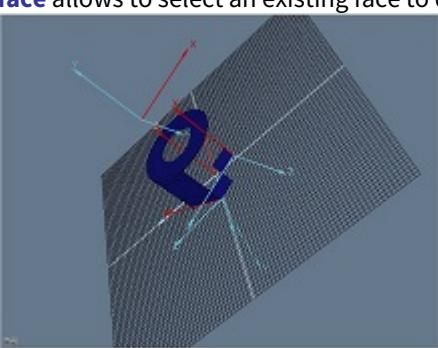
Option	Use
Fit all	Adjusts the view to the whole scene, which means all the loaded models
Camera – Focal point	Allows you to zoom in or out on the model's focal point: * First select the option and the point to be used(the current focal point is indicated within the location bar) <ul style="list-style-type: none"> Then hold the left mouse button down and slide the mouse forward/backward
Camera – Clipping plane	Allows you to zoom in or out on the model's clipping plane
Camera – Direction	Allows you to move through the model following a direction determined by an option: * First select the option and the direction to be used(the current vector is indicated within the location bar) <ul style="list-style-type: none"> Then hold the left mouse button down and slide the mouse forward/backward

Help items

Help items creation

Creates geometrical objects that can be used to help you create points and new surfaces (see example of use in "Creating a new geometrical model based on an existing one")

You can either use the icon bar, the contextual menu or the menu "Modeler/Create help item". Then, you will have to select points or faces to define the position of these help items.

	<i>Note: To select a face, you must click on it in the 3D view or in the browser. To select a point, you must press the "Alt" key, and click on the point in the 3D view</i>
Option	Use
Help Items – Grid	<p>Allows you to draw a grid following a plan determined by an option:</p> <ul style="list-style-type: none"> Create grid by 3 points allows to select three points to define a plan (instructions are given in the log bar) Create grid by normal allows to select two points as the normal of your grid, then one point on the grid. Create grid by face allows to select an existing face to define a plan. Example: 

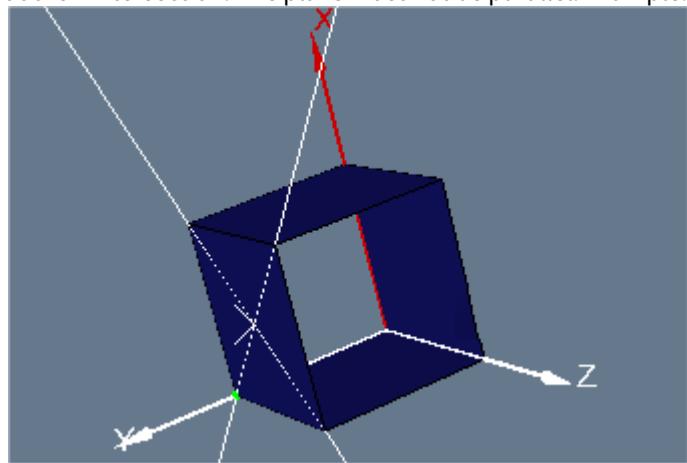
Every point on the grid can be selected to create shapes

i From Systema 4.9.1, it is possible to set the size of a cell and the number of cells in a direction. In previous versions, the grid is always composed of 50 cells with a size of norm(Point1Point2)/5.

Help Items – Line

Draws a line:

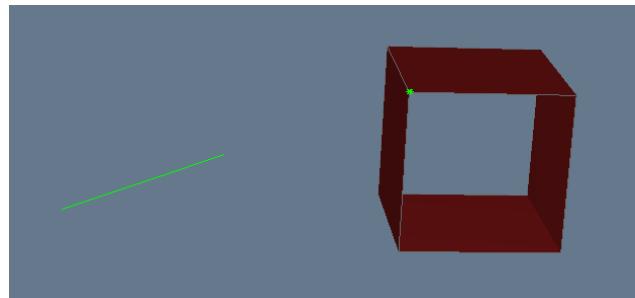
- **Create line** allows to select two points to define the line
- **Create line by direction** allows to select two points as direction of your line, then a third point defining the position of the line.
- **Create line by face** allows to select two faces to create the line at their intersection. The plans must not be parallel. Example:



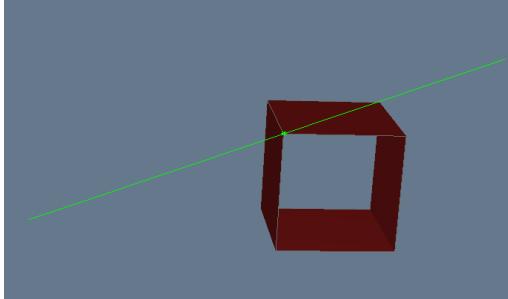
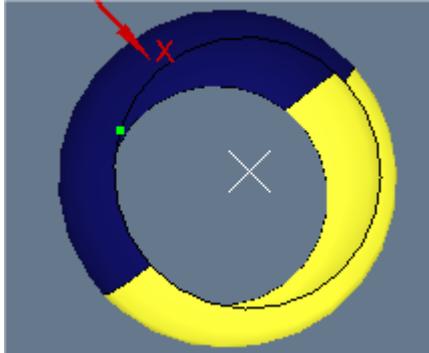
The point of intersection between two lines defines a new point that can be selected to create shapes

Help Items – Segment

Create a segment between two points using the **Create segment** option. You can adjust its length in the edition panel. The length can be negative.

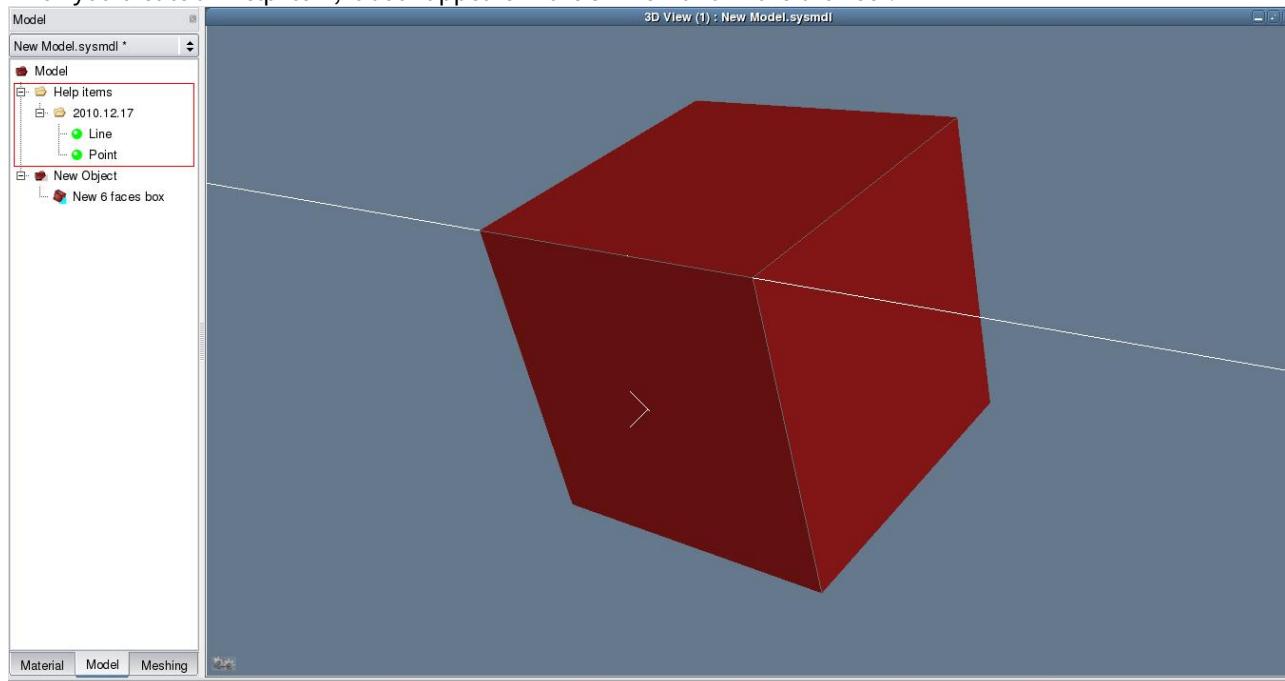


Origin		Target	
X 0	<input type="text"/>	X 2	<input type="text"/>
Y 0	<input type="text"/>	Y -0.75	<input type="text"/>
Z 0	<input type="text"/>	Z 0	<input type="text"/>
Length			
1	<input type="text"/>		

	 <table border="1"> <tr> <td>Origin</td><td></td><td>Target</td></tr> <tr> <td>X 0</td><td><input type="text"/></td><td>X 2</td></tr> <tr> <td>Y 0</td><td><input type="text"/></td><td>Y -0.75</td></tr> <tr> <td>Z 0</td><td><input type="text"/></td><td>Z 0</td></tr> <tr> <td>Length</td><td></td><td></td></tr> <tr> <td colspan="3"><input type="text" value="4"/></td></tr> </table>	Origin		Target	X 0	<input type="text"/>	X 2	Y 0	<input type="text"/>	Y -0.75	Z 0	<input type="text"/>	Z 0	Length			<input type="text" value="4"/>		
Origin		Target																	
X 0	<input type="text"/>	X 2																	
Y 0	<input type="text"/>	Y -0.75																	
Z 0	<input type="text"/>	Z 0																	
Length																			
<input type="text" value="4"/>																			
Help Items – Middle	Creates a point that is the middle of a segment using the Create middle option: you have to select two points in the 3D view The created point can be selected to create shapes																		
Help Items – Curve center	Creates a point that is the center of a curve using the Create curve center option: you have to select three points in the 3D view Example:  The created point can be selected to create shapes																		
Help Items --Sphere center	Creates a point that is the center of a sphere using the Create sphere center option: you have to select four points belonging to the sphere in the 3D view. The created point can be selected to create shapes																		
Help Items – Point	Creates a point that is the intersection of 3 planes using the Create point by face option: you have to select three faces that are not parallel in the 3D view. The created point can be selected to create shapes																		

Help items modification and suppression

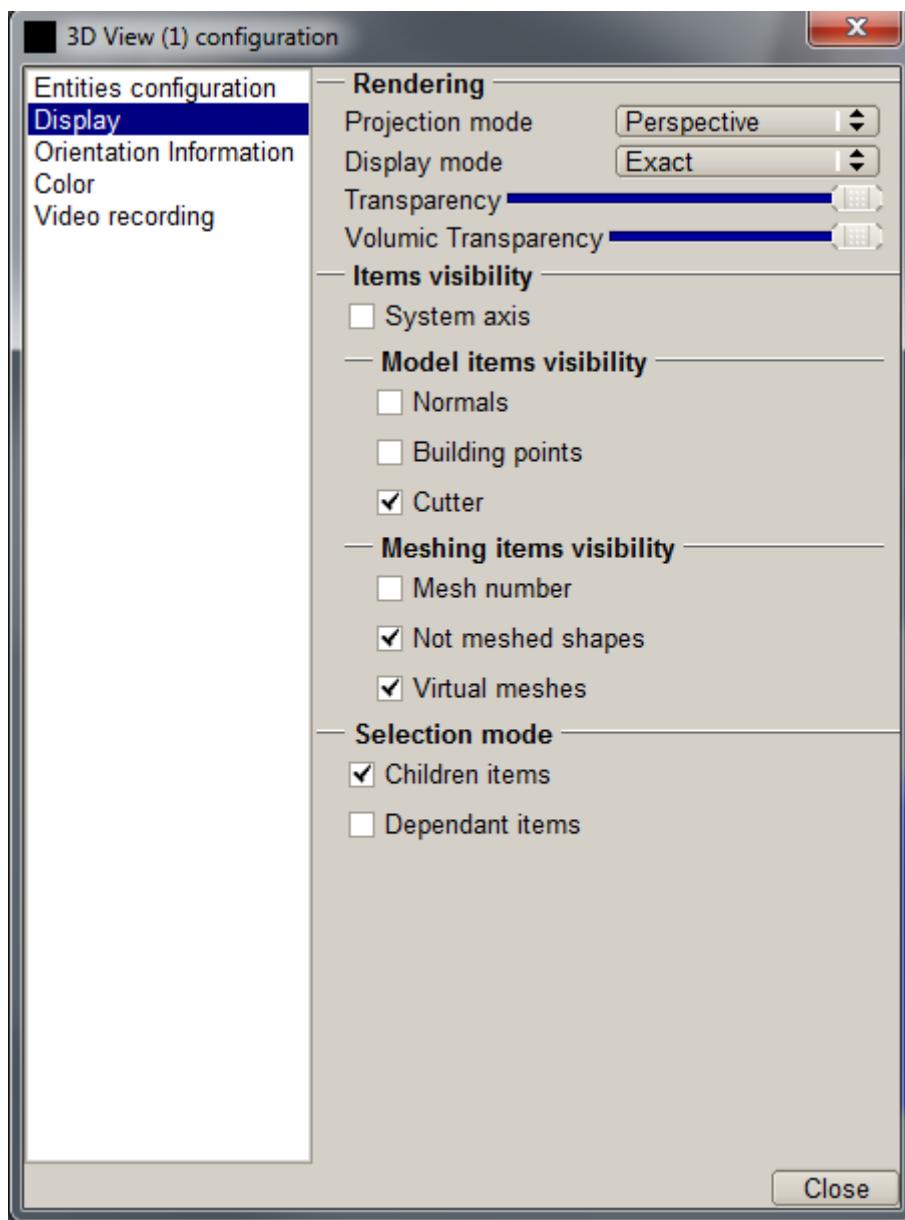
When you create an help item, it both appears in the 3D view and in the browser.



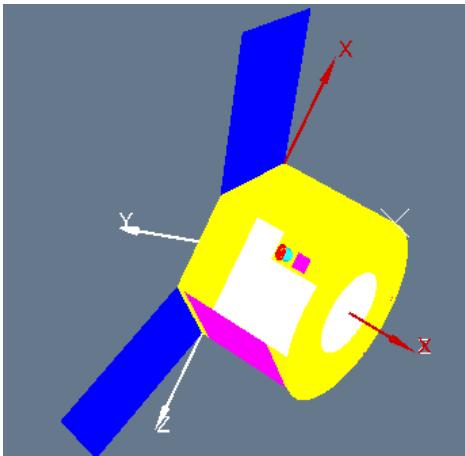
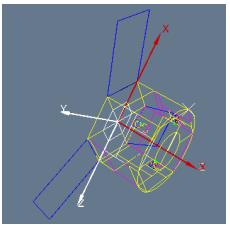
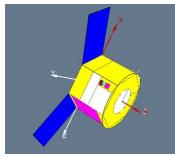
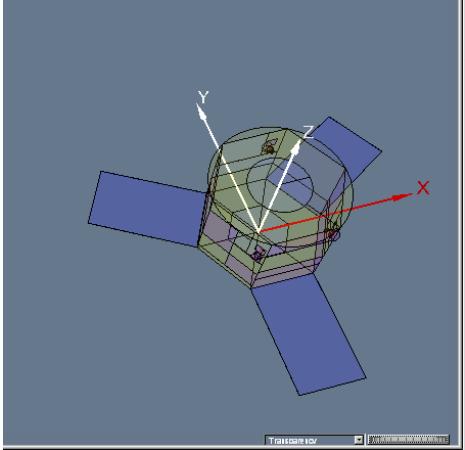
By editing it, you can modify and delete it just like you do with shapes and objects.

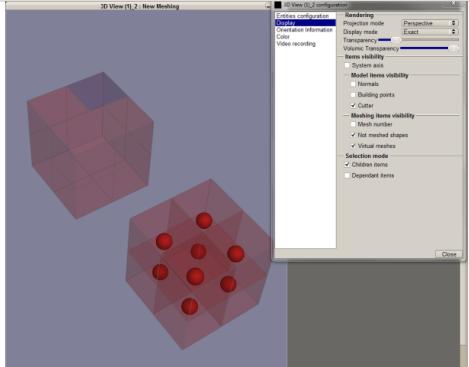
Configuration window

Display facilities



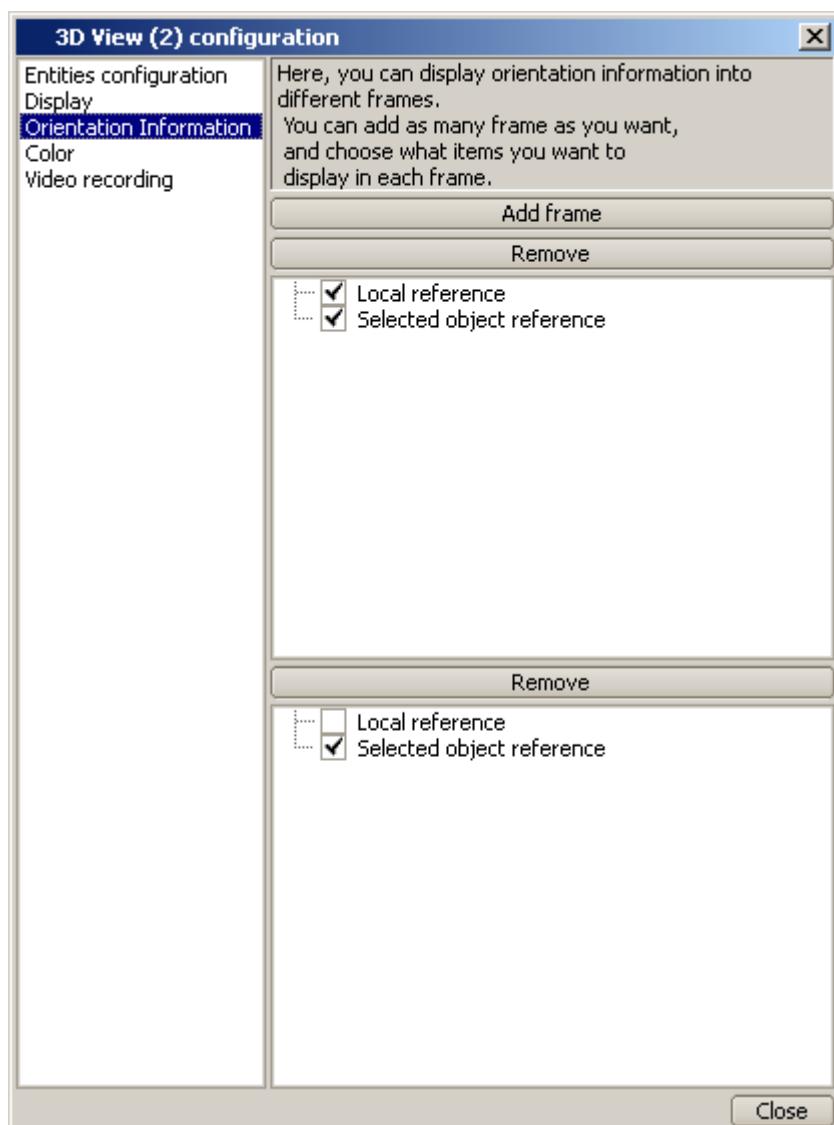
Option	Use		
Projection mode	Allows you to select one of two display modes: Perspective or Parallel		
Display mode	Allows you to select one of three modes:		

Option	Use		
	Solid: 	Wireframe: 	Exact: 
Transparency	<p>Applies a level of transparency from transparent to opaque</p>  <p>Transparency may also be applied interactively. The transparency option applies to all the items appearing in the 3D view, except to volumic meshes.</p>		
Volumic transparency	Applies a level of transparency to volumic meshes only, from transparent to opaque.		

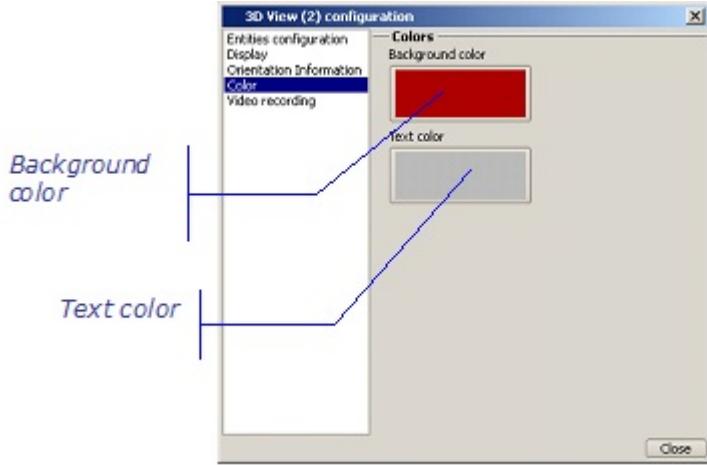
Option	Use		
	 <p data-bbox="493 676 838 707">It may be applied interactively.</p>		
System axis visibility	<p><i>Checked by default. The default values can be changed with the settings (see General User Interface)</i></p> <p>Check/uncheck the box to display the system axis.</p> <p>The orientation frames are always displayed. (for more information, refer to Axis system display)</p>		
Model items visibility	<p>Allows to select the model items visibility</p> <ul style="list-style-type: none"> • Normals (<i>start and end of normal vector can be selected so that it can be used to create interactive shapes</i>) • Building points • Cutter (<i>checked by default</i>) <p><i>The default values can be changed with the settings (see General User Interface)</i></p>		
Meshing items visibility	<p>Allows to select the meshing items visibility for a meshing model</p> <ul style="list-style-type: none"> • Mesh number Displays all mesh number for all meshes in the viewport • Selected meshes numbers Displays only mesh number of selected meshes • Not meshed shapes (<i>checked by default</i>) Allows to display the original shapes, when the shapes are not meshed • Virtual meshes (<i>checked by default</i>) Allows to display the virtual meshes (bounding box and computation point) <p><i>The default values can be changed with the settings (see General User Interface)</i></p>		

Option	Use		
Selection mode	<p>Allows to select the selection mode*</p> <p>Children items In the 3D view, allows to display all the children items of the selected item in the selection mode color</p> <ul style="list-style-type: none"> • Dependant items In the 3D view, allows to display items depending on the selected item in the selection mode color Example: the bounding box referenced shapes are displayed in the selection mode color <p><i>The default values can be changed with the settings (see General User Interface)</i></p>		

Orientation Information: used to create different frames to display orientation information



Color

Option	Use
Color	Allows to select the background or text colors by clicking the background color or the text color 

Video recording

The video recording is relevant for model meshing only when the time concept is associated with the 3D view. Refer to chapter "General User Interface", § Recording a video.

Copy to the clipboard

This action allows to copy to the clipboard the displayed 3D view in which the model is loaded.

Working with material files

General

When working with **Material** tab in the **Modeler** tabs, you can open an existing material database or create a new one.

When a material database is opened, you can:

- modify any material,
- create a new material,
- delete an existing material.

Creating a new material database

To create a new material database :

- In the **File** menu of the **Material** tab, select the **New** option.

Result: In the browser window, a new material database is opened with the root object named "Material database". You can then build the material database and save it.

Opening an existing material database

You can open a material database which has previously been saved within a ".sysmtr" file.
To edit an existing database :

- In the **File** menu of the **Material** tab, select the **Open** option and select the file containing the database.

Result: In the browser window, the database is added to the list of databases and the list of materials is displayed.
The previously edited databases can be accessed from the list of databases.
You can then modify the material database and save your modifications.

Modifying a material database

You can modify a database by:

- creating new materials.
- editing existing materials
- deleting existing materials

Creating materials

There are two different ways to create materials in the current database:

- Within the browser window using the **Create material** option from the right click menu.

Result: The material is created. In the browser window, it is placed at the end of the list if the root was selected or behind the selected material otherwise

- Using the **Create material** option from the **Modeler** menu.

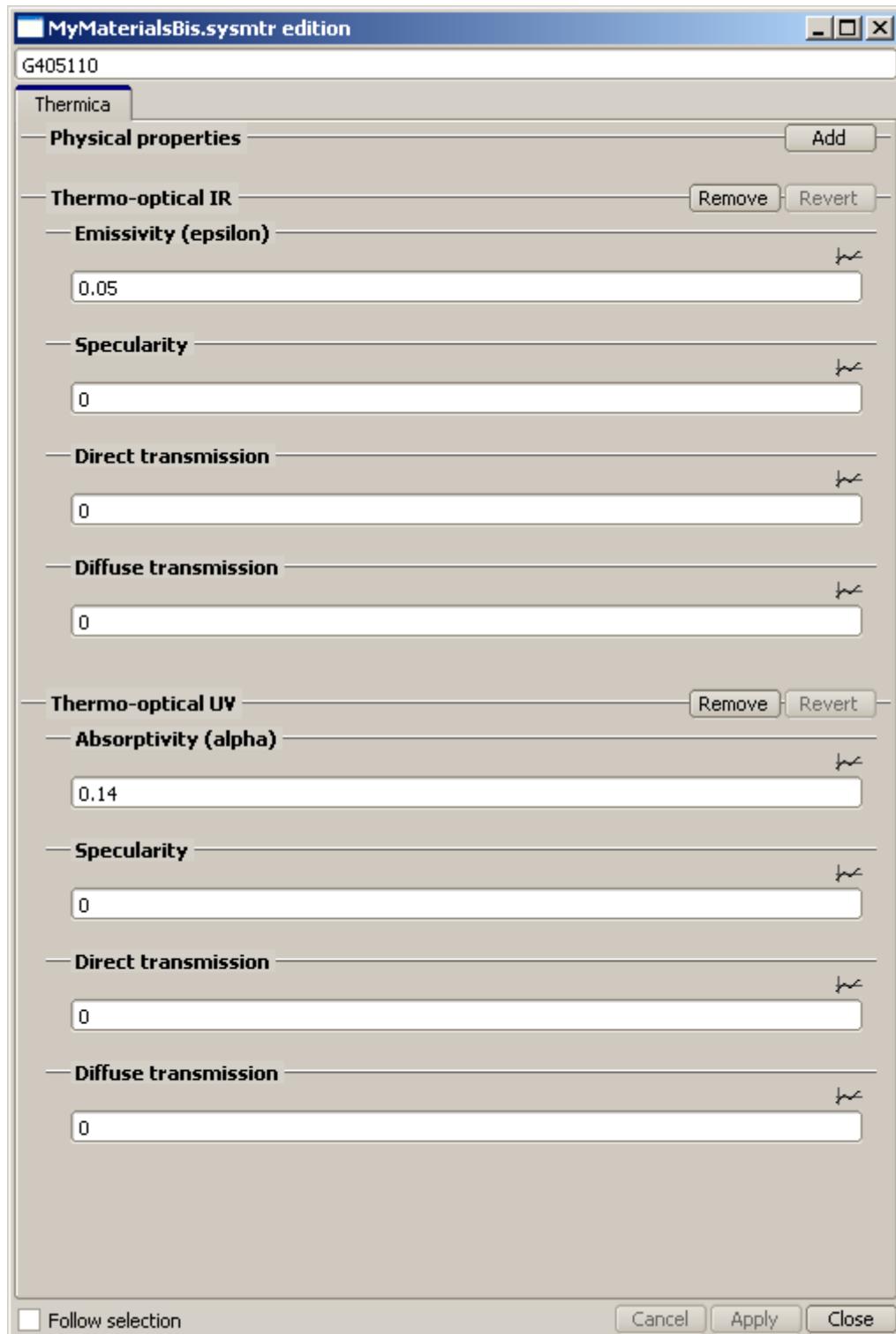
Result: The material is created. In the browser window, it is placed at the end of the list
(For more information, refer to "Working with the browser window" in the "General User Interface" chapter).

Editing materials

There are three different ways to edit materials in the current database:

- Within the browser window using the **Edit** option from the right click menu.
- Use the **Edit Selection** option of the **Edit** menu
- Double click on a material in the browser window

Result: a edition window is opened for the selected material and properties can be added according to the application specifications.



Deleting materials

There are two different ways to delete materials in the current database:

- Within the browser window using the **Delete** or **Cut** options from the right click menu.
- Use the **Delete** or **Cut** option of the **Edit** menu

Result: The material is removed from the list in the browser window.

Working with geometrical models

General

When working with a model in the **Modeler** tab, you can open an existing model or create a new one.
When a geometrical model is opened, you can:

- load it in any 3D view,
- modify it,
- create a meshing for the model,
- create cutters of an object/shape in the current model,

Creating a new model

To create a new model:

- In the **File** menu of the **Modeler** tab, select the **New** option.

Result: In the viewer area, a new 3D view is opened containing an axis system.
You can then build the geometrical model and save it.

Opening an existing model

You can open a geometrical model which has previously been saved within a ".sysmdl" file.

To edit an existing model:

- In the **File** menu of the **Modeler** tab, select the **Open** option and select the file containing the model.

Result: In the viewer area, a new 3D view is opened with the geometrical model displayed in it.
In the browser window, the hierarchical organization of this model is displayed. The previously edited models can be accessed from the list of models.
You can then modify the geometrical model and save your modifications.

Loading a model in a 3D view

In 3D views, you can load geometrical models which have been opened.



Note: Several models may be loaded in the same 3D view.

To load a model in a 3D view:

- In the 3D view, in the right click menu:
1. Select **Configuration window**.



1. Select the model to load.
 2. Click **Load**.
- Result: The selected model is then displayed in the 3D view.

Modifying a model

You can modify a model by:

- creating new objects and shapes (refer to the "Building a geometrical model" topic),
- editing existing objects and shapes.

Building a geometrical model

How to build a model

In order to build a model, you have to create and organize objects and shapes.

Creating objects

You can only create objects within the browser window using the **Create object** option from the right click menu. (For more information, refer to "Working with the browser window" in the "General User Interface" chapter).

Creating shapes

There are two different ways to create shapes:

- *Statically*, using the browser window or the **Modeler** menu.
- *Interactively*, using options from the **Modeler** menu, or the shape icons in the icon bar.

(Details are provided in the section below).

Creating cutters

There are two different ways to create cutters of the object/shape in the current model:

- Within the browser window using the **Create Cutter** option from the right click menu.
- Using the **Create Cutter** option from the **Modeler** menu.

Refer to "Creating cutters" below.

Adding shapes statically

Shapes are created in the current model.

A new shape is created with default values in a default position. The shape attributes (name, coordinates and transformation) can then be modified by editing the shape.

How to "statically" add a shape with default coordinates

To add a shape from the browser window:

1. In the browser window, right click the object/shape you wish to add a new shape to.
2. In the right click menu, select **Create shape**, the position of the new shape (after, before or in) and then the shape type.

Result: The new shape is created. In the browser window, it is placed in the selected position.

To add a shape from the Modeler menu:

1. Select the object/shape you wish to add a new shape to.
2. In the **Modeler** menu, select the **Create shape** option and then the shape type.

Result: The new shape is created. In the browser window, its position depends on the initially selected element:

- if it was created from a selected shape, the new shape is positioned just after the selected one, in the same container;
- if it was created from a selected object, the new shape is positioned as the last one in the selected object.

Adding shapes interactively

Shapes are created interactively in the current model.

Creating shapes interactively consists in directly selecting points to define shapes.



Note: All types of shapes cannot be created interactively but only some of them.

How to add a shape interactively

To create a shape interactively using the Modeler menu:

1. In the **Modeler** menu, select the **Interactive create shape** option, then select the shape to be added.
2. Follow the instructions in the log bar to create each point of the shape by selecting it in the 3D view (Alt + left click).
3. When the shape is added, Systema will try to create a new shape automatically. You can stop the shape creation at any time with the **Abort Interactive creation** command or its shortcut "Esc".



Note: You can cancel point selection as long as the shape has not been completely terminated; then you can modify a shape using the edition window.

Result in the 3D views

When the shape has been created, it is displayed in each 3D view in which the model is loaded.

Result in the hierarchical model organization

When the shape has been created, it is added to the hierarchical model organization, in a position relative to the element that was selected before the shape was created:

if...	...then, in the model browser...
no element was selected	the new shape is created as the last child of the model
an object was selected	the new shape is created as the last child of the selected object
a shape was selected	the new shape is created just after the selected shape

Creating cutters

Cutters are created in the current model. The cutters are associated with the selected object/ shape in the current model.

Cutters allow you to cut items and all the associated sub-items.

- The cutter, linked to a shape, cut the shape only.
- The cutter, linked to an object, cut all items associated with the object.

There are two cutter types:

- Cylinder.
- Extrusion (extruded polygon).

A cutter is created with default values in a default position. The cutter attributes (name, coordinates) can then be modified by editing the cutter.

Note that cutters cannot be applied to volumic shapes.

	<i>Note: Several cutters may be created in the same entity.</i>
---	---

How to create a cutter of an object/shape in the model

To create a cutter from the browser window:

1. In the browser window, right click the object/shape you wish to create a cutter.
2. In the right click menu, select **Create cutter**, the position of the cutter (after, before or in) and then the cutter type.

Result: The cutter is created. In the browser window, it is placed in the selected position.

To create a cutter from the Modeler menu:

1. Select the object/shape you wish to create a cutter.
2. In the **Modeler** menu, select the **Create cutter** option and then the cutter type.

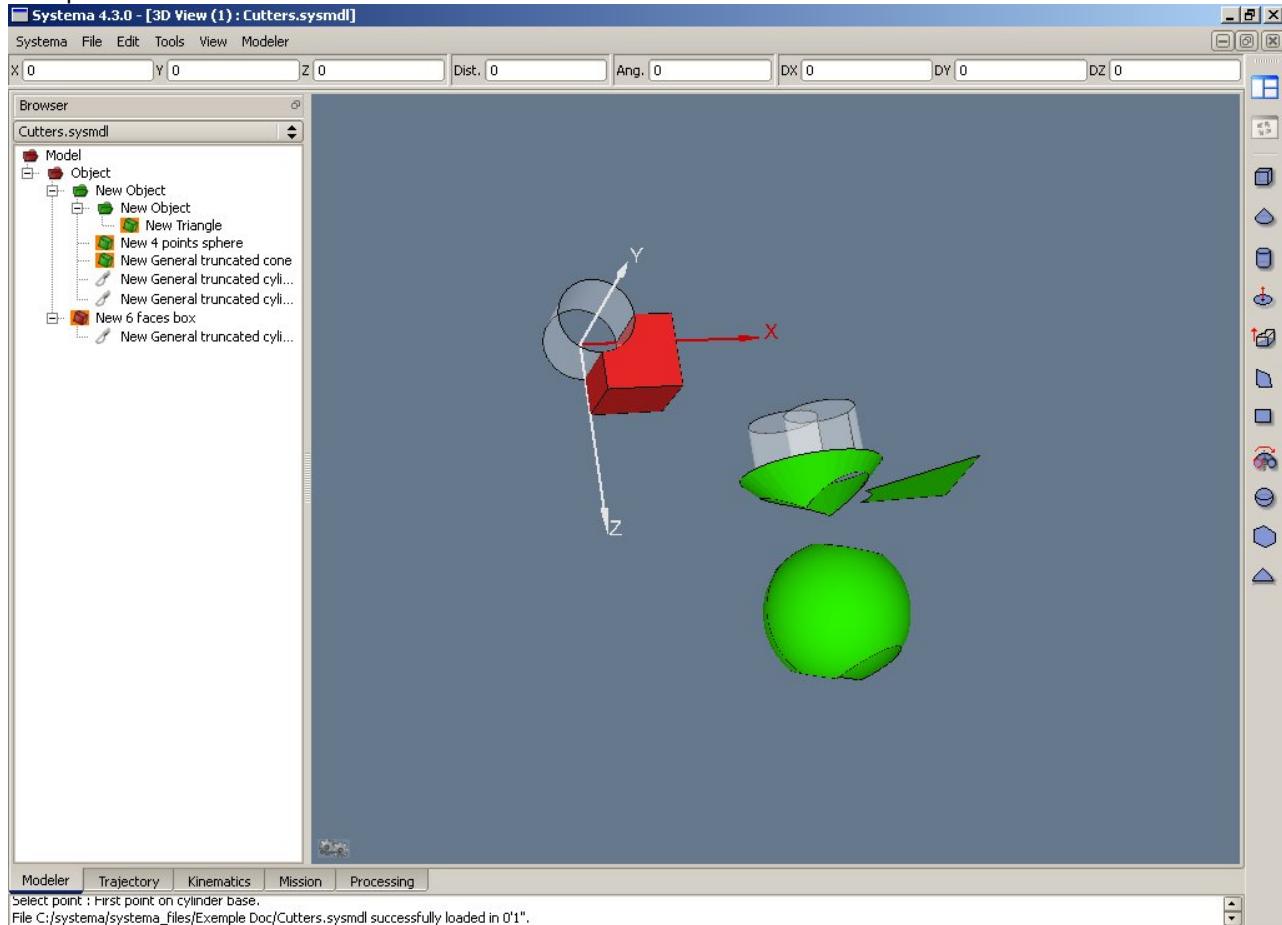
Result: The cutter is created. In the browser window, its position depends on the initially selected element:

- if it was created from a selected object, the cutter is positioned just after the selected one, in the same container.
- if it was created from a selected shape, the cutter is positioned as the last one in the selected object.

Result in the 3D views

When the cutter has been created, it is displayed in each 3D view in which the model is loaded. The shapes subjected to cutting operation are displayed.

Example:



Example1: Simplifying a model using help items

Principles

If you wish to work with a CAD model in SYSTEMA, you first have to import it into SYSTEMA.

A model that has been built with CAD tools consists of shapes that cannot be modified in SYSTEMA. If you wish to work on it, you first have to create a SYSTEMA model, based on CAD and consisting of SYSTEMA shapes.

You have to proceed as follows:

- First, open the CAD model,
- Secondly, create a new model,
- Thirdly, reproduce the CAD shapes in the new model.

You can use 3D display options such as help items, transparency, etc, to define SYSTEMA shapes.

Supported file formats for models

With the present version of SYSTEMA, you can import files in different formats.
The following table presents the extension and format of files you can import:

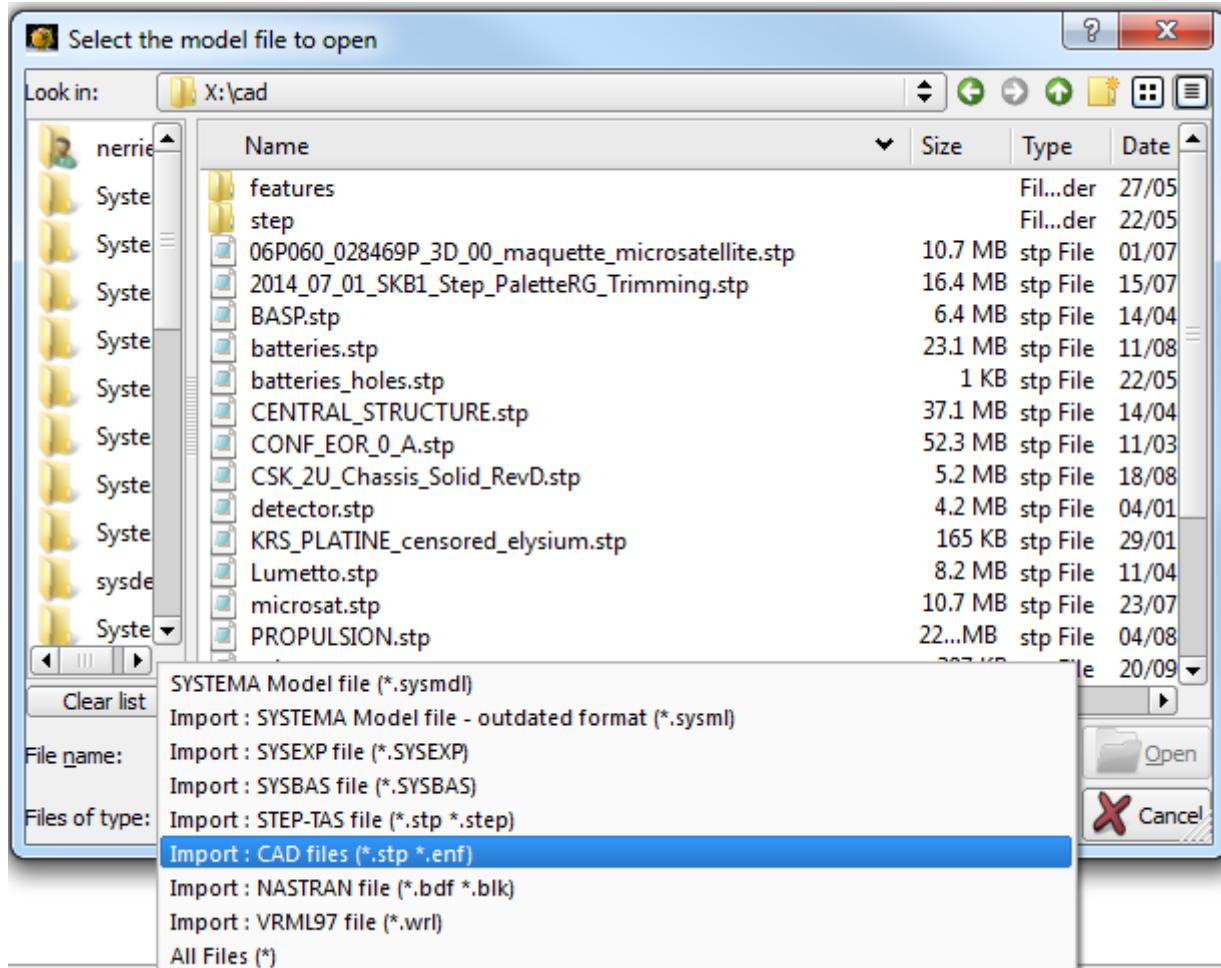
File extension	File format
.blk	NASTRAN file
.wrl	VRML 97 file
.stp or .step	STEP TAS files
.stp or .enf	STEP AP203 and STEP AP214
.sysml	SYSTEMA V4.1 model file
.SYSBAS	SYSBAS file
.SYSEXP	SYSEXP file
	<i>Note: You import a model using the File - Open option.</i>

Opening the CAD model and creating a new model

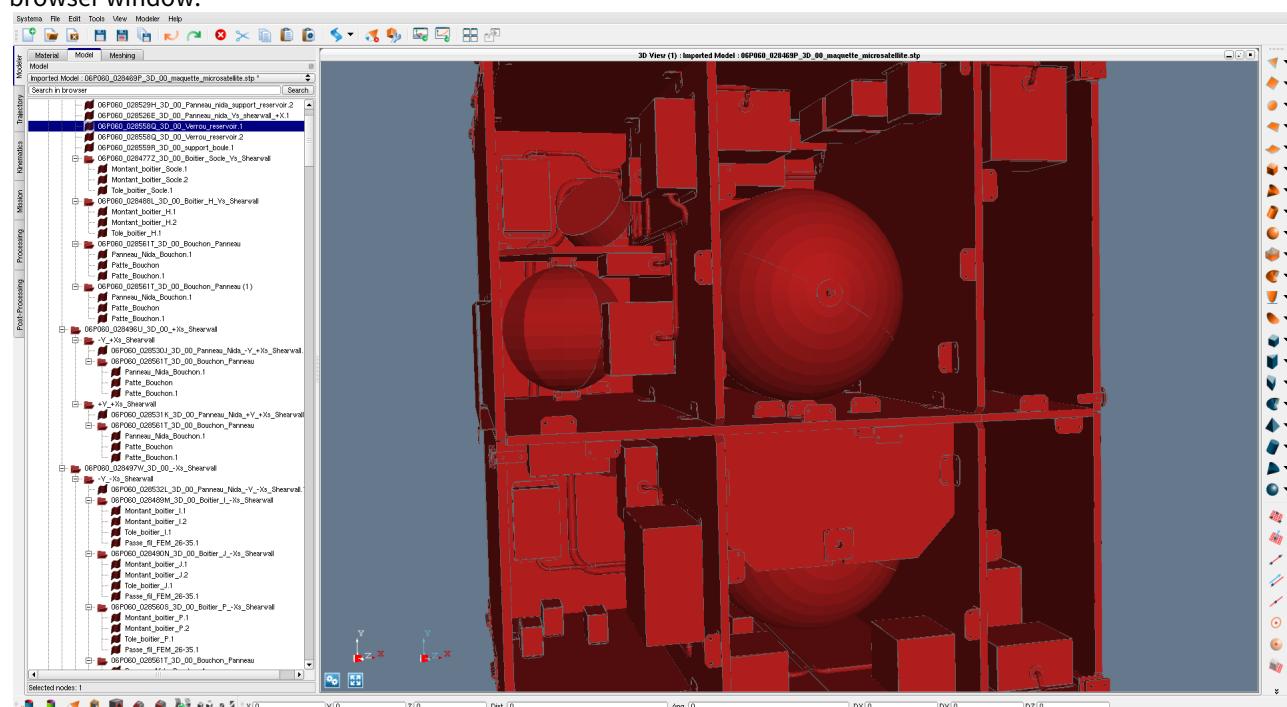
Opening the CAD model

To open the CAD model, you proceed as for opening a SYSTEMA model:

1. In the [File](#) menu, select the [Open](#) option.
2. Select the STEP file (*.stp) corresponding to the CAD model to be opened.
You can restrict the list by selecting the file type, if you know it (eg .stp or .enf):



After selecting the file, the CAD model is displayed in a new 3D view. Its hierarchical organization is displayed in the browser window:



Note that the model's shapes are not SYSTEMA shapes.

Simplification of the CAD model

Once the CAD model is opened, some functionalities are available to make defeaturizing:



Selection of small shapes from a defined threshold, the shape volume (by default 10mm³). After the shapes selection, you can remove them by clicking on "suppr"



Remove holes with a diameter inferior to a defined threshold (by default 10 mm)



Remove chamfers with a defined maximum length (by default 10 mm)



Remove fillets with a defined maximum length (by default 10 mm)

All these actions are available :

- on all the model
- on a selection of shapes
- on all shapes under an object

Convert a CAD shape into Systema Triangles

Once the model is simplified, if you want to increase or decrease the precision of the model, you can apply



tessellation on your model :

Two threshold must be defined :

- chord height tolerance must be included between 0 and 1000mm, by default 5mm
- angle tolerance in degree must be included between 0 and 90°, by default 30°



Note: if you use the tessellation on your CAD model, you can not do simplification after.

The facetisation action will divide each faces of the CAD model into tessellated faces: . Then you can use the

action which transforms each face into Systema triangles: .

All these actions are available :

- on all the model
- on a selection of shapes
- on all shapes under an object

Using help items (lines) to create a copy of a CAD object

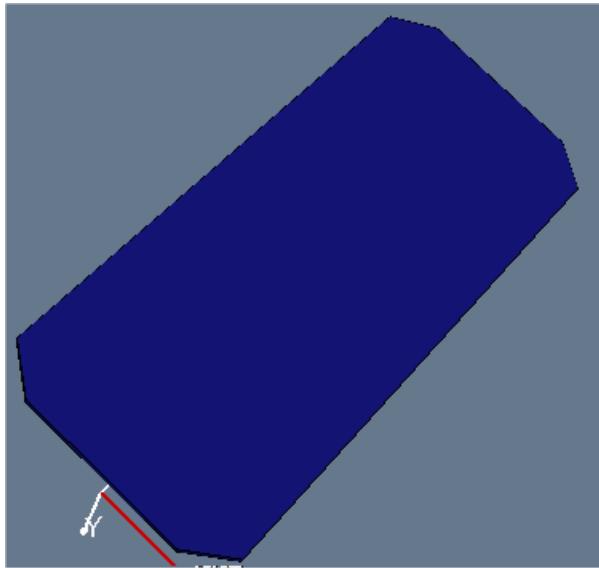
In this example, you will see how you can use "help items" to create a SYSTEMA shape from a CAD shape.

Introduction

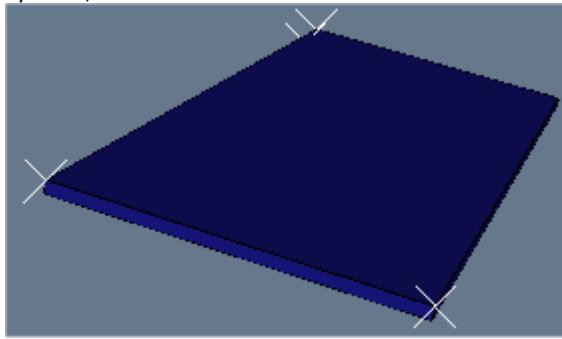
We are going to describe how to isolate a CAD shape and use it to create a SYSTEMA shape that has almost the same appearance.

In our example, the CAD shape to be copied looks like a six-sided box, but with beveled corners. We are going to use a six-sided box to represent it, as shown below:

The CAD shape that we want to reproduce is as follows:

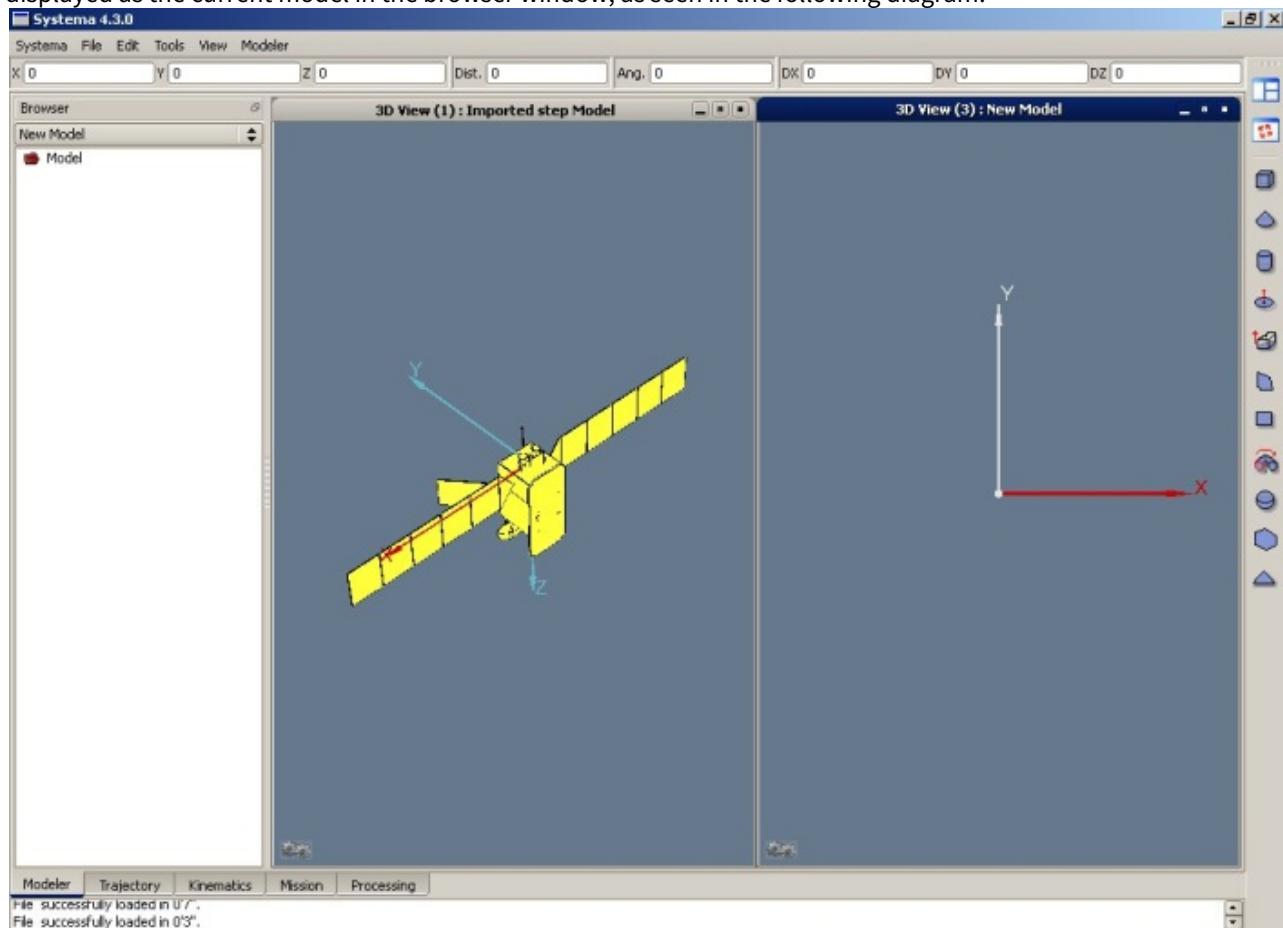


The shape we want to obtain is a six-sided box (with squares):



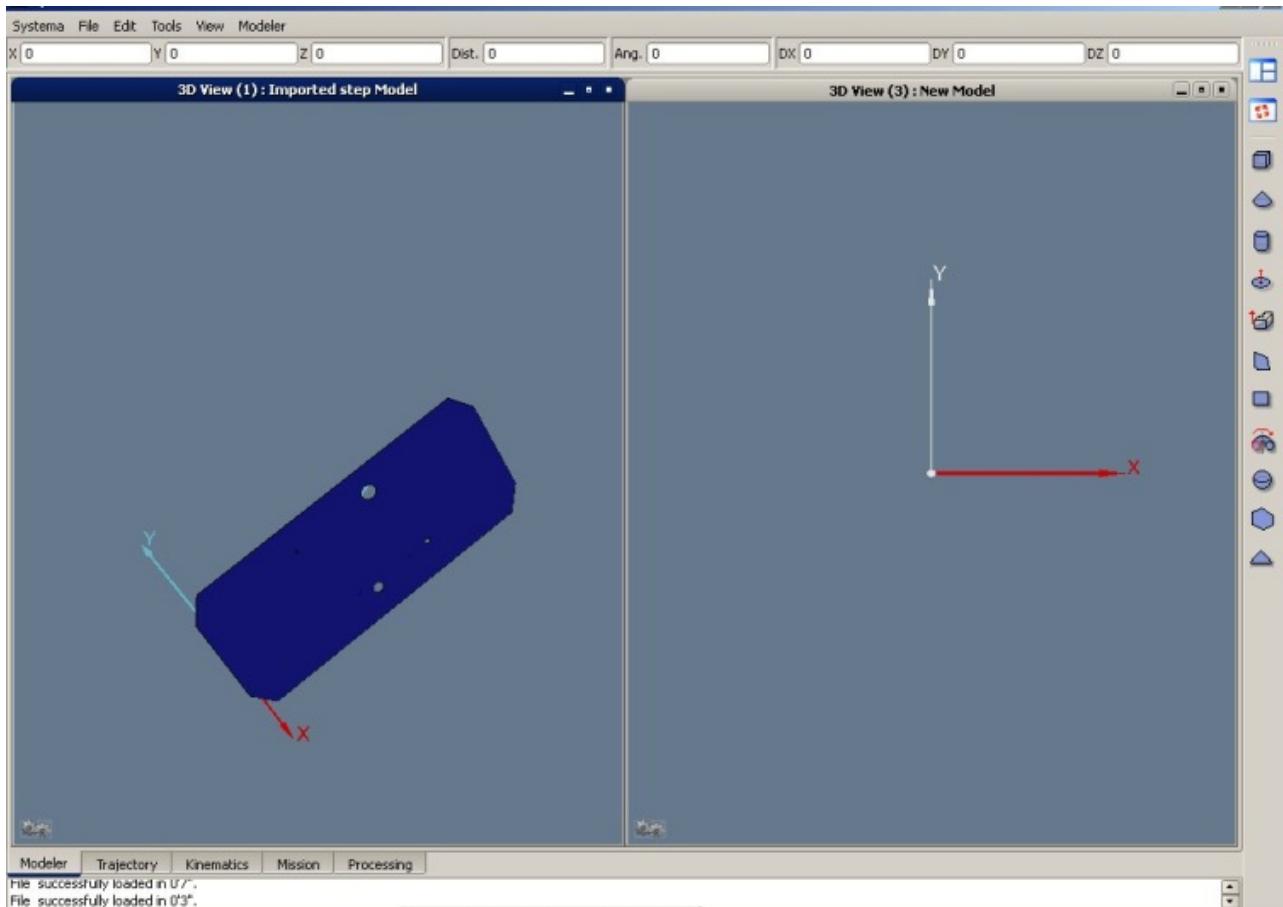
Context

The CAD model has been loaded in a 3D view; the new SYSTEMA model has been loaded in another 3D view and is displayed as the current model in the browser window, as seen in the following diagram:

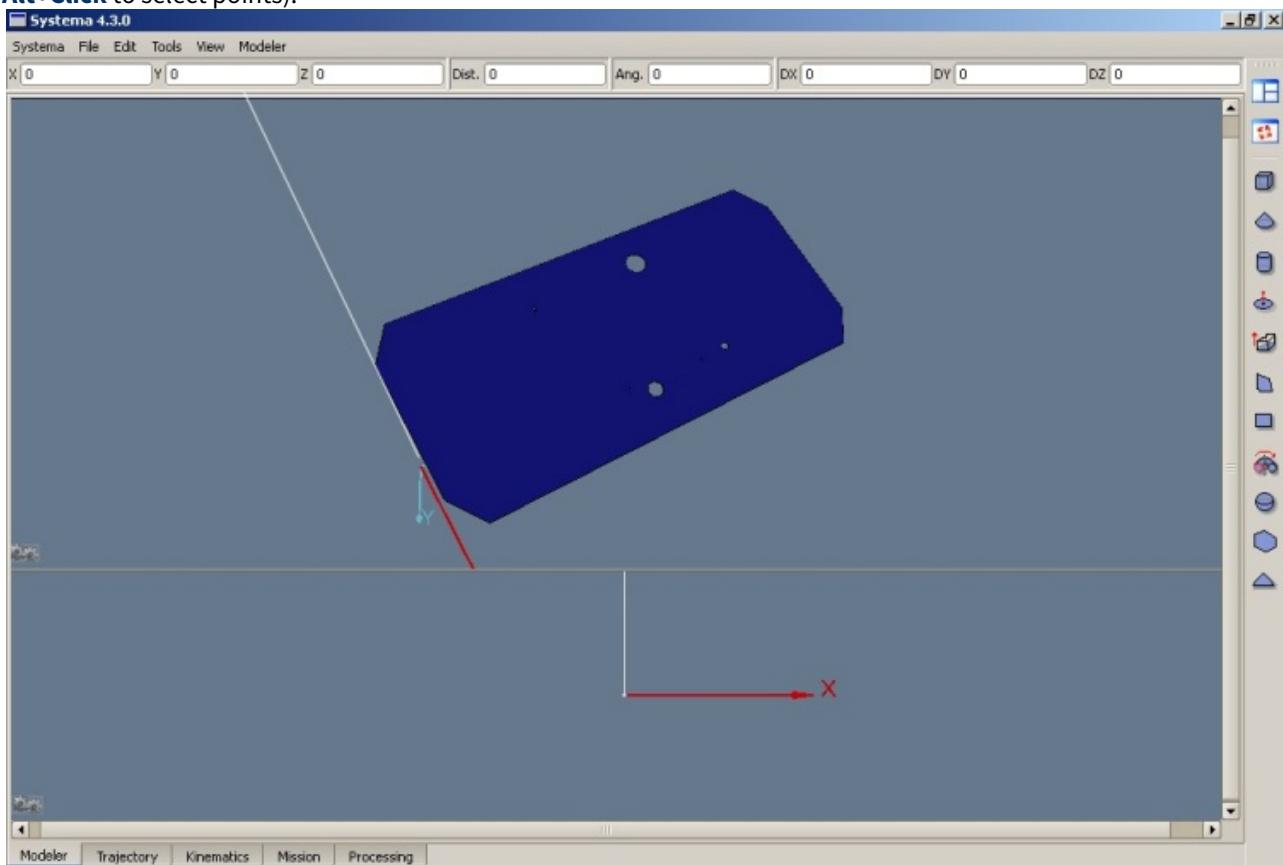


Procedure

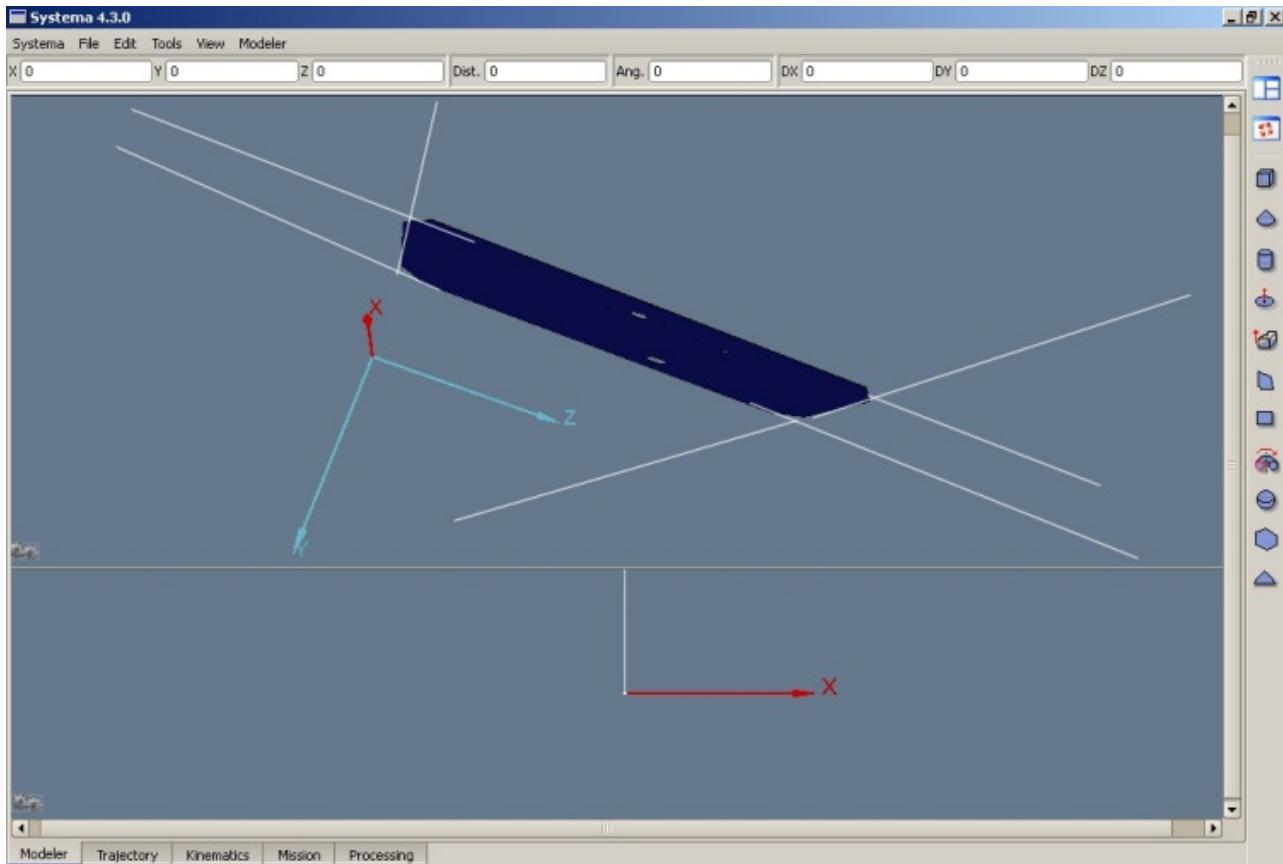
1. In the 3D view containing the CAD model, select the CAD shape and hide all other shapes. You can also fit the 3D view onto the selected shape for a better display:



2. Draw a line on one edge of the selected shape (in the right click menu, select **Help items – Create line ...** and use **Alt+Click** to select points).



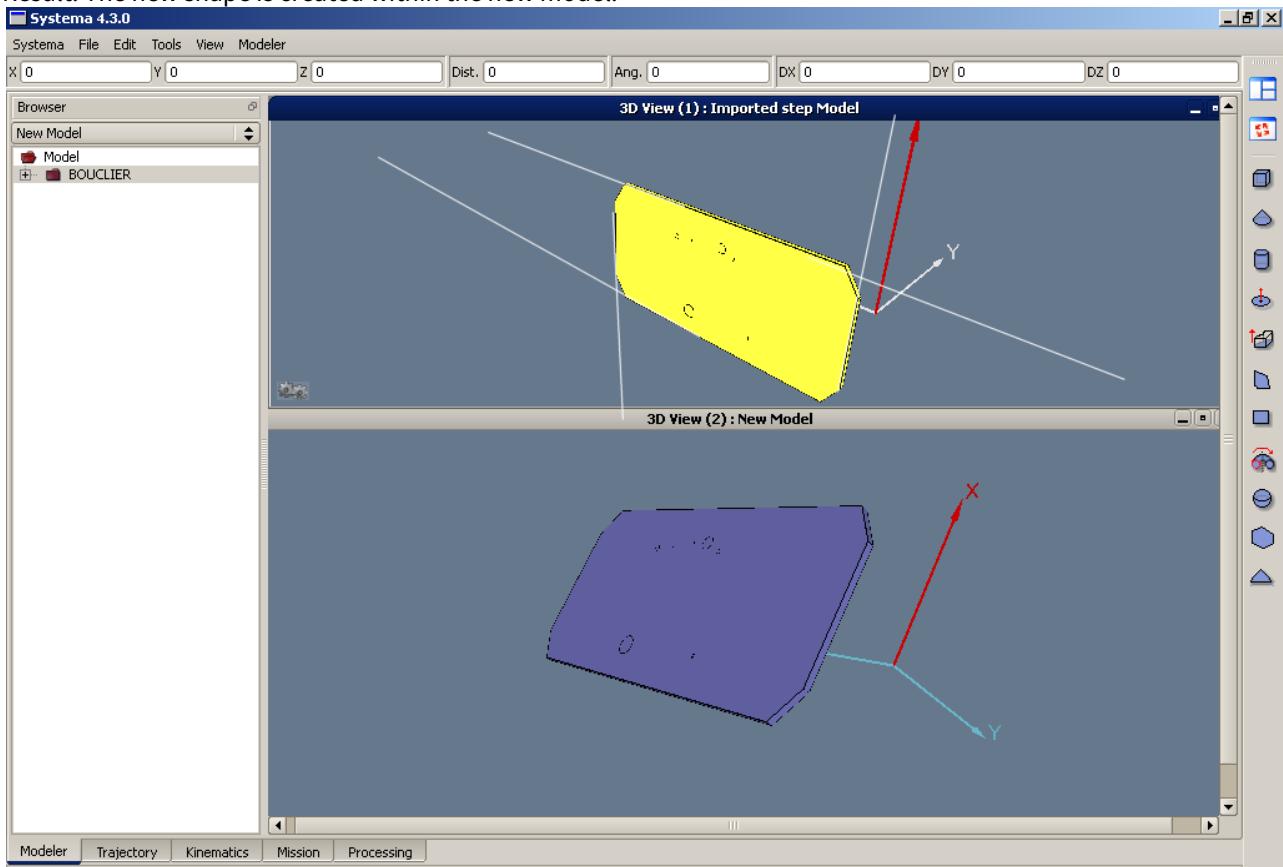
3. Draw three other lines in order to define one side of the volume.



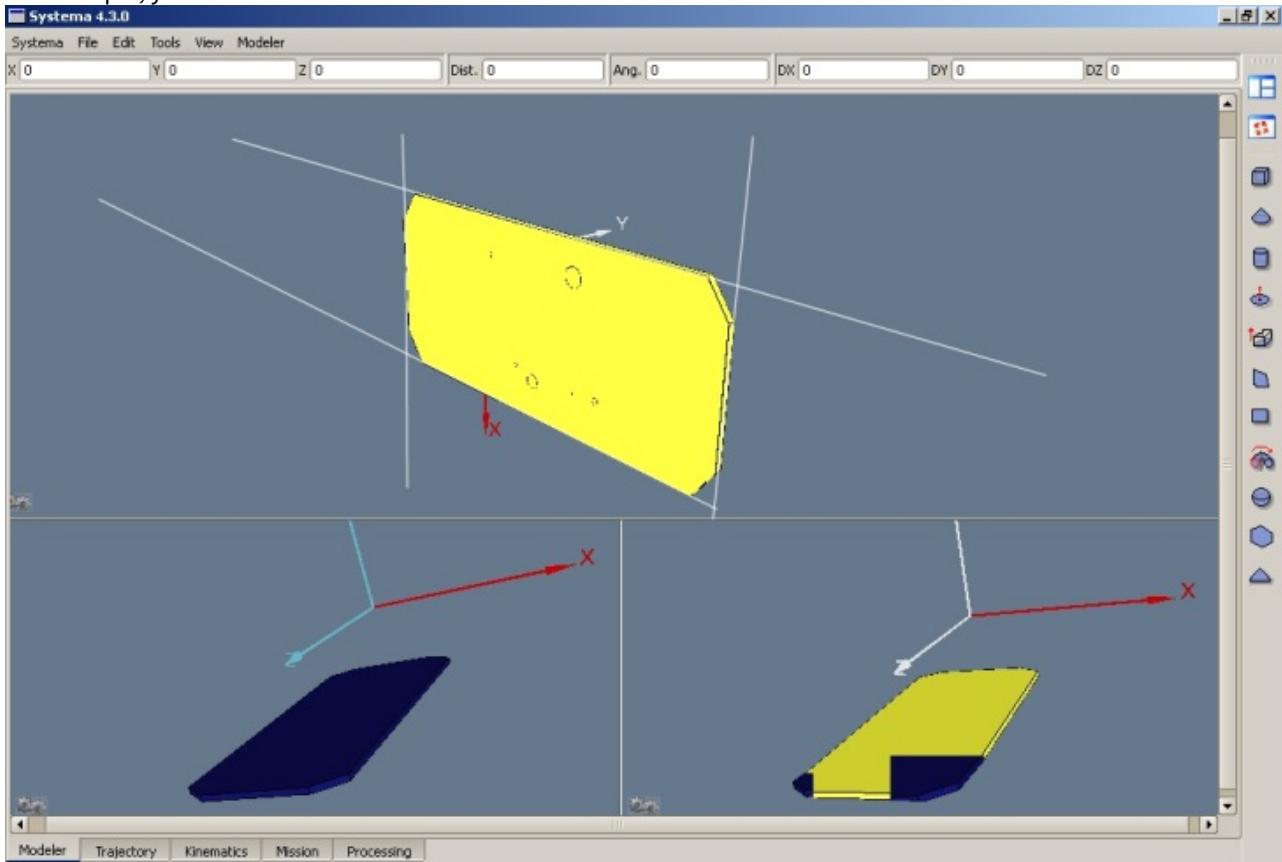
Note: Each intersection of two lines defines a point which can then be selected if required.

4. If necessary, set the new model as the active one (load it in the browser window).
5. Use the **Modeler** menu or the icon bar to create a six-sided box interactively.
6. Use the points at the intersection of the item's lines to define the new shape.

Result: The new shape is created within the new model:



You can superimpose the two models in order to verify that the newly created shape fits to the initially selected one. For example, you can create a new 3D view and load both models in it:



You can then use the same principle to create other SYSTEMA shapes from the CAD model.

Working with meshings

What are meshings for?

A meshing, used to perform an analysis, generates identified nodes:

- From the division of geometrical model shapes into meshes (i.e smaller parts).
- From bounding boxes (if the DOSRAD application is installed).

A meshing allows to prepare the items used for a computation.

There are two types of elements:

- Image of geometrical model divided into sub items.
- Virtual nodes (no physical reality):
 - Space node (3D view)
 - Bounding boxes
 - Computation point

A meshing is defined for a given application.

SYSTEMA allows you to create a meshing associated with a model, save it and modify it, if necessary. Meshing can also be used to view the result of processes applied to the meshes of a meshing.

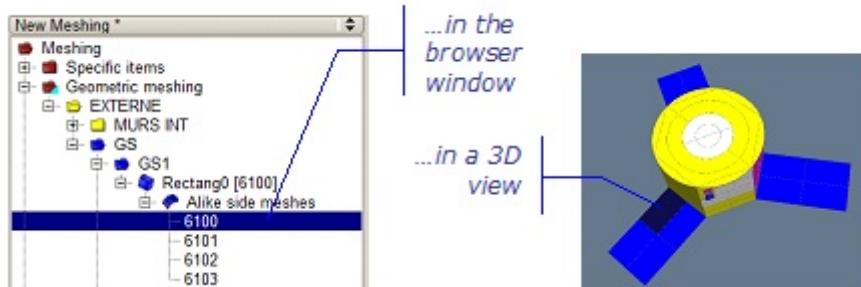
Displaying a meshing

The meshing can be displayed in two ways:

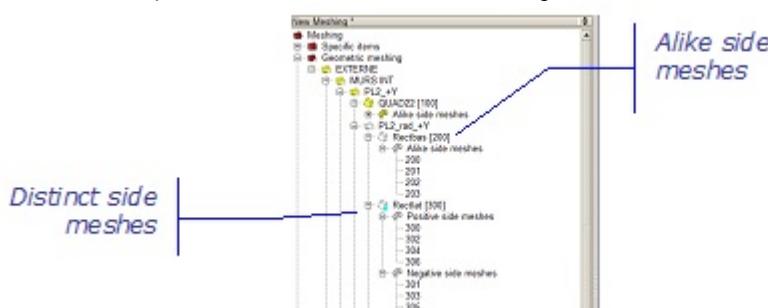
- the browser window shows the structure of the meshing

- the 3D view shows a 3D representation of the model outlining the meshes.

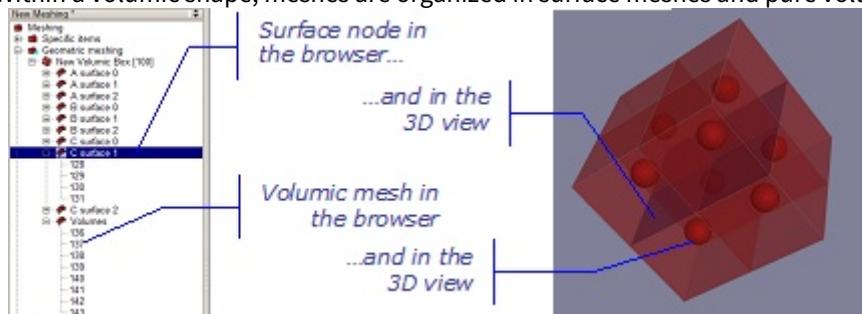
The example below shows the meshing of a rectangle in the browser window and in a 3D view:



Within a shell shape, meshes are organized either in *Alike side meshes* or *Positive side meshes* and *Negative side meshes*, if the option *Enable distinct side numbering* is activated.



Within a volumic shape, meshes are organized in surface meshes and pure volumic meshes.



Defining a meshing when working with the geometrical model shapes

When working with a meshing in the **Modeler** tab, you can open an existing meshing or create a new one. You then have to define the meshing attributes.

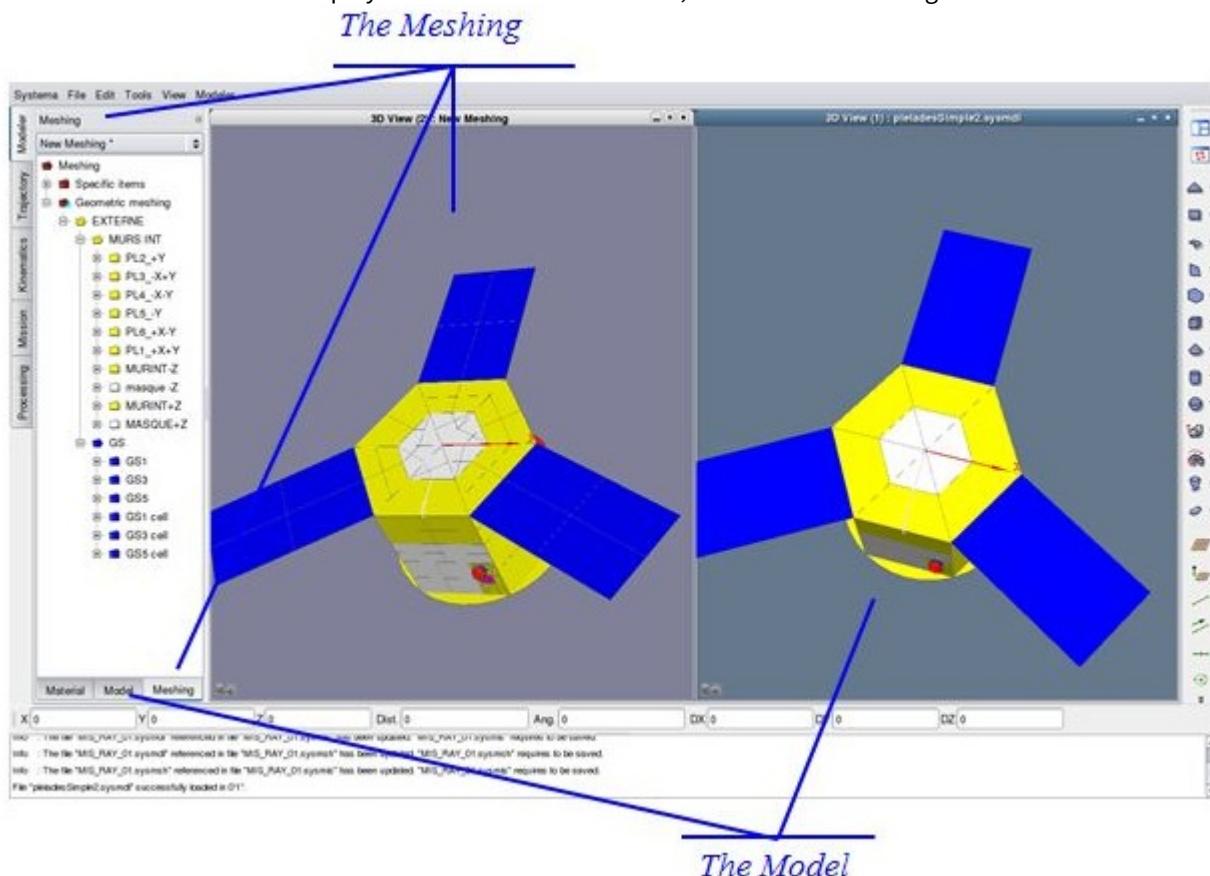
How to create a meshing

- Open the model that you wish to be meshed.
 - Create the new meshing (in the **File** menu select **Meshing-New**).
 - In case you have multiple applications installed, choose the one that will be used for the meshing. Otherwise, this is done automatically.
- In the browser window, the hierarchical structure of the "New meshing" is displayed. It is an exact copy of the model structure.
- A new 3D view "New meshing" is displayed.
- Save the new meshing (in the **File** menu select **Meshing-Save**).

How to open an existing meshing

- Open the required meshing (in the **File** menu, select **Meshing-Open "name.sysmsh"**).

In the browser window, you have access to two sub-tabs that allow you to switch from Model to Meshing. Two new 3D views are also displayed: one contains the model, the other the meshing.



Since Systema 4.7.0, the numbering methods change.

If your meshing file contains shapes and objects defined at the same level, numbering will change compared to the original file. Old numbering can be re-apply to new meshing file by using 'meshingConverter' python script.

Here is an example of how to convert a meshing created with Systema 4.5.3 version to be loaded by Systema 4.8.3 version :

- Edit the meshingConverter453.py script installed by Systema 4.8.3 in python\publicScripts folder and modify the following values :
 - meshingToConvertPath* : set the absolute path of the 4.5.3 meshing file.
 - numberFilePath* : set the absolute path of the numbering file that will be created by this script
- Save the modifications and launch it with Systema4.5.3 by :
 - Selecting **Tools → Launch python script** menu
 - Or directly in batch mode (command line with -script meshingConverter453.py option). When the execution of the script is complete, the file '*numberFilePath*' is created.
- Then launch the meshingConverter.py script installed by Systema 4.8.3 in python\publicScripts folder by :
 - Selecting **Tools → Launch python script** menu. In the displayed dialog, set :
 - inputMeshing* : Absolute path to the 4.5.3 meshing file
 - numberingFile* : same value than previously
 - action* : import
 - outputMeshing* : Absolute path to the new meshing file

- Or by launching Systema 4.8.3 in batch mode with the following parameters -parameters
`inputMeshing ="Absolute path to the 4.5.3 meshing file" numberingFile="same value than previously"`
`action="import" outputMeshing="Absolute path to the new meshing file".`

Once the execution of the script is complete, the new meshing file is generated and can be loaded in Systema4.8.3. The numbering of the new meshing file is the same than the 4.5.3 one. The input meshing file is closed at the end of the script. If you prefer to keep it opened, please comment the following lines in meshingConverter.py :

```
meshing.close()
model.close()
```



Note: Executing a script in batch mode (i.e. in a command line) is the most efficient mode

On Windows, please use \ as separator in file path (for example : 'c:\folder1\meshing.sysmsh') in python scripts

❗ meshingConverter limitation

Since 4.8.2, Custom method for the numbering parameters is no longer supported for objects. During export phase, a warning message is displayed for each of these objects whose numbering method is 'Custom'.

For these objects, the method will be change automatically to 'Generic' but custom numbers list need to be reported manually to all of their child shape.

Defining the meshing attributes

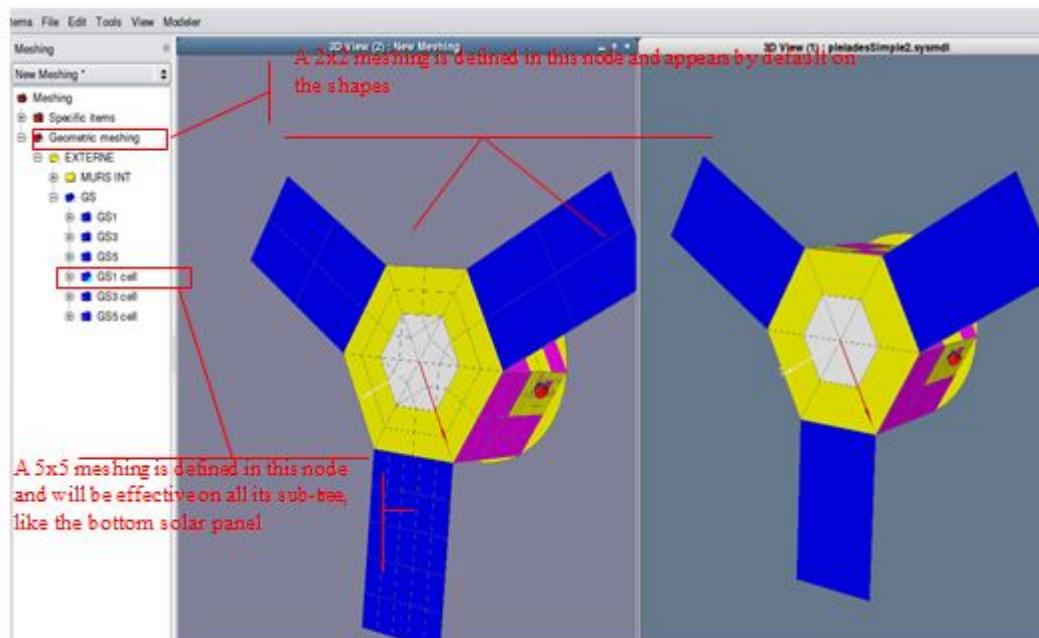
The meshing attributes

The meshing attributes allow you to define how each shape is meshed.

Each node has several attributes: the object name, the meshing and numbering parameters, submodel name, condensation... Default values are defined in the "Geometric meshing" node, but can be redefined for each object.

Principles

The properties of the different objects and shapes will define the meshing of your model. Each definition will take effect in all the sub-tree of the node, except if a contradictory property is defined under it. Using these inheritances properties, you can easily define different meshings depending on the precision you need on the different parts of your model.



Note: The technique of mesh numbering is defined by the application which is associated with the meshing.

How to modify the meshing for an object or a shape

Note: If you modify the meshing attributes for an object, these modifications will apply to every shape within this object.

1. From the browser window, display the edition window for the object/shape you wish to modify.

2. Modify as required the meshing parameters for the selected element:

In the **Meshing parameters** area, you can modify the parameters to be applied to the meshing type (this has been defined at meshing root level).

In the **Label** area, you can specify a label to be used in the results files in order to shorten the meshes path:

With a label "RAD +X" the path "Model/My/path/to/the/rad +x/panel1/objects" becomes "RAD +X/panel1/objects".

The label can contain underscores, spaces, plus and minus characters.

Labels can be displayed in the meshing tree along side the meshes entries. By default they are hidden, to display or hide them : right click anywhere in the meshing tree to access the functionality.

Warning: some actions like renaming, renumbering, remeshing, adding or removing objects or shapes in the tree will take longer if the labels are displayed. The update of labels takes the whole tree into account.

In the **Numbering parameters** area, you can change the numbering rule for the selected object. Several numbering options and methods are available:

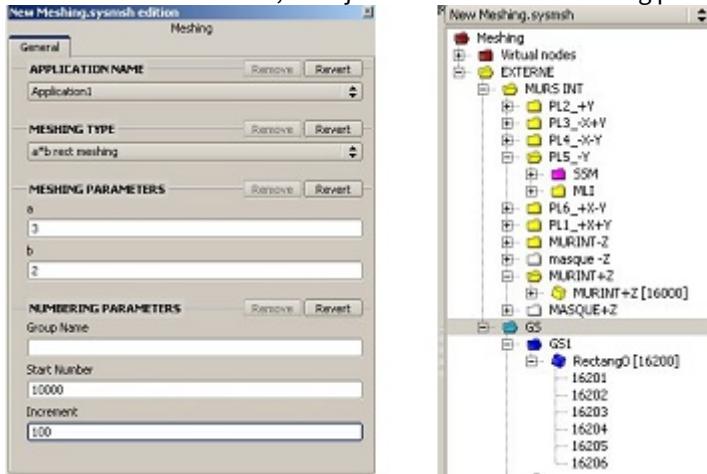
1. "Start number" is the first meshing number attributed
2. "Shape increment" is the increment that will be applied between two shapes
3. "Global Numbering" is an option to not modify the numbering outside of a local renumbering. By default this option is not selected
4. You can select the "Enable distinct side numbering" option: both sides of the shapes will be distinctly meshed
5. Generic/Specific/Custom define the way the meshing numbers will be attributed to the meshes of a shape. The "Custom" method allows you to define a discrete list of numbers. This list shall contain the same number of elements than the number of nodes. Example of custom numbers list for 2*2 meshing : "100 50 25 10"

The **Authorize condensation** option allows you to authorize that two meshes have the same number. Meshes with the same number are considered like one, so this option must be used carefully.

The **Submodel name** allows you to give a name to the current submodel.

3. Click Apply to validate.

In the following example, meshes have been redefined for the GS object, with 3x2 meshes, numbered from 10,000 with an increment of 100, at object level. The new meshing parameters are then applied to the shape "Rectang0":

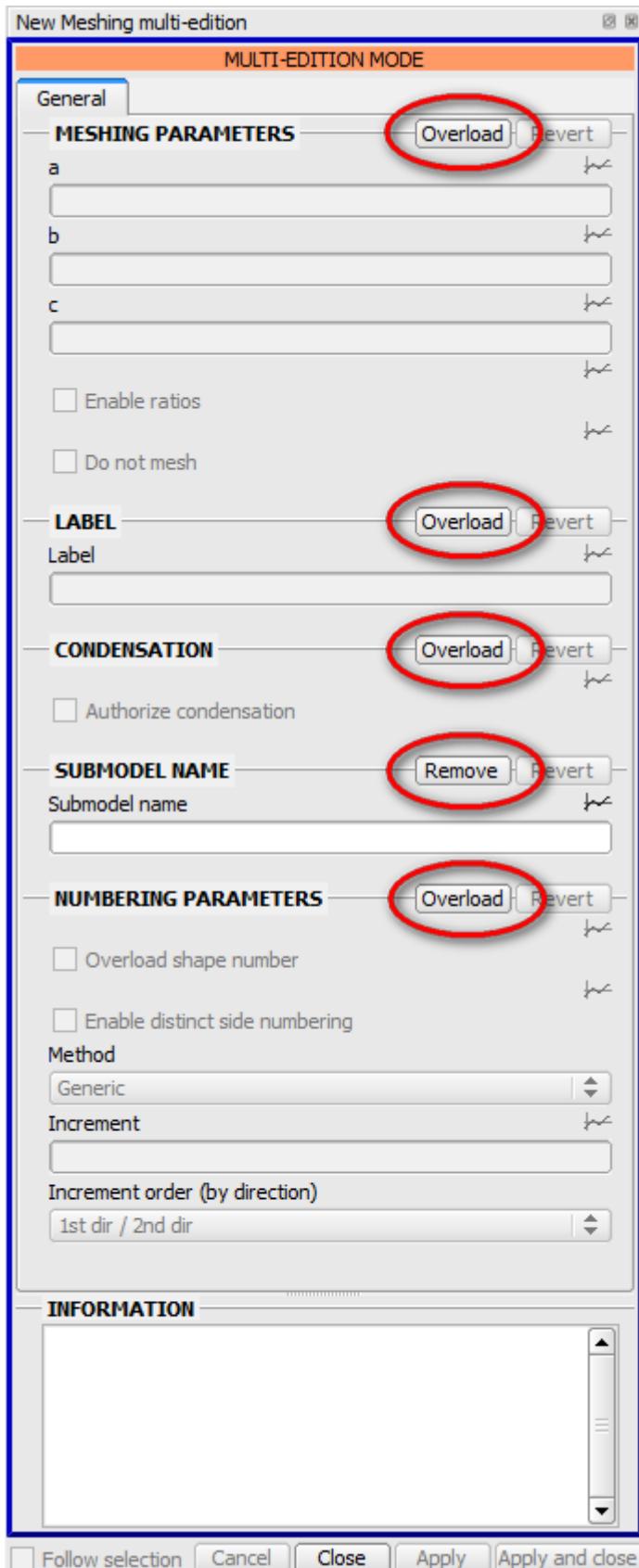


Subsequent shapes in the GS object will be numbered from 16200, then 16400, and so on.

How to modify the meshing for a list of objects or shapes

Using the multi edition option you can modify more than one shape's or object's attribute at the same time :

1. Select components, in the browser window or in one of the viewports by using the CTRL key.
2. Open the multi edition window by selecting the Multi Edition option either in the Edit menu of the main window or in the browser window, or use CTRL+SHIFT+E shortcut.



This multi edition window allows to set, update or remove each attribute of the multi selection.

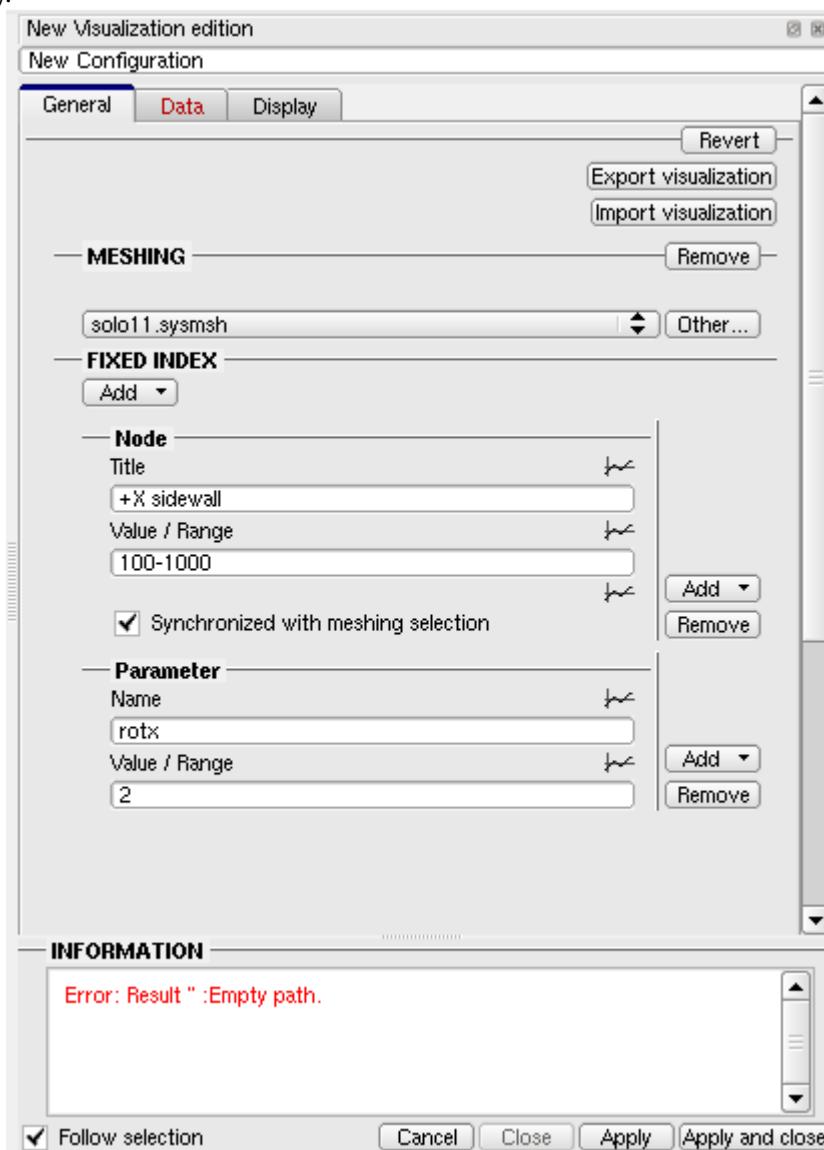
- When the Overload button is visible, it means that for this attribute, no specific value is set in the multi selection. You can then set the attribute values by Overloading them.

- When the Remove button is visible, it means that this attribute is set for at least one of the components in the selection. You can then update attributes values (setting them for components where attribute is not defined) or remove them (no effects for components where attribute is not defined).

Displaying the result of a process in tables and curves

How to display a result in tables and in graphs

- Open a viewport of graph type (In the **Viewport** menu, select **Add - Graph**).
- From the graph viewport, open the configuration window (in the right click menu, select **Configuration window**).

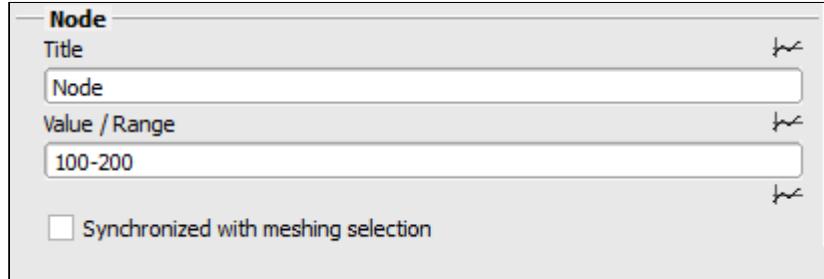


- In the configuration window, you can import or export the configuration in a file with the extension *.sysvisu. You can export this configuration to reuse it or import it in the Post-Processing (Cf [Post-Processing - Visualization box](#))
- In the tab "General", you can choose a meshing file in order to select directly the nodes from the meshing browser.
- The Fixed index part represents the result values you want to define and will not change. You can define node selection, parameters or time.

Definition of Node index:

- one node : 100

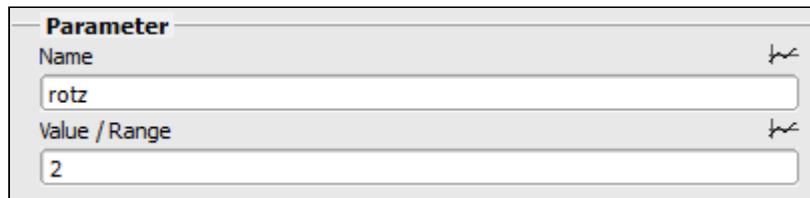
- more than one node : 100,200
- nodes between 2 values : 100-200
- multi-definition : 100,200,300-400
- all nodes : all



- select "Synchronized with meshing selection" if a meshing file has been previously selected in the meshing section and you want to select the nodes directly from the meshing browser.

Definition of a Parameter index :

- enter the name of parameter
- enter the value of parameter



Definition of a Time index :

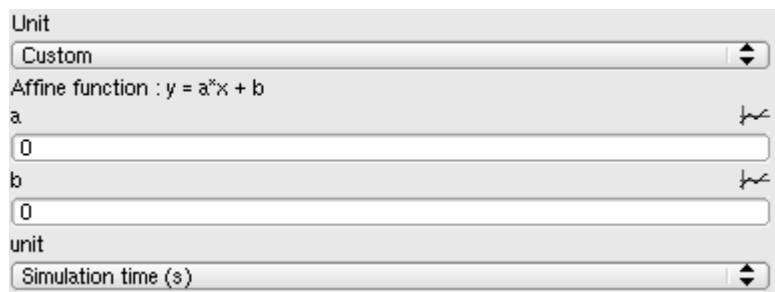


- enter a title
- enter a value or a range (same syntax as the node one). You can also work only on the last orbit. When it is chosen, you have to define the orbit : its period and its unit (second, minute, hour, day)

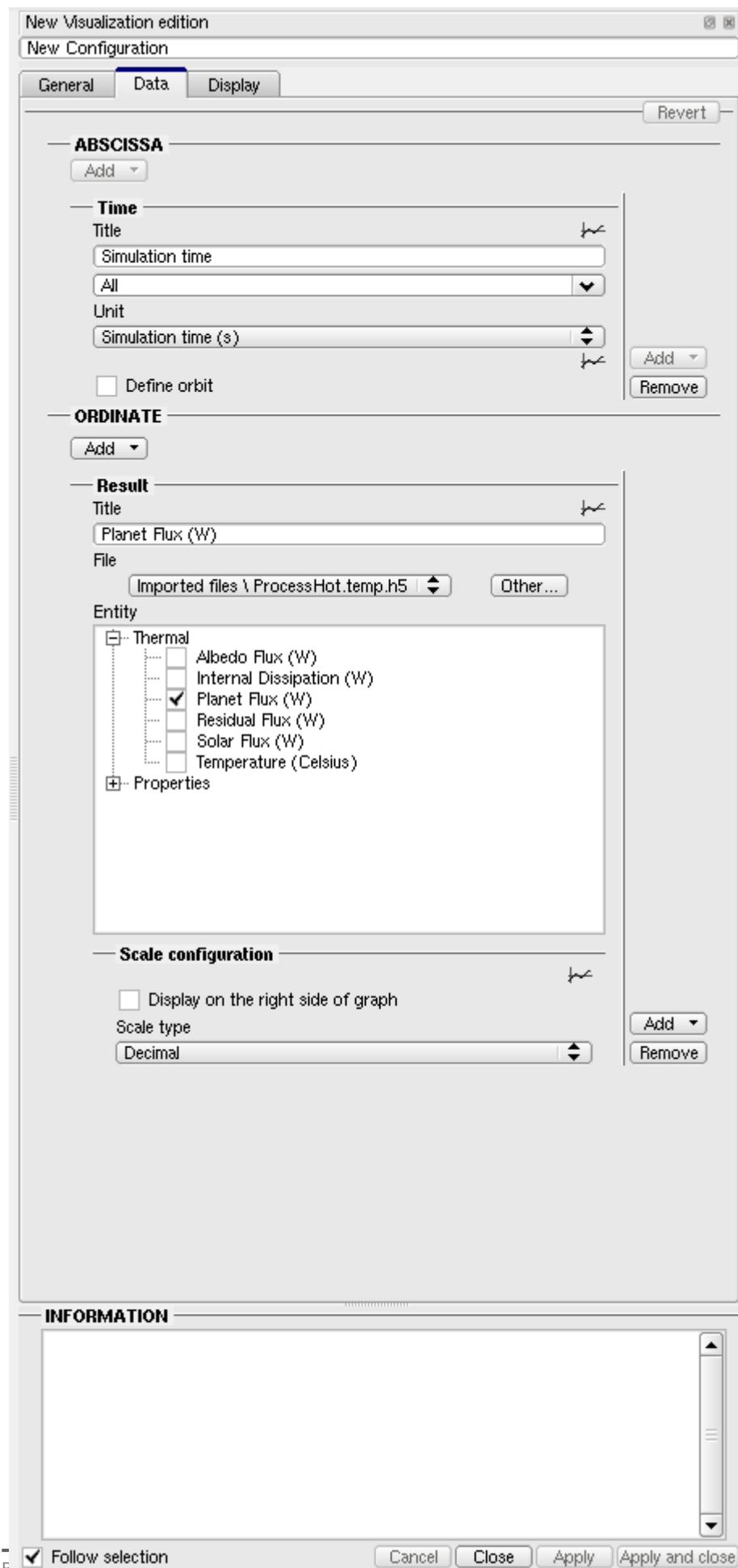


- select the unit you want to display (gamma50, second, minute, hour, day, orbit, or custom). When "orbit" is chosen, you have to define the orbit : its period and its unit (second, minute, hour, day)

You can select the "Custom" unit which will create the time axis based on an affine function:



6. In the "Data" tab, you can define the different axis.



For the abscissa, only time and node values are available. The definition is the same as in the Fixed Index section.

For the ordinate, only results are available. You can select the result(s) you want to display in the graph viewport by choosing a H5 file and an associated entity. Then, you can display the result in the left ordinate (by default) or in the right one. And the scale can be decimal or logarithm.

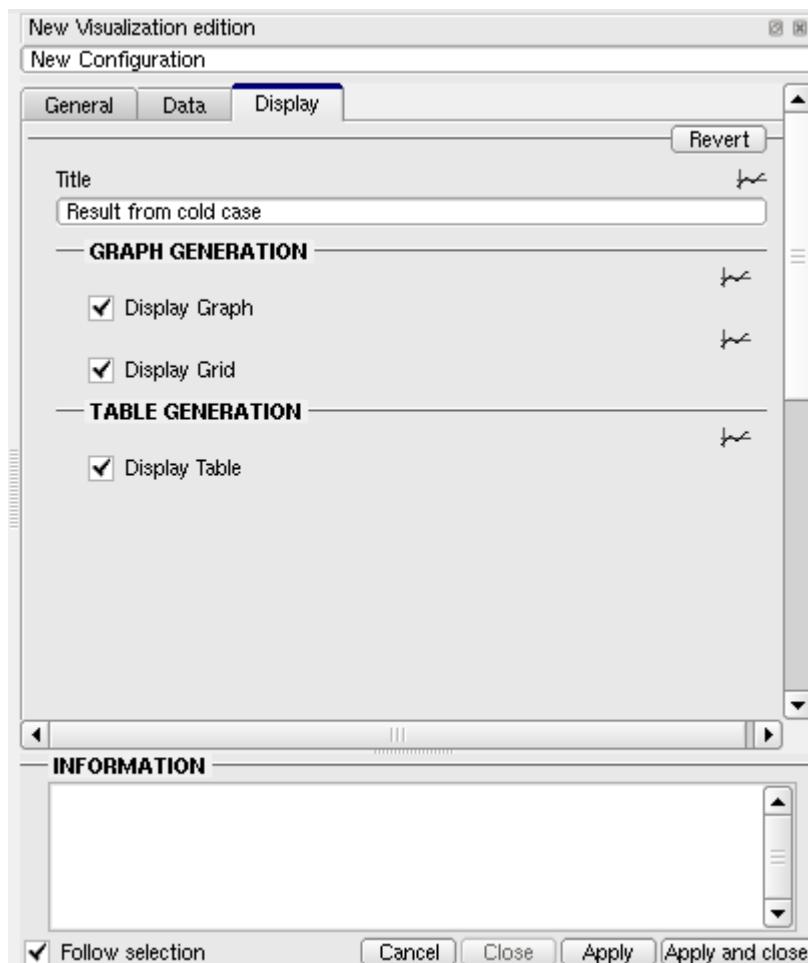
You can display more than one result.

The titles of abscissa and ordinates can be modified.

The scale is the same for both the graph and the table.

7. In the "Display" tab, you can configure the display of the graph View:

- modify the title of graph and table
- display only the graph, only the table or both
- display the grid



8. Then apply and close the configuration window (click **Apply and close**).

9. In the meshing, if you have checked "synchronized with meshing selection", select the mesh(es) you wish to display the results for. You can select the mesh either from the browser window or from the 3D view. The configuration window will be updated.

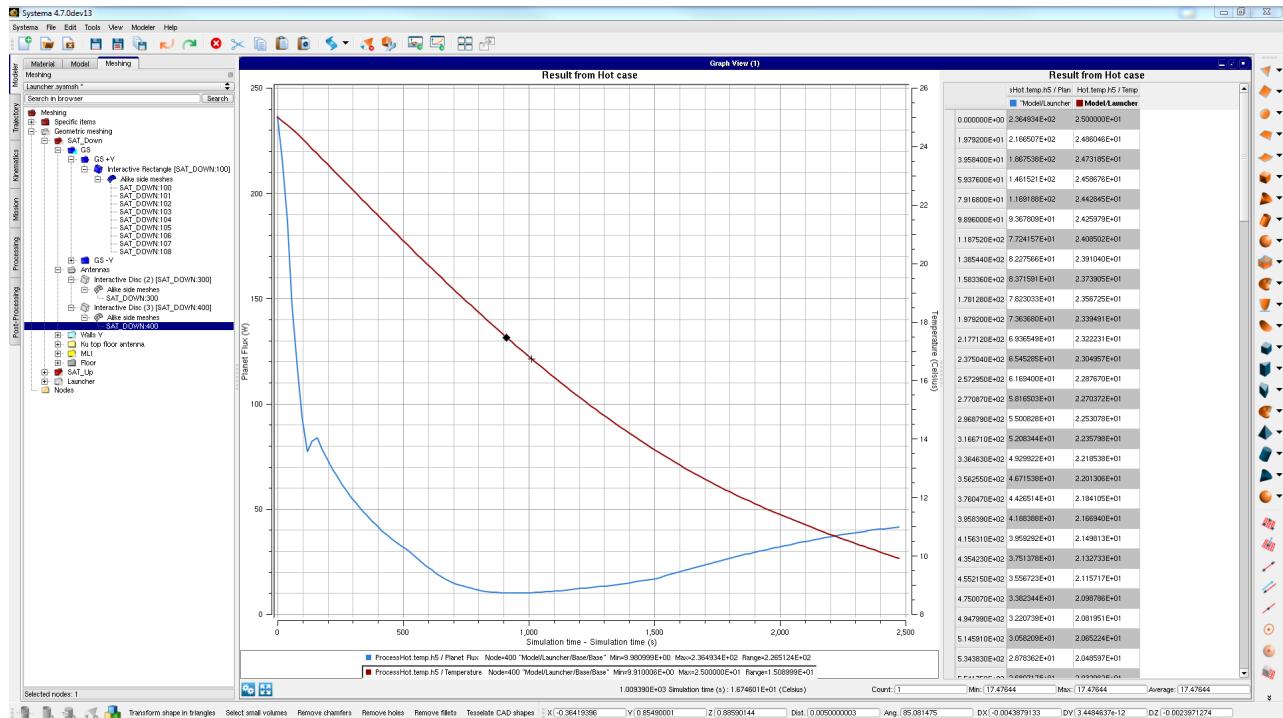
Result: In the graph viewport, a curve panel is displayed showing the result(s) for the selected mesh(es), a table is also displayed showing the points values in a table.

You can select several meshes and results (even if the units are different). A curve is displayed for each one. If there is no value for a mesh at a specific date in the table the cell is empty.

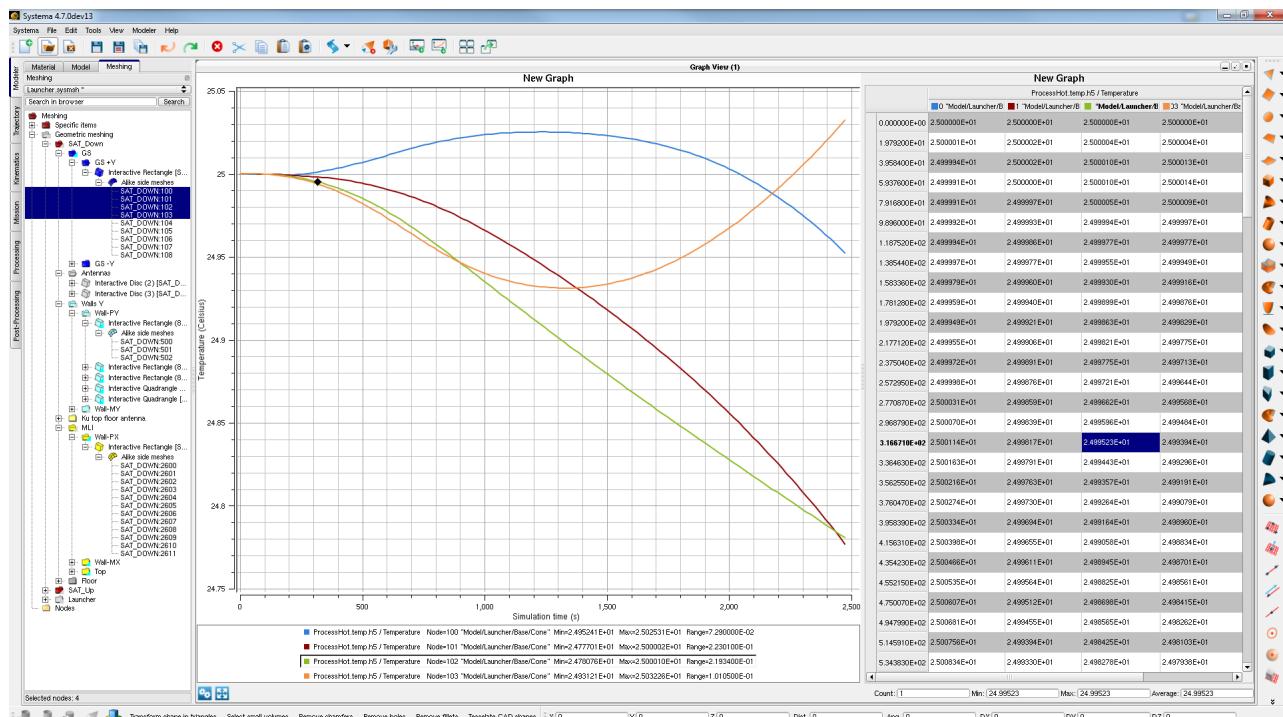
The graph and the table are separated by a bar that can be moved to arrange the size of each part.

Example 1: The following diagram contains curves displaying two results for a mesh.

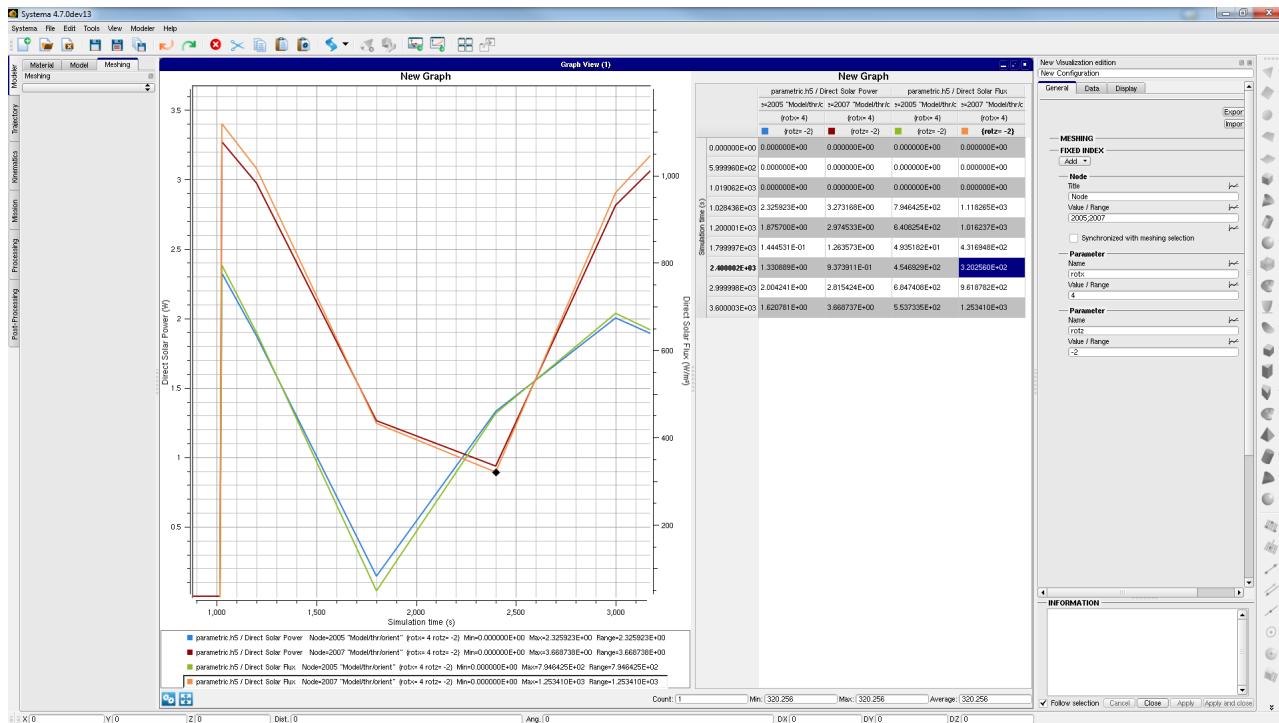
SYSTEMA V4.9 User Guide



Example 2: The following diagram contains curves displaying a result for four meshes.

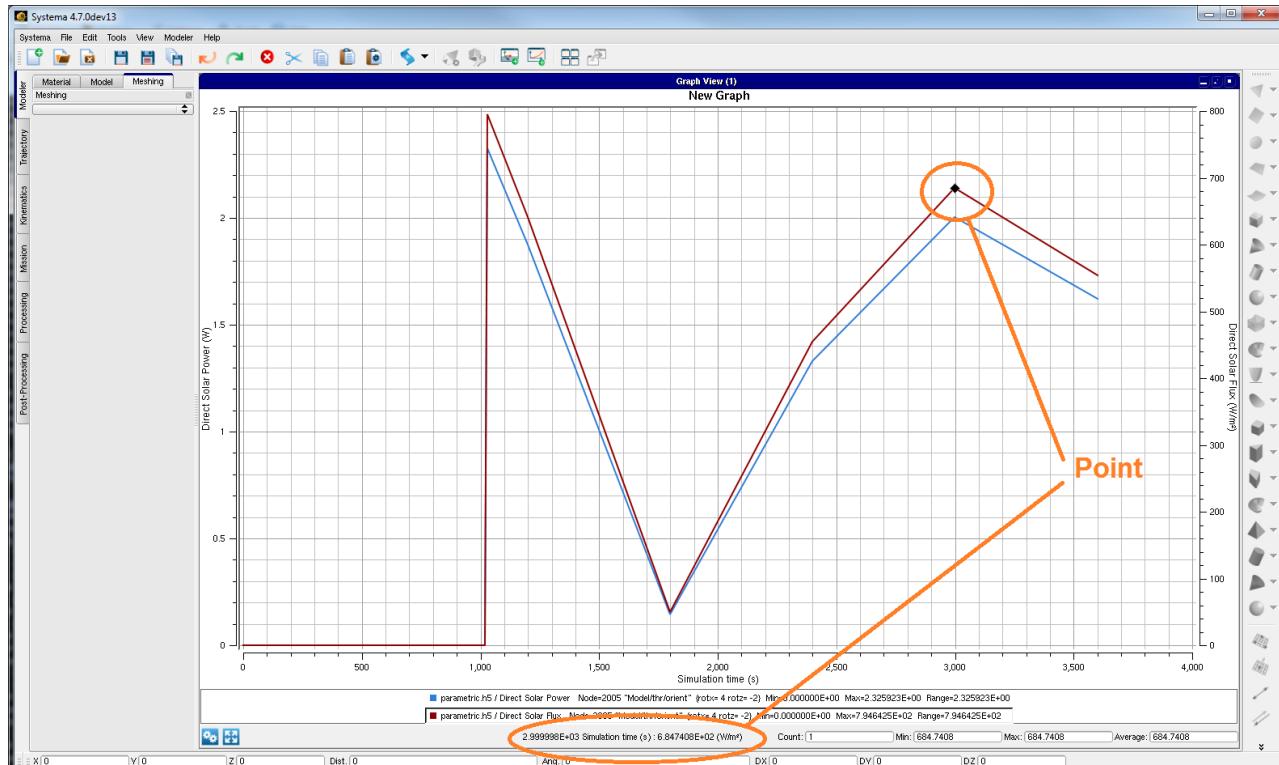


Example 3: The following diagram contains curves displaying two parametric results for two meshes.



Reading points on the curve

You can read the coordinates of a point by moving the cursor over the curve. Each point that corresponds to a calculated result is indicated by a cross on the curve and the corresponding coordinates are displayed in the information bar at the bottom.



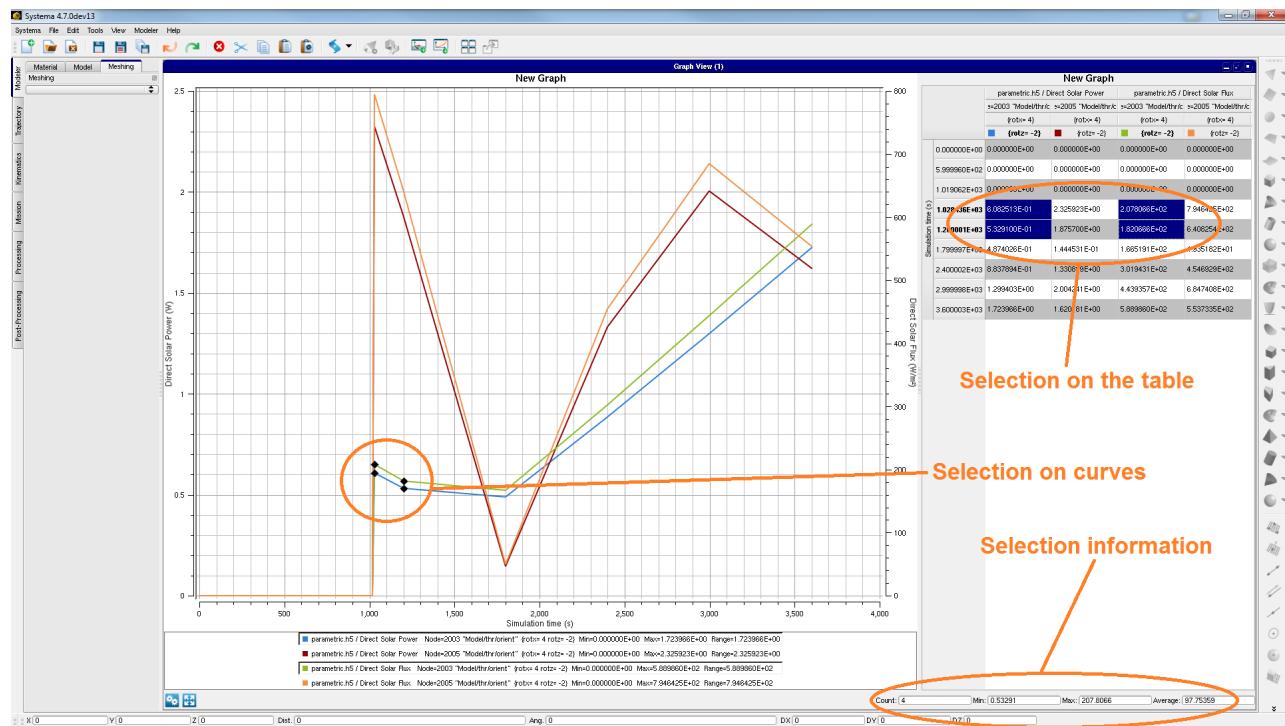
Note : it is also possible to get information about a point by selecting it and reading the values in the table. See the following section.

Selection in the graph and the table

If the left button of the mouse is clicked when a cross is displayed on the curve, the point is selected in both the curve and the table. A point selected in a curve is displayed with a marker.

The selection is synchronized between the table and the curves : if a point is selected in the graph, the matching cell will be selected in the table. And if a cell is selected in the table, the matching point will be selected in the graph. It is possible to use the 'control' keyboard key to cumulate selection. You can select a whole line or column in the table by clicking on the table header. You can select a whole curve in the graph by clicking on the legend of this curve.

The information bar at the bottom of the window display information about the selection : number of selected items, minimum, maximum and average.

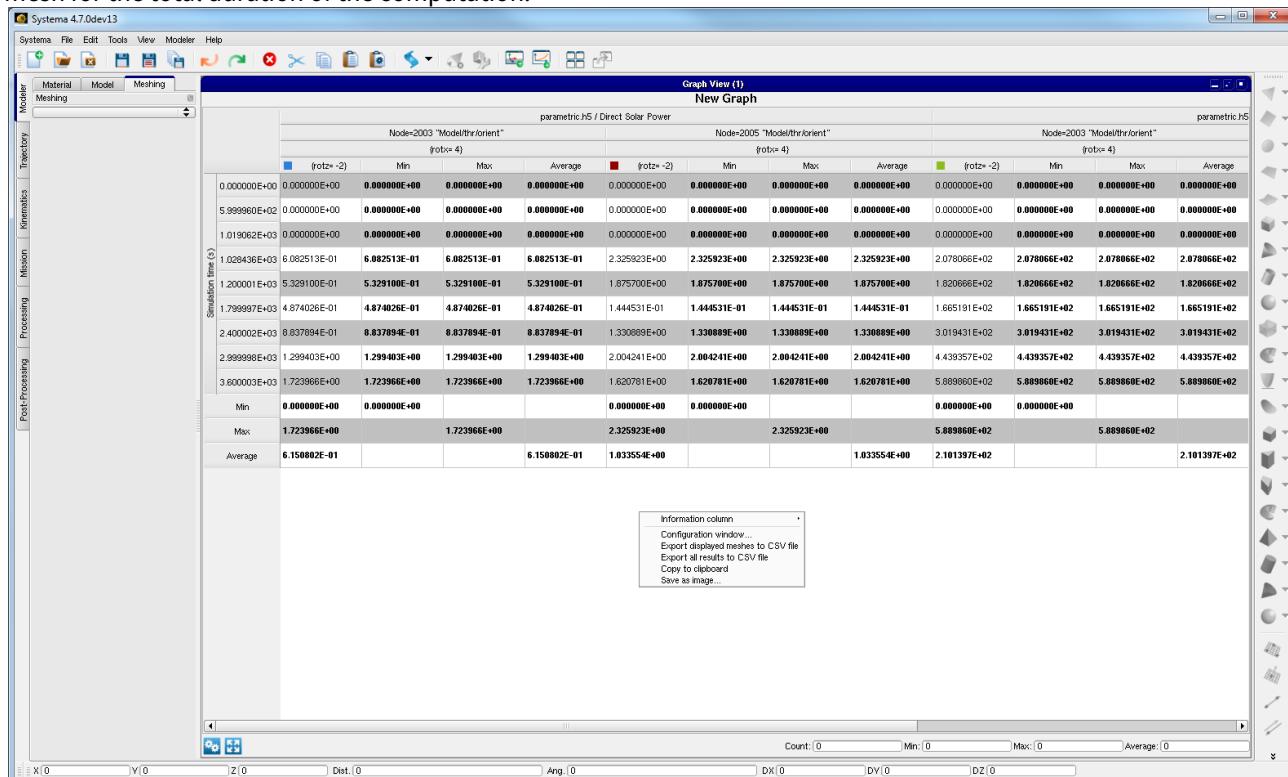


Minimum, maximum and average in the table

It is possible to display specific values in the table : minimum, maximum and average on properties and mesh. To display these information, right click with the mouse , and choose an information column. The information values are displayed in bold.

e.g. you can see the minimum of the property at a specific date for all meshes, or the average of a property on a

mesh for the total duration of the computation.

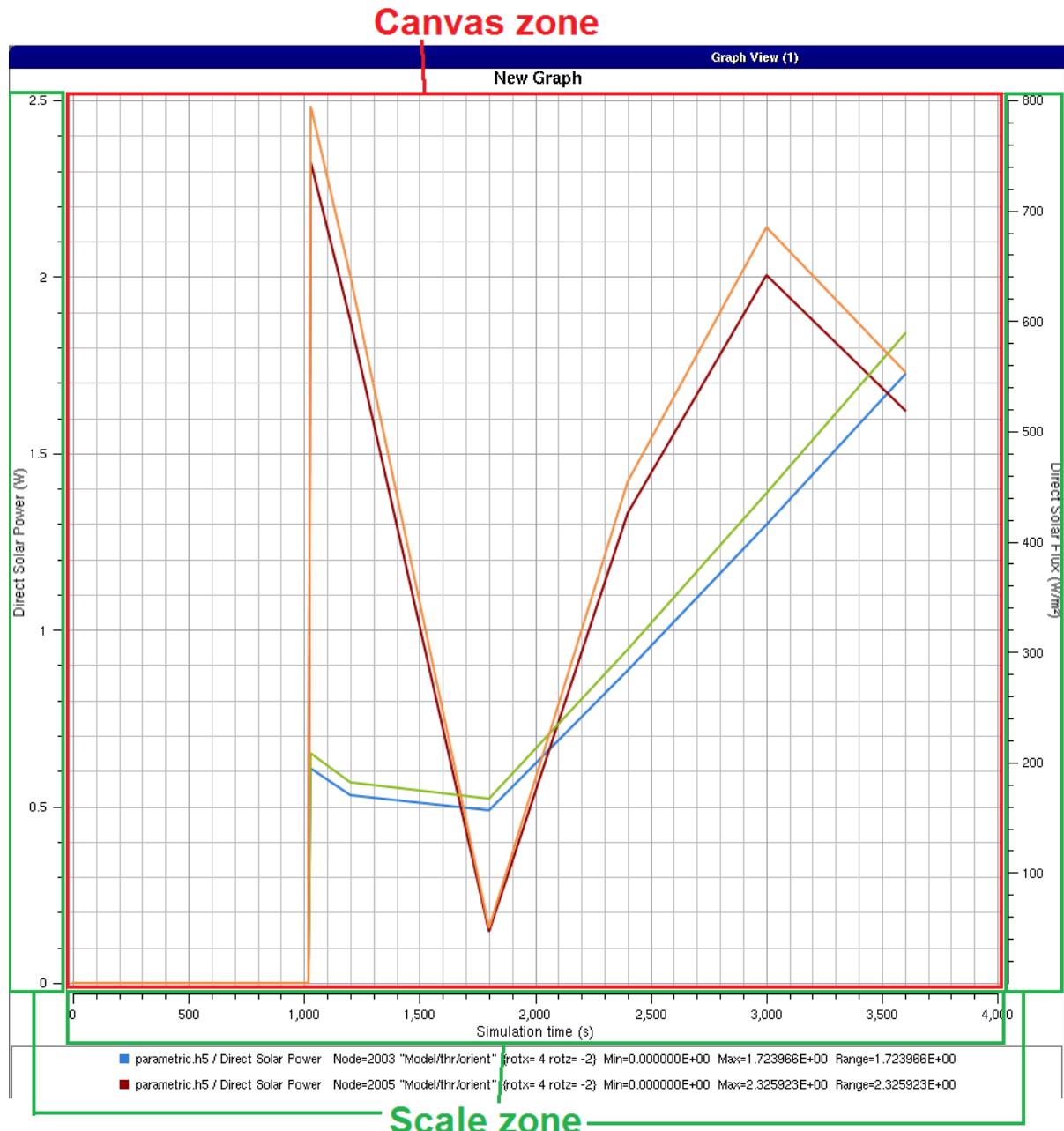


Data export

You can export data in two formats:

- A picture by right click menu, **Copy to clipboard**.
- Data in a CVS file by right click menu, **Export to CSV file**.

Navigation in curve view



- Several navigation features are available (zoom, fit, drag).
- Use mouse wheel on canvas zone to zoom all axis.
- Use mouse wheel on a scale zone to zoom only the corresponding axis.
- Use mouse drag on canvas zone to move all curves for all axis.
- Use mouse drag on a scale zone to move only the corresponding axis.

Selected meshes management

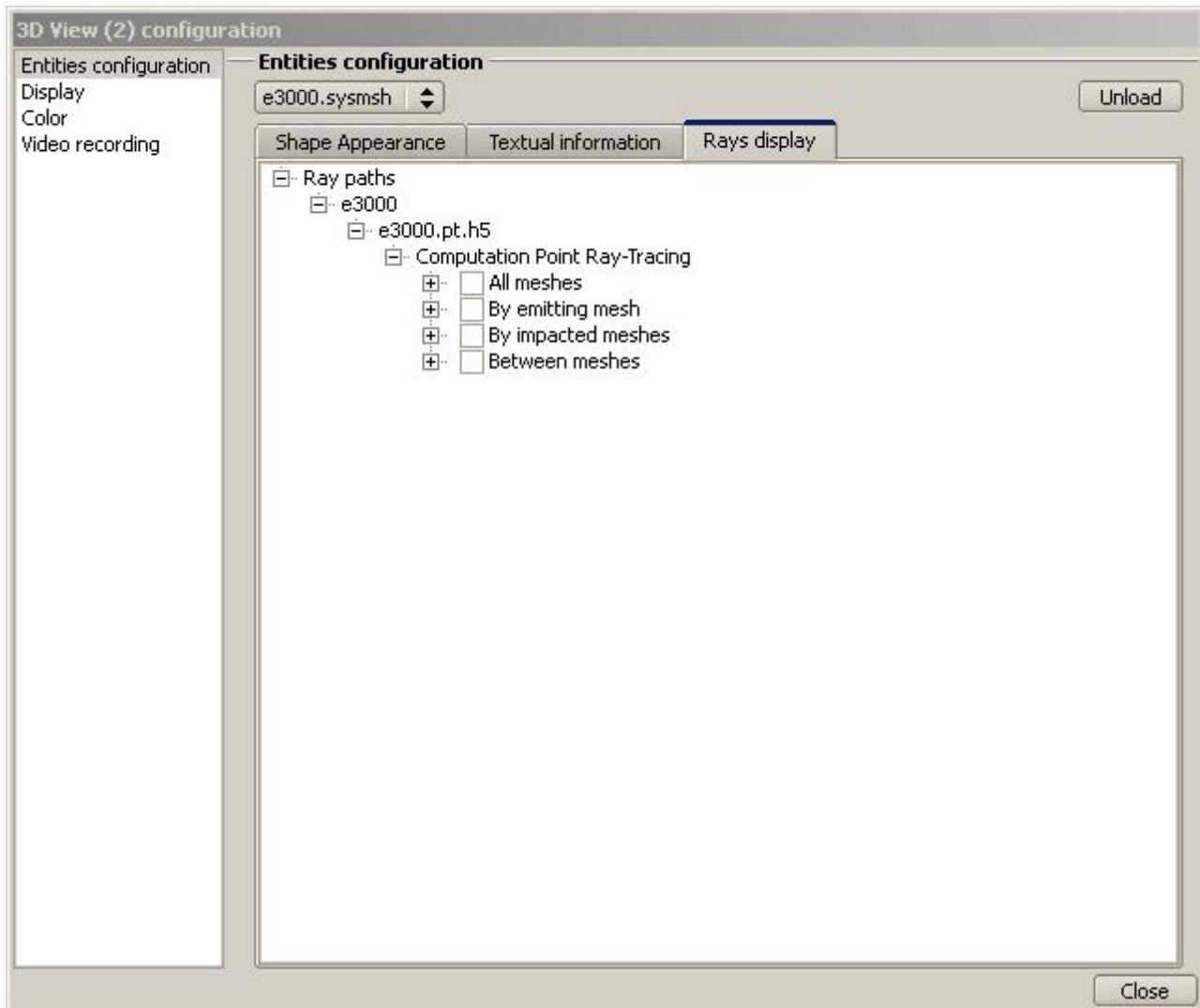
Possibility to lock the selected meshes for a curve view, in order to select other meshes without modifying the display of the 2D view. Display the configuration window (In the right click menu, select **Configuration window**). After selected interactively the meshes, in **Fixed index** section, selected meshes are displayed. Unselect "Synchronized with meshing selection" and apply, Selected meshes are still displayed. Then if you want to add interactively other meshes, select back "Synchronized with meshing selection" and apply. Next selected meshes will be added to the previously ones.

Displaying the computation rays used in the application

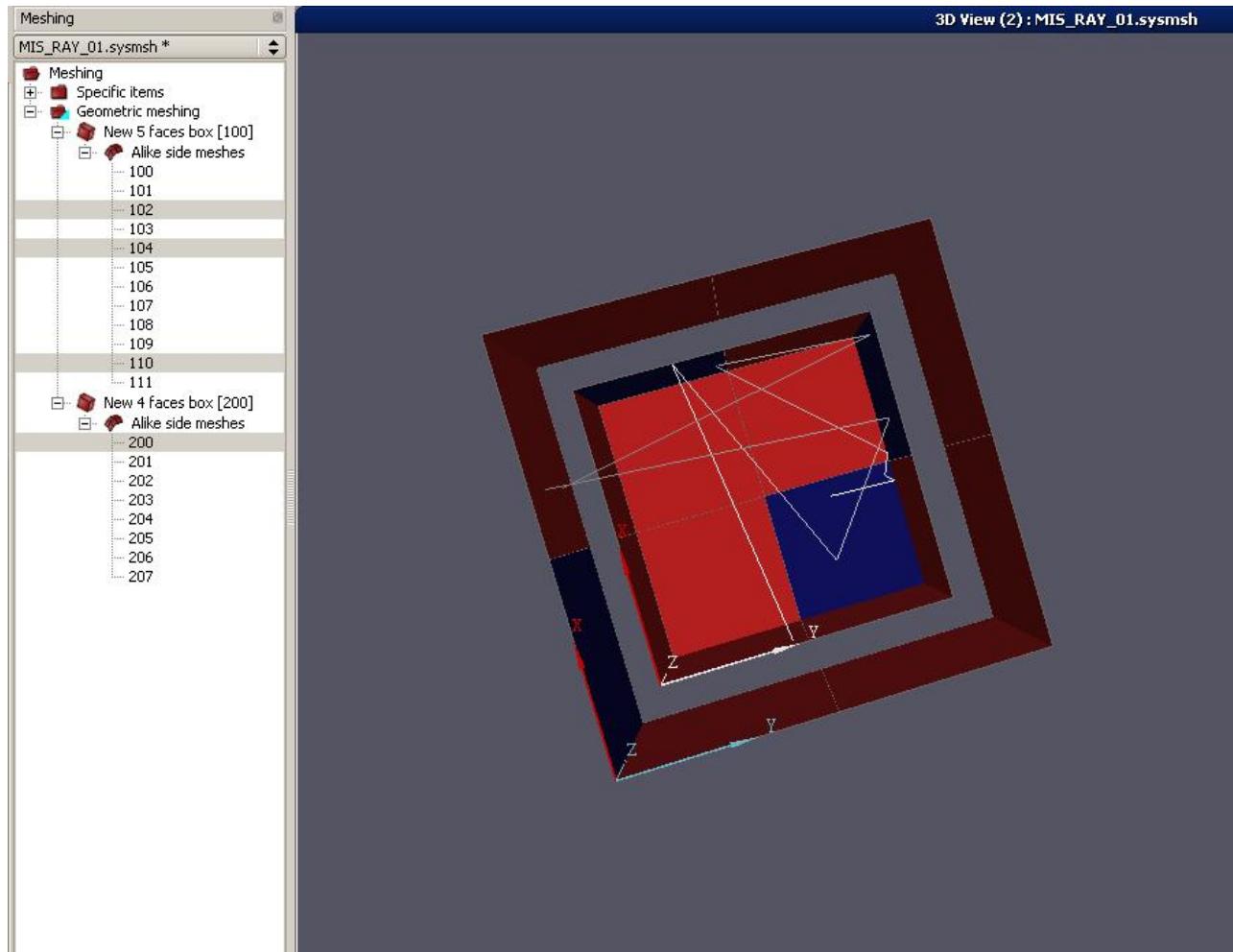
Some applications allow you to visualize the rays they use for their computation.

How to display the application's computation rays

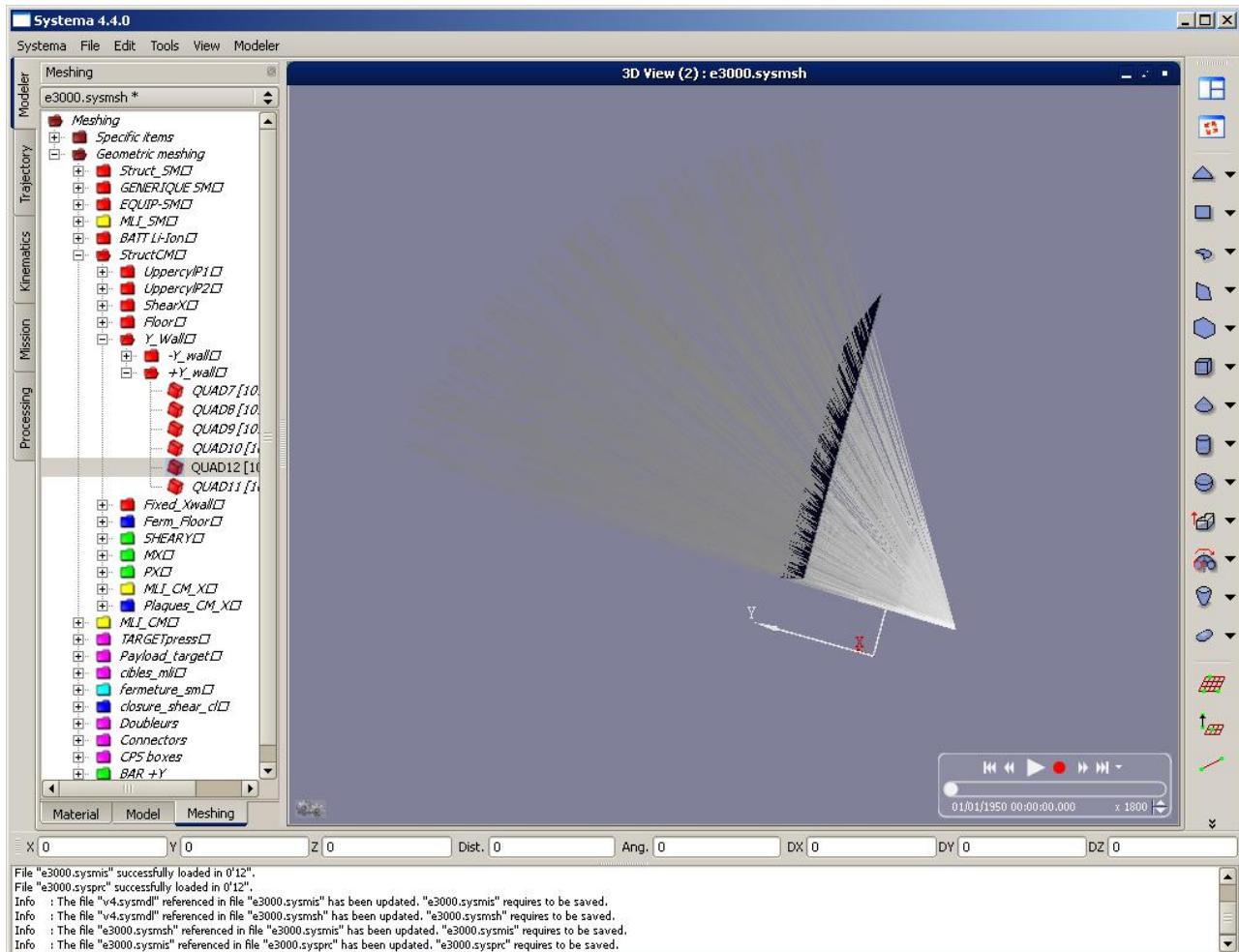
1. Open the meshing.
2. In the processing tab, load or compute results related with that meshing that contain rays data. Refer to your application's documentation for further information.
3. In your meshing 3D view, select **Configuration – Configuration window – Entities configuration – Rays display**. You have then four options:



- **All meshes** allows you to see all rays emitted by all meshes.
- **By emitting mesh** allows you to see all rays emitted by the currently selected mesh. When activated, the 3D view will display the rays launched by the mesh you select in the view or in the browser. If you select several meshes, you will see all the rays emitted by all these meshes.
- **By impacted mesh** compute all rays received by the currently selected mesh. If several meshes are selected, only the rays that impact all these meshes will be displayed.



- **Between meshes** compute all rays that where launched or impact all the selected meshes.
1. The two options **Impacted meshes** and **Between meshes** can be configured by selecting or not **Omit hidden meshes** which can be accessed by the **Configure** button in the configuration window. If enabled, the visualisation of rays will not display the rays that go through meshes that are hidden in the current view. If disabled, the shown or hidden meshes will be both considered in the ray visualisation computation.



- Because these results depend on the time, you have access to the animation panel, that allows you to control it.

- The set of animation buttons**

The animation buttons are described in the following table:

Button	Use
	Starts running (or re-starts) the animation
	Stops running the animation (just pauses it, time and position are kept)
	Starts recording video
	Moves the spacecraft to the previous event (computation point, penumbra or eclipse)
	Moves the spacecraft to the start date

Button	Use
	Moves the spacecraft to the next event (computation point, penumbra, eclipse...)
	Moves the spacecraft to the end date
	Allows selecting the type of event (computation point, penumbra, eclipse...)

- **The player speed**

To modify the player speed :

1. To increase the player speed, click .
2. To decrease the player speed, click .
3. The speed value can be modified manually

Schematic management

Schematic management

Overview

Introduction

This chapter describes how to use SYSTEMA in order to create, load, save, view schematic diagrams. With SYSTEMA, you use the **Schematic** tab to work 2D schematic diagrams. Everything described in this chapter takes place within the **Schematic** tab.

In this chapter

This chapter covers the following topics:

- Schematic management
 - Overview
 - Introduction
 - In this chapter
 - The Schematic tab
 - What is the Schematic tab for?
 - The different parts of the Schematic tab
 - Example
 - The menu bar
 - The Schematic diagram area
 - The component browser
 - Schematic diagrams in SYSTEMA
 - What is a Schematic diagram?
 - Working with Schematic
 - General
 - Creating a new schematic diagram
 - Opening an existing schematic diagram
 - Modifying a diagram
 - Building a diagram
 - How to build a diagram
 - Adding standard item
 - How to add a component
 - To add a component from the browser:
 - To add a component from the menu:
 - Result
 - Organizing schematic components
 - Joining connectors
 - How to join connectors
 - To join two files:
 - How to split a connector
 - To split a connector link using the right click menu:
 - Manipulate a component
 - How to move a component
 - How to rotate a component
 - How to delete a component
 - How to edit a component
 - Manage points on line

- Use the selection rectangle
 - How to create a selection rectangle
 - How to move a selection rectangle
 - How to rotate a selection rectangle
 - How to delete a selection rectangle
- Define variables
- Numbering of nodes
 - How to display the numbering
 - How to modify the numbering
- Management of connector label

The Schematic tab

What is the Schematic tab for?

You can use the **Schematic** tab of the SYSTEMA application to:

- open existing schematic diagrams and modify them when necessary,
- create new diagrams by combining components and links.

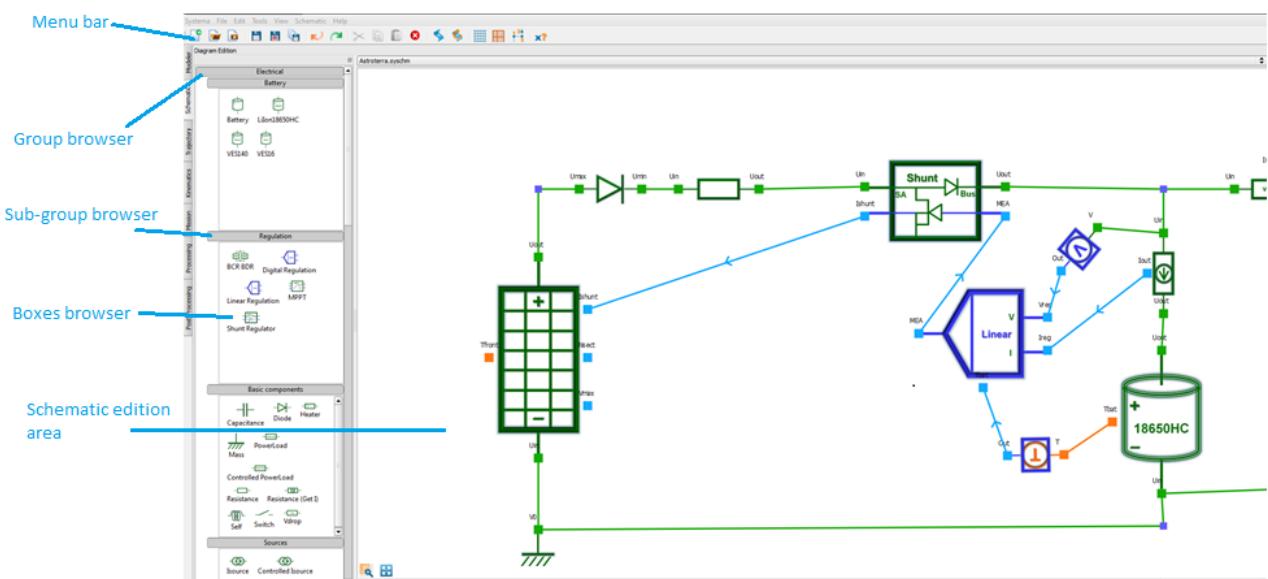
The different parts of the Schematic tab

The **Schematic** tab is divided into 3 main parts:

- the *Schematic* area, containing the diagram,
- the *Edition* area, containing the elements definition of the edited element.

Example

The following diagram shows an example of the **Schematic** tab. The *Schematic* area contains a diagram, the *Edition* area contains the parameters of an item:



The menu bar

The menu bar contains:

- The menus which are common to all SYSTEMA tabs: the **File**, **Edit**, **Tools**, **View** and **Help** menus.

For more information, refer to the "General User Interface" chapter

- One specific menu, the **Schematic** menu, which can be used to build the current post-processing.

The following table presents the options available from the **Schematic** menu:

Option	Use
Import a schematic component	Adds a component to the current schematic with default attribute values <i>Available schematic component here are the same as those in the Components browser displayed in the Schematic area</i>

The undo/redo and the cut/copy/paste functionalities are not available.

The Schematic diagram area

The *Schematic* area is divided into two parts:

- the right-hand part contains the *schematic viewer* which is used to display, build and modify the diagram for the current schematic;
- the left-hand part contains the *components browser* which displays the list of components that can be added to the current schematic

The schematic viewer can be manipulated with the following actions:

Action	Control
Zoom	Ctrl + mouse wheel
Pan	Mouse middle clic and drag
Fit Selected	 button is available in the diagram area
Fit All	 button is available in the diagram area

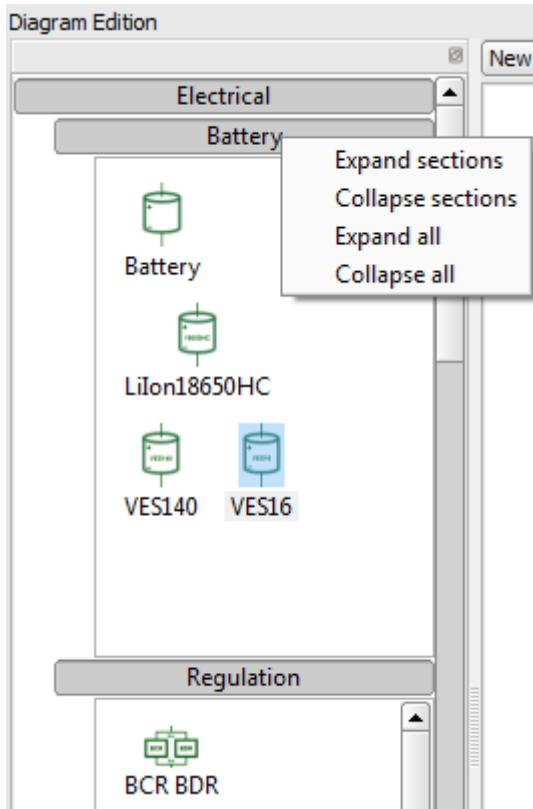
A line grid can be displayed or hidden by clicking on 

A point grid can be displayed or hidden by clicking on 

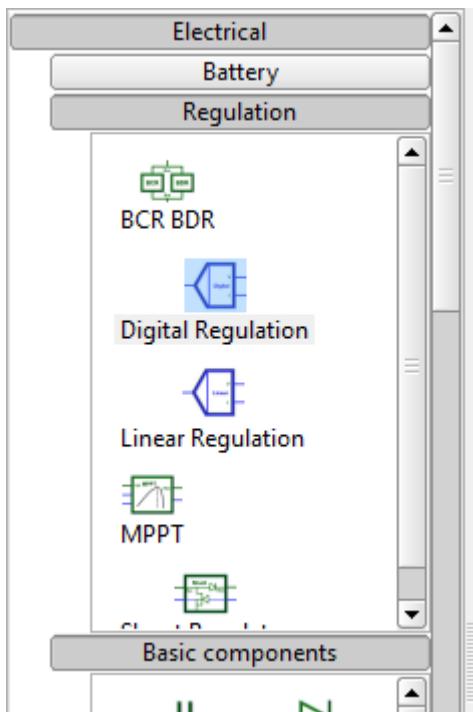
The component browser

The component browser is divided in component groups. By default the browser is expanded. But it is possible to collapse some groups by right click on a group name and select "Collapse section".

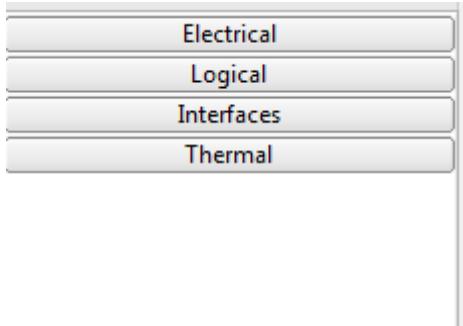
Before collapsing a group in the browser



After collapsing a group in the browser



All the browser can be collapsed :



Schematic diagrams in SYSTEMA

What is a Schematic diagram?

A SYSTEMA schematic diagram is a 2d diagram representing the electrical, thermal and logical interactions between components.

The electrical network is represented by the green components and connectors.

The thermal network is represented by the orange components and connectors.

The logical network is represented by the blue components and connectors.

Working with Schematic

General

In the **Schematic** tab, you can open an existing schematic or create a new one.

When a schematic is opened, you can:

- modify it,
- save it.

Creating a new schematic diagram

To create a new schematic:

- In the **File** menu of the **Schematic** tab, select the **New** option.

Result: The *Schematic diagram* area displays an empty editable space.

You can then build the diagram and save it.

Opening an existing schematic diagram

You can open a schematic which has previously been saved within a ".syschm" file.

To edit an existing diagram:

- In the **File** menu of the **Schematic** tab, select the **Open** option and select the file containing the required diagram.

Result: The diagram is displayed in the *Schematic* area.

You can then modify it and save your modifications.

Modifying a diagram

You can modify a diagram by :

- adding new components,
- linking them together with the other components,
- Copying/Cutting and pasting sub-diagrams,
- editing and modifying components and connectors.

Building a diagram

How to build a diagram

You build a diagram by adding and organizing components in a diagram.

Adding standard item

There are two different ways to add standard boxes:

- using the *Components* browser,
- or using the **Schematic** menu.

How to add a component

To add a component from the browser:

- In the browser, drag and drop it on the diagram part.

To add a component from the menu:

- In the **Schematic** menu, select the **Import a schematic component** option, then select the component you want to add.

Result

The selected component with its associated connectors is added with default values. Their attributes can then be modified by editing them.

Organizing schematic components

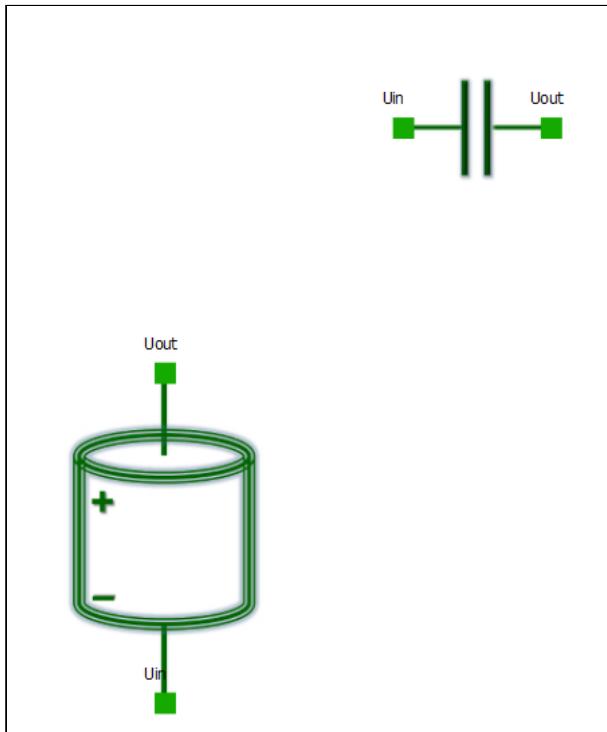
To organize components , you can join or split connectors.

Joining connectors

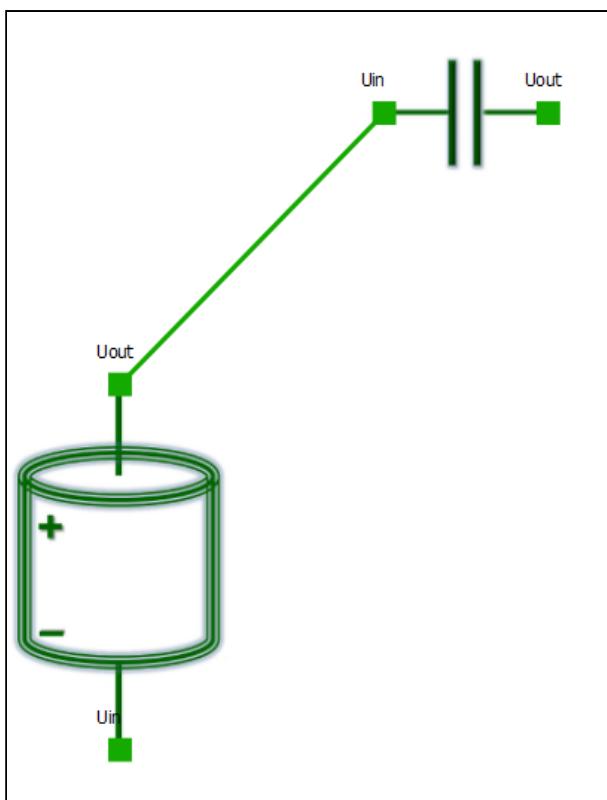
You can join connectors in the following case:

- When the connectors of one component is from the same network than the connector of another component (in the electrical and thermal network):

Before joining the connectors:



After joining the connectors:



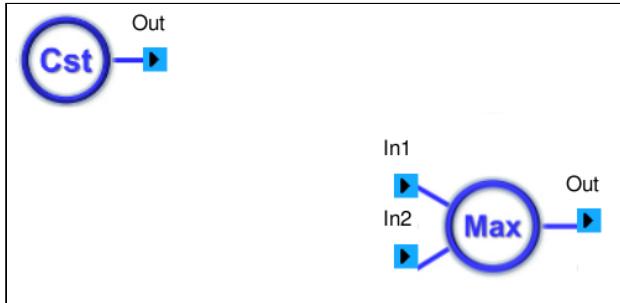
- When the connectors are an output of one component and an input of another component (in the logical networks):

In that case, an arrow appears on the connector to indicate that it is an input or an output.

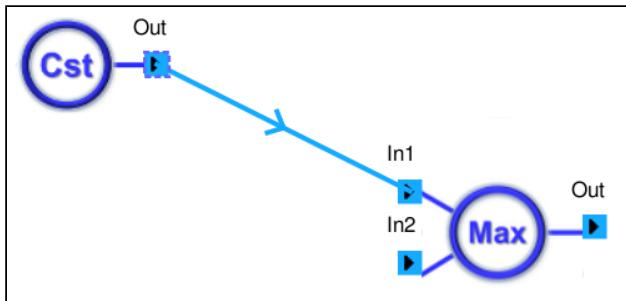
Links can only be created from the output to the input.

After line creation, another arrow appears on the line to show the orientation of the link (from the output to the input).

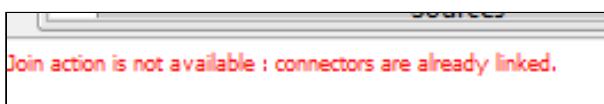
Before joining the connector:



After joining the connector:



If a connection is not valid, an error message appears in the log view

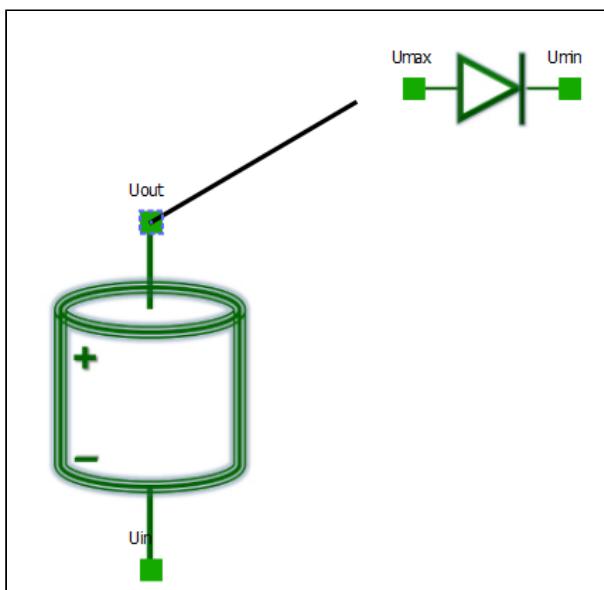


How to join connectors

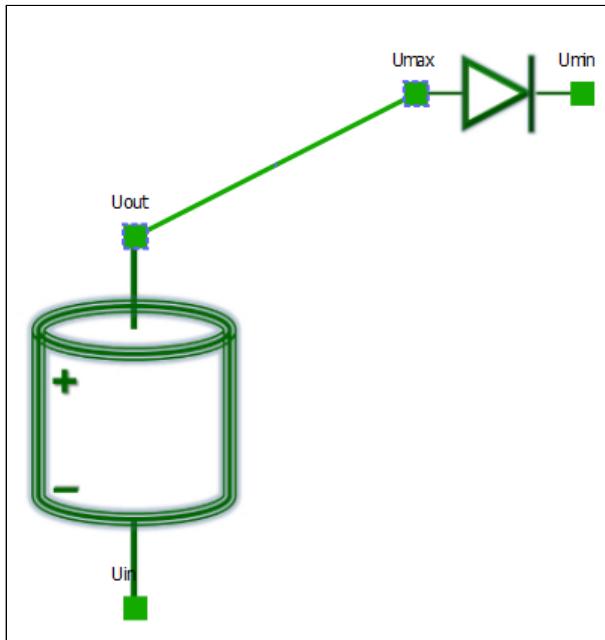
To join two files:

1. Move the mouse to the connector and click on it
2. Hold down the left mouse button while moving the mouse toward the other connector to be joined

Before joining the connectors:



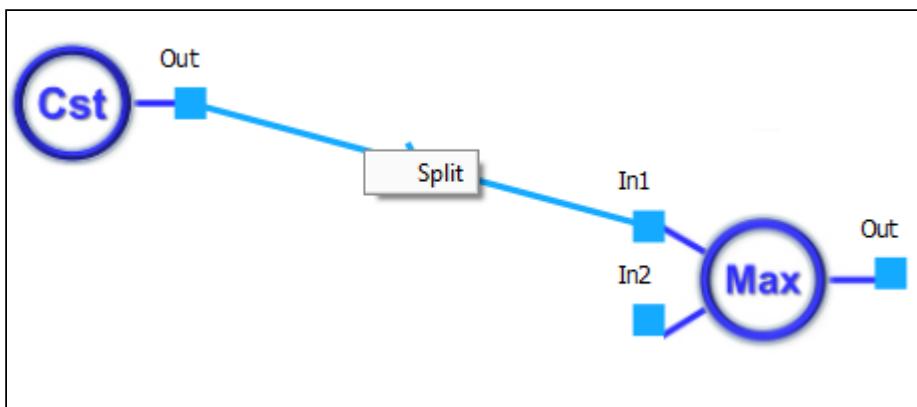
After joining the connectors:



How to split a connector

To split a connector link using the right click menu:

1. Move the mouse to the link between the connectors.
2. Using the right click menu, select the **Split** option.

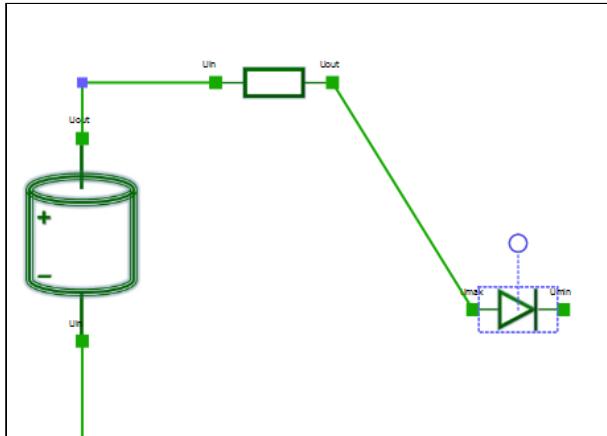


Manipulate a component

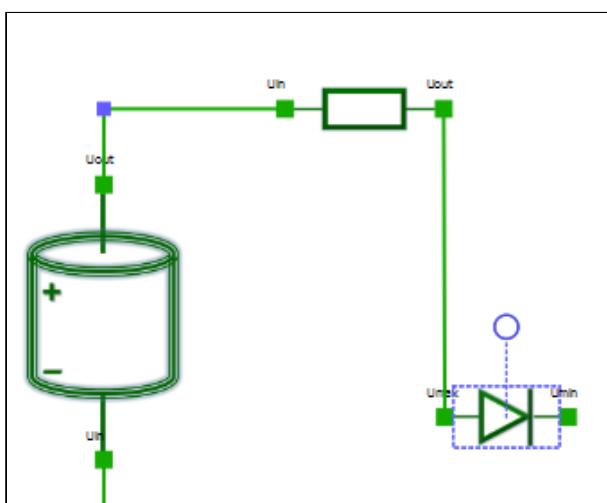
How to move a component

1. Move the mouse to the component and click on it
2. Hold down the left mouse button while moving the mouse toward the new location

Before moving a component:



After moving a component:



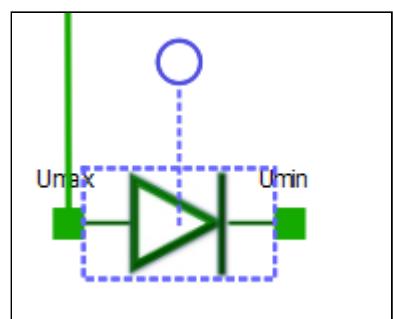
How to rotate a component

1. Move the mouse to the component and click on it

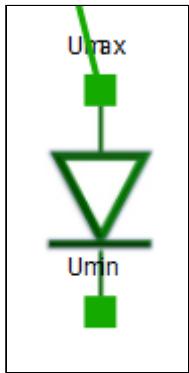


2. Hold down the left mouse button on the the rotation handle while moving the mouse toward the new angle

Before rotating a component:



After rotating a component

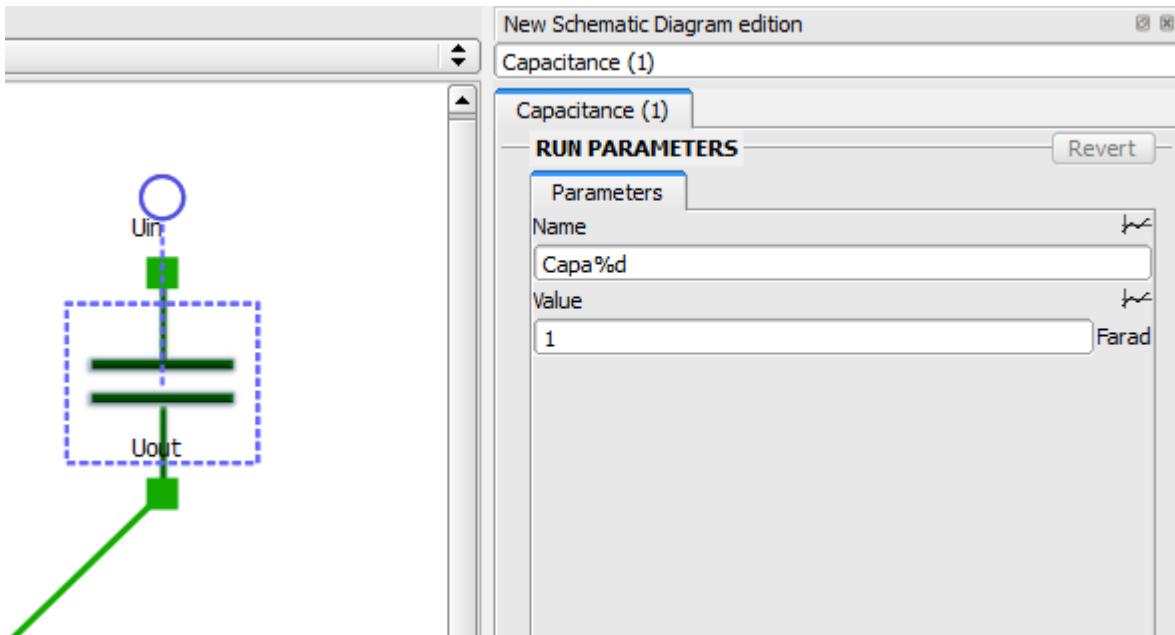


How to delete a component

To delete a component, select it and click on **Suppr** key or on the toolbar icon

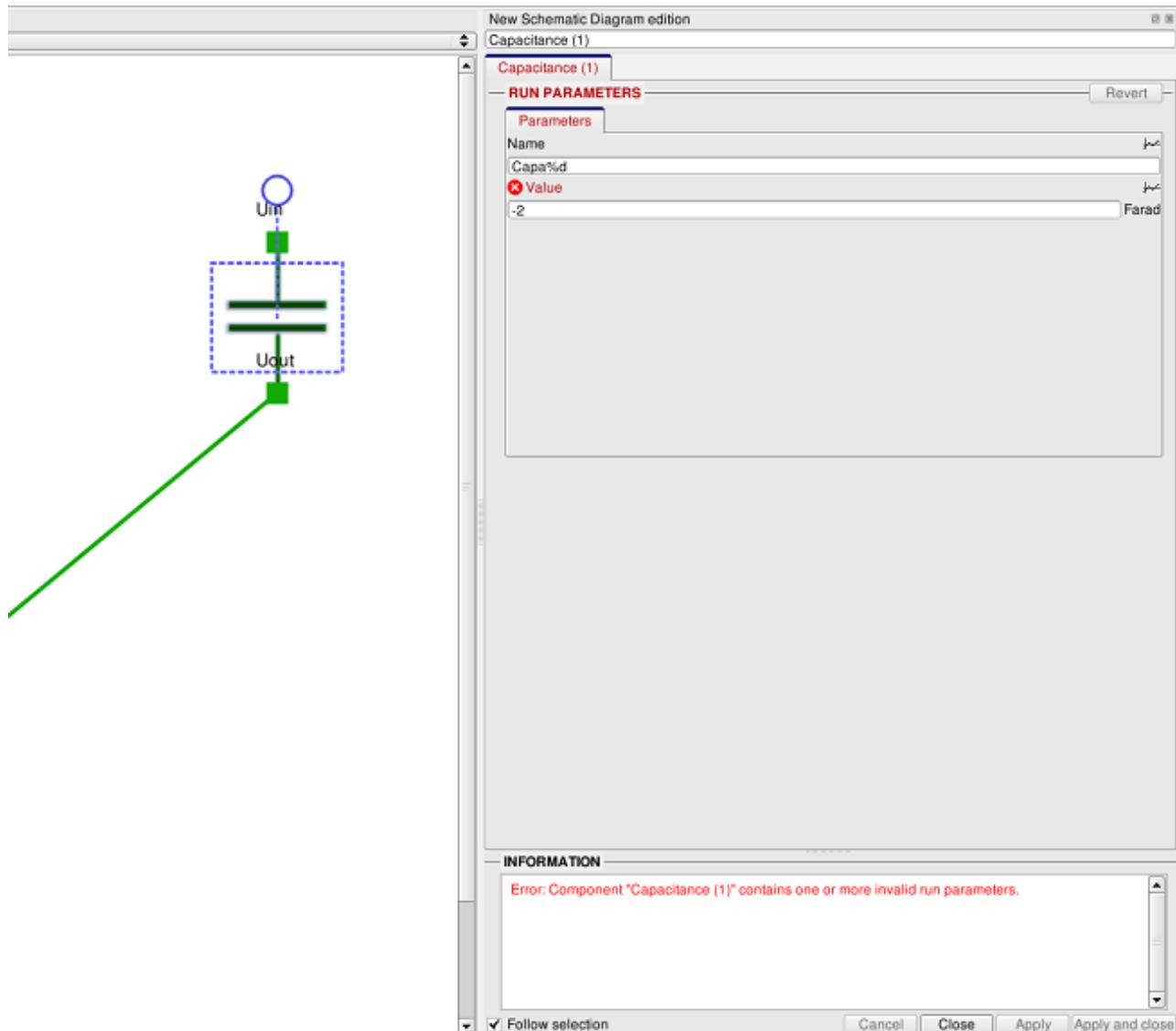
How to edit a component

To edit a component, double click on the component and the edition Window will open.



Whenever component parameters fall out of range or invalid for some reasons, their display turns red in the edition window.

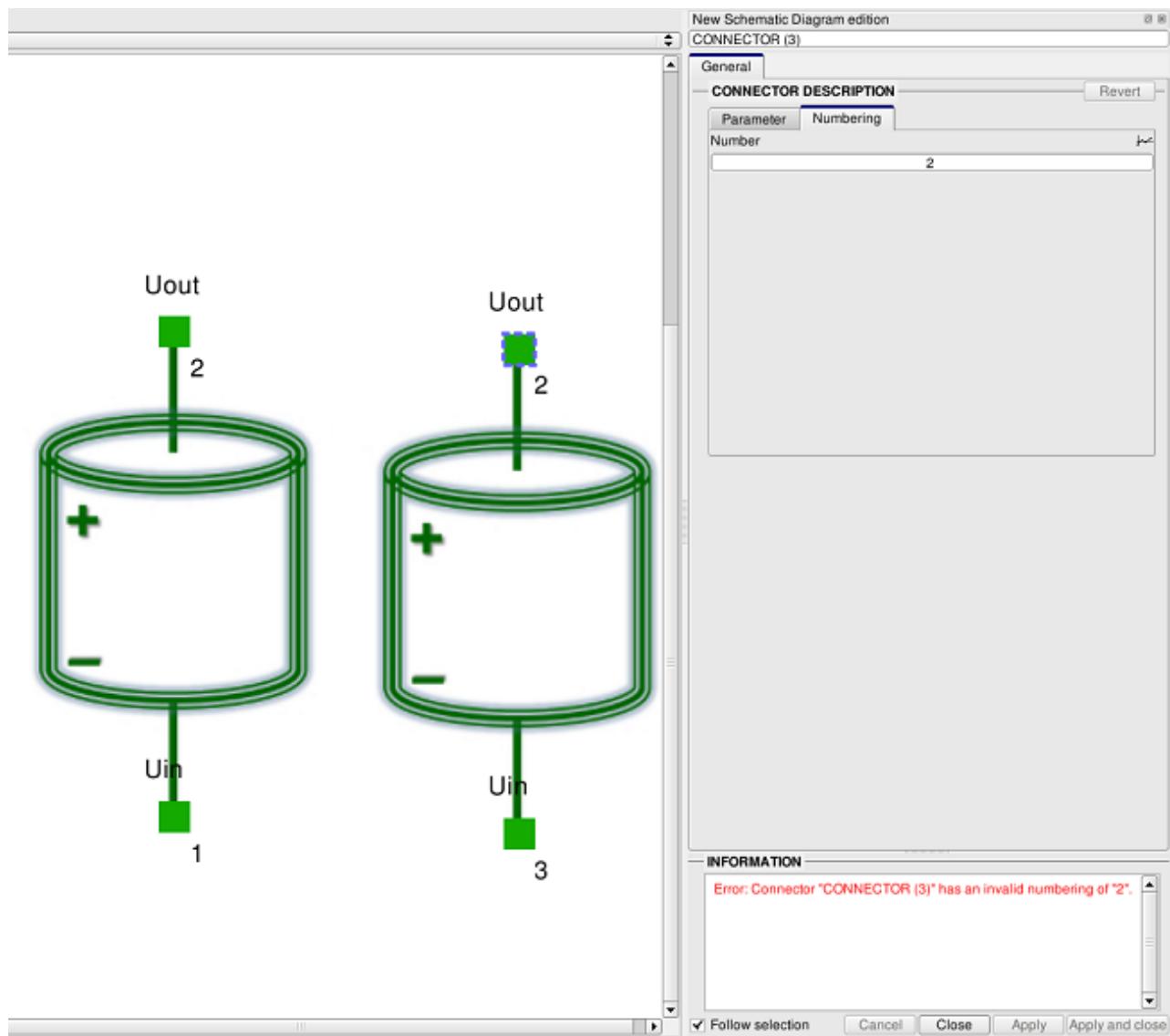
The error is described in the bottom-right log section of the edition window.



The same error message is also displayed in the console view.

Component "Capacitance (1)" contains one or more invalid run parameters.

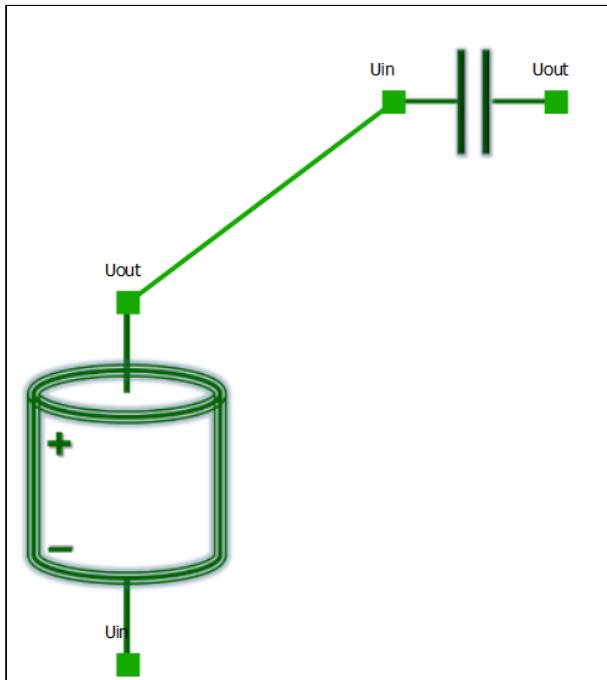
The error detection mechanism is similar for the connectors parameters.
Moreover, it addresses numbering errors such as negative or redundant numberings.



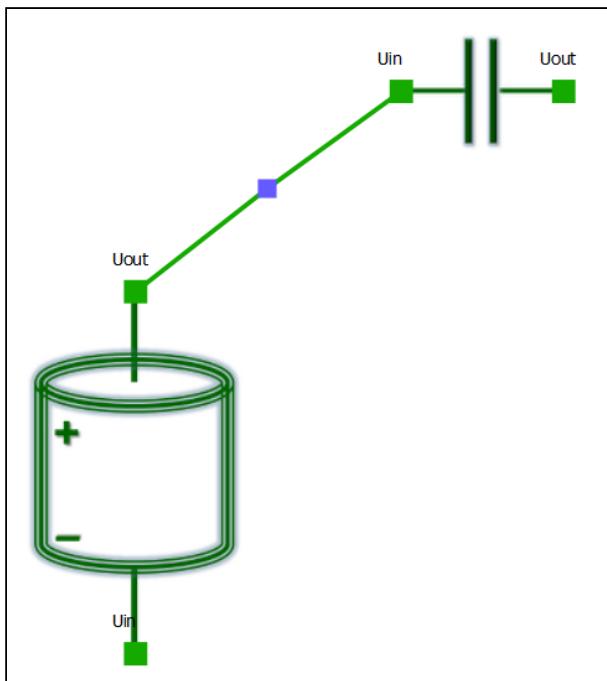
Manage points on line

In the schematic, it is possible to create point on the lines with **Shift+left** click on the line.
 This function is disabled for logical networks.
 This function is not handled by the copy/cut/paste operations.

Before creating the point:

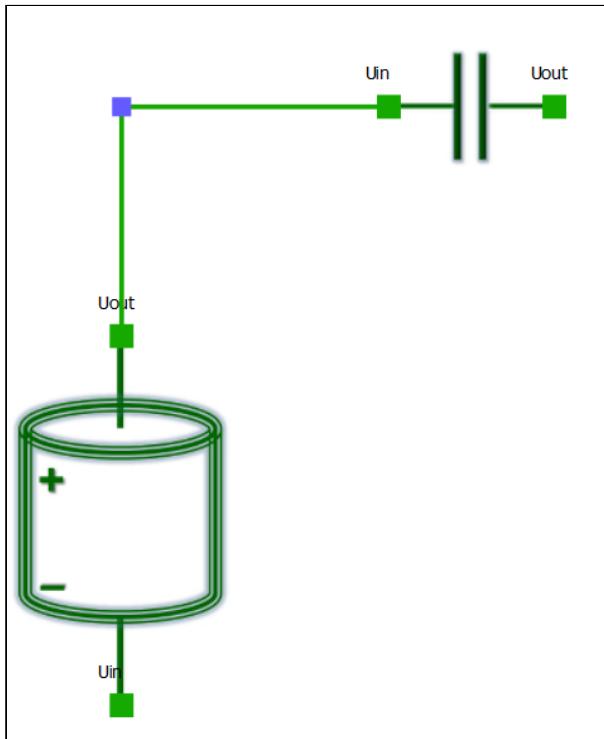


After creating a point:



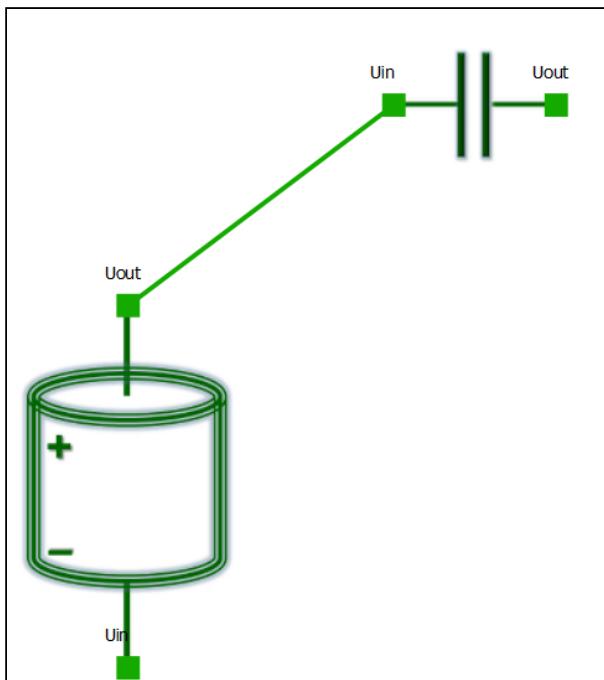
Points can be moved by selecting them and drag them on the diagram area

After moving the point:



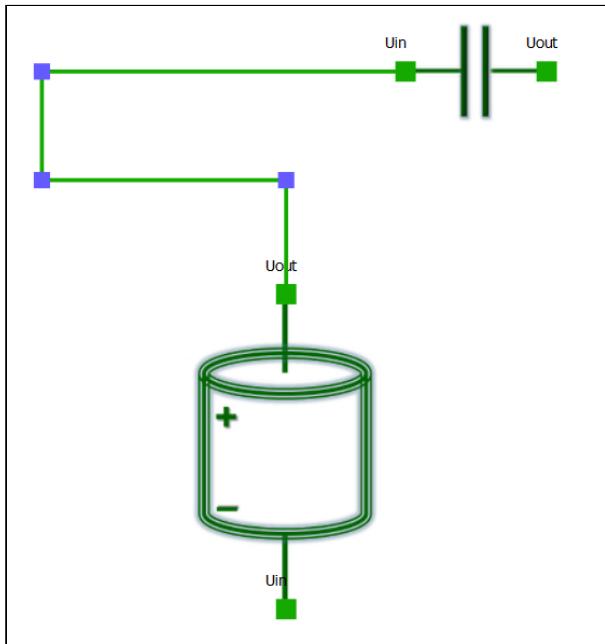
Points can be deleted by selecting them and click on **Suppr** key or on the toolbar icon . When a point is deleted, the line still exists

After deleting the point:



It is possible to create multiple points on a line.

After the creation of multiple points:

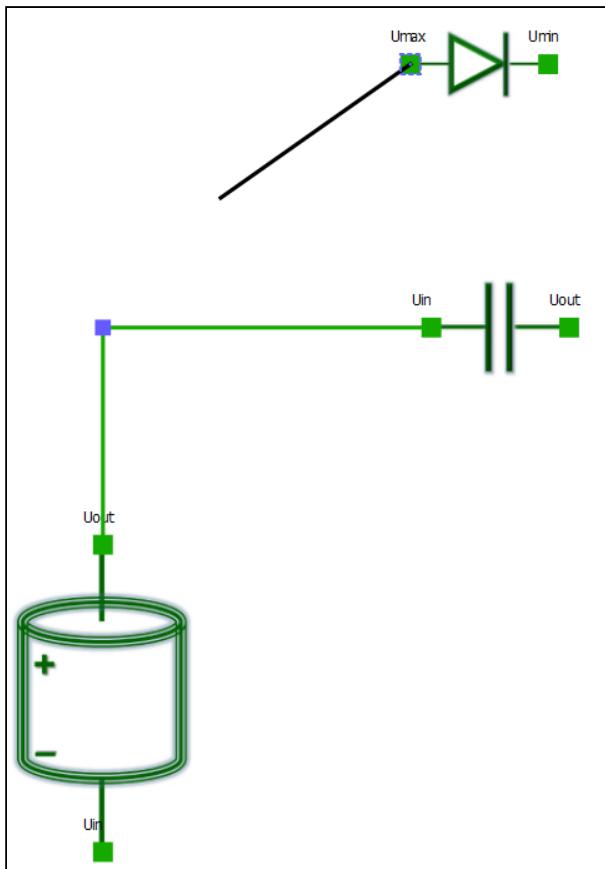


It is possible to link a connector directly to a point by selecting the connector and create a line to the point:

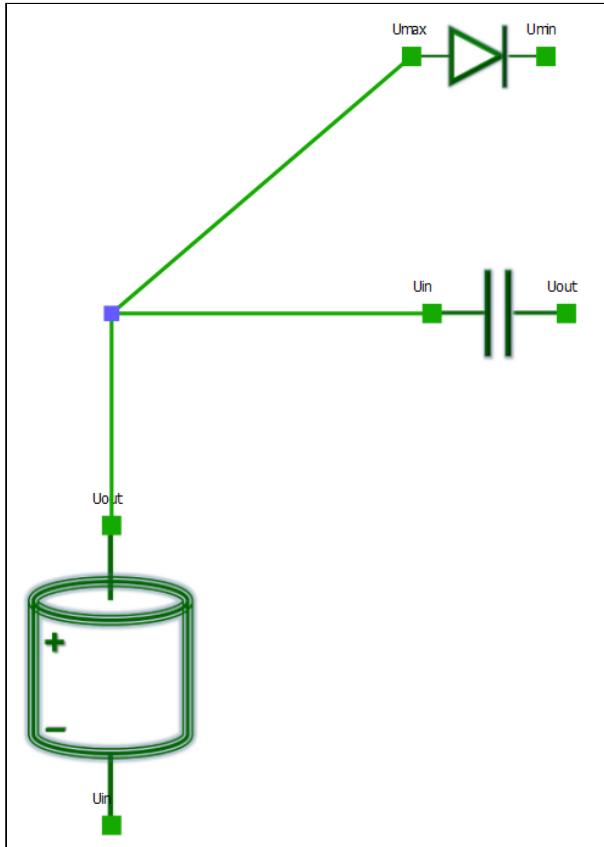
1..Move the mouse to the connector and click on it

2..Hold down the left mouse button while moving the mouse toward the point to be joined

Before joining a connector to a point:



After joining a connector to a point:



It is also possible to link directly a point to a connector by selecting the point, then click on **Shift+left** on the point and drag the mouse until the connector.

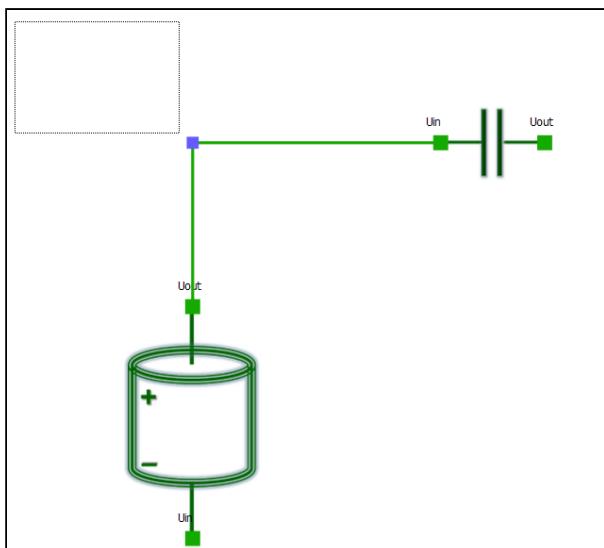
Use the selection rectangle

A selection rectangle is available to manipulate a set of elements.

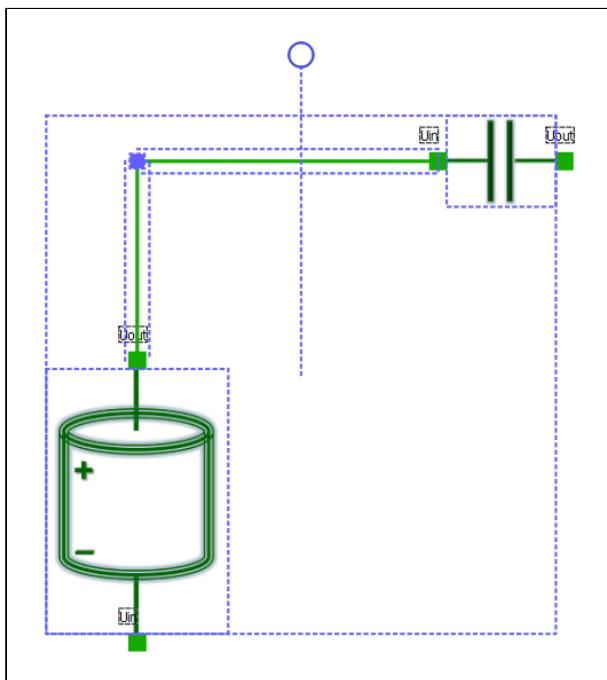
How to create a selection rectangle

Hold down the left mouse button while moving the mouse toward the elements to select

Before selecting a set of elements:



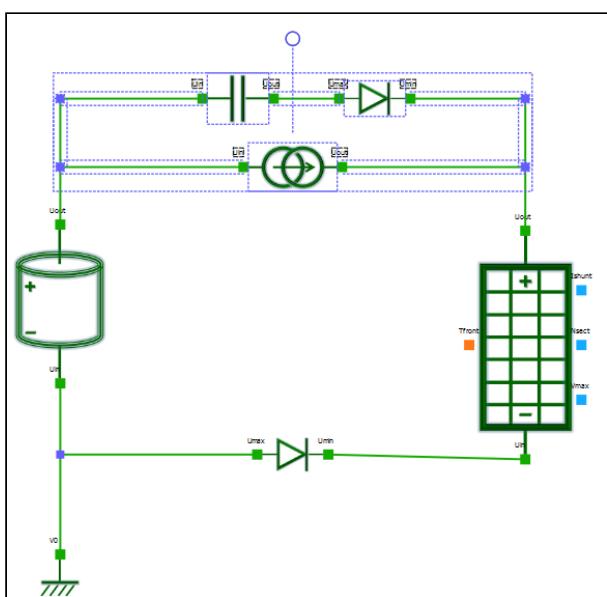
After selecting a set of elements:



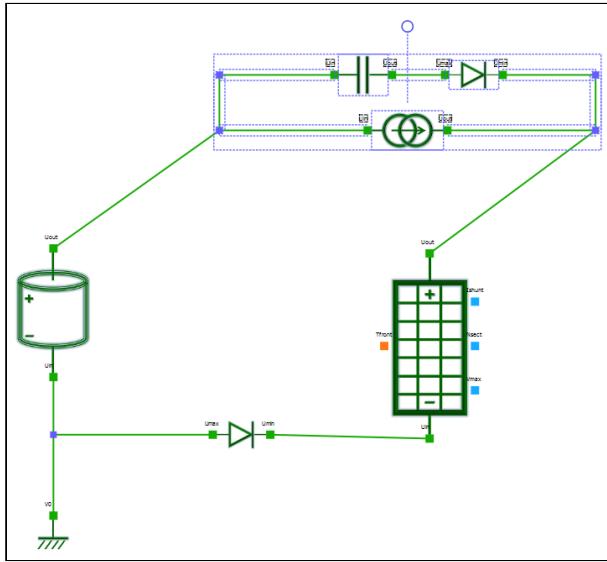
How to move a selection rectangle

When elements are selected, it is possible to move these elements: hold down the left mouse button while moving the mouse toward the new location

Before moving the elements:



After moving the elements:



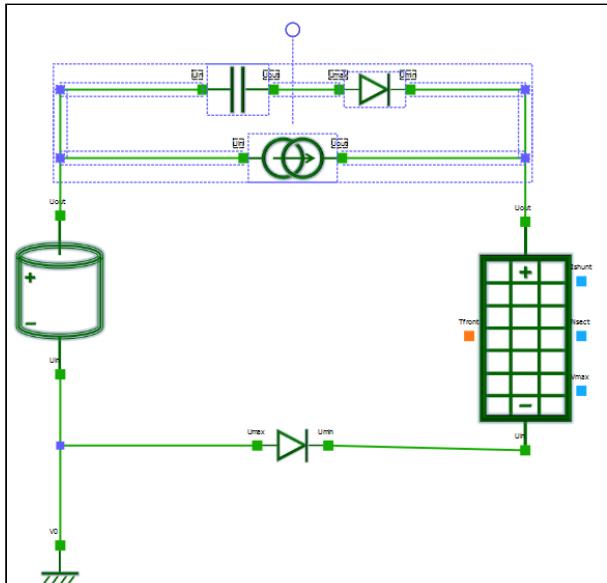
How to rotate a selection rectangle

When elements are selected, it is possible to rotate these elements: hold down the left mouse button on

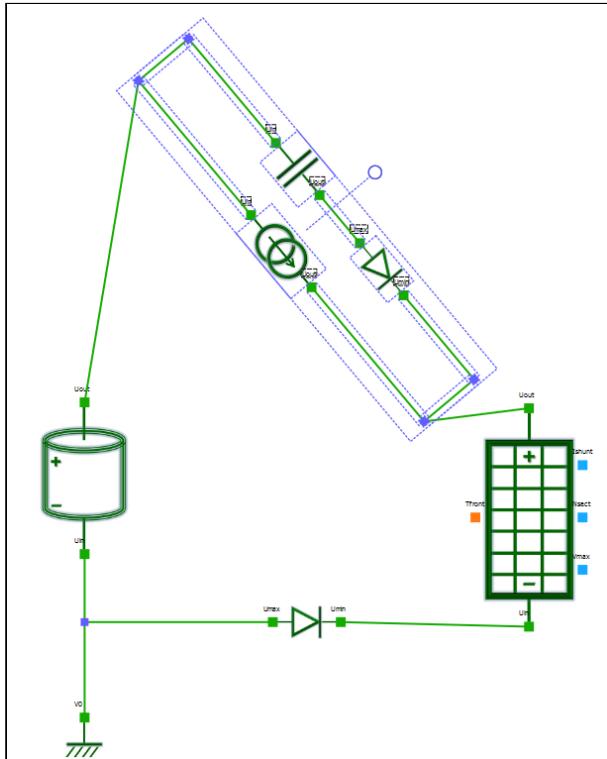


the rotation handle while moving the mouse toward the angle

Before rotating the elements:



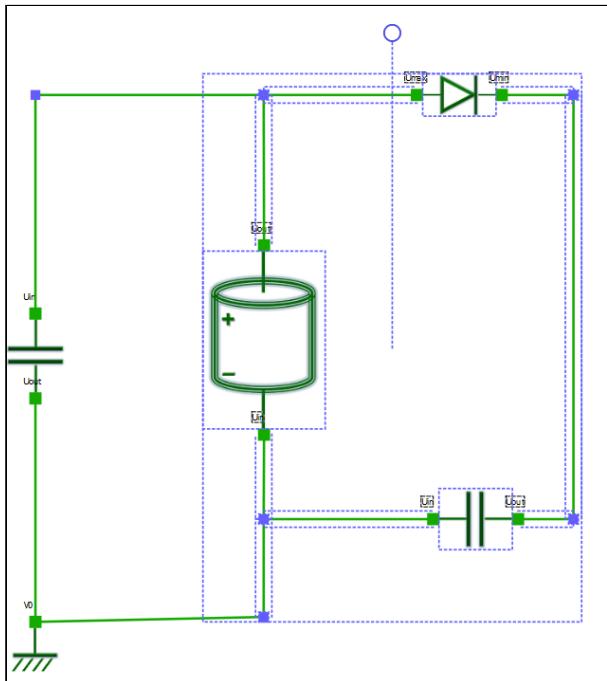
After rotating the elements:



How to delete a selection rectangle

It is possible to delete selected elements by clicking on **Suppr** key or on the toolbar icon .

Before deleting the selected elements:



After deleting the selected elements:



Define variables

Cf [Working with variables](#)

Numbering of nodes

Numbering is not available for logical connectors.

Each node is numbered automatically.

When a component is added, the number of each connector is automatically calculated.

When two connectors are linked for the electrical and thermal connectors, the number from the start of the link will replace the number of the connector at the end of the link

When a component is deleted, the numbering will re-start from the last highest number displayed in the diagram.

How to display the numbering



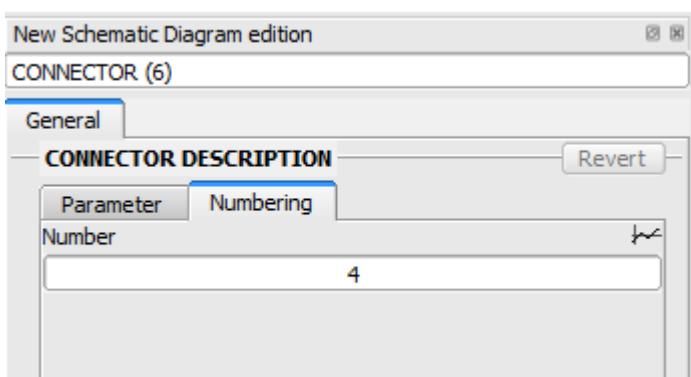
To display the numbering, click on icon in the toolbar. Numbers will be displayed close to graphical node.



To hide the numbering, click on icon in the toolbar. The numbering will be hidden.

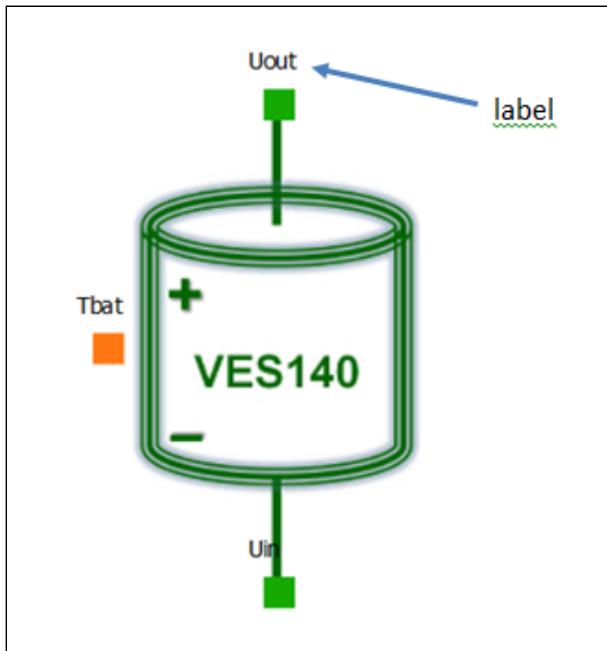
How to modify the numbering

It is possible to update the number of a node. Double click on the connector. An edition window will appear. On the Numbering tab, modify the number and click on apply. The number will be updated in the diagram view.



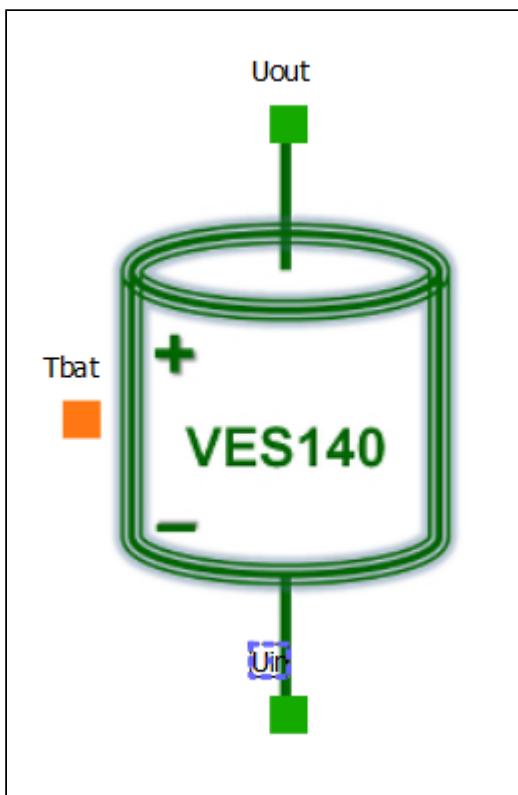
Management of connector label

Labels are displayed close to the connectors. Labels can not be modified.

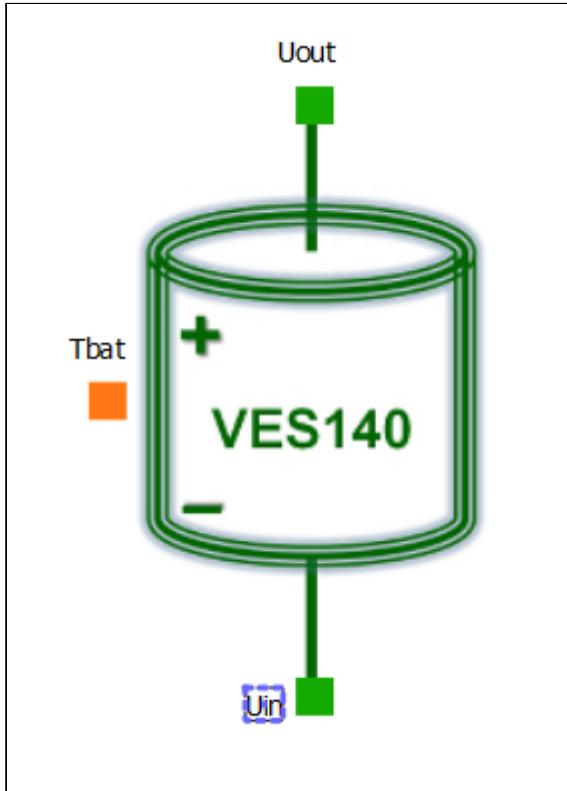


But label can be moved by selecting it and drag the mouse to the new position:

Before moving the label



After moving the label



The position of the label is not kept in the schematic file. That means if the user save the schematic file, close and reopen it, the labels will be located at their original positions.

Trajectory management

Trajectory management

Overview

Introduction

This chapter describes how to use SYSTEMA in order to create, load, save, display and run trajectories. It also describes how to display ground tracks for a given trajectory.

With SYSTEMA, you use the **Trajectory** tab to work on trajectories. Everything described in this chapter takes place within the **Trajectory** tab.

In this chapter

This chapter covers the following topics:

- Trajectory management
 - Overview
 - Introduction
 - In this chapter
 - The Trajectory tab
 - What is the Trajectory tab for?
 - The different parts of the Trajectory tab
 - Example
 - The menu bar
 - The viewer area
 - The icon bar
 - Trajectories in SYSTEMA
 - What is a trajectory?
 - The available arc types
 - The Ariane 5 GTO
 - The Planet stationary
 - The general keplerian
 - The Sun Synchronous
 - The Time Position Velocity
 - The trajectory in the browser window
 - The trajectory in the 3D views
 - The trajectory in the timeline window
 - Trajectory attributes
 - Celestial bodies
 - Variables
 - Arc attributes
 - Arc's orbit parameters
 - Arc time parameters
 - How to define the Start
 - How to define the End
 - Display options in the 3D views of the Trajectory tab
 - Right click menu in the 3D views
 - Characteristics of trajectory displays in the 3D views
 - Specific display options for viewing trajectories

- Zoom facilities
- Camera Lock in options
- Configuration window specific options
 - Display options
 - Video recording
- Interactive 3D animation tool
 - The set of animation buttons
 - The player speed
- Working with trajectories
 - General
 - Creating a new trajectory
 - Opening an existing trajectory
 - Loading a trajectory in a 3D view
 - Modifying a trajectory
- Building a trajectory
 - How to build a trajectory
 - Creating arcs
 - Organizing arcs
 - Adding arcs
 - How to add an arc with default values
 - To add an arc from the browser window or from the Trajectory menu:
 - To add an arc from the icon bar:
 - Result in the 3D views
- Viewing trajectory ground tracks
 - How to view ground tracks for a given trajectory
- Viewing trajectory textual information
 - How to view textual information for a given trajectory

The Trajectory tab

What is the Trajectory tab for?

You can use the **Trajectory** tab of the SYSTEMA application to :

- open existing trajectories and modify them when necessary,
- create new trajectories by combining arcs,
- test a trajectory by displaying a 3D animation in a 3D view,
- display ground tracks for a given trajectory.



*Note: In the **Trajectory** tab, you define and view a trajectory for a spacecraft. When a trajectory has been defined, you can then associate it with geometrical models and kinematics in order to define missions (refer to the "Mission management" chapter).*

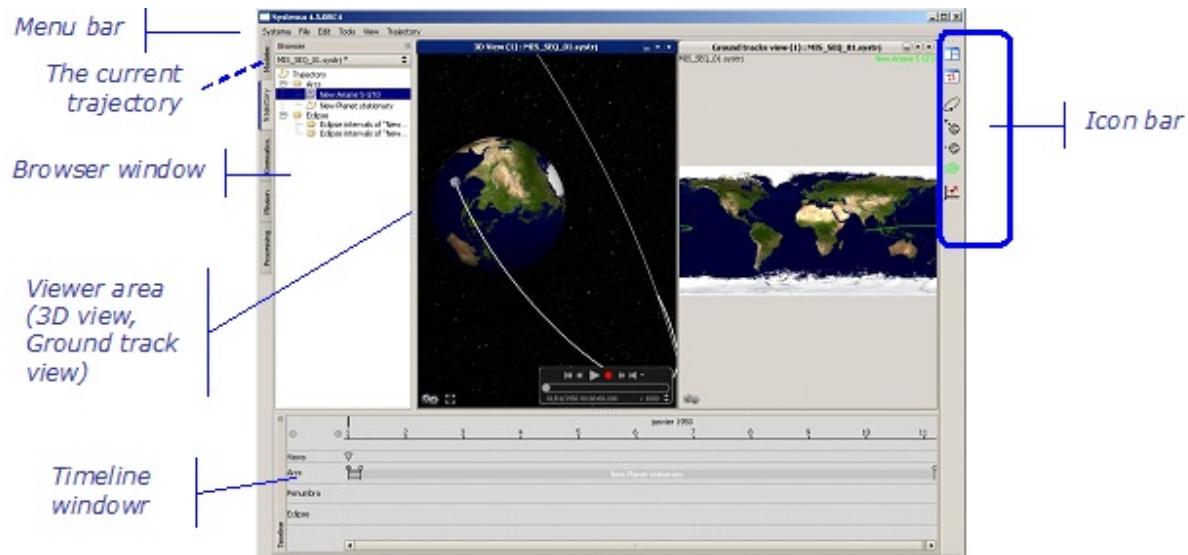
The different parts of the Trajectory tab

When trajectories have been opened in the **Trajectory** tab, information about them is provided in different areas:

- the browser window, containing the trajectory's hierarchical description,
- the viewer area, containing the viewports.
- the timeline window used to display all the components linked to the time such as orbital arcs, eclipse and penumbra intervals.

Example

The following diagram shows an example of the **Trajectory** tab. The viewer area contains two viewports, a 3D view with a trajectory and a Ground Track view:



The menu bar

The menu bar contains:

- The menus which are common to all SYSTEMA tabs: the **File**, **Edit**, **Tools**, **View** and **Help** menus.

For more information, refer to the "General User Interface" chapter

- One specific menu, the **Trajectory** menu, which can be used to add arcs to the current trajectory.

The **Trajectory** menu contains the **Create arc** option in which there are five available arc types. This option creates arcs with default attribute values that can be changed via the edition window.

The viewer area

The viewer area may contain two types of viewports, 3D views and *Ground Track* views. These are used to display the ground tracks of a spacecraft along a trajectory.

One or more viewports can be opened at the same time.

The following rules (which are general to all tabs) are applicable to 3D views and *Ground Track* views of the **Trajectory** tab:

- In a viewport, you can load any trajectory which has been opened.
- Several trajectories may be loaded at the same time in the same viewport.

The icon bar

The icon bar contains the standard icons to tile the 3D views using all the available space in the viewer area and to show the next 3D view

For more information, refer to the "General User Interface" chapter

The icon bar also contains icons that can be used to create arcs in the current trajectory by selecting the arc type directly, without using the Trajectory menu or the right click menu.

Trajectories in SYSTEMA

What is a trajectory?

A trajectory is a sequence of orbital arcs which represents the movement of a spacecraft in relation to celestial bodies.

Trajectories and arcs have properties (called *attributes* in SYSTEMA) that can be consulted or modified using the edition window.

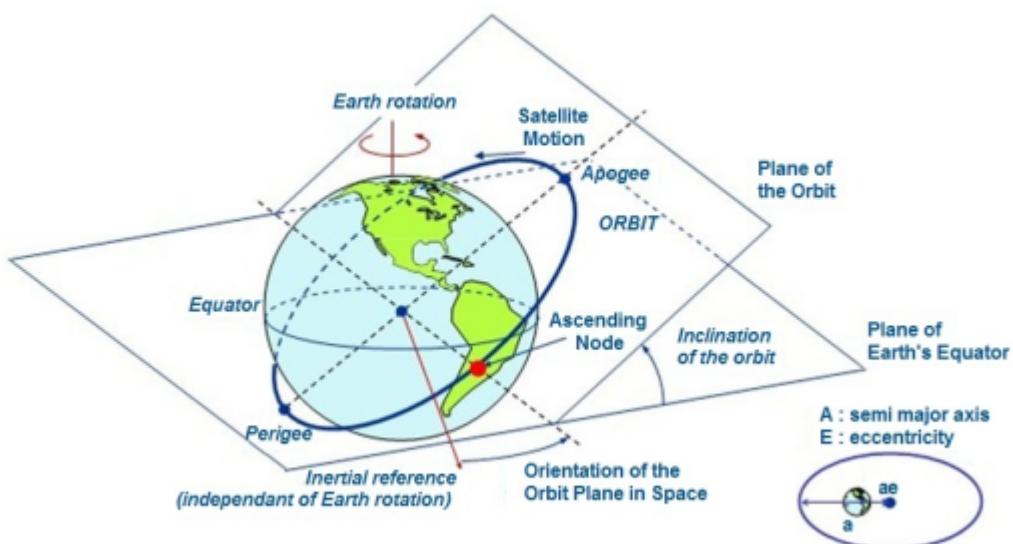
Each arc may be configured independently: **Continuity between arcs is not necessary either for spatial parameters or for time parameters.**

The available arc types

There are five available arcs types.

All types of arc's orbit are defined by

- an orbit reference which is a celestial body
- a begin and an end date,
- a reference date except for planet stationary arcs



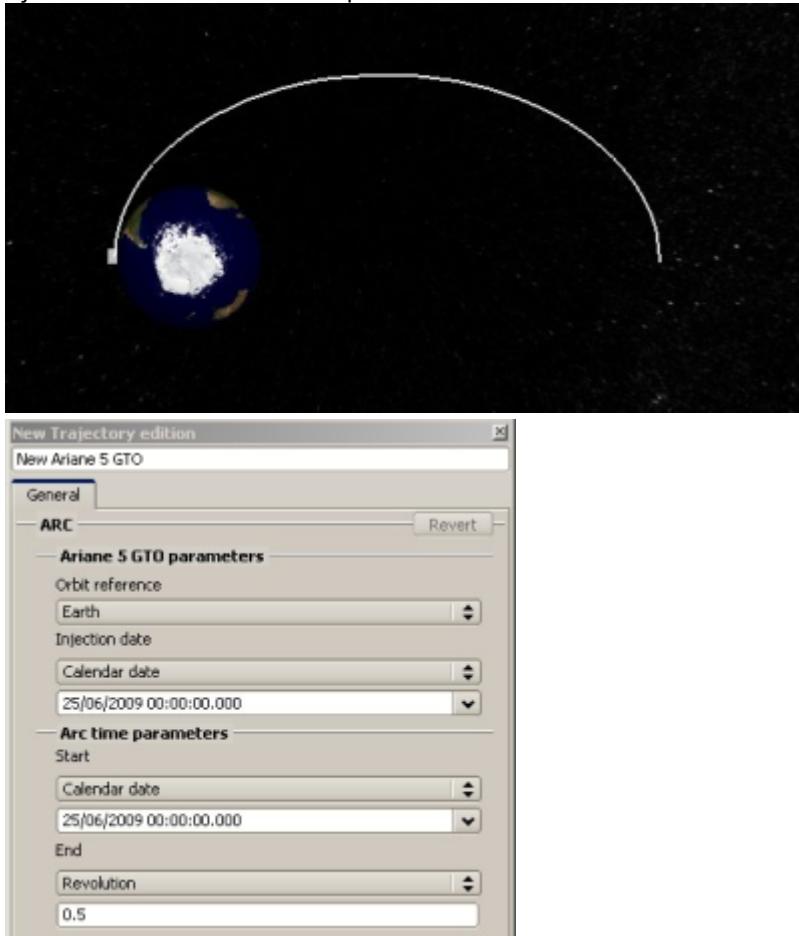
The following table presents the list of available arcs:

Arc type	Description
Ariane 5 GTO	Temporary orbit described by satellites launched by Ariane 5 and intended to be geostationary.
Planet stationary	This keplerian orbit is equatorial and circular. Its period of revolution is equal to the selected planet period of rotation.

Arc type	Description
General Keplerian	Orbit described by a body around a celestial body under the only action of their mutual gravitational attraction.
Sun synchronous	The plan of this keplerian orbit keeps permanently a constant angle with the direction Planet-Sun.
Time-Position-Velocity	Orbit around the Earth created from an ASCII external file

The Ariane 5 GTO

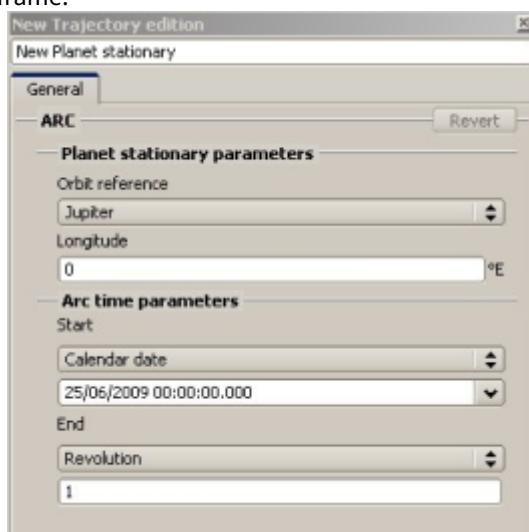
Temporary orbit described by satellites launched by Ariane 5 and intended to be geostationary. Only the Earth can be defined as orbit reference. It's a highly [elliptical](#) earth [orbit](#) with apogee at about 35,975 km and perigee at 202 km. This leads to a semi major axis $a = 24,466.644$ km and an excentricity $e = 0.731$. Because of the latitude of Kourou, the inclination of this orbit is 7° and the ascending node is 179.82° . The single parameter to define is the injection date which will correspond to the reference date of the orbit.



The Planet stationary

This keplerian orbit is equatorial (inclinaison = 0) and circular(excentricity = 0). Its period of revolution is equal to the selected planet period of rotation. Only Solar System planets can be defined as orbit reference (no Sun and no Moon). As the on ground track is a point and with reference to the defined celestial body constants, the single

parameter to define is the longitude in the planet frame.



The general keplerian

Orbit described by a body around a celestial body under the only action of their mutual gravitational attraction:
The specific parameters to define are :

- The semi major axis (a) : it is the half of the ellipse longest diameter, given in kilometres. If the orbit is circular, it corresponds to the radius of the circle
- The eccentricity (e) : it measures how much the orbit deviates from the circle. It is given by the formula

$$e = \sqrt{1 - \left(\frac{b}{a}\right)^2}$$

where a is the semi major axis and b is the semi minor axis. The eccentricity belongs to the interval [0 ; 1[, a null eccentricity means that the orbit is circular. There is no unit. Note that an eccentricity equal to 1 leads to a parabolic trajectories and an eccentricity superior to 1 leads to hyperbolic trajectories. Parabolic and hyperbolic trajectories are not treated with Systema. Be careful of the combination of the value a and e : Systema doesn't check if there is an intersection between the arc and the celestial bodies.

The inclination : it is the angle between the orbital plane and the equator plane of the celestial body, given in degrees. The inclination varies from 0° to 180° around the ascending node of the orbit

- The ascending node longitude (O) : it is the angle between the ascending node and the vernal point. The ascending and descending nodes are the two intersection points between the orbit and the equator plan. There is an ascending node (South to North) and a descending node (North to South). The vernal point is defined at a particular date (for example 01/01/ 2000 for J2000). The ascending node longitude belongs to the interval [-180°; 180°].
- The argument of periapsis : it is the angle between the ascending node and the periapsis. It belongs to the interval [-180°; 180°]. The periapsis is the point of closest approach to the celestial body. The periapsis could be named perigee if the celestial body is the Earth and perihelion if the celestial body is the Sun.
- The anomalies
 - True anomaly : it is the angle between the direction of periapsis and the current position of the body, as seen from the main focus of the ellipse. The radius (distance from the focus of attraction and

$$r = \frac{a(1-e^2)}{1+e \cdot \cos(v)}$$

the orbiting body) is related to the true anomaly by the formula :

- Mean anomaly (M): it is not a geometric angle but a fictive one. It is defined as the time since the

$$M = \frac{2\pi}{T}(t - t_p)$$

last periapsis (t_p) times $2p / T$, where T is the orbital period:

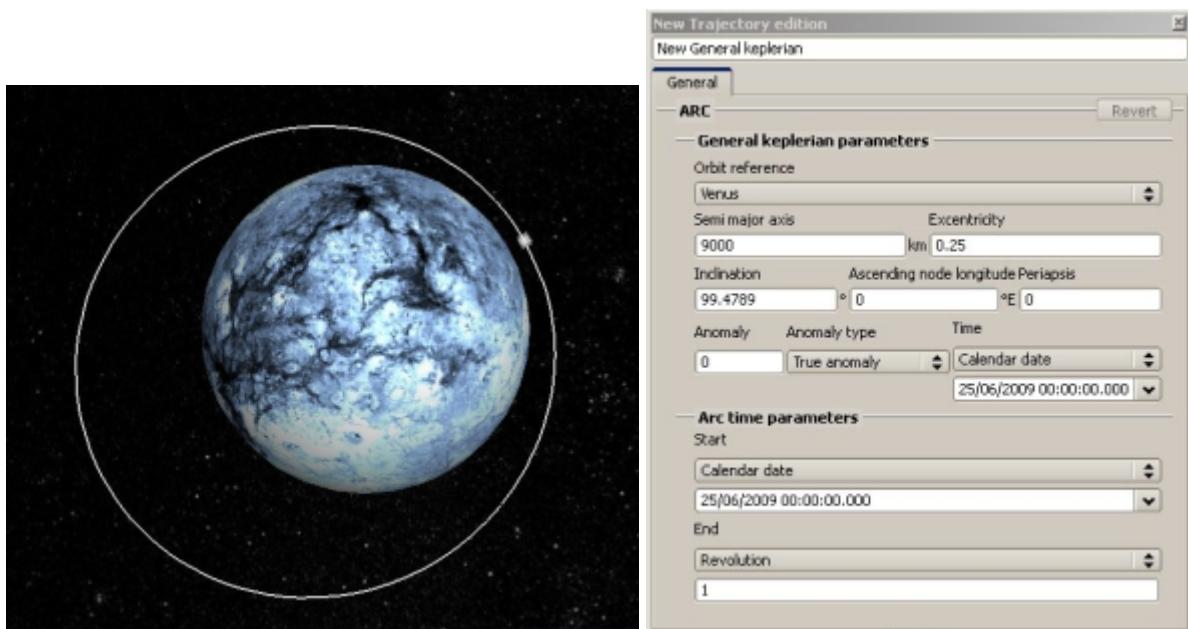
- Eccentric anomaly (E): it is the angle between the direction of periapsis and the vector CS' (with C centre of the ellipse and S' the projection of the point on the major circle of the ellipse). The relation

between eccentric anomaly and true anomaly is defined by the Kepler equation:

$$M = E - e \cdot \sin(E)$$

- Orbital position (PSO): it is the angle between the direction of ascending node and the current position of the satellite. If the argument of periapsis is 0, true anomaly and orbital position are equal.

The PSO is linked to the true anomaly by : $PSO = v + \omega$



The orbital period computation of a Keplerian arc takes into account the gravitational perturbations linked to the Earth flattening (also known as the J2 influence).

As a result, the orbital period values given by Systema are slightly different from the ones given by the Kepler law.

The formula is use are given here:

$$T = \frac{2\pi}{n} \quad , \quad n = \sqrt{\frac{\mu}{a^3}} (1 + \varepsilon) \quad , \quad \varepsilon = 3J_2 R_e^2 \frac{3\cos^2(i)-1}{4a^2(1-e^2)^{1.5}}$$

- T (s): orbital period
- μ ($km^3 \cdot s^2$): planet standard gravitational parameter
- a (km): orbit semi-major axis
- e : orbit eccentricity
- i (rad): orbit inclination
- R_e (km): planet radius
- J_2 : second spherical harmonic of the planet gravitational field (linked to planet flattening)

For example, we get $T = 6310$ s for an orbit around the Earth with the following parameters:

- $\mu_{Earth} = 398601.3 \text{ km}^3 \cdot \text{s}^2$
- $a = 7378.14 \text{ km}$
- $e = 0.01$
- $i = 1.309 \text{ rad } (75^\circ)$
- $R_e = 6378.144 \text{ km}$
- $J_2 = 0.00108265$

The same calculation with the Kepler law ($\varepsilon = 0$) would provide us $T = 6307$ s instead.

The Sun Synchronous

The plan of this keplerian orbit keeps permanently a constant angle with the direction Planet-Sun. Only the Earth can be defined as orbit reference. The parameters to define are:

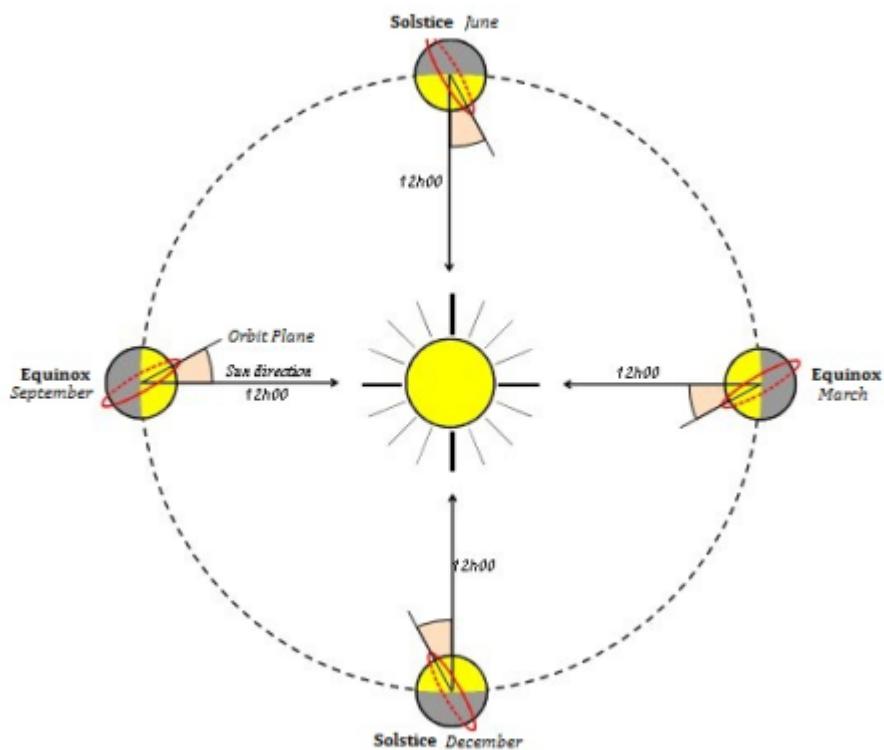
- the mean local solar time : it allows to define the angle between the nodes line and the mean direction Sun / Earth. Notice that this angle is constant because of the sun synchronous conditions. The mean local solar time being considered, rather than the true one, the equation of time is automatically taken into account with its 2 components - obliquity of the planet ecliptic and the eccentricity of the planet orbit.
- either the altitude. Because of the sun synchronous character of the orbit, the semi major axis (altitude + planet equatorial radius (6378.144 km for Earth)) is linked to inclination by the formula :

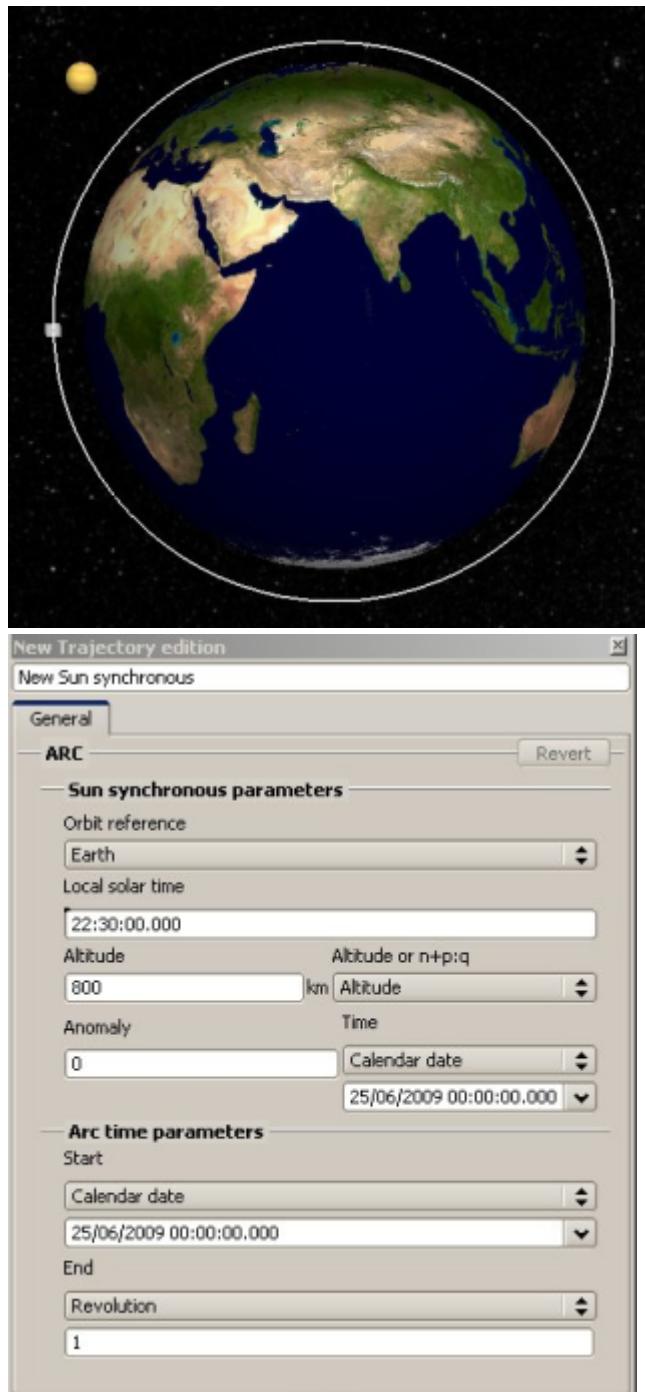
$$-\frac{3}{2} \cdot n \left(\frac{R_p}{a} \right)^2 \cdot J_2 \cdot \frac{\cos(i)}{(1-e^2)^3} = 0,985$$

. In Systema, the sun synchronous orbit are circular so the eccentricity is null and the relation between semi major axis and inclination is :

$$-9,97 \left(\frac{R_p}{a} \right)^{\frac{3}{2}} \cos(i) = 0,985$$

- either the N, P, Q values. For a sun synchronous orbit, the number of orbits in one day is equal to N + P/Q. N is the number of complete revolutions per day, Q is the cycle duration in days and P is the track step (also called under cycle). Remember that a cycle of Q days means that the tracks are repeated after Q days.
- the anomaly : here it is the true anomaly which should be given. Notice that the three anomalies are strictly the same if the orbit is circular (and sun synchronous orbits are circular under Systema)

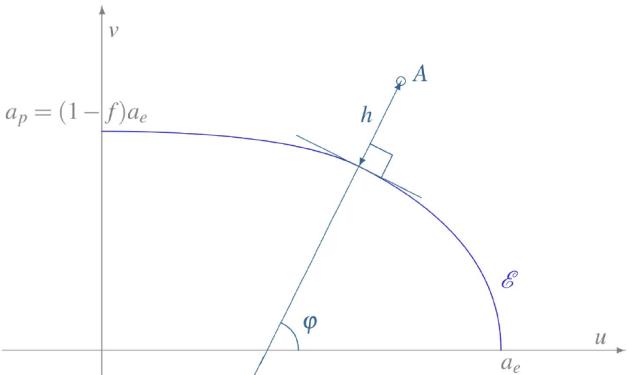




The Time Position Velocity

This type of arc allows importing orbits from file which data format is described in the Data format panel of Systema's orbit properties edition window. ASCII CSV files representing dates and position in xyz or longitude, latitude and altitude can be imported and interpreted in regard to data format have been selected. An orbit is created by importing a file and defining what data format to use to parse and handle the informations. The supported format are :

File type	Description
STK ephemeris file (.e)	<p>Interface to and from the STK software. It consists in a header describing the content of the file (stk version, number of points, epoch, coordinate system and type of data : EphemerisTimePos).</p> <p>Then data array with number of seconds till epoch as first column (double) followed by the position in meter on X Y and Z. If epoch is not defined (with ScenarioEpoch), epoch is set to 01/01/1970 00:00:00.</p> <p>Since Systema 4.9.2, CoordinateSystem field is taken into account as follow :</p> <ul style="list-style-type: none"> • B1950 → Veis1950 • J2000 → J2000 (earth centered frame in Systema). Note that in STK, it is can be centered on an other body defined by CentralBody field. This is ignored • ICRF → ICRF (solar system barycenter centered frame in Systema). For STK it is can be centered on an other body defined by CentralBody field. This is ignored • Other values or no header == Planet inertial (Veis1950 for the Earth. See "Solar system frames" in https://www.orekit.org/site-orekit-11.1.2/architecture/frames.html for more details) <p>CentralBody field is ignored by Systema.</p> <p>See https://help.agi.com/stk/index.htm#stk/importfiles-02.htm for more details.</p> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <pre>Mandatory information in the header: : stk.v.9.0 BEGIN Ephemeris NumberOfEphemerisPoints 1786 ScenarioEpoch 01 Jan 2000 0:0:0.000000 CoordinateSystem B1950 EphemerisTimePos Then the data: - column 1 : number of seconds till Epoch if defined otherwise till 01/01/1970 00:00:00 (1 double) - column 2-4 : position in meters (3 doubles) END Ephemeris</pre> </div>
CNES CIC based OEM ephemeris file (.txt)	<p>This file is based on the CCSDS norm and is dedicated to CNES interfaces. The OEM file consists of a header, a part describing the content of the file and the data as an array of doubles:</p> <p>Date in modified julian days followed by seconds of the day, position (x y z) in km and the optional velocity values in km/sec.</p> <p>The time reference used by those file is 1858/11/17@00:00:00. To convert to Gamma50 epoch, you need to withdraw 33282 days and to convert the second file column from seconds to day fraction.</p> <p>The center planet is read in the header of the file (ex: CENTER_NAME = EARTH_2020)</p>

File type	Description
	<p>The reference frame is read in the header with REF_FRAME keyword :</p> <ul style="list-style-type: none"> • EME2000 → J2000 • ICRF → ICRF • Other → Planet Inertial (Veis1950 for the Earth)
Other types of file	<p>Any file with known structure can be imported by one of the other data format:</p> <ul style="list-style-type: none"> ▪ Cartesian position and velocity in Veis1950 ("Cartesian position and velocity in planet inertial reference Gamma50" before Systema-4.9.2) : date followed by positions (km) and speed (km/s) in Veis1950 reference frame (inertial and earth centered frame) ▪ Cartesian position and velocity in J2000 ("Cartesian position and velocity in planet inertial reference J2000" before Systema-4.9.2) : date followed by positions (km) and speed (km/s) in J2000 reference frame (inertial and earth centered frame) ▪ Cartesian position and velocity in ICRF : date followed by positions (km) and speed (km/s) in ICRF (inertial and solar system barycenter centered frame) ▪ Cartesian position and velocity in planet inertial : date followed by positions (km) and speed (km/s) in the planet reference frame (Veis1950 for the Earth.) ▪ East longitude, latitude and altitude (spherical coordinates) in planet rotational reference : date then east longitude in degree, latitude in degree and altitude (km) from the ground level. Note that the Systema definition is planetocentric (not planetographic), meaning that all planets are considered as spherical (no flattening). <p>As a result, we recommend to the users that request a high precision to convert their geodesic data to cartesian coordinates before importing it into Systema. This can be done by using the following formulae:</p>  <p>Illustration of geodesic latitude (a_e (equatorial radius) = 6378137.0m, a_p = pole radius, f (Earth flattening parameter) = 1.0 / 298.257223563)</p>

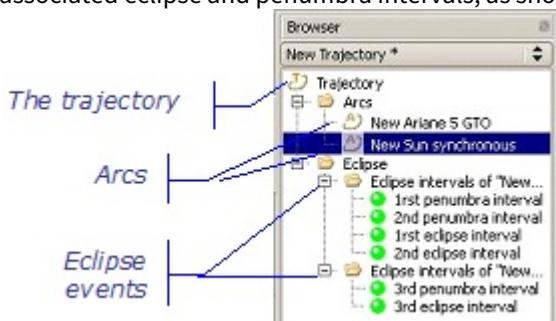
File type	Description
	$u = \left(\frac{a_e}{\sqrt{1-f(2-f)\sin^2(\varphi)}} + h \right) \cos(\varphi)$ $v = \left(\frac{(1-f)_e^{2a}}{\sqrt{1-f(2-f)\sin^2(\varphi)}} + h \right) \sin(\varphi)$ <p>The (x,y,z) coordinates can be computed from the (u, v) coordinates as follows (Phi = latitude, Lambda = longitude, h = altitude):</p> $\begin{cases} x = u \cos(\lambda) \\ y = u \sin(\lambda) \\ z = v \end{cases}$ <p>Moreover, the celestial radius in use is the equatorial radius (not the mean radius).</p> <ul style="list-style-type: none"> • Cartesian position and velocity in a planet inertial reference (frozen planet rotational reference) : date, position (km) and velocity (km/s) in the planet rotation frame defined at the given longitude and latitude.



For more details about frames and date formats, please refer to [Trajectory technical annex](#).

The trajectory in the browser window

The browser window displays the hierarchical organization of the trajectories arranged by successive arcs and the associated eclipse and penumbra intervals, as shown in the example below:



Trajectories are symbolized by the icon followed by the name (example **Trajectory**).

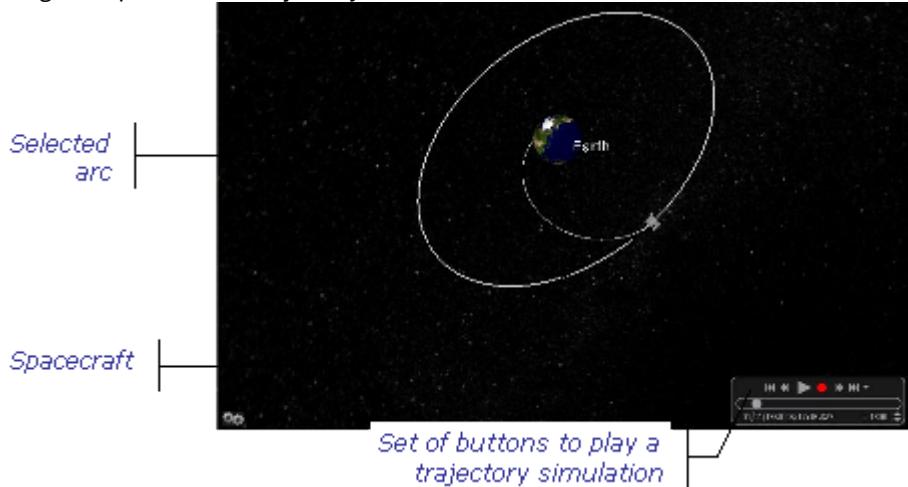
Arches are symbolized by the icon followed by the name (example **New orbit**).



Note: The color of the arc icon's background indicates if there are any warnings (orange) or errors (red) associated with the corresponding arc.

The trajectory in the 3D views

The 3D views show the trajectory in a three-dimensional form. They are used to view and run the trajectory. The following example shows a trajectory in a 3D view:

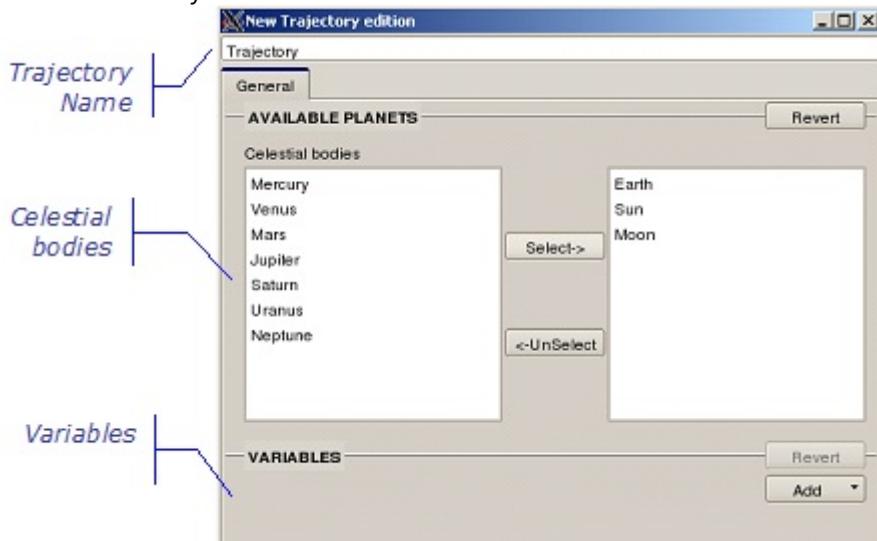


The trajectory in the timeline window

Refer to "Working with the timeline window" in the "General User Interface" chapter to get the description of the timeline window.

Trajectory attributes

At root level, the trajectory has three attributes: the trajectory name, the celestial bodies and the Variables attribute. You modify them via the edition window:



Celestial bodies

Each trajectory is associated with the celestial bodies. Each of them may then be chosen as the orbit reference for arcs in the trajectory.

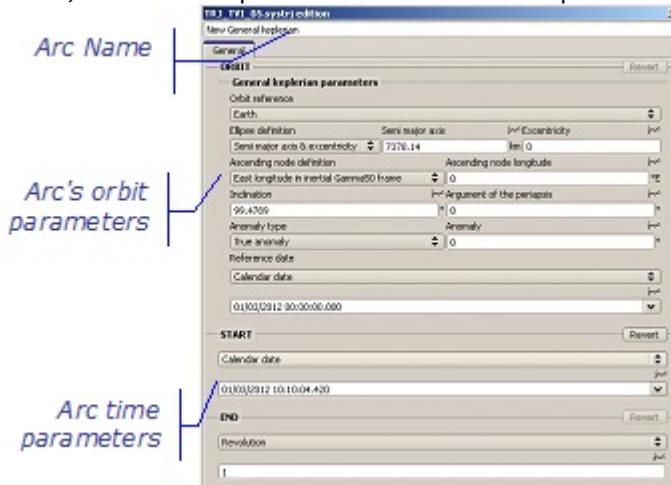
The available celestial bodies are the Sun, the Moon and the planets: Earth, Neptune, Uranus, Saturn, Jupiter, Mars, Venus, Mercury

Variables

Cf [Working with variables](#)

Arc attributes

Arcs have three main attributes: the arc name, the arc's orbit parameters and the arc time parameters. You modify



them via the edition window:

Arc's orbit parameters

Each arc's orbit is defined by:

- an **Orbit reference**, which can be one of the celestial bodies associated with the trajectory,
- a set of additional parameters which define the orbit form and are specific to each type of arc.

Arc time parameters

Each arc has two time parameters:

- **Start**, the date on which the spacecraft begins to follow the arc,
- **End**, the date on which the spacecraft leaves the arc.

The default value for the **Start** date is the **End** of the previous arc in the trajectory.

	<p><i>Note: In some cases, warnings are displayed in the Information area, for example,</i></p> <ul style="list-style-type: none"> • <i>when Start and End values are identical,</i> • <i>when arcs are partly superimposed.</i>
--	--

How to define the Start

Refer to "Attributes of a date" in "Working with the timeline window" in the "General User Interface" chapter to get the definition of the date attributes.

How to define the End

Refer to "Attributes of a date" in "Working with the timeline window" in the "General User Interface" chapter to get the definition of the date attributes.

Display options in the 3D views of the Trajectory tab

Right click menu in the 3D views

Within the 3D views, the right click menu contains various display options.

Some of them are options common to all the 3D views of every SYSTEMA tab:

- **Hide, Hide others, Hide all** and **Show all** options concern either the selected arcs or all arcs;

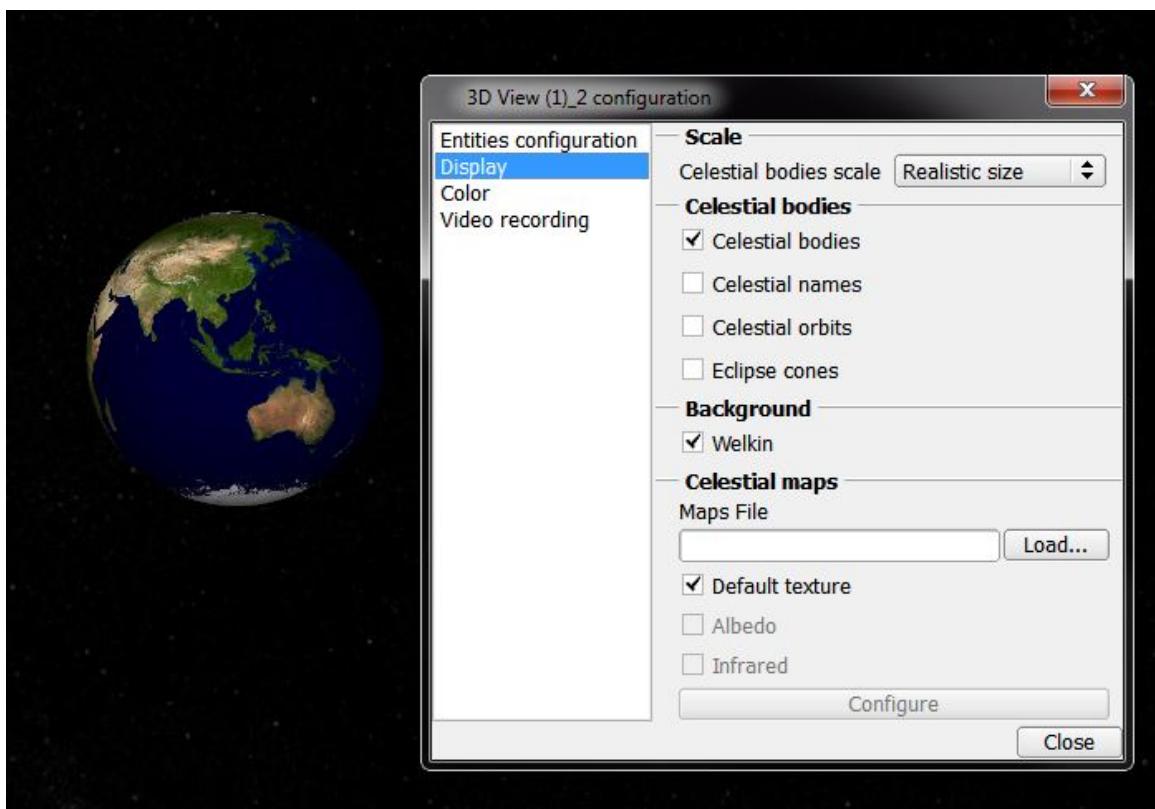
	<i>Note: Hide / show actions are reflected in all 3D views in which the trajectory is loaded, and also in the browser window.</i>
--	---

- the **Configuration window** option.

	<i>Note: This action is local to the viewport (several trajectories may be loaded in a viewport)..</i>
--	--

For more information about these options, refer to the "General User Interface" chapter.

- A section of the configuration window is dedicated to **Celestial maps**. You can select a maps file that contains Albedo, IR flux and Temperature for celestial bodies. If you load a new maps file, all the previous loaded data are replaced.

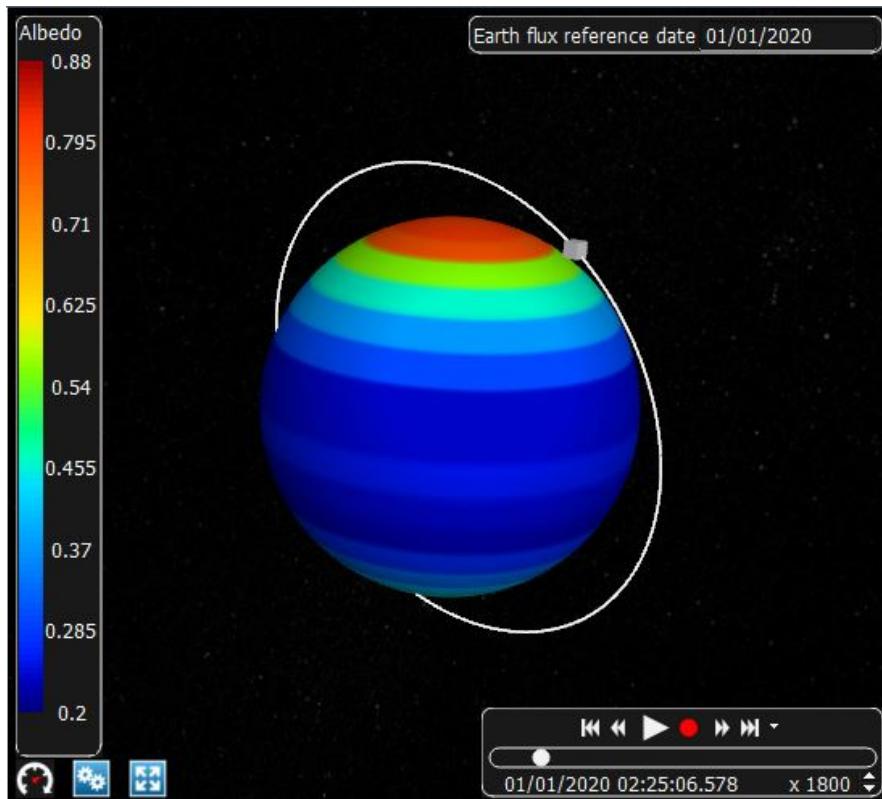


Please refer to the user manual of Thermica for more details about maps files.

Once a valid maps file is loaded, you can select the display mode of the celestial bodies :

- The default texture of a planet (default mode)
- The albedo values (if available in the maps file)
- The Infrared temperatures values (if IR fluxes or temperatures are available in the maps file).

The date when the values (albedo or IR flux) are provided in the maps file can be displayed with the "Planet flux reference date" textual information.



! The "SUN_ORIENTED" frame is not yet taken into account by the 3D view. This limitation does not affect the computations performed by Thermica.

Characteristics of trajectory displays in the 3D views

In the 3D views of the **Trajectory** tab, two different aspects of the available scenes are not realistic: the **Configuration window** option.

- Light comes from the SYSTEMA user observing the scene, whenever the scene orientation is changed, and does not represent solar light (this is not the case for the **Mission** tab, which does show realistic lighting).
- The scale used to draw the sun is different from the scale used for the other bodies. The sun is shown on a scale of 1. The other planets and their orbits are drawn on a scale of 250 (they are 250 times greater than in reality).

Specific display options for viewing trajectories



Note: All these actions are local to the 3D view.

Zoom facilities

Option	Use
Fit - Selected	<p>Adjusts the zoom in order to fit entirely the selected arc in the 3D view:</p> 
Fit -- Loaded Trajectory	<p>Adjusts the zoom in order to fit entirely the imported trajectory (whose name is "Loaded Trajectory") in the 3D view:</p> 
Fit - Celestial Body	<p>Allows you to select a celestial body in order to fit it entirely in the 3D view</p>
Fit All	<p>Adjusts the zoom in order to fit entirely the entire scene (arcs and celestial bodies) in the 3D view</p>

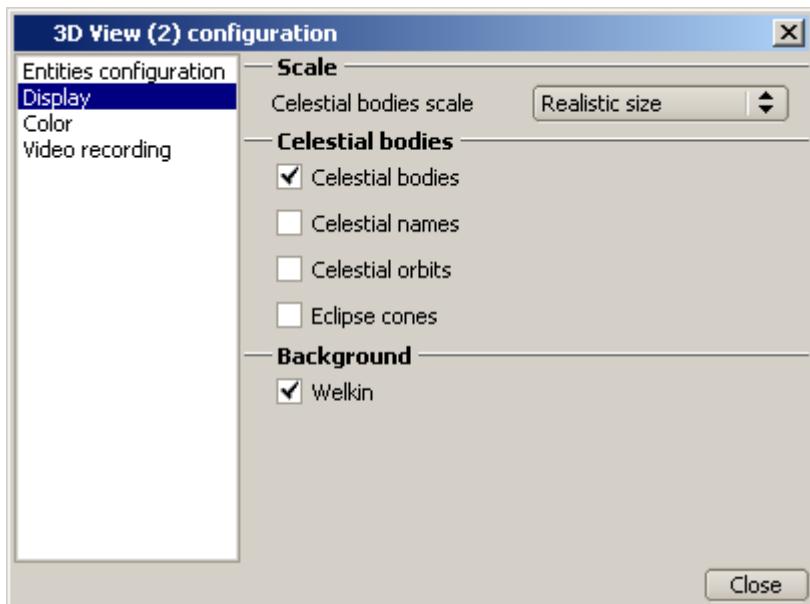
Camera Lock in options

SYSTEMA allows you to choose the referential frame to be used when positioning camera:

Option	Use

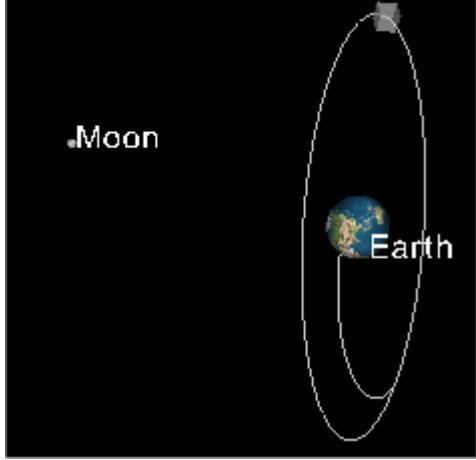
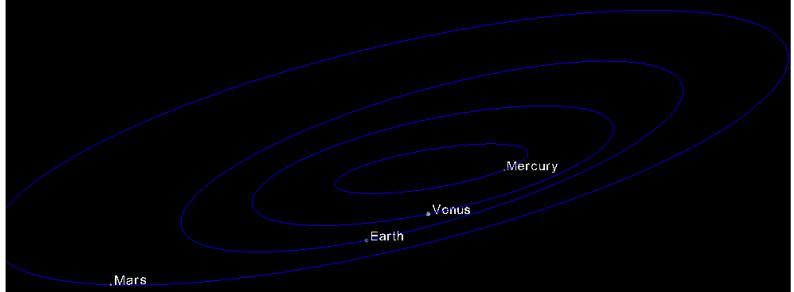
Planet inertial frame	Allows you to select a celestial body to display the trajectory in the corresponding <i>inertial system</i> . In the displayed scene, the camera is fixed in the planet inertial frame.
Planet rotational frame	Allows you to select a celestial body to display the trajectory in the corresponding <i>rotational system</i> . In the displayed scene, the camera is fixed in the planet rotational frame.
Local orbital frame	Allows to fix the camera in the local orbital frame of the trajectory.

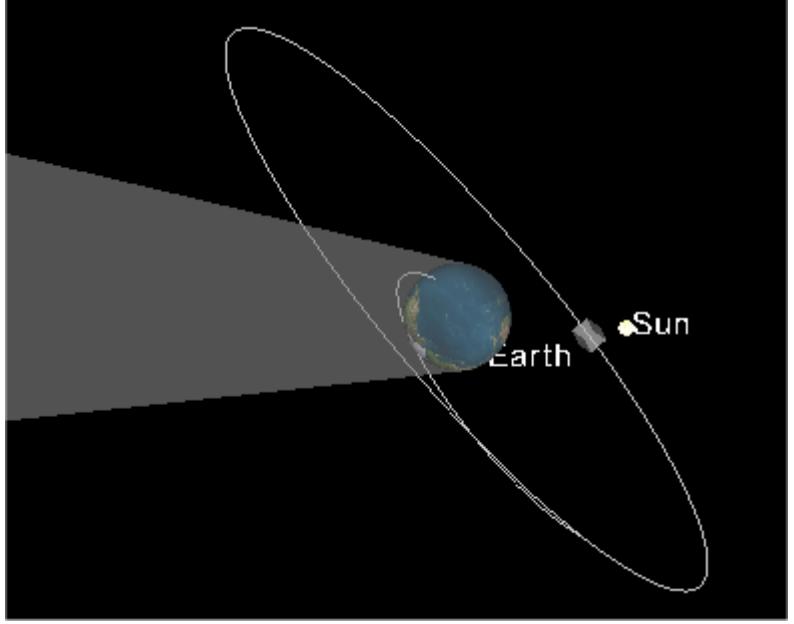
Configuration window specific options



Display options

Option	Use
Celestial bodies	By default, the box is checked (celestial bodies are shown) Uncheck the box to hide the celestial bodies associated with the trajectory in order to show only the trajectory's arcs Check the box to show the celestial bodies associated with the trajectory.
Celestial names	By default, the box is unchecked (celestial names are hidden) Check the box to display the name of every celestial body associated with the trajectory:

Option	Use
	 <p>Uncheck the box to hide the name of every celestial body associated with the trajectory</p>
Celestial orbits	<p>By default, the box is unchecked (celestial orbits are hidden) Check the box to show the celestial orbits (in blue) of every celestial body associated with the trajectory. Uncheck the box to hide the orbits of every celestial body associated with the trajectory:</p>  <div data-bbox="635 1381 1421 1493" style="border: 2px solid red; padding: 10px;"> <p>⚠ Computing the orbits of every celestial bodies could be quite long the first time this option is activated</p> </div>
Eclipse cones	<p>By default, the box is unchecked (eclipse cones are hidden) Check the box to show the eclipse cone:</p>

Option	Use
	 <p>This option can be used to see which elements are in shadow and when.</p>

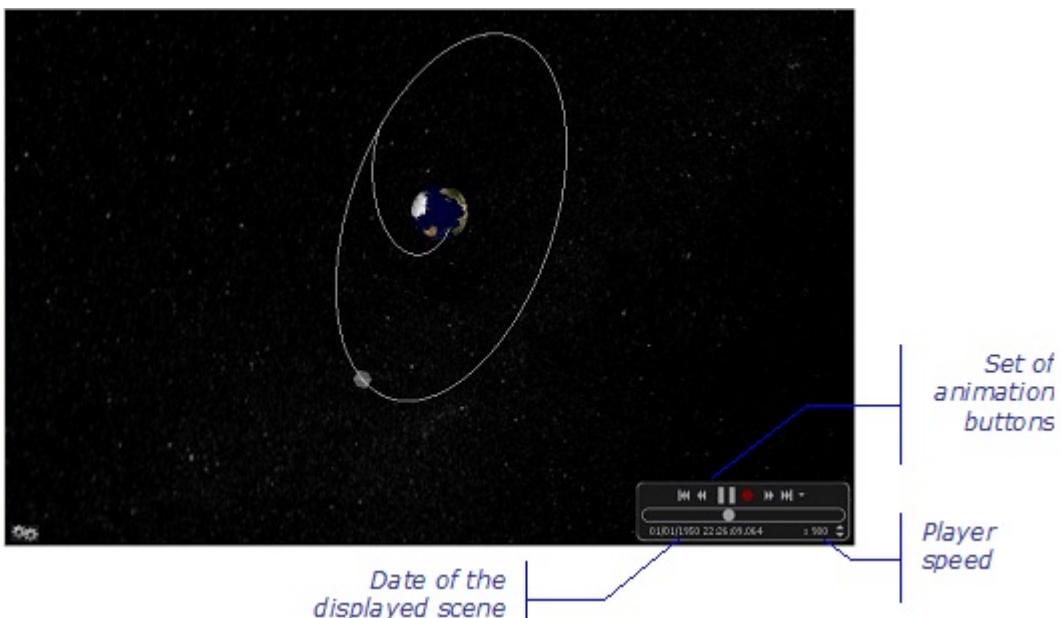
Video recording

To record a video, refer to the chapter "General User Interface", § Recording a video.

Interactive 3D animation tool

In the 3D views, you can interactively display an animation of the trajectory: a spacecraft is displayed as a grey cube moving along the trajectory according to the defined dates.

At the bottom of the 3D view, a set of buttons allows to the animation to be run:



The set of animation buttons

The animation buttons are described in the following table:

Button	Use
	Starts running (or re-starts) the animation
	Stops running the animation (just pauses it, time and position are kept)
	Starts recording video
	Moves the spacecraft to the previous event (arc, penumbra or eclipse)
	Moves the spacecraft to the start date of the first arc
	Moves the spacecraft to the next event (arc, penumbra or eclipse)
	Moves the spacecraft to the end date of the last arc
	Allows to select the type of event ((arc, penumbra or eclipse))
	Allows to select how to display the time (absolute time in calendar date, elapsed time in sec)

The player speed

To modify the player speed :

1. To increase the player speed, click .
2. To decrease the player speed, click .

Working with trajectories

General

When working with a trajectory in the **Trajectory** tab, you can open an existing trajectory or create a new one. When a trajectory is opened, you can:

- load it in any 3D view,
- modify it,

- display an animation of the trajectory,
- view ground tracks.

Creating a new trajectory

To create a new trajectory:

- In the **File** menu of the **Trajectory** tab, select the **New** option.

Result: In the viewer area, a new 3D view is opened containing the default space environment (the *Earth* is the default celestial body).

You can then build the trajectory and save it.

Opening an existing trajectory

You can open a trajectory which has previously been saved within a ".systrj" file.

To edit an existing trajectory:

- In the **File** menu of the **Trajectory** tab, select the **Open** option and select the file containing the required trajectory.

Result: In the viewer area, a new 3D view is opened with the trajectory displayed in it.

In the browser window, this trajectory's hierarchical organization is displayed. The previously edited trajectories can be accessed from the list of trajectories.

You can then modify the trajectory and save your modifications.

Loading a trajectory in a 3D view

In the 3D views, you can load trajectories which have been opened.

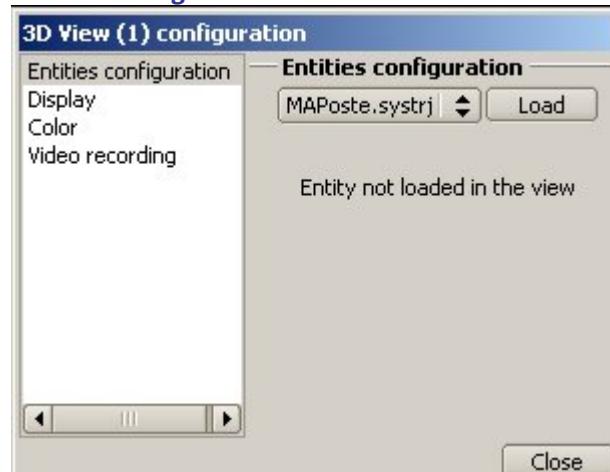


To load a trajectory in a 3D view:

- In the 3D view, in the right click menu:

In the browser window, select a component (from which the new component will be created).

1. Select Configuration window.



2. Select the trajectory to load.

3. Click **Load.**

Result: The selected trajectory is then displayed in the 3D view.

Modifying a trajectory

You can modify a trajectory by:

- creating new arcs (refer to the "Building a trajectory" section),
- editing and modifying existing arcs.

Building a trajectory

How to build a trajectory

You build a trajectory by creating and organizing arcs.

Creating arcs

There are three different ways to create arcs:

- using the browser window,
- or using the **Trajectory** menu,
- or using the arc icons in the icon bar.

(Details are provided in the section below).

Organizing arcs

To organize arcs, you have to edit the information window for each one and enter the appropriate parameters.

Adding arcs

Arcs are created in the current trajectory (the one displayed in the browser window).

A new arc is created with default values. Its attributes (name, position and dates) can then be modified by editing it.

How to add an arc with default values

To add an arc from the browser window or from the Trajectory menu:

1. In the browser window, right click on the arc after which the new one is to be added.
2. In the right click menu or in the **Trajectory** menu, select **Create arc** and then the arc type.
Result: The new arc is created. In the browser window, it is placed in the selected position. The corresponding edition window is displayed.

To add an arc from the icon bar:

You can also create arcs using the icons in the icon bar (there is one icon per arc type).
The arc icons are described in the following table:

Icon	Arc type
	General Keplerian

Icon	Arc type
	Sun synchronous
	Planet stationary
	Ariane 5 GTO
	Time-Position-Velocity

Result in the 3D views

When the arc has been created, it is displayed in each 3D view in which the trajectory is loaded.

Viewing trajectory ground tracks

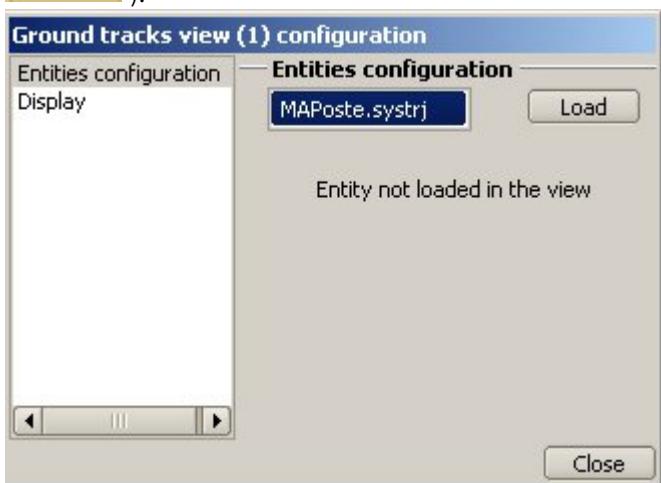
You can display ground tracks for every trajectory opened in the **Trajectory** tab.



Note: You can display ground tracks for several trajectories at the same time.

How to view ground tracks for a given trajectory

1. Open the trajectory whose ground tracks you wish to view.
2. Open a viewport of a *Ground tracks* view type (In the **Viewport** menu, select **Add - Ground tracks view**).
3. From the *Ground Tracks* view viewport, load the trajectory (In the right click menu, select the trajectory and select **Load**).



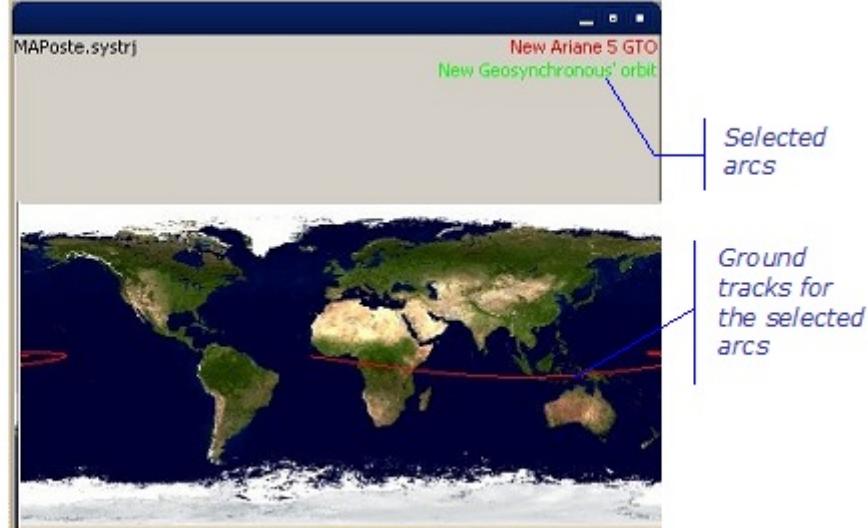
4. From the browser window, select the arc you want to display the ground tracks for.

Result: In the *Ground tracks* view viewport, you can see tracks for the selected arc.

You can select several arcs. A track is displayed for each one. If no arcs are selected, the whole trajectory is displayed.

- (i) If the selected arcs use different celestial bodies, an error is displayed.

Example: The following diagram contains ground tracks for two arcs (from one trajectory).

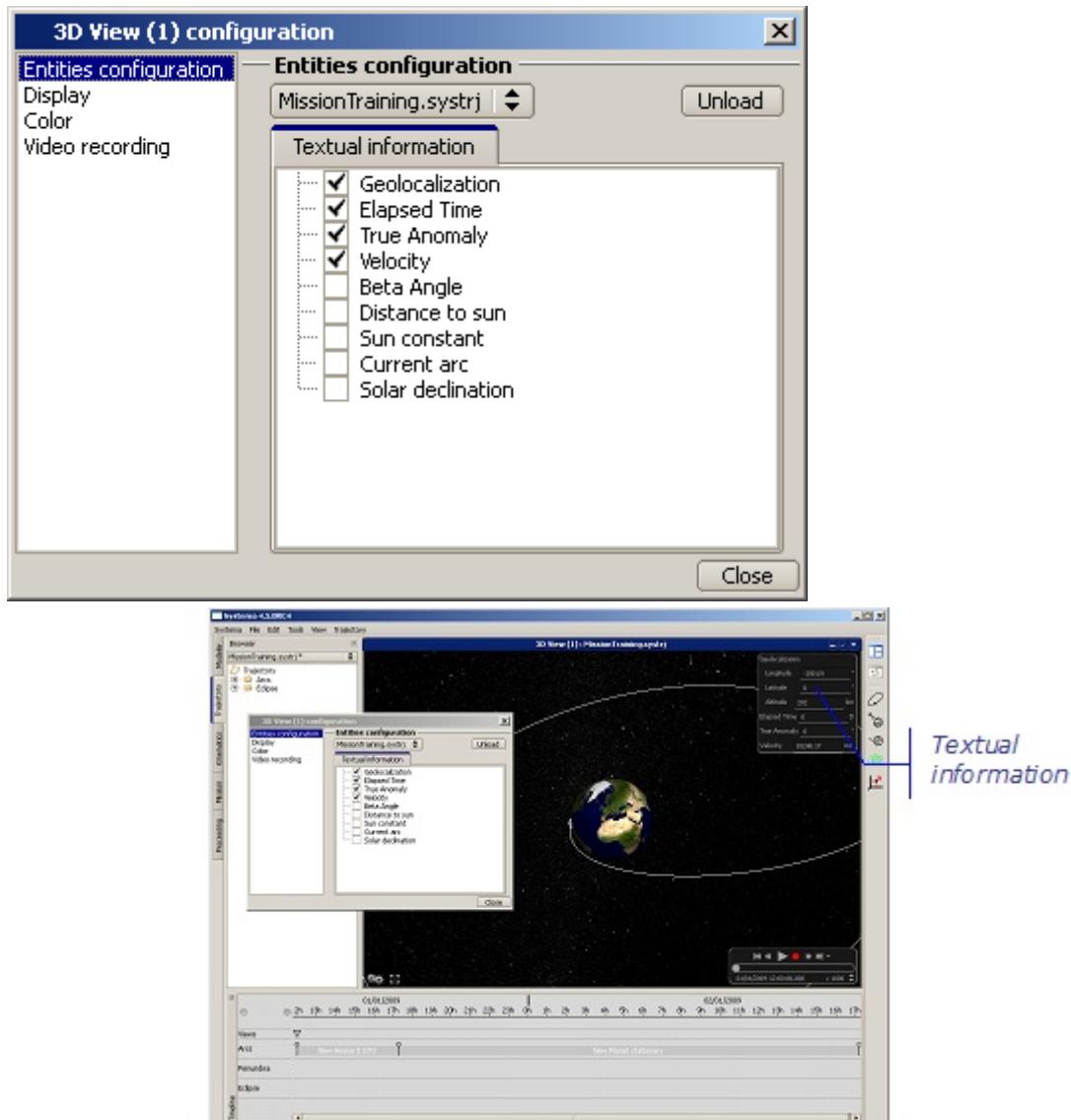


Viewing trajectory textual information

You can display textual information for every trajectory opened in the **Trajectory** tab.

How to view textual information for a given trajectory

1. Open the trajectory whose textual information you wish to view. A 3D view is automatically created
2. Open the configuration window. A list named textual information is displayed in 'Entities configuration' option. Check any information to display it in the 3D viewport. It is immediately added. Uncheck it to remove it from the display



⚠ The "Sun direction" textual information is given from the planet reference frame (not from the satellite frame)

⚠ Since Systema 4.9.0, the planet velocity vector is computed in the Sun inertial frame whereas it was computed in the ecliptic plane before.



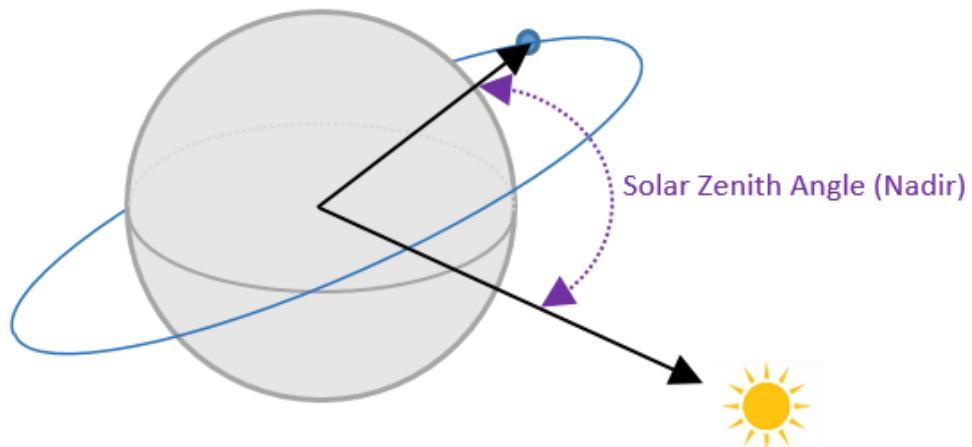
Systema-4.9.0 - Earth velocity 17/08/2020
00:00:00.000



Systema-4.8.3 - Earth velocity 17/08/2020
00:00:00.000

Please note that the norm of both vectors is similar.

Since Systema-4.9.2, the Solar Zenith Angle is available. It is the angle between the nadir direction and the sun direction :



In the Systema Trajectory tab, it is possible to display the current beta angle of the orbit. In the case of orbits defined from file, the value of this beta angle is valid only if the input file contains the satellite velocity. The beta angle calculation has indeed resort to the cross-product of the position and velocity vectors.

Kinematics management

Kinematics management

Overview

Introduction

This chapter describes how to use SYSTEMA in order to create, load, save, display and run kinematic models. With SYSTEMA, you use the **Kinematics** tab to work on kinematic models. Everything described in this chapter takes place within the **Kinematics** tab.

In this chapter

This chapter covers the following topics:

- Kinematics management
 - Overview
 - Introduction
 - In this chapter
 - The Kinematics tab
 - What is the Kinematics tab for?
 - The different parts of the Kinematics tab
 - Example
 - The menu bar
 - The viewer area
 - The icon bar
 - The location bar
 - Kinematic models in SYSTEMA
 - What is a kinematic model?
 - The kinematic model in the browser window
 - The kinematic model in the 3D views
 - The SYSTEMA kinematic bodies
 - The phases in SYSTEMA
 - The kinematic laws in SYSTEMA
 - Transformation defined in a file
 - Kinematic model root attributes
 - The kinematic law attribute
 - Variables
 - Kinematic body attributes
 - The SYSTEMA kinematic body connection attribute
 - The kinematic body origin attribute
 - The kinematic law attribute
 - Display options in the 3D views of the Kinematics tab
 - Right click menu in the 3D views
 - Specific display options for viewing kinematic models
 - Zoom facilities
 - Configuration window specific options
 - Display option
 - Video recording
 - Interactive 3D animation tool

- Running a kinematic simulation
- Example of kinematic model animation
- Working with kinematic models
 - General
 - Creating a new kinematic model
 - Opening an existing kinematic model
 - Loading a kinematic model in a 3D view
 - Modifying a kinematic model
- Building a kinematic model
 - How to build a kinematic model
 - Creating kinematic bodies
 - Organizing kinematic bodies
 - Adding/Removing phases
 - Adding kinematic bodies
 - How to add a kinematic body with default values
 - To add a kinematic body from the browser window:
 - To add a kinematic body from the Kinematics menu:
 - Result in the 3D views
- Exporting transformations
 - How to export transformations for a given kinematic body

The Kinematics tab

What is the Kinematics tab for?

You can use the **Kinematics** tab of the SYSTEMA application to:

- open existing kinematic models and modify them when necessary,
- create new kinematic models by combining kinematic bodies,
- test a kinematic model by displaying a 3D animation in a 3D view,
- export body transformations into a ASCII file.



Note: Kinematic models which have been created in the Kinematics tab can be associated with geometrical models and trajectories in order to define missions (refer to the "Mission management" chapter).

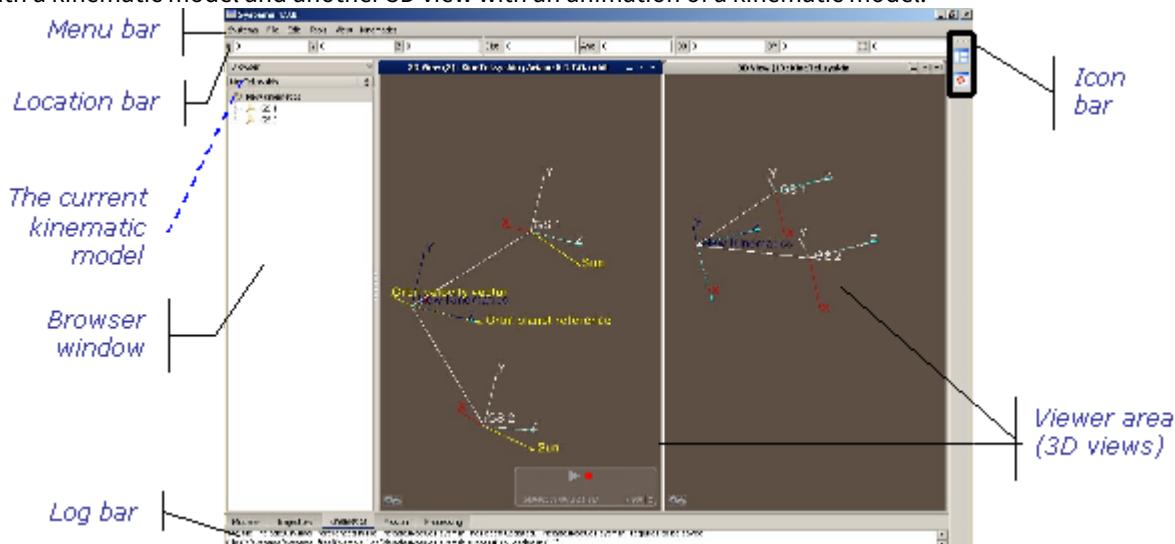
The different parts of the Kinematics tab

When kinematic models have been opened in the **Kinematics** tab, information about them is provided in different areas:

- the browser window, containing the kinematic model's hierarchical description,
- the viewer area, containing the viewports,
- the location bar, containing information about the most recently selected point(s) (from the **Modeler** tab).

Example

The following diagram shows an example of the **Kinematics** tab. The viewer area contains two viewports, a 3D view with a kinematic model and another 3D view with an animation of a kinematic model:



The menu bar

The menu bar contains:

- The menus which are common to all SYSTEMA tabs: the **File**, **Edit**, **Tools**, **View** and **Help** menus.

For more information, refer to the "General User Interface" chapter

- One specific menu, the **Kinematics** menu, which can be used to add kinematic bodies to the current kinematic model.

The **Kinematics** menu contains the **Create kinematic bodies** option, in which there are three available body types. This option creates bodies with default attribute values that can be changed via the edition window.

The viewer area

The viewer area may contain one type of viewport, 3D views.

One or more viewports can be opened at the same time.

The following rules (which are general to all tabs) are applicable to 3D views of the **Kinematics** tab:

- In a viewport, you can load any kinematic model which has been opened.
- Several kinematic models may be loaded at the same time in the same viewport.

The icon bar

The icon bar contains the standard icons to tile the 3D views using all the available space in the viewer area and to show the next 3D view

For more information, refer to the "General User Interface" chapter

The location bar

The information in the location bar can be used to fill in kinematic body attributes, from selections performed in the **Modeler** tab.

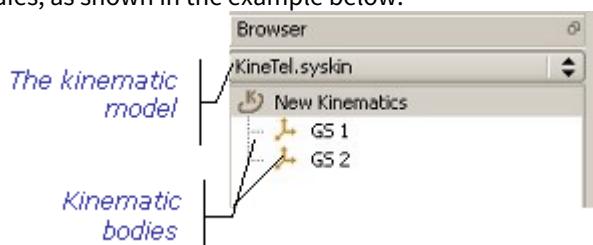
Kinematic models in SYSTEMA

What is a kinematic model?

A kinematic model is a set of kinematic bodies which represent a mechanical structure and its motion. The root of a kinematic model is also called the *main body*. Each kinematic body may itself contain kinematic bodies which are configured in relation to their parent. Kinematic models and kinematic bodies have properties (called *attributes* in SYSTEMA) that can be consulted or modified using the edition window.

The kinematic model in the browser window

The browser window displays the hierarchical organization of the kinematic models arranged by sets of kinematic bodies, as shown in the example below:



Kinematic models are symbolized by the icon followed by the name (example

New Kinematics).

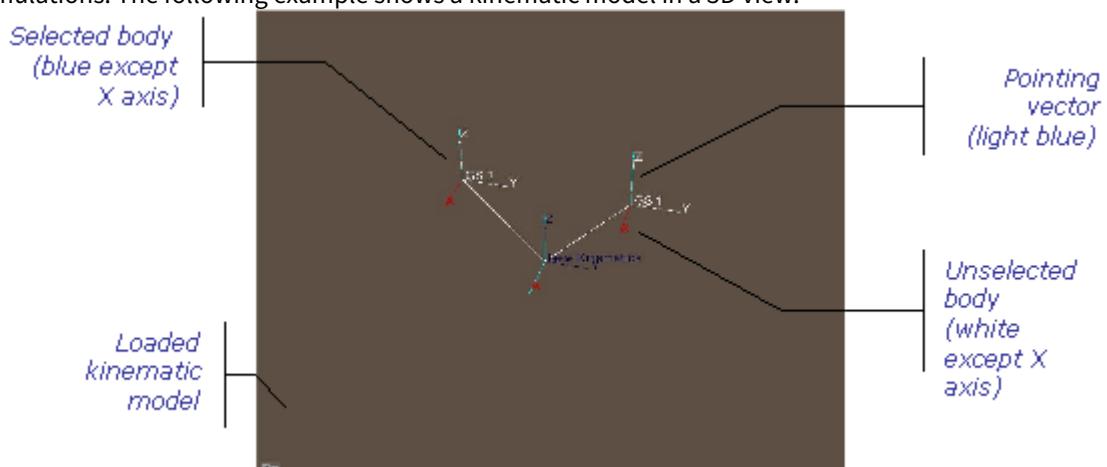
Kinematic bodies are symbolized by the icon followed by the name (example GS 1).



Note: The color of the body icon's background indicates if there are any warnings (orange) or errors (red) associated with **the corresponding body**.

The kinematic model in the 3D views

The 3D views show the kinematic model in a three-dimensional form. They are used to view it and to run simulations. The following example shows a kinematic model in a 3D view:



By default, the names of the model root and bodies are displayed.

In the 3D view, each kinematic body is displayed with a connection to its parent. In the example above, the model root is the "New kinematic model"; it is linked to each of its children which are GS1 and GS2.

The SYSTEMA kinematic bodies

The following table presents the list of available types of SYSTEMA kinematic bodies:

Kinematic body type	Description
Pivot connection (1 dof)	Has one degree of freedom in rotation
Slot ball joint (2 dof)	Has two degrees of freedom in rotation
Ball joint (3 dof)	Has three degrees of freedom in rotation
Free connection	Free connection

The phases in SYSTEMA

Several phases can be defined for a kinematic model. A phase is automatically created and defined as the default one. Each phase is defined for the all structure and defines a moving behaviour of the satellite

The kinematic laws in SYSTEMA

Kinematic laws are used to govern a kinematic body in relation to the environment.

The following table presents the list of available kinematic laws (grouped into types of kinematic laws) for the kinematic model root and every kinematic body:

Type of kinematic law	Kinematic law(s)	Description
Space pointing standard laws	Orbit angular momentum Planet Orbit planet reference north direction Orbit planet reference rotational direction Sun Orbit velocity vector Orbit planet reference vernal direction X/Y/Z direction of Gamma50 X/Y/Z direction of J2000 Orbit planet reference Another satellite trajectory 2 pointing vectors law Frozen rotational orientation	These laws are defined with one or two pointing directions and the associated pointing vectors. The pointing vector is an axis of the kinematic body and the pointing direction is defined by the following laws.
Linear attitude laws	Linear attitude 3 attitudes laws	These laws are defined on one, two or three specific axis of the kinematic body. The attitude angles around the specific axis are linearly interpolated according to the selected variable (time, true anomaly or longitude).

Type of kinematic law	Kinematic law(s)	Description
Rotating laws	Regular spin around axis Fast spin around axis	Rotation around a specific axis of the kinematic body.
Transformation defined in a file		External data files indicating attitude or transformation of the object over time (CSV, STK or text files)
Fixed attitude laws	Fixed position Fixed last position	Useful to stay in a given position or in the last position of the previous phase
	<p><i>Note: For the import of STK attitude files (through "Transformation defined in a file" law), we only import the attitude in the current referential. You must define the kinematics body that will import the STK file in the same referential. See Kinematics technical annex for more details.</i></p>	

Transformation defined in a file

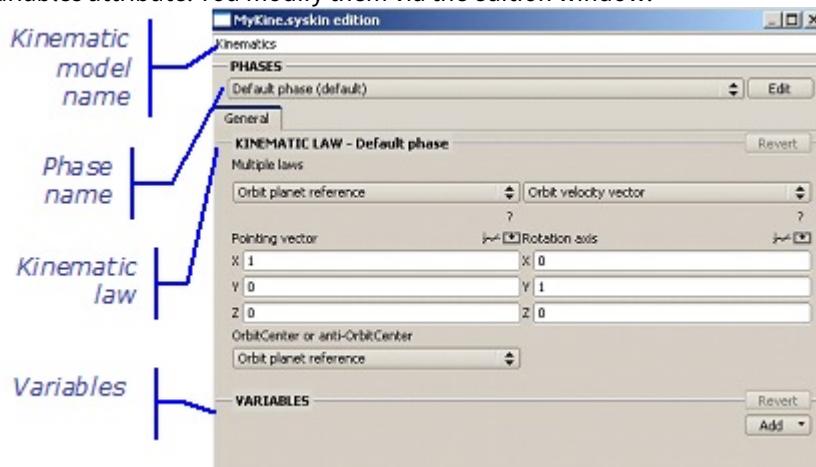
File type	Description
stk attitude file (.a)	<p>File to and from the STK software. It contains the attitude of a node at given times.</p> <p>It consists in a header describing the data and an array of values: epoch followed by a quaternion. The convention for quaternions is that the fourth element of the quaternion is the scalar component. The quaternions represent the rotation from the reference coordinate frame to the vehicle's body coordinate frame</p> <p>See https://help.agi.com/stk/index.htm#stk/importfiles-01.htm for more details.</p> <p>The structure of the file must be :</p> <pre> stk.v.9.0 {optional} BEGIN Attitude {mandatory} NumberOfAttitudePoints [int] {not supported} ScenarioEpoch 01 Jan 2000 00:00:00.0 {optional. if not defined, 01/01/1970 00:00:00.0 will be used} CoordinateSystem B1950 {Optional, but only this coordinate system is supported} AttitudeTimeQuaternions {mandatory, only this format is supported} {data:} - column 1 : date in Epoch (number of seconds since ScenarioEpoch) (1 double) - column 2-5 : quaternion(x,y,z,w) (4 doubles) END Attitude {optional} </pre> <p>Two simple options to import "STK " attitude files:</p>

File type	Description
	<ul style="list-style-type: none"> Provide a file with no header and set the Kind of date value to "Phase elapsed time (sec)" The start date will then be the one of the associated trajectory Provide a file with a header containing a reference date The date must respect the following syntax given : ScenarioEpoch dd Jan yyyy hh:mm:ss.zzz where d, y h, m, s, z are digits. The month format must be: Jan/Feb/Mar/Apr/May/Jun/Jul/Aug/Sep/Oct/Nov/Dec In this case the Kind of date must be "Absolute time Gamma50 (day)"
CNES CIC based AEM spacecraft attitude file (.txt)	<p>CIC based file modified to suit the interfaces of the CNES and based on the CCSDS standard. It contains the attitude of the spacecraft at given times.</p> <p>It consists in a header describing the data and an array of values: date in julian days followed by seconds of the day and the quaternion describing the attitude of the spacecraft.</p> <p>It requires to set the relevant reference axis for the parent item of the kinematics body depending of the reference frame defined with REF_FRAME keyword :</p> <p style="margin-left: 20px;">EME2000 :</p> <ul style="list-style-type: none"> X = "X direction of J2000" (vernal direction at J2000 epoch) <i>For Systema before 4.9.2 release</i> : Z = "Orbit planet reference North direction" The "Z direction of J2000" corresponds to the Sun-centered referential which is oriented towards to ecliptic North. This would not fit the EME2000 definition which has a planetary North Z. <i>For Systema 4.9.2 release and later</i> : Z = "Z direction of J2000" <p style="margin-left: 20px;">ICRF</p> <ul style="list-style-type: none"> X = "X direction of ICRF" Y = "Y direction of ICRF" Z = "Z direction of ICRF"
CNES CIC based MEM moving body attitude file (.txt)	<p>CIC based file modified to suit the interfaces of the CNES and based on the CCSDS standard It contains the attitude of a moving part at given times.</p> <p>It consists in a header describing the data and an array of values: date in julian days followed by seconds of the day and an angle in degrees. The axis around which the rotation angle will be applied is defined in the edition window of the kinematics body.</p> <p>While AEM file describes the attitude of the whole spacecraft, MEM describes the attitude of a single moving part, therefore a spacecraft using this type of files is defined as follows in Systema:</p> <ol style="list-style-type: none"> Root => reference frame to use a. Spacecraft attitude kinematic node => AEM file <ol style="list-style-type: none"> Moving body kinematic node 1 => MEM file

File type	Description
	<p>ii. Moving body kinematic node 1 => MEM file iii. ...</p> <p><u>Parameters to be modified include:</u></p> <ul style="list-style-type: none"> Reference mark transformation files : choice of the ASCII file which contains all transformations. Kind of date : choice between absolute and relative date (ignored on some files). Reference date : in case of relative date, choice of the reference date between Gamma50, J2000, Calendar date and Earth Season date. Data format : describes how to handle input data Adaptation : definition of a transformation to be considered before each transformation defined in the file. <p>As for AEM files, the relevant reference axis shall be defined (see AEM files).</p>
Transformation matrix	<p>ASCII file that contains:</p> <ul style="list-style-type: none"> column 1 : date (1 double) ^[P] column 2-17 : transformation 4x4 matrix in meters (16 doubles) written by columns ^[P] <ul style="list-style-type: none"> rotation matrix from line 1 col 1 to line 3 col 3 (orthogonal direct) ^[P] translation vector from line 1 col 4 to line 3 col 4 ^[P] last line is equal to "0 0 0 1" ^[P]
Rotations in transformed axis Z-Y'-X'' Rotations in transformed axis X-Y'-Z'' Rotations in fixed axis Z-Y-X Rotations in fixed axis X-Y-Z	<p>ASCII file that contains:</p> <ul style="list-style-type: none"> column 1 : date in gamma50 (1 double) ^[P] column 2-4 : rotation angle ^[P] <ul style="list-style-type: none"> rotation around X axis in degree (1 double) ^[P] rotation around Y axis in degree (1 double) ^[P] rotation around Z axis in degree (1 double)

Kinematic model root attributes

The model root has several attributes: the kinematic model name and the kinematic law defined for each phase, the Variables attribute. You modify them via the edition window:



The kinematic law attribute

At root level for a given phase, the kinematic model is associated with a kinematic law which governs the global object in relation to its environment.

When a kinematic law attribute has been defined for a model, the associated directions are marked in the 3D views where the model is displayed:

- a *light blue arrow* indicates the pointing vector relating to the kinematic law,
- a *yellow arrow* indicates the direction aimed at (relating to the kinematic law) and is labelled with the kinematic law's name

Variables

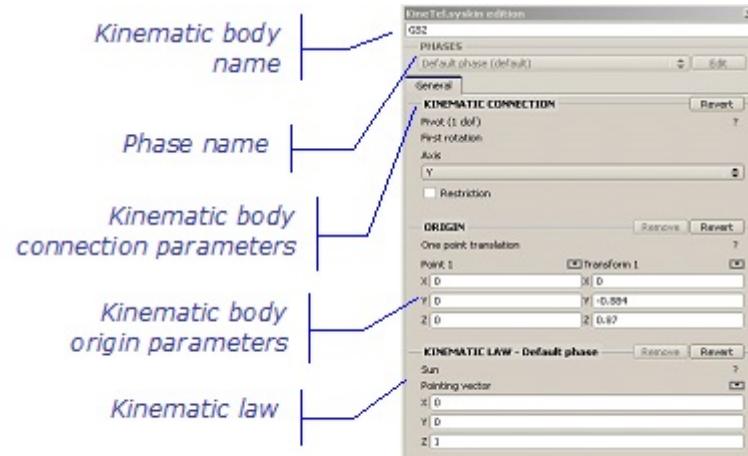
Cf [Working with variables](#)

Kinematic body attributes

Kinematic bodies have four main attributes: the kinematic body name, the kinematic body connection, origin and the kinematic law defined for each phase attributes.



Note: The kinematic connection and origin parameters define the mechanical links inside the relevant model. Only the kinematic law depends from the phase



You modify them via the edition window:

The SYSTEMA kinematic body connection attribute

Every body is linked to its parent by means of a connection, which is specific to the body type.

The kinematic body origin attribute

This parameter is used to define the position of a kinematic body in relation to its parent (distance and direction). For example, if the kinematic body is associated with a solar panel, you have to define the distance to the satellite's center of gravity, and the direction in relation to the main axis system.

The kinematic law attribute

For each phase, each kinematic body is associated with a kinematic law which governs it in relation to environment and/or to the time.

When a kinematic law attribute has been defined for a kinematic body for a phase, the associated directions are marked in the 3D views where the model is displayed for this phase:

- a *light blue arrow* indicates the pointing vector relating to the kinematic law,
- a *yellow arrow* indicates the aimed at direction (relating to the kinematic law) and is labelled with the kinematic law name.

Display options in the 3D views of the Kinematics tab

Right click menu in the 3D views

Within the 3D views, the right click menu contains various display options.

Some of them are options common to all the 3D views of every SYSTEMA tab:

- **Hide, Hide others, Hide all** and **Show all** options concern either the selected bodies or all bodies;



Note: Hide / show actions are reflected in all 3D views in which the kinematic model is loaded, and also in the browser window.

- the **Configuration window** option.



Note: This action is local to the viewport (several kinematic models may be loaded in a viewport).

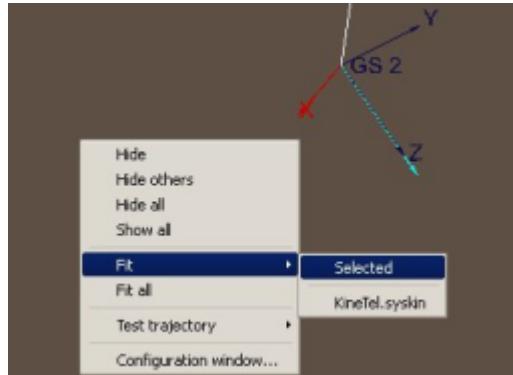
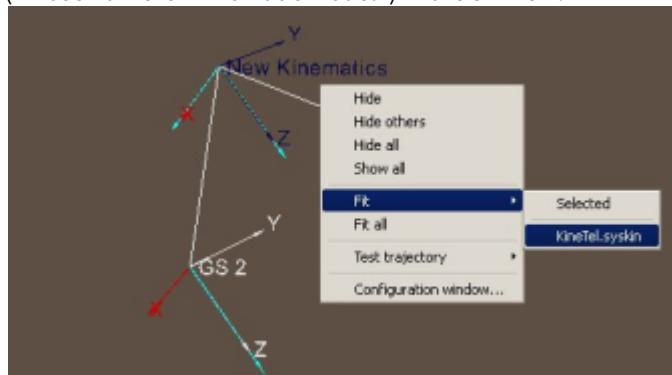
For more information about these options, refer to the "User Guide Interface" chapter.

Specific display options for viewing kinematic models



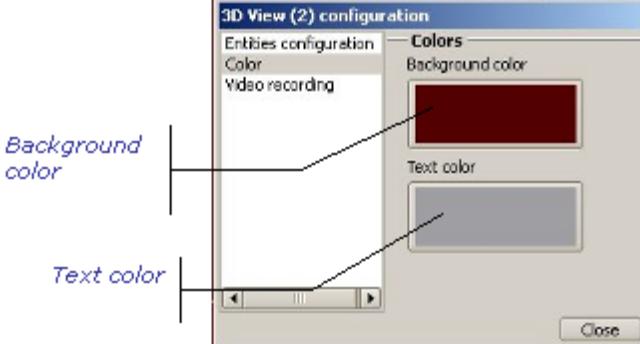
Note: All these actions are local to the 3D view.

Zoom facilities

Option	Use
Fit - Selected	<p>Adjusts the zoom in order to fit entirely the selected body in the 3D view:</p> 
Fit -- Loaded Kinematic model	<p>Adjusts the zoom in order to fit entirely the imported model (whose name is "Kinematic model") in the 3D view:</p> 
Fit All	<p>Adjusts the zoom in order to fit entirely all imported models in the 3D view</p>

Configuration window specific options

Display option

Option	Use
Color	Allows to select the background or text colors by clicking the background color or the text color. 

Video recording

To record a video, refer to the chapter "General User Interface", § Recording a video.

Interactive 3D animation tool

In the 3D views, you can interactively display an animation of the kinematic model for the selected phase.

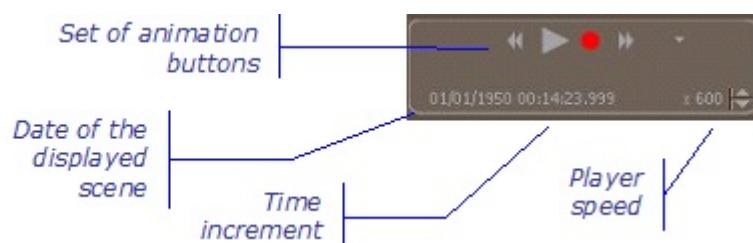
Running a kinematic simulation

To run a kinematic simulation:

1. In a 3D view, load the kinematic model you want to animate.
2. In the right click menu, select a trajectory in the **Test trajectory** menu.
3. In the configuration window, select the phase you want to display

At the bottom of the 3D view, a set of buttons allows the animation to be run:

Example:



The animation buttons are described in the following table:

Button	Use
	Starts running (or re-starts) the animation

	Stops running the animation (just pauses it, time and position are kept)
	Starts recording the animation
	Moves the display time to the previous event
	Moves the display time to the next event
	Allows to select the type of event (no filter, file law)

To modify the player speed :

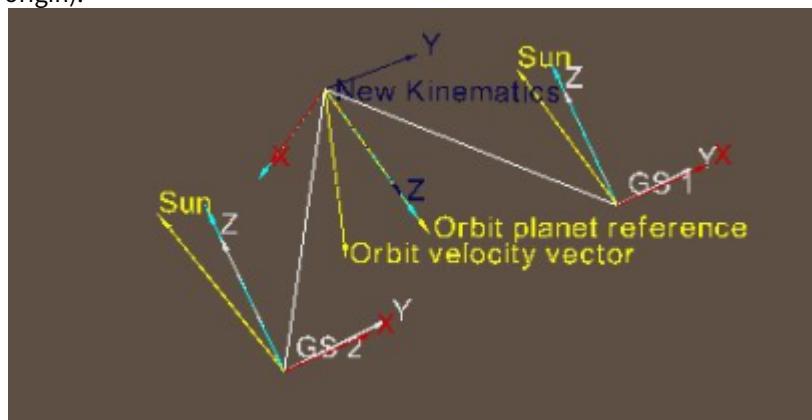
1. To increase the player speed, click .
2. To decrease the player speed, click .

Example of kinematic model animation

In the following example, the kinematic model is divided into three parts:

- New Kinematic model which corresponds to the model root and which follows two compound kinematic laws (Orbit planet reference law and Orbit velocity vector law),
- the GS1 body which follows a Sun kinematic law (indicated by a yellow arrow) with the $(0, 0, 1)$ vector as the pointing vector,
- the GS2 body which follows a Sun kinematic law (indicated by a yellow arrow) with the $(0, 0, 1)$ vector as the pointing vector.

The GS1 and GS2 kinematic bodies have their origins defined by the **ORIGIN** attribute (different from the root origin).



Working with kinematic models

General

In the **Kinematics** tab, you can open an existing kinematic model or create a new one. When a kinematic model is opened, you can:

- load it in any 3D view,
- modify it,
- display an animation of the kinematic model,
- export body transformations into a file.

Creating a new kinematic model

To create a new model:

- In the **File** menu of the **Kinematics** tab, select the **New** option.

Result: In the viewer area, a new 3D view is opened.

You can then build the kinematic model and save it.

Opening an existing kinematic model

You can open a kinematic model which has previously been saved within a ".syskin" file.

To edit an existing model:

- In the **File** menu of the **Kinematics** tab, select the **Open** option and select the file containing the required model.

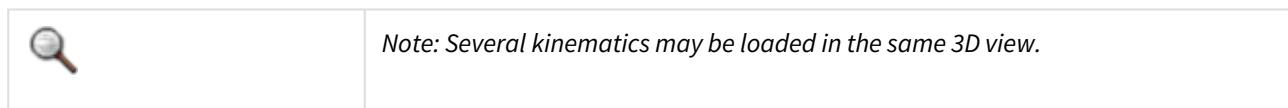
Result: In the viewer area, a new 3D view is opened with the kinematic model displayed in it.

In the browser window, this model's hierarchical organization is displayed. The previously edited models can be accessed from the list of models.

You can then modify the kinematic model and save your modifications.

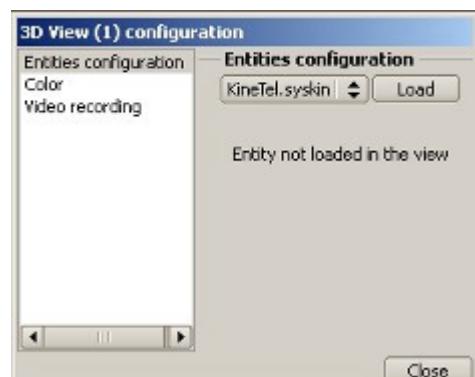
Loading a kinematic model in a 3D view

In the 3D views, you can load kinematics which have been opened.



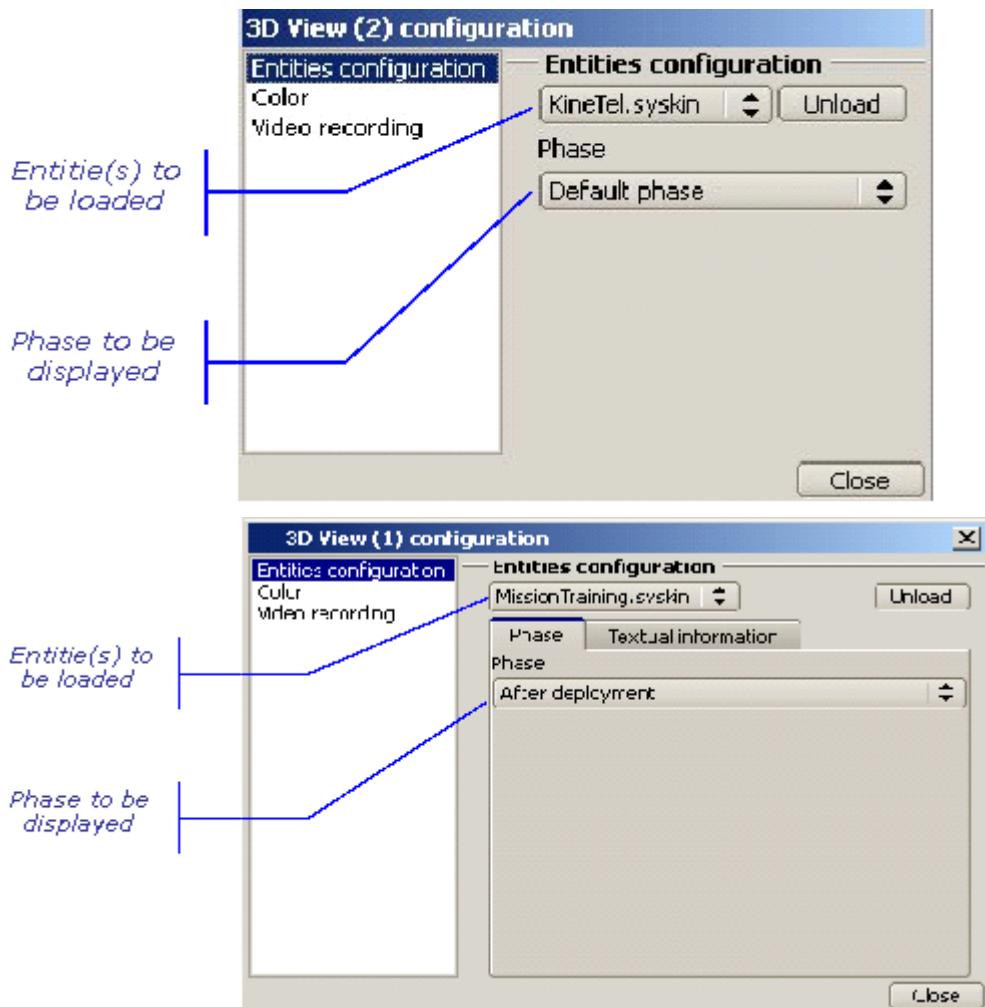
To load a kinematic in a 3D view:

- In the 3D view, in the right click menu:
1. Select **Configuration window**.



2. Select the kinematic to load.

3. Click **Load**



4Select the phase to be displayed

Result: The selected kinematic model is then displayed in the 3D view.

Modifying a kinematic model

You can modify a kinematic model by:

- creating new kinematic bodies (refer to the "Building a kinematic model" section),
- editing and modifying existing kinematic bodies.

Building a kinematic model

How to build a kinematic model

You build a kinematic model by creating and organising kinematic bodies.

Creating kinematic bodies

There are three different ways to create kinematic bodies:

- using the browser window
- or using the **Kinematics** menu.

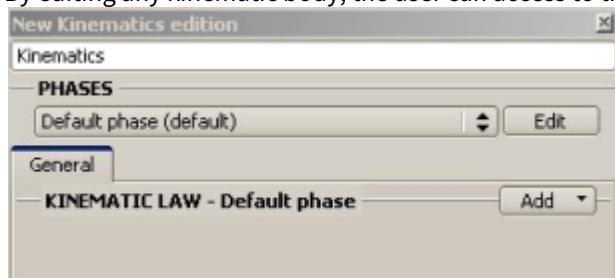
(Details are provided in the section below)

Organizing kinematic bodies

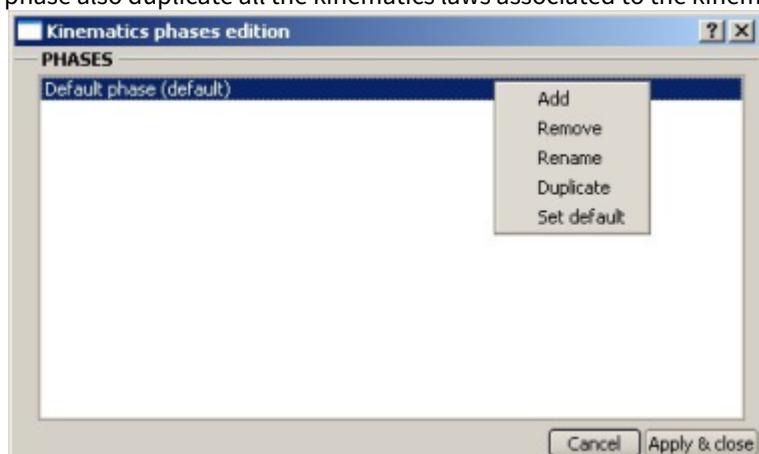
To organize kinematic bodies, you have to edit the information window for each one and enter the appropriate parameters.

Adding/Removing phases

A default phase is automatically created and set as the default one. The phases are defined for the all structure. By editing any kinematic body, the user can access to the edition of the phase.



The phases' edition window presents the list of defined phases. By right click menu in the list, a phase can be added, removed, renamed, duplicated, set as default. The default phase cannot be removed. The duplication of a phase also duplicate all the kinematics laws associated to the kinematics bodies



Adding kinematic bodies

Kinematic bodies are created in the current model (the one displayed in the browser window).

A new kinematic body is created with default values. Its attributes (name, connection, origin and kinematic law) can then be modified by editing it.

How to add a kinematic body with default values

To add a kinematic body from the browser window:

1. In the browser window, right click on the kinematic body you wish to add a new one to.
2. In the right click menu, select **Create kinematic body**, the position of the new body (after, before or in) and then the body type.

Result: The new kinematic body is created. In the browser window, it is placed in the selected position.

To add a kinematic body from the Kinematics menu:

1. Select the kinematic body you wish to add a new body to.
2. In the **Kinematics** menu, select the **Create kinematic body** option and then the body type.

Result: The new kinematic body is created. In the browser window, it is positioned as the last one in the selected element.

Result in the 3D views

When the kinematic body has been created, it is displayed in each 3D view in which the kinematic model is loaded.

Exporting transformations

You can export body transformations into a file. This file may be used later to define a kinematic law. For a given orbit type, the generated file contains all the body transformations during a given time interval and computation step. The generated file is suffixed with ".trs".

How to export transformations for a given kinematic body

1. Open the kinematic model containing the body you wish to export.

2. In the **File** menu, select **Export points**.

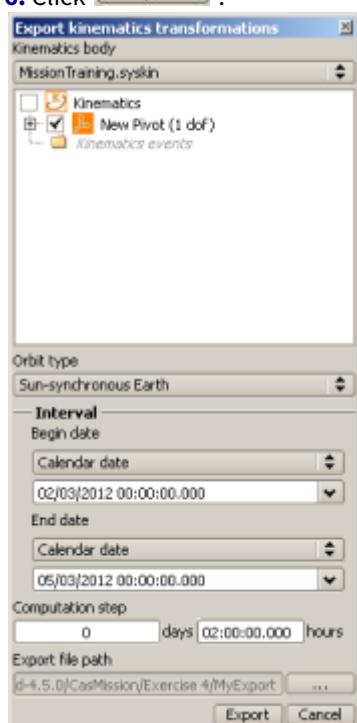
3. Define the transformation parameters:

- Select the orbit type
- Enter the time interval (**Begin date** and **End date**)
- Enter the computation step.

4. Select a node (i.e. a body) in the hierarchical view of the *Export transformation* window.

5. Enter the name of the file to be created, using the  button.

6. Click .



Mission management

Mission management

Overview

Introduction

This chapter describes how to use SYSTEMA in order to create, load, save, display and test missions. With SYSTEMA, you use the **Mission** tab to work on missions. Everything described in this chapter takes place within the **Mission** tab.

In this chapter

This chapter covers the following topics:

- Mission management
 - Overview
 - Introduction
 - In this chapter
 - The Mission tab
 - What is the Mission tab for?
 - The different parts of the Mission tab
 - Example
 - The menu bar
 - Options from the Mission menu
 - The viewer area
 - The icon bar
 - Missions in SYSTEMA
 - What is a mission?
 - The mission in the browser window
 - The mission in the 3D views
 - Orientation frame display
 - The mission in the timeline window
 - Mission attributes
 - Main attributes of the mission root
 - The geometrical model attribute
 - The meshing attribute
 - The schematic attribute
 - The trajectory attribute
 - The kinematic model attribute
 - The kinematic body link attribute
 - The geometrical to kinematic attribute
 - Variables
 - The list of sequences
 - The list of computation dates (computation events and computation event sets)
 - The list of custom events, custom intervals, custom events list and custom intervals sets
 - Attributes of the model elements
 - The Kinematic link attribute

- The Initial move attribute
- Attributes of a sequence
 - How to define the Referenced Phase
 - How to define the Start of the Time interval
 - How to define the End of the Time interval
- Attributes of an event
- Attributes of an event set
- Attributes of an interval
- Attributes of an interval set
- Attributes of a date
- Display options in the 3D views of the Mission tab
 - Right click menu in the 3D views
 - Characteristics of mission displays in the 3D views
 - Specific display options for viewing missions
 - Zoom facilities
 - Camera Lock in options
 - Configuration window specific options
 - Display options
 - Orientation Information
 - Color
 - Video recording
 - Interactive 3D animation tool
 - The set of animation buttons
 - The player speed
- Working with missions
 - General
 - Creating a new mission
 - Opening an existing mission
 - Loading a mission in a 3D view
 - Modifying a mission
 - Adding/ Deleting a sequence
 - Adding/ Deleting computation dates (computation events and computation event sets)
 - Adding/ Deleting custom events, event sets, intervals and interval sets
- Building a mission
 - How to build a mission
 - How to define the mission components at root level
 - How to define the motion parameters of a model element
- Viewing mission textual information
 - How to view textual information for a given mission
- The Mission Log
 - Content of the Mission Log
 - Transformations
 - References frames

The Mission tab

What is the Mission tab for?

You can use the **Mission** tab of the SYSTEMA application to:

- open existing missions and modify them when necessary,
- create new missions,
- test a mission by displaying a 3D animation in a 3D view.



*Note: In the **Mission** tab, you define and view a mission for a geometrical model, a trajectory and a kinematic model.*

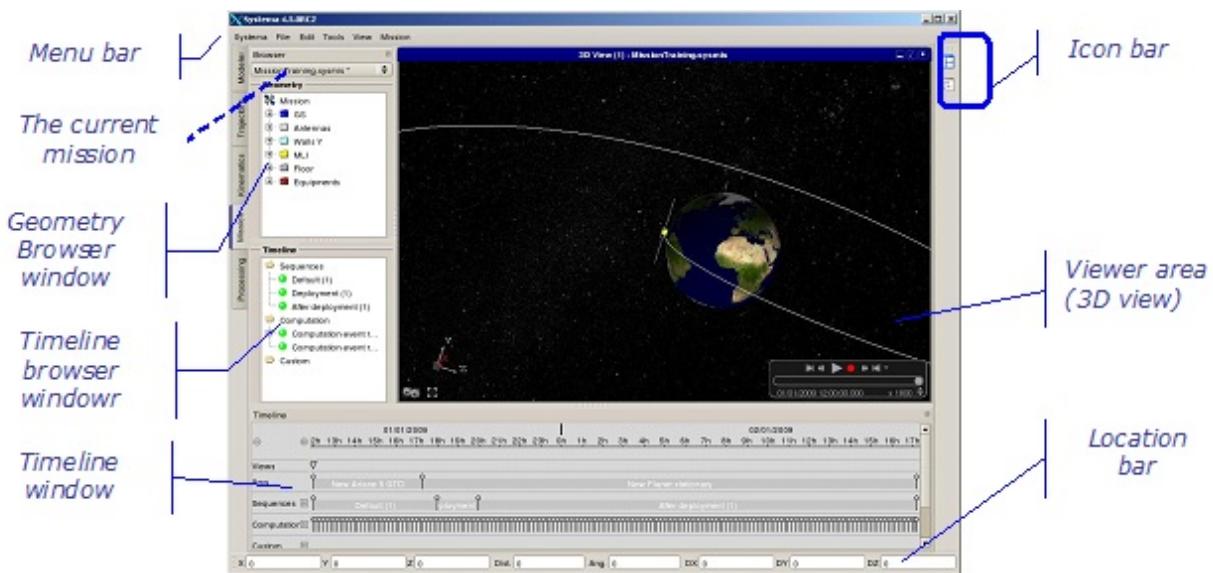
The different parts of the Mission tab

When missions have been opened in the **Mission** tab, information about them is provided in different areas:

- the geometry browser window, containing the model's hierarchical description,
- the meshing browser window, containing the meshing's hierarchical description which is optional, not activated by default. Show it by checking the **View/Meshing browser** option in the menu bar
- the timeline browser window, containing the components managed by the timeline grouped in categories
- the viewer area, containing the viewports,
- the timeline window used to display all the components linked to the time such as orbital arcs, eclipse and penumbra intervals.
- the location bar, containing information about the most recently selected point(s) (from the **Modeler** tab).
- The log bar which is optional, not activated by default. Show it by checking the **View/Log** option in the menu bar

Example

The following diagram shows an example of the **Mission** tab. The viewer area contains a 3D view with a model on its trajectory:



The menu bar

The menu bar contains:

- The menus which are common to all SYSTEMA tabs: the **File**, **Edit**, **Tools**, **View** and **Help** menus.

For more information, refer to the "General User Interface" chapter

- One specific menu, the **Mission** menu, which can be used to add events, intervals, events sets and intervals sets to the current mission.

Options from the Mission menu

Creates components managed in the timeline in the current mission.

The following table presents the options available from the **Mission** menu:

Option	Use
Create sequence intervals	Allow you to create sequence interval(s) <ul style="list-style-type: none"> • Interval : creates a sequence interval • Intervals set : creates a set of sequence intervals
Create computation events	Allow you to create computation point(s) <ul style="list-style-type: none"> • Event : creates a computation point • Events set : creates a set of computation points
Create custom events	Allow you to create custom events <ul style="list-style-type: none"> • Event : creates a custom event • Events set : creates a set of custom events
Create custom intervals	Allow you to create custom intervals <ul style="list-style-type: none"> • Interval : creates an interval • Intervals set : creates a set of intervals

The viewer area

The viewer area may contain one type of viewport, 3D views.

One or more viewports can be opened at the same time.

The following rules (which are general to all tabs) are applicable to 3D views of the **Mission** tab:

- In a 3D view, you can load any mission which has been opened.
- Several missions may be loaded at the same time in the same 3D view.

The icon bar

The icon bar contains the standard icons to tile the 3D views using all the available space in the viewer area and to show the next 3D view

For more information, refer to the "General User Interface" chapter

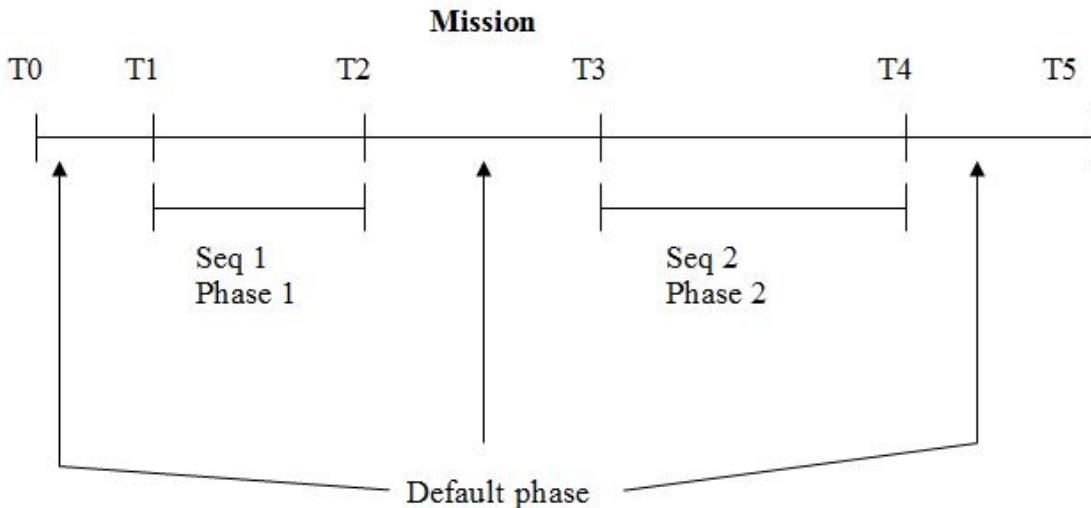
Missions in SYSTEMA

What is a mission?

A mission consists of a geometrical model, a trajectory and a kinematic model. A mission allows to simulate a geometrical model movement in a given simulation environment. Some sequences can be defined in a mission in order to reference different phases of the kinematic. By default, no sequence is defined and the entire mission references the default kinematic phase. Each created sequence has a name automatically computed from the phase's name, a start and an end date, an associated kinematic phase. If the end of a sequence does not match the start of the following one, the mission references the default phase in the interval

Missions have properties (called *attributes* in SYSTEMA) that can be consulted or modified using the edition

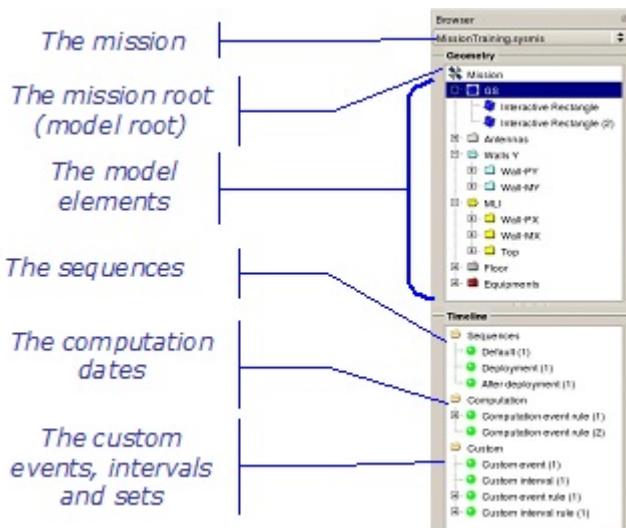
window.



The mission in the browser window

The browser window is split in two or three windows as shown in the example below :

- the hierarchical organization of the mission's model (the model browser window)
- the hierarchical organization of the components managed in the timeline (the timeline browser window).
- the hierarchical organization of the meshing model (the meshing browser window) can be optionally displayed by the user using the option **View/Meshing browser**



Note: The browser window contains an image of the model and its elements whose attributes relate to the mission. They cannot be modified from this view.

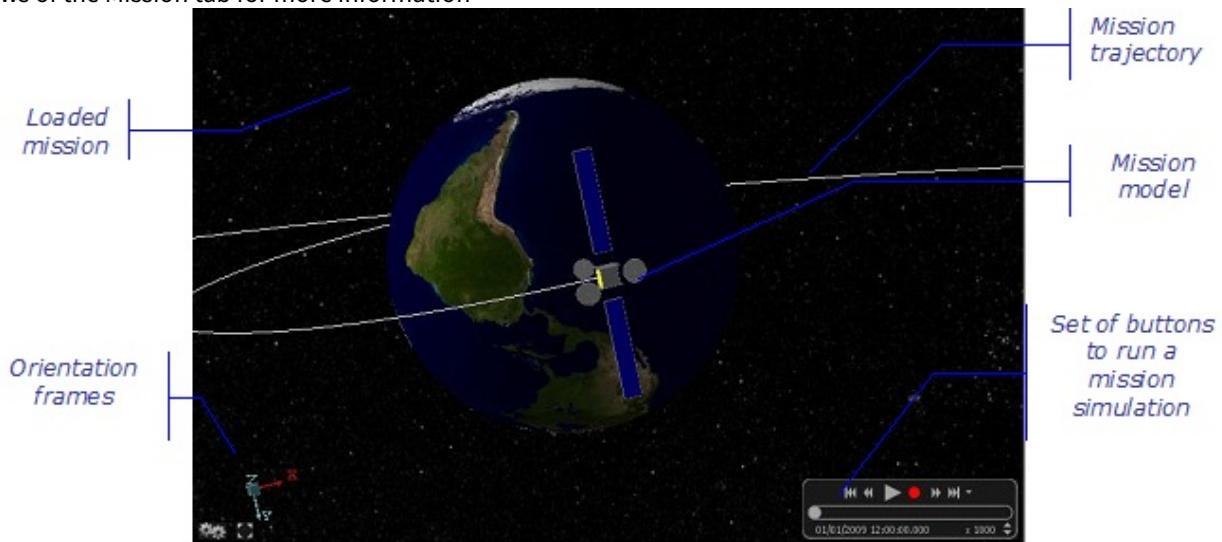
Missions are symbolized by the  icon followed by the name (example  Mission).

The mission in the 3D views

The 3D views show the trajectory and the SYSTEMA model in a three-dimensional form. They are used to view and run the trajectory when applying kinematic motions to the model. The following example shows a model on its trajectory in a 3D view:

Orientation frame display

The user can add as many frames as necessary to display orientation information. Refer to Display options in the 3D views of the Mission tab for more information



The mission in the timeline window

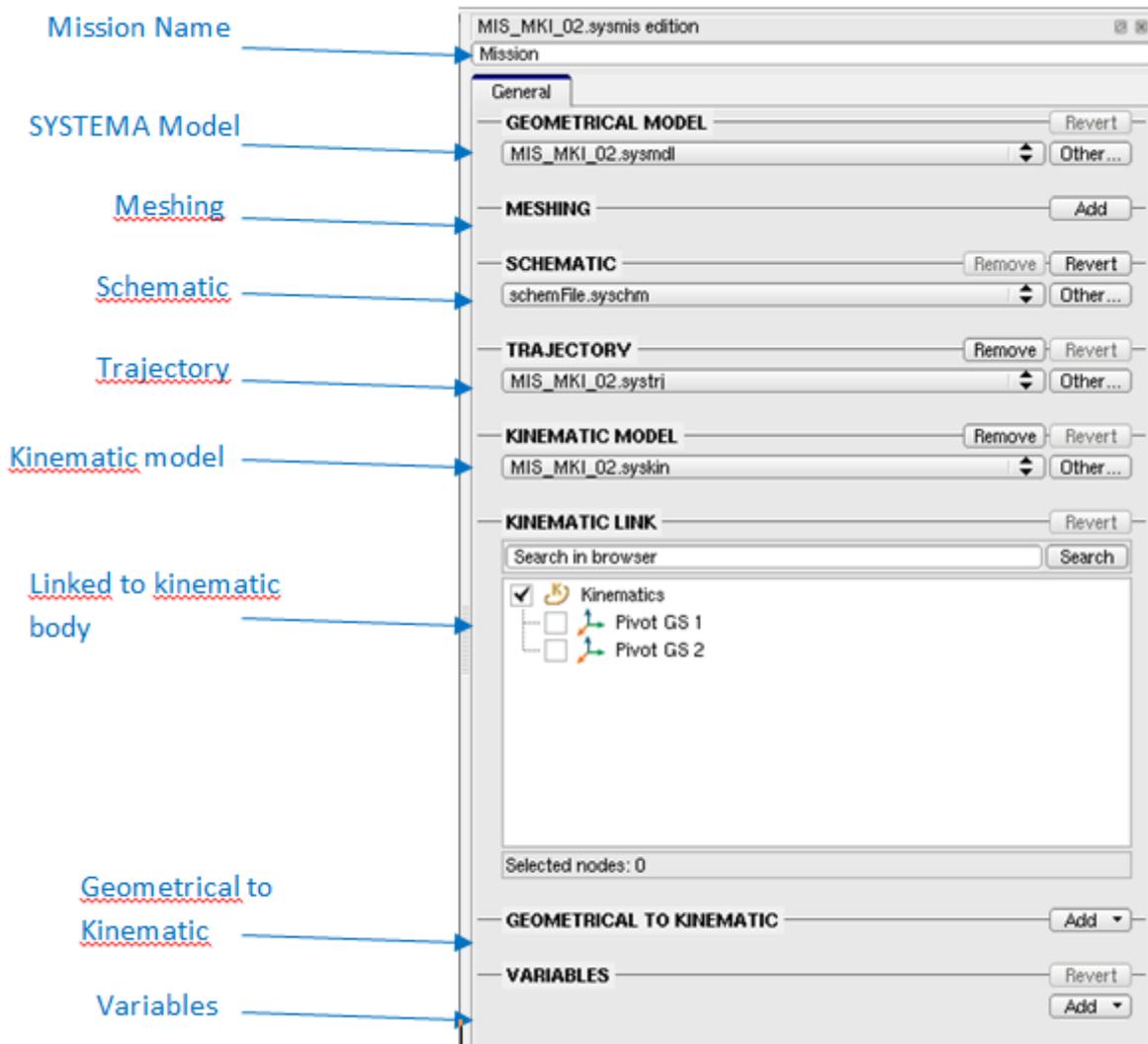
Refer to "Working with the timeline window" in the "General User Interface" chapter to get the description of the timeline window.

Mission attributes

The mission root has several attributes:

- the mission name,
- the linked files: the geometrical model, the meshing associated to this model, the trajectory, the kinematic model
- the kinematic body link, the geometrical to kinematic attribute and the Variables attribute,

You modify them via the object edition window :



Main attributes of the mission root

The geometrical model attribute

Each mission is associated with a SYSTEMA model.

Once a model has been associated with the current mission, it is loaded in the **Model** tab.

The meshing attribute

The Model meshing attribute defines the meshing to which processes will be applied. The meshing must be associated with the geometrical model.

Once a model meshing has been associated with the current mission, it is loaded in the **Model** tab.

The schematic attribute

Each mission may be associated with a SYSTEMA schematic.

Once a schematic has been associated with the current mission, it is loaded in the **Schematic** tab.

The trajectory attribute

Each mission is associated with a SYSTEMA trajectory.

Once a trajectory has been associated with the current mission, it is loaded in the **Trajectory** tab.

The kinematic model attribute

Each mission may be associated with a SYSTEMA kinematic model.

Once a kinematic model has been associated with the current mission, it is loaded in the **Kinematics** tab. Each of its kinematic bodies may then be associated with a model element (object or shape).

The kinematic body link attribute

The mission root (main body of the SYSTEMA model) may be associated with a kinematic body of the selected kinematic model.

Once a kinematic body has been associated with the mission root, the model is placed on the trajectory as required by the kinematic body parameters (position and orientation). When running the mission, the model will follow the kinematic motion and position defined by the kinematic body parameters.

The geometrical to kinematic attribute

The Geometrical to kinematic attribute defines a transformation applicable to the mission root (main body of the SYSTEMA model) and consequently to all shapes and objects contained in the SYSTEMA model.

Variables

Cf [Working with variables](#)

The list of sequences

A list of sequences is associated to the mission. It is displayed in the timeline browser window

The list is empty when a new mission is created.

The list of computation dates (computation events and computation event sets)

A list of computation dates is associated to the mission. It is displayed in the timeline browser window. It contains computation events and generated computation events belonging to an events set defined by a computation rule. The computation dates attribute defines trajectory points which will be used when running processes on the mission.

The list may be empty when a new mission is created or may contain default computation dates if the **Create default computation dates** option have been used at the creation.

For more information, refer to "Creating a new mission" in "Working with mission" chapter

The list of custom events, custom intervals, custom events list and custom intervals sets

A list of custom events, custom intervals, custom events sets and custom intervals sets is associated to the mission. It is displayed in the timeline browser window

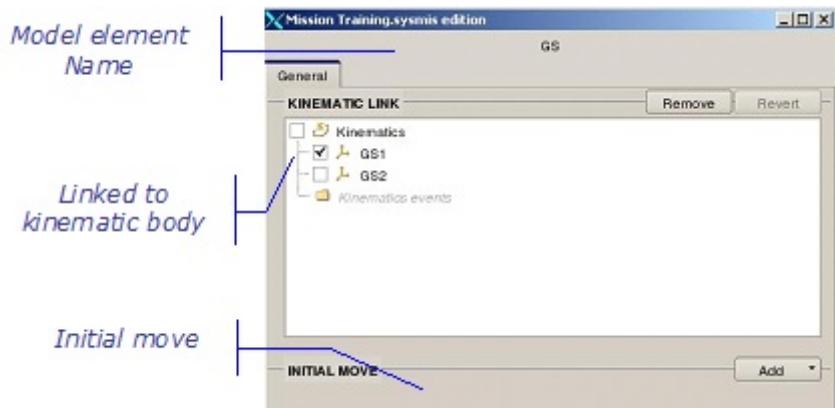
The list is empty when a new mission is created.



Note: refer to "Working with the timeline window" in the "_General User Interface" chapter to get the description of the attributes of the events, intervals, events sets and intervals sets._

Attributes of the model elements

Model elements have two attributes: the kinematic link and the initial move attributes. You modify them via the edition window: The model element name is displayed.



The Kinematic link attribute

Each element of the model hierarchy may be associated with a kinematic body related to the mission via the kinematic link attribute (at the root level).

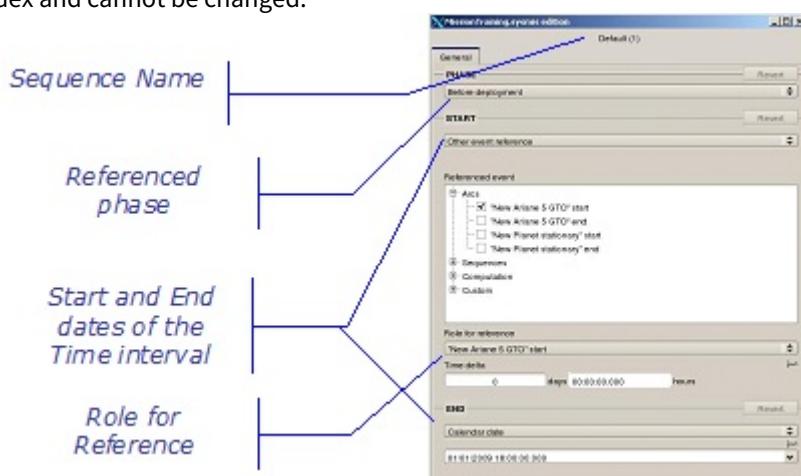
Once a kinematic body has been associated with an element of the model, this element and its sub-elements are placed as required by the kinematic body parameters (position and orientation). When running the mission, the element will follow the kinematic motion defined by the kinematic body parameters.

The Initial move attribute

The Initial move attribute defines a transformation applicable to the model element and which defines its initial position in its environment.

Attributes of a sequence

A sequence has three attributes : the sequence name, the associated kinematic phase and the sequence time interval defined through the start and end dates. The name is automatically computed from the phase name with an index and cannot be changed.



How to define the Referenced Phase

The referenced phase is the default kinematic phase until another phase is chosen through the list containing all the names of the kinematic phases.

How to define the Start of the Time interval

Refer to "*Attributes of a date*" in "*Working with the timeline window*" in the "*General User Interface*" to get the definition of the date attributes.

How to define the End of the Time interval

Refer to "*Attributes of a date*" in "*Working with the timeline window*" in the "*General User Interface*" chapter to get the definition of the date attributes.

Attributes of an event

Refer to "*Attributes of an event*" in "*Working with the timeline window*" in the "*General User Interface*" chapter to get the definition of the event attributes.

Attributes of an event set

Refer to "*Attributes of an event set*" in "*Working with the timeline window*" in the "*General User Interface*" chapter to get the definition of the event set attributes.



Note: If the step involves a great number of computation points, running a process for the mission may take a very long time

Attributes of an interval

Refer to "*Attributes of an interval*" in "*Working with the timeline window*" in the "*General User Interface*" chapter to get the definition of the interval attributes.

Attributes of an interval set

Refer to "*Attributes of an interval set*" in "*Working with the timeline window*" in the "*General User Interface*" chapter to get the definition of the interval attributes.

Attributes of a date

Refer to "*Attributes of a date*" in "*Working with the timeline window*" in the "*General User Interface*" chapter to get the definition of the interval attributes.

Display options in the 3D views of the Mission tab

Right click menu in the 3D views

Within the 3D views, the right click menu contains various display options.

Some of them are options common to all the 3D views of every SYSTEMA tab:

- the **Configuration window** option.



Note: This action is local to the viewport (several trajectories may be loaded in a viewport).

For more information about these options, refer to the "User Guide Interface" chapter.

- the **Load maps file** and **Display maps file** options. Please refer to [Trajectory management chapter](#) for more details.

Characteristics of mission displays in the 3D views

In the 3D views of the **Mission** tab, the scenes present the following characteristics:

- The scene in the **Mission** tab shows realistic lighting (i.e. solar light).
- Like in the 3D views of the **Trajectory** tab, the scale used to draw the Sun is different from the scale used for the other bodies:
- the Sun is shown on a scale of 1;
- the other planets and their orbits are drawn on a scale of 250 (they are 250 times larger than in reality); the model size may be determined via the **Configuration** option.



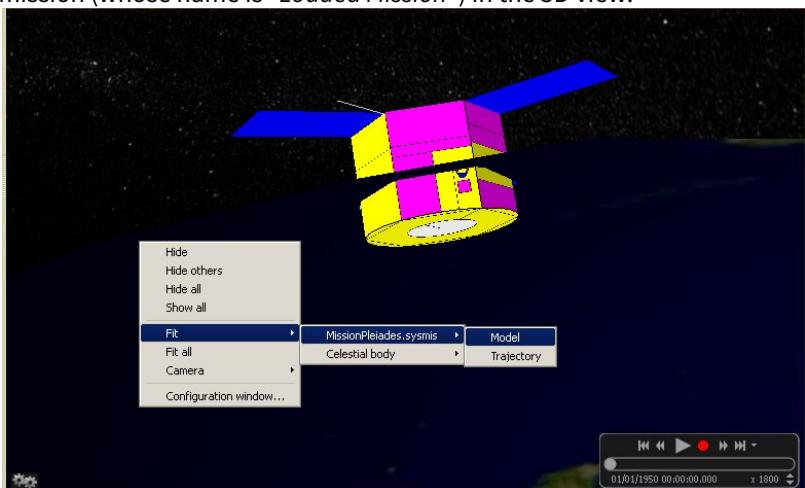
Note: Considering the scale values, it may be that orientations of model elements do not correspond to reality (for example, a solar panel may appear to be incorrectly directed).

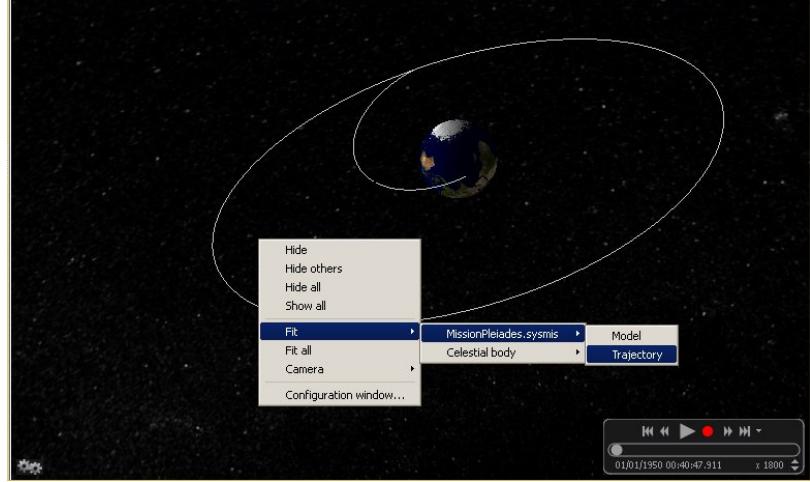
Specific display options for viewing missions



Note: All these actions are local to the 3D view.

Zoom facilities

Option	Use
Fit -- Loaded Mission - Model	Adjusts the zoom in order to fit entirely the model of the imported mission (whose name is "Loaded Mission") in the 3D view: 

Option	Use
Fit -- Loaded Mission - Trajectory	<p>Adjusts the zoom in order to fit entirely the trajectory of the imported mission (whose name is "Loaded Mission") in the 3D view:</p> 
Fit _ Celestial body	<p>Allows you to select a celestial body in order to fit it entirely in the 3D view. The available planets are declared in the trajectory root element.</p>
Fit All	<p>Adjusts the zoom in order to fit entirely the entire scene (models, trajectories and celestial bodies) in the 3D view</p>

Camera Lock in options

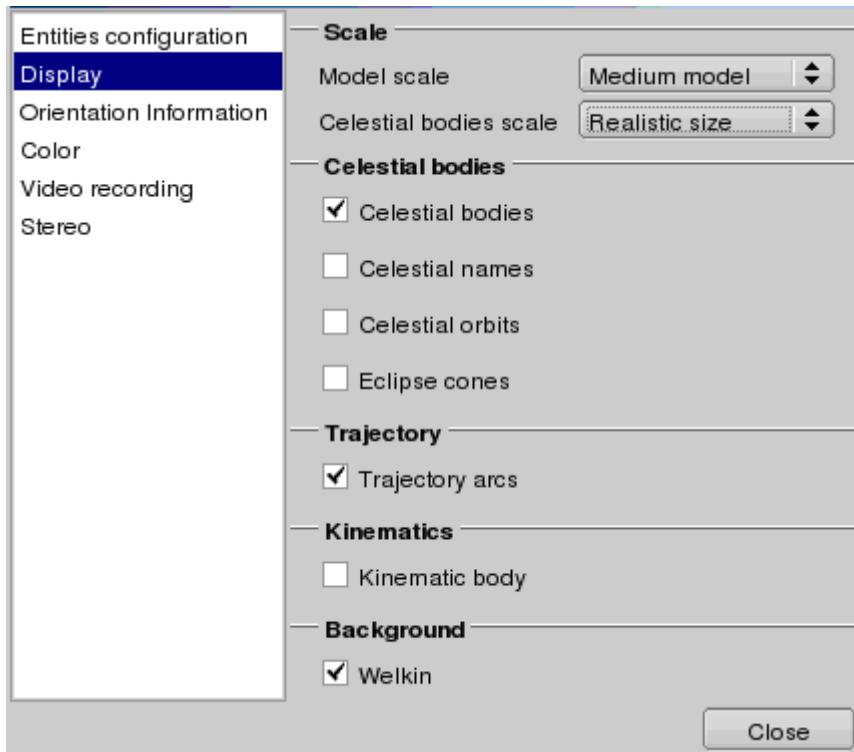
SYSTEMA allows you to choose the referential frame to be used when positioning camera:

Option	Use
Planet inertial frame	<p>Allows you to select a celestial body to display the mission in the corresponding <i>inertial system</i>. In the displayed scene, the camera is fixed in the planet inertial frame.</p>
Planet rotational frame	<p>Allows you to select a celestial body to display the mission in the corresponding <i>rotational system</i>. In the displayed scene, the camera is fixed in the planet rotational frame.</p>
Local orbital frame	<p>Allows to fix the camera in the local orbital frame of the trajectory.</p>
Model frame	<p>Allows you to display scene, with the camera fixed in the model system frame.</p>
To velocity direction	<p>Allows to fix the camera to the velocity direction</p>
From velocity direction	<p>Allows to fix the camera from to the velocity direction</p>

From Sun or planet direction	Allows to fix the camera from to the Sun or <i>planet</i> direction
To Sun or planet direction	Allows to fix the camera to the Sun or <i>planet</i> direction

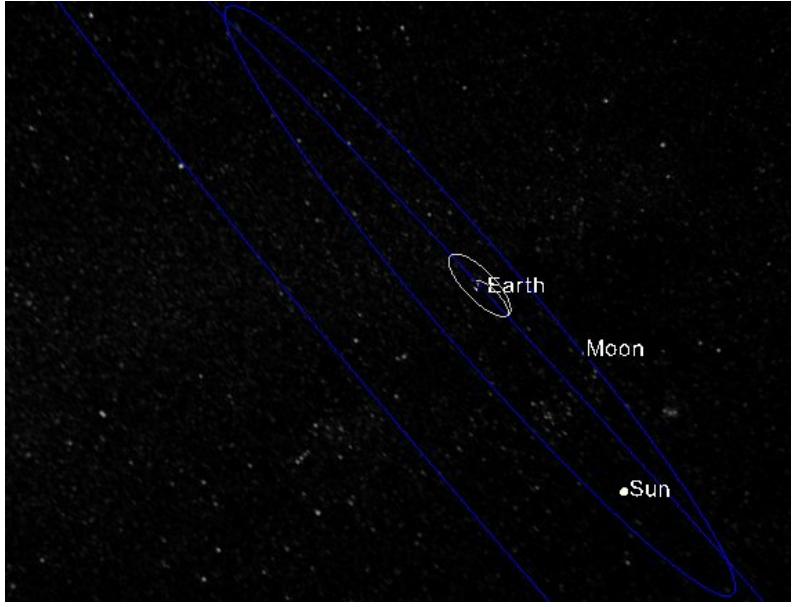
As many options as necessary exist according to the available planets declared in the trajectory root element.

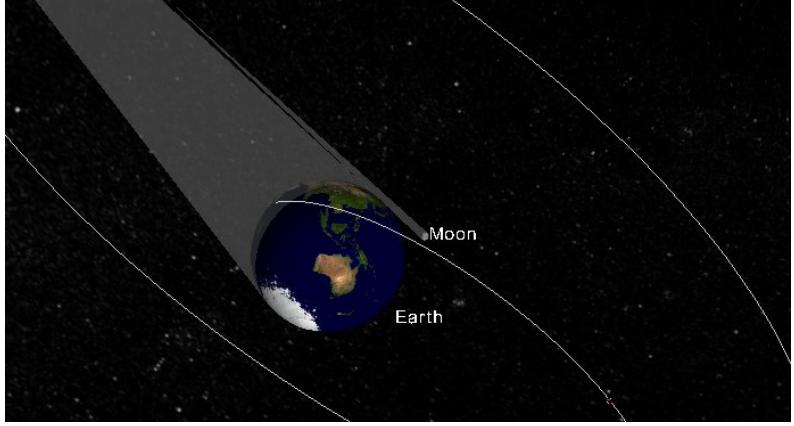
Configuration window specific options



Display options

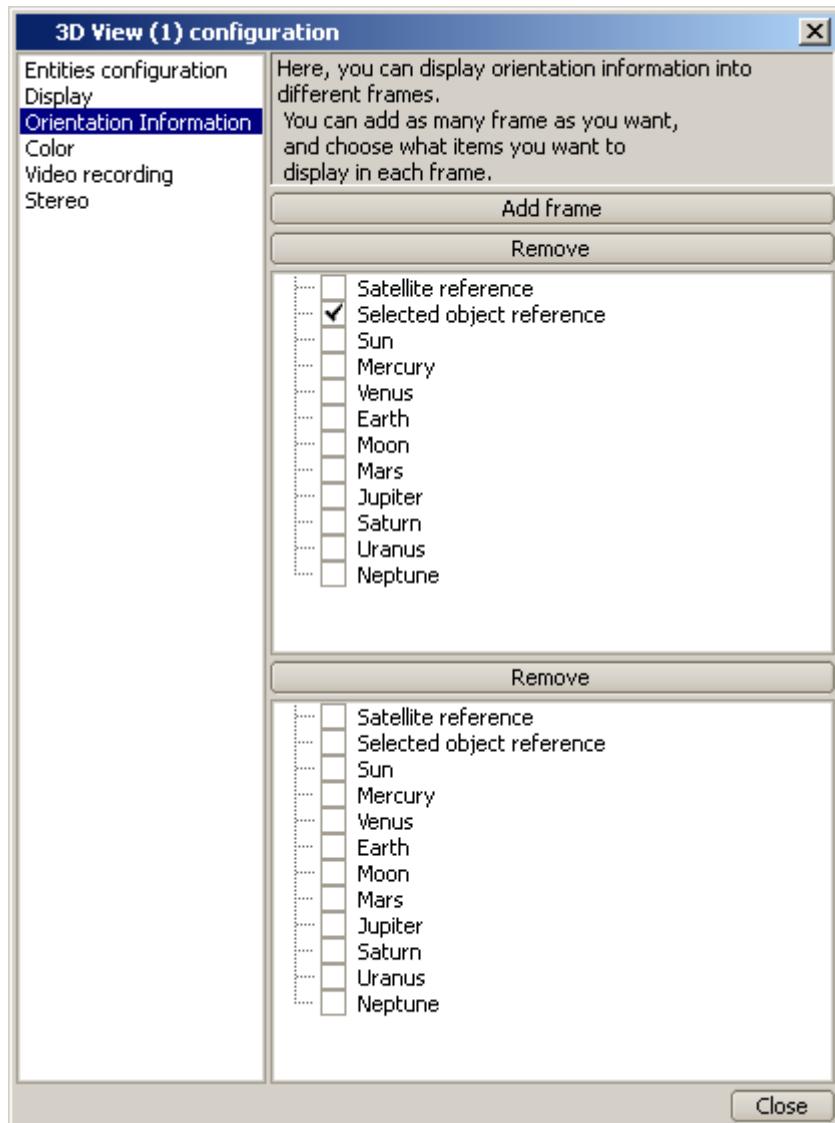
Option	Use
Celestial bodies	By default, the box is checked (celestial bodies are shown). Uncheck the box to hide the celestial bodies associated with the mission in order to show only the trajectory and the model. Check the box to show the celestial bodies associated with the mission.
Celestial names	By default, the box is unchecked (celestial names are hidden). Check the box to display the name of every celestial body associated with the

Option	Use
	<p>mission</p> 
Celestial orbit	<p>By default, the box is unchecked (celestial names are hidden.). Check the box to display the orbits (in blue) of every celestial body associated with the mission.</p>  <div data-bbox="647 1612 1429 1724" style="border: 1px solid red; padding: 10px;"> <p>⚠ Computing the orbits of every celestial bodies could be quite long the first time this option is activated</p> </div>
Eclipse cones	<p>By default, the box is unchecked (eclipse cones are hidden). Check the box to display the eclipse cone:</p>

Option	Use
	
	<p>This option can be used to see which elements are in shadow and when.</p>
Trajectory Arc	<p>Checked by default. Allows you to display, or not, the arc trajectories you defined.</p>
Kinematic bodies	<p><i>Unchecked by default.</i> Allows you to display kinematics information, like the direction of the referentials, pointing vectors...</p>

Orientation Information

The user may add as many frames as necessary to display orientation information. He can choose the items that he want to display in each frame.



Color

Option	Use
Color	<p>Allows to select the background or text colors by clicking the background color or the text color.</p>

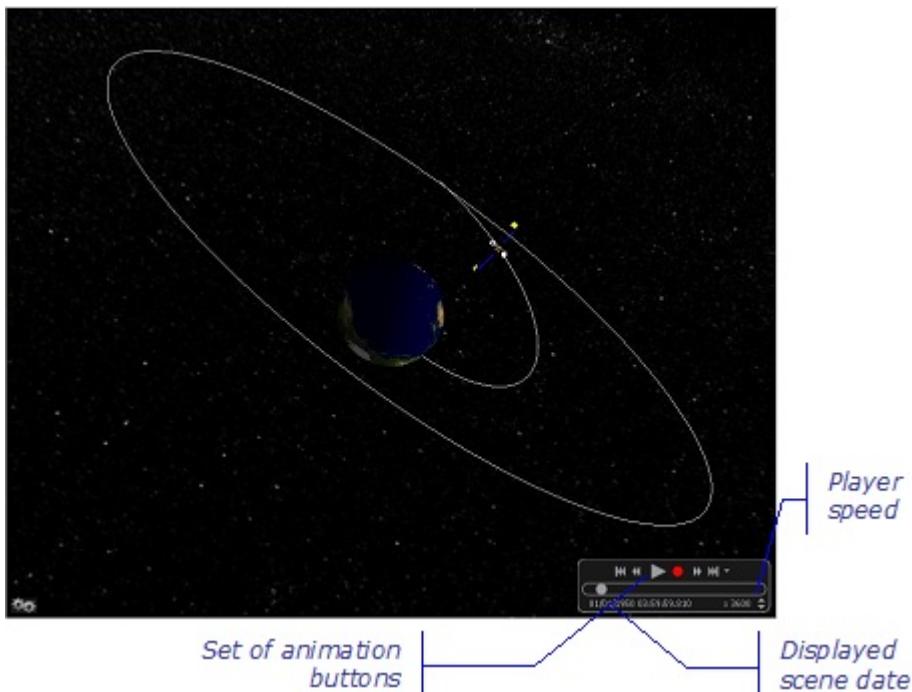
Video recording

To record a video, refer to the chapter "General User Interface", § Recording a video.

Interactive 3D animation tool

In the 3D views, you can interactively display an animation of the mission: the SYSTEMA geometrical model moves along the trajectory according to the kinematic associations.

At the bottom of the 3D view, a set of buttons allows the animation to be run:



Note: The animation tools (set of animation buttons, player speed) are the same as those available in the Trajectory tab.

The set of animation buttons

The animation buttons are described in the following table:

Button	Use
	Starts running (or re-starts) the animation
	Stops running the animation (just pauses it, time and position are kept)
	Starts recording video
	Moves the spacecraft to the previous event (arc, penumbra or eclipse)
	Moves the spacecraft to the start date of the first arc

Button	Use
	Moves the spacecraft to the next event (arc, penumbra or eclipse)
	Moves the spacecraft to the end date of the last arc
	Allows to select the type of event (arc, penumbra or eclipse)

The player speed

To modify the player speed :

1. To increase the player speed, click .
2. To decrease the player speed, click .

Working with missions

General

When working in the **Mission** tab, you can open an existing mission or create a new one.

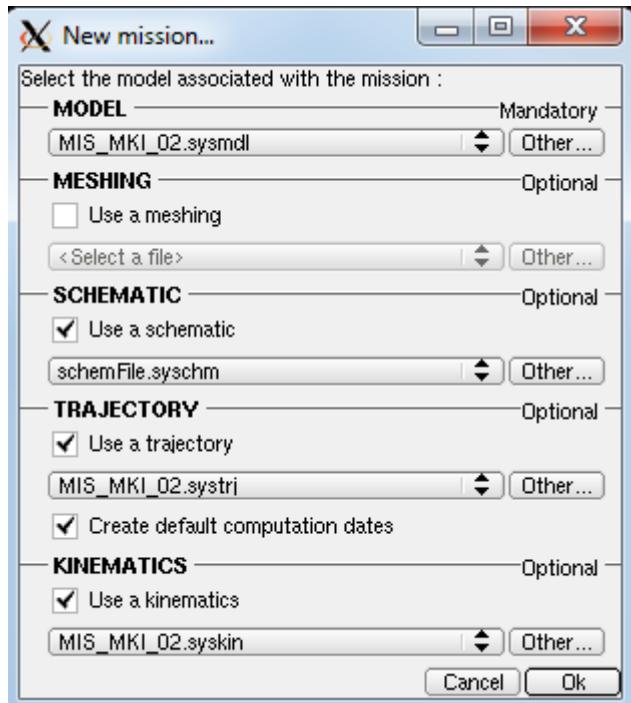
When a mission is opened, you can:

- load it in any 3D view
- modify it,
- display an animation of the mission.

Creating a new mission

To create a new mission:

- In the **File** menu of the **Mission** tab, select the **New** option. A window is displayed with the linked files (the geometrical model, the meshing associated to this model, the schematic, the trajectory, the kinematic model) selected by default for the new mission. The user can choose other files in the list of loaded models or through the **Other** option.



- Click **OK** to confirm the new mission creation.

Result: In the viewer area, a new 3D view is opened.
You can then build the mission and save it.

Opening an existing mission

You can open a mission which has previously been saved within a ".sysmis" file.

To edit an existing mission:

- In the **File** menu of the **Mission** tab, select the **Open** option and select the file containing the required mission.

Result: In the viewer area, a new 3D view is opened with the mission displayed in it.
In the browser window, the hierarchical organization of the model associated with the mission is displayed. The previously edited missions can be accessed from the list of missions.
You can then modify the mission and save your modifications.

Loading a mission in a 3D view

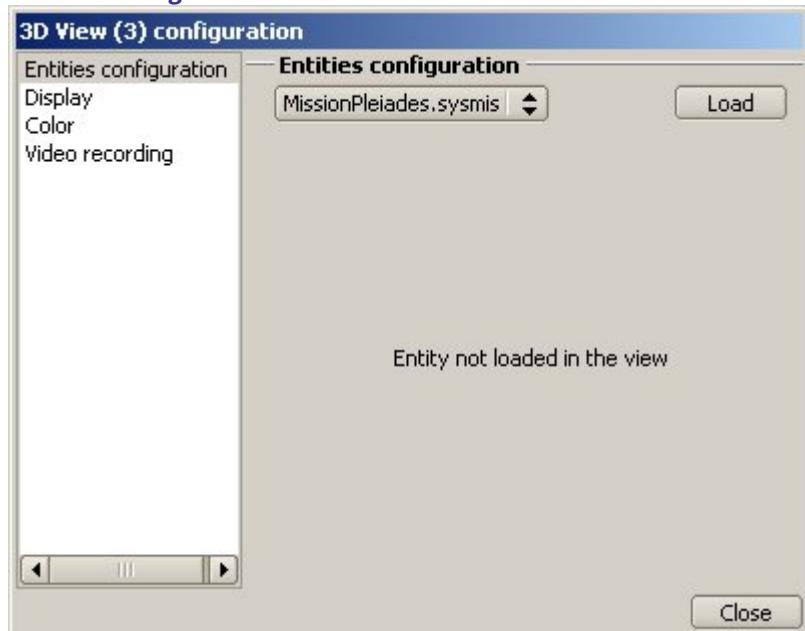
In the 3D views, you can load missions which have been opened.

	<i>Note: Several missions may be loaded in the same 3D view.</i>
--	--

To load a mission in a 3D view:

- In the 3D view, in the right click menu:

1. Select Configuration window.



2. Select the mission to load.

3. Click **Load.**

Result: The selected mission is then displayed in the 3D view.

Modifying a mission

You can modify a mission by:

- modifying the attributes of the mission root,
- editing and modifying the attributes of the model elements.

Adding/ Deleting a sequence

There are three different ways to create sequences in the current mission model:

- within the timeline browser window use the **Create sequence intervals** option from the right click menu.
- use the **Create sequence intervals** option from the **Mission** menu.
- use the icon of the **Sequences** line in the timeline window :: refer to "How to create a component from the timeline window" in "Working with the timeline window" in the "General User Interface"

Two options are available in all cases:

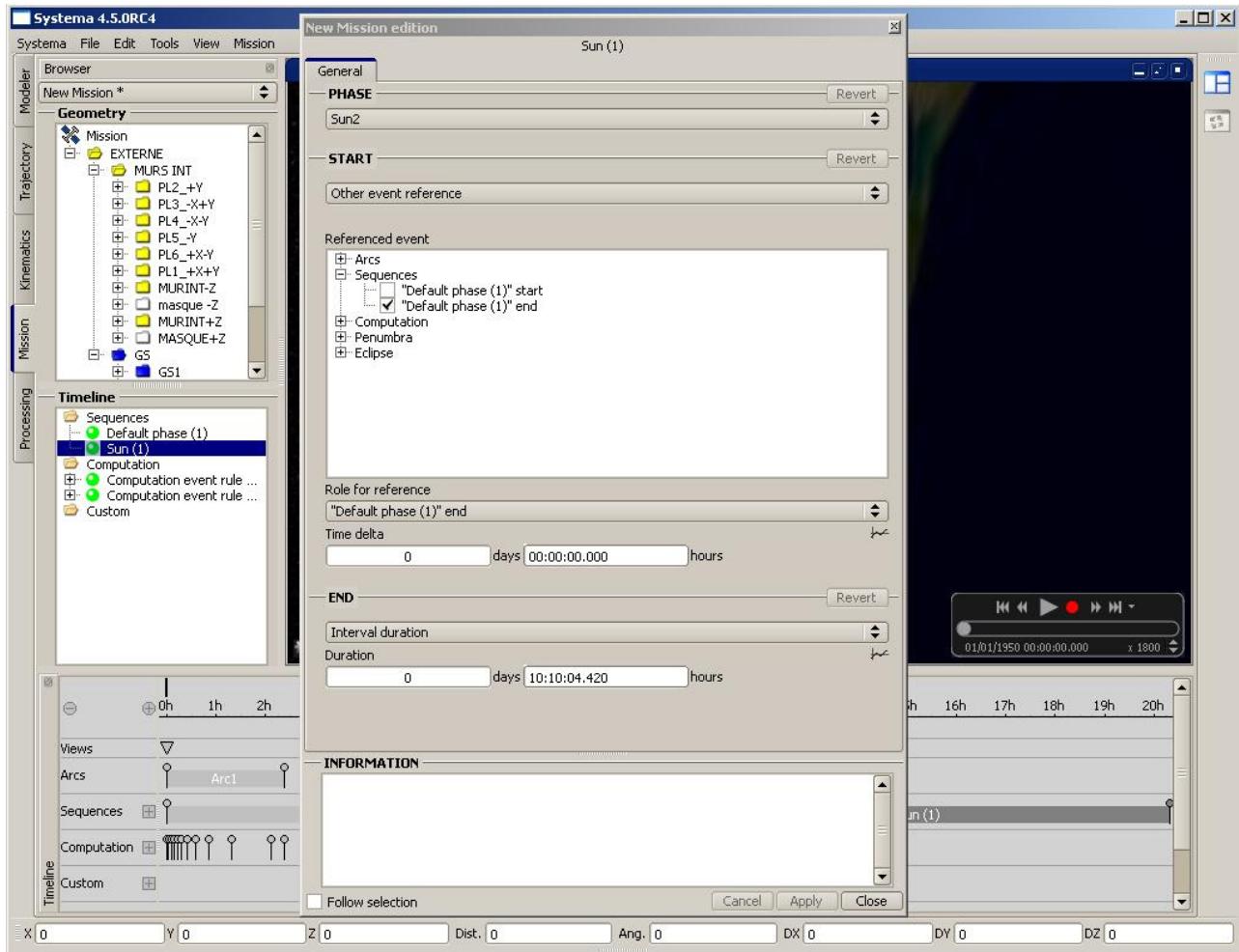
Option	Use
Interval	Creates an interval defined by start and end dates. Each date can be defined by an absolute date or the reference to another event
Intervals set	Creates a set of intervals defined by an intervals computation method

The new sequence is automatically created at the end of the list of sequences and the sequence edition window is opened'. It references the default phase. If it is not the first sequence, the start date automatically references the end of the previous sequence in the list. The name is automatically computed from the phase name with an index

and cannot be changed.

A sequence can be deleted by

- right click menu on it in the timeline browser and access to the **Delete** option
- select it in the timeline window or in the timeline browser and clicking the [Delete] button or use the **Delete** option of the **Edit** menu



Adding/ Deleting computation dates (computation events and computation event sets)

There are three different ways to create computation dates in the current mission model:

- within the timeline browser window use the **Create computation events** option from the right click menu.
- use the **Create computation events** option from the **Mission** menu.
- use the  icon of the **Computation** line in the timeline window :: refer to "How to create a component from the timeline window" in "Working with the timeline window" in the "General User Interface"

Two options are available in all cases:

Option	Use
Event	Creates an event defined by an absolute date or the reference to another event

Events set	Creates a set of events defined by the intervals definition and the events computation method
-------------------	---

A computation event or event set can be deleted by

- *right click menu* on it in the timeline browser and access to the **Delete** option
- select it in the timeline window or in the timeline browser and clicking the *[Delete]* button or use the **Delete** option of the **Edit** menu. A generated event in an event set cannot be deleted.

	<i>Note: refer to "Attributes of an event" and "Attributes of an event set" in "Working with the timeline window" in the "General User Interface" chapter to get the description of the attributes of the events and events sets._</i>
---	--

Adding/ Deleting custom events, event sets, intervals and interval sets

There are three different ways to create custom components in the current mission model:

- within the timeline browser window use the **Create custom events** or **Create custom intervals** options from the right click menu.
- use the **Create custom events** or **Create custom intervals** options from the **Mission** menu.
- use the  icon of the **Custom** line in the timeline window : : refer to "How to create a component from the timeline window" in "Working with the timeline window" in the "General User Interface"

Depending of the selected menu, options are available :

Option	Use
Option	Use
Event	Creates an event defined by an absolute date or the reference to another event
Interval	Creates an interval defined by start and end dates. Each date can be defined by an absolute date or the reference to another event
Events set	Creates a set of events defined by the intervals definition and the events computation method
Intervals set	Creates a set of intervals defined by an intervals computation method

A custom event or event set or interval or interval set can be deleted by

- *right click menu* on it in the timeline browser and access to the **Delete** option
- select it in the timeline window or in the timeline browser and clicking the *[Delete]* button or use the **Delete** option of the **Edit** menu. A generated event in an event set and a generated interval in a interval set cannot be deleted.



Note: refer to "Working with the timeline window" in the "_General User Interface" chapter to get the description of the attributes of the events, intervals, events sets and intervals sets._

Building a mission

How to build a mission

You build a mission by defining its components, i.e.:

- model, trajectory, kinematic model ..., at root level,
- motion parameters at the level of the SYSTEMA model elements.

How to define the mission components at root level

1. Open the edition window of the mission at root level.

2. In the **Attributes** tab, select the main components of the mission:

- the model,
- the meshing
- the schematic
- the trajectory,
- the kinematic model,
- if necessary, the link between the mission root and a kinematic body of the selected diagram,
- the transformation, which changes the main axis system of the model to the axis system of the linked kinematic body.

3. In the *applicative* tab, select the applicative parameters of the mission: the computation points and the meshing to which processes will be applied.

Result: Once a model, trajectory, kinematic model or meshing has been associated with the mission, it is loaded in the corresponding SYSTEMA tab.



Note: While a mission is opened, the model, trajectory, kinematic model or meshing related to the mission, cannot be saved in a new file ("save as" function) in the appropriate SYSTEMA tab.

How to define the motion parameters of a model element

1. Open the edition window for the relevant model element.

2. If necessary, select the kinematic body you want to associate with the model element.

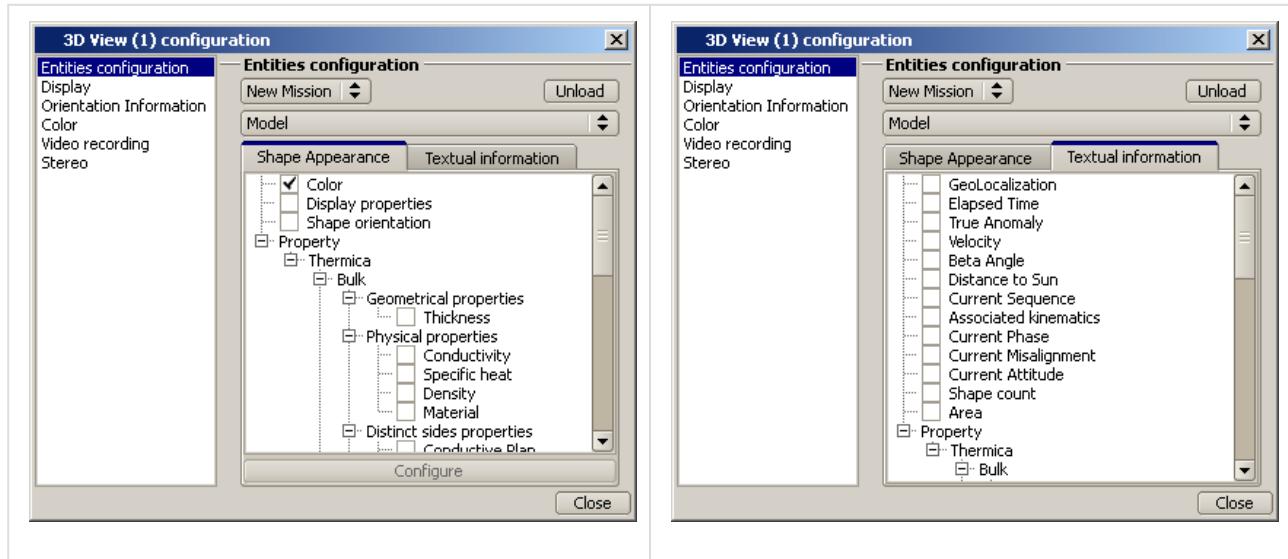
3. If necessary, define the model element's initial move.

Viewing mission textual information

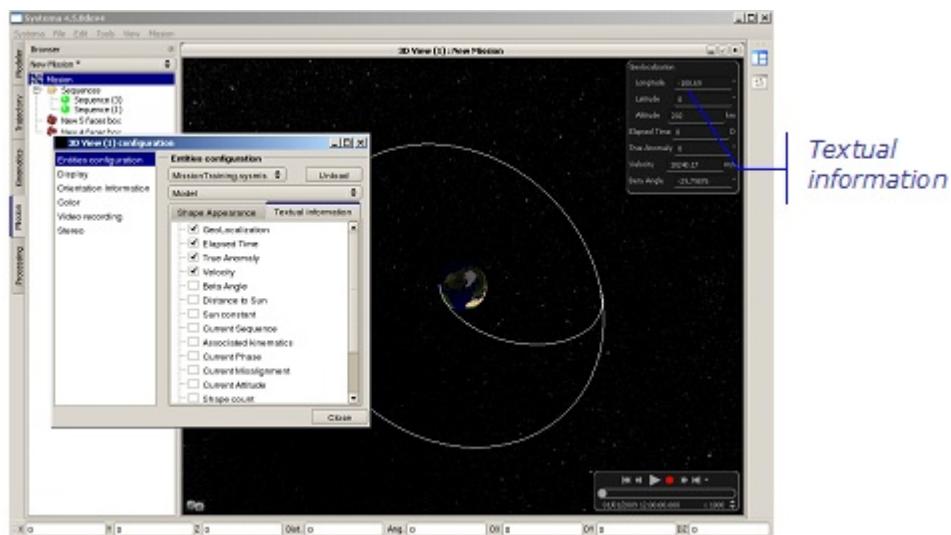
You can display textual information for every mission opened in the **Mission** tab.

How to view textual information for a given mission

1. Open the mission whose textual information you wish to view. A 3D view is automatically created
2. Open the configuration window. A tab named textual information is displayed in 'Entities configuration' option. Click on 'Textual information' tab. A window containing the list of available textual information appears.



1. Check a textual information in the list. The selected information now appears in the viewport in a special area Repeat the operation by checking other textual information. Any information can be removed from the special area by unchecking it in the list.



The Mission Log

The Mission Log assemble all the information contained by the Mission, so that the user can check the validity of the different inputs, and use the trajectory and kinematics information to perform light system analysis. The Mission Log is exported as an HTML file, easily readable by the user, and fully compatible with spread sheet applications like Excel. This allows the user to recomputed data based on the information provided in the Mission Log, display graphs, etc.



The Mission Log can be exported in the Mission tab, in the File/Mission Log menu, or using the icon . The Mission Log is also automatically exported when executing processes, in the Process tab (see the Process management chapter)

Content of the Mission Log

The Mission Log contains:

- The generation date, and the path of the files used in the mission
- The list of the different shapes of the modeler, their geometric, meshing parameters, as well as the materials applied on it
- The list of the arcs of the trajectories, with their names and parameters
- The ephemeris of the trajectory, containing, for each computation date of the mission:
 - The computation point name
 - The date, in calendar, Gamma50, J2000
 - The elapsed time, in days and seconds
 - The name of the current arc
 - The longitude, latitude, altitude of the spacecraft
 - The velocity norm, the velocity vector, in M50 and J2000
 - The distance to the sun
 - The true anomaly
 - The beta angle (angle between the orbit plane and the sun direction)
 - The sun direction (from planet reference frame, not satellite)
 - The solar declination (angle between the rays of the Sun and the plane of the Earth's equator)
 - The solar zenith angle (angle between the nadir direction and the sun direction) (Available since Systema-4.9.2)
 - The illumination (light, eclipse, or penumbra and if relevant the occulting body)
 - The orbit angular momentum (in ecliptic M50)
 - The transformation between the Sun and the different planets of the trajectory (from ecliptic M50 to M50)
 - The planet velocity vectors (in ecliptic M50)
- The list of the kinematic bodies
- The transformation from the spacecraft local reference frame to the different kinematic bodies local frames at each computation date
- The misalignment of the different kinematic bodies at each computation date. The misalignment is the angle, between the vector to be pointed (e.g. the Sun, the Earth) and the vector actually pointed by the kinematic body. This applies only for kinematic bodies with pointing laws, otherwise, the misalignment is equal to 0.
- The ephemeris for the mission, containing, for each computation date:
 - The name of the current kinematic phase
 - The name of the current sequence
 - The sun constant
 - The transformation between the Sun and the spacecraft local frame, in ecliptic M50, with the distance from the Sun
 - The transformation between the different planets and the spacecraft local frame, in M50, J2000, and rotational frame

Transformations

The transformation are exported with a translation vector (Tx, Ty, Tz) in meters, and the three Euler angles XY'Z" (intrinsic rotations), in degrees.

References frames

Different reference frames are used :

- M50 : Centered on the Earth, the x-axis is aligned with the vernal equinox on 1 January 1950 at midnight, the z-axis is aligned with the Earth's spin axis.
- J2000 : Centered on the Earth, the x-axis is aligned with the vernal equinox on 1 January 2000 at noon, the z-axis is aligned with the Earth's spin axis.
- Ecliptic M50 : Centered on the Sun, the x-axis is aligned with the vernal equinox on 1 January 1950 at midnight, the z-axis normal to the ecliptic.

Process management

Process management

Overview

Introduction

This chapter describes how to use SYSTEMA in order to create, load, save, view and run processes. It also describes how to apply the process results to models and missions.

With SYSTEMA, you use the **Processing** tab to work on processes. Everything described in this chapter takes place within the **Processing** tab.

In this chapter

This chapter covers the following topics:

- Process management
 - Overview
 - Introduction
 - In this chapter
 - The Processing tab
 - What is the Processing tab for?
 - The different parts of the Processing tab
 - Example
 - The menu bar
 - The Process area
 - The Result Infos and Variables area
 - The Run area
 - The Result area
 - Processes in SYSTEMA
 - What is a processing diagram?
 - The process in the Process area
 - Run parameters attributes
 - File attribute
 - Working with process
 - General
 - Creating a new process
 - Opening an existing process
 - Modifying a process
 - Building a process
 - How to build a process
 - Adding basic processes
 - How to add a basic process
 - To add a basic process from the Basic process browser:
 - To add a basic process from the Process menu:
 - Result
 - Organizing basic processes
 - Joining files
 - How to join files
 - To join two files:

- [Splitting files](#)
- [How to split a file](#)
 - To split a file using the right click menu:
 - To split a file using the files joining link
- [How to deactivate/activate a file](#)
 - To deactivate a file
 - To activate a file
- [Edit an input file](#)
- [Running a process and editing results](#)
 - [Principles](#)
 - [Define the result parameters](#)
 - [Prerequisite](#)
 - [How to define the result parameters](#)
 - [Define variables](#)
 - [Overload variables](#)
 - Overload variables using the edition window
 - Overload variables importing a file
 - Export the file with overload variables
 - [Run the current process](#)
 - How to run the current process
 - Result
 - [Export mission data](#)
 - [Editing process results](#)
 - How to load run results
 - How to edit run results
 - How to unload run results
- [Using process results](#)
 - [Using process results in the Modeler tab](#)
 - How to read result data directly from file
 - How to display a result in a 3D view
 - How to display some application properties on computation rays, if available
 - [Using process results in the Mission tab](#)
 - How to display a result on a model in the Mission tab

The Processing tab

What is the Processing tab for?

You can use the **Processing** tab of the SYSTEMA application to:

- open existing processing diagrams and modify them when necessary,
- create new processes by combining applicative processes,
- run processes on missions.



*Note: You define processes in the **Processing** tab. Once a process has been defined, you can then apply it to missions which have been loaded in the **Mission** tab.*

The different parts of the Processing tab

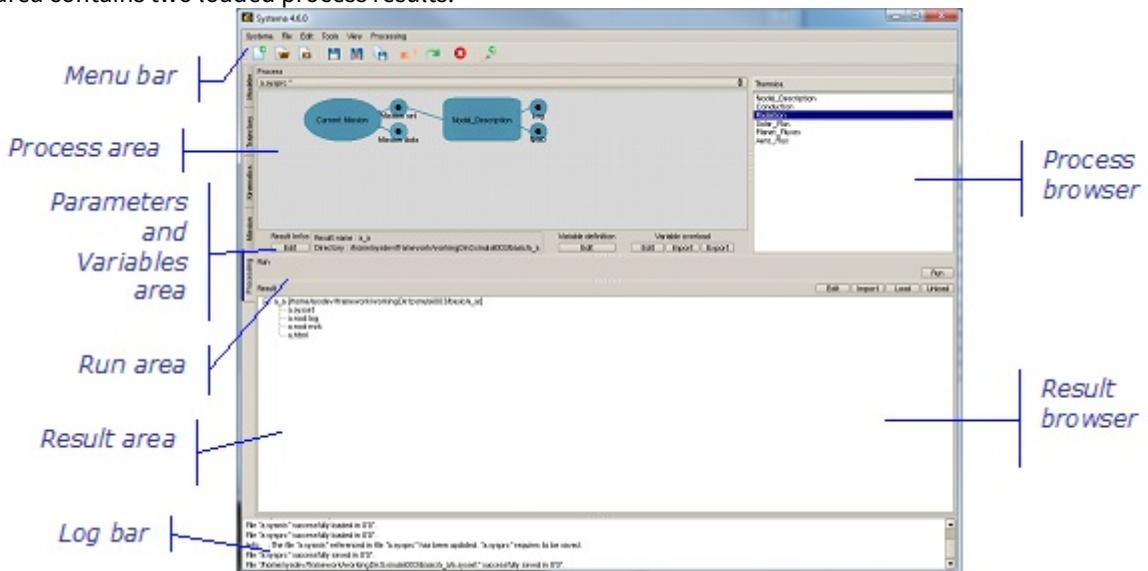
The **Processing** tab is divided into three main parts:

- the *Process* area, containing the process description,

- the *Run area*, containing the elements necessary to run a process,
- the *Result area*, from which process results can be loaded and edited.

Example

The following diagram shows an example of the **Processing** tab. The *Process area* contains a process, the *Result area* contains two loaded process results:



The menu bar

The menu bar contains:

- Five menus which are common to all SYSTEMA tabs: the **Systema**, **File**, **Edit**, **Tools**, and **View** menus.

For more information, refer to the "General User Interface" chapter

- One specific menu, the **Process** menu, which can be used to build the current process.

The following table presents the options available from the **Process** menu:

Option	Use
Import a process (for each applicative process of each application)	Adds an applicative process to the current process with default attribute values <i>Available processes here are the same as those in the process browser displayed in the Process area</i>
Insert current mission (only for some shapes)	Adds a "Current mission" process to the current process. A maximum of 2 "Current Missions" can be added to a process.

The Process area

The *Process area* is divided into two parts:

- the left-hand part contains the *process viewer* which is used to display, build and modify the process diagram for the current process;

- the right-hand part contains the *basic process browser* which displays the list of basic processes that can be added to the current process.

A zoom is available to increase the size of the diagram area.

The Result Infos and Variables area

The *Result Infos and Variables* area is used to:

- define the result file's name and directory,
- define variables usable in processing,
- overload values of all variables defined in other entities (modeler, meshing, schematic, trajectory, kinematics, mission) and in processing entity.

The following table presents the elements of the *Parameters and Variables* area:

Element	Use
Result infos area	Displays the result file name and directory <i>In this area, the names cannot be changed</i>
 Edit button (Result Infos)	Is used to define the result name file and directory
 Edit button (Variable definition)	Is used to define variables
 Edit button (Variable overload)	Is used to overload values of variables defined in all entities
 Import button (Variable overload)	Is used to import a file to overload several variables
 Export button (Variable overload)	Is used to export a file with all variable overloaded

The Run area

The *Run* area is used to:

- run the current process on the selected mission.

The following table presents the elements of the *Run* area:

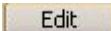
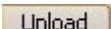
Element	Use
 button	Applies the current process to the selected mission and runs

The Result area

The *Result* area is divided into two parts:

- the upper part contains buttons which are used to edit, import, load and unload result files;
- the lower part contains the *result browser* which displays the list of loaded results.

The following table presents the buttons of the *Result* area:

Button	Use
 Edit button	Is used to edit result files
 Import button	Is used to import any type of file <i>The corresponding files are then displayed in the result browser</i>
 Load button	Is used to load result by selecting a ".sysres" file <i>The corresponding files are then displayed in the result browser</i>
 Unload button	Is used to unload a result which has been previously loaded in the <i>result browser</i>

Processes in SYSTEMA

What is a processing diagram?

A SYSTEMA processing diagram is a sequence of processes which are specific to the associated applications.

The process in the Process area

Each basic process generally consists of:

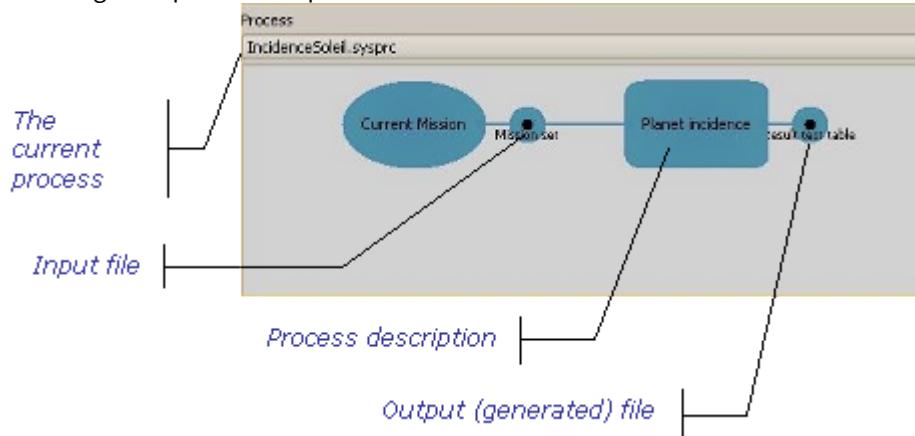
- input file(s),
- a process description,
- output file(s).

Some processes include the current mission as input, for certain types of processes, two current missions are required.

The color of the entry/output icon indicates whether there are any warnings (orange) or errors (red) associated with **the corresponding file**:

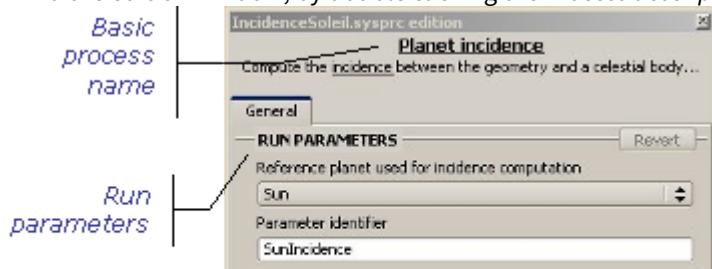
Icon	Meaning
	The file exists in the expected format
	The file is not in the expected format
	The input/output file is optional and can be deactivated
	The input/output file is optional and has been deactivated

The following example shows a process in the *Process* area:



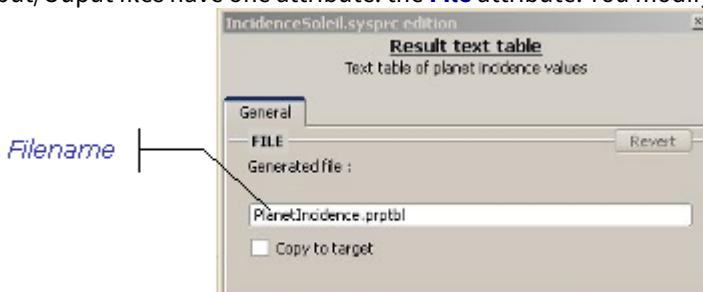
Run parameters attributes

Each *Process description* has **Run parameters** attributes depending on the purpose of the process. You modify them via the edition window, by double clicking the *Process description* box:



File attribute

The Input/Ouput files have one attribute: the **File** attribute. You modify it via the edition window:



By default, the output files are generated, in the *Result* directory (defined in the *Run* area). This directory can be changed by ticking the **Copy to target** checkbox.

Working with process

General

In the **Processing** tab, you can open an existing process or create a new one.
When a process is opened, you can:

- modify it,
- run it,
- consult the process results.



*Note: The color of the entry/output icon indicates if there are any warnings (orange) or errors (red) associated with **the corresponding file**.*

Creating a new process

To create a new process:

- In the **File** menu of the **Processing** tab, select the **New** option.

Result: The *Process* area displays an empty editable space.

You can then build the process and save it.

Opening an existing process

You can open a process which has previously been saved within a ".sysrc" file.

To edit an existing process:

- In the **File** menu of the **Processing** tab, select the **Open** option and select the file containing the required process.

Result: The process diagram is displayed in the *Process* area.

You can then modify the process and save your modifications.

Modifying a process

You can modify a process by :

- adding new basic processes (refer to the "Building a process" section),
- editing and modifying input/output files and process descriptions.

Building a process

How to build a process

You build a process by adding and organizing basic processes.

Adding basic processes

There are two different ways to add basic processes:

- using the *Basic process browser*,
 - or using the **Processing** menu.

How to add a basic process

To add a basic process from the Basic process browser:

- In the *Basic process browser*, double click the basic process you wish to add to the process diagram.

To add a basic process from the Process menu:

- In the **Processing** menu, select the **Import a process** option, then select the application and the basic process.

Result

The selected basic process with its associated input/output files are added with default values. Their attributes can then be modified by editing them.

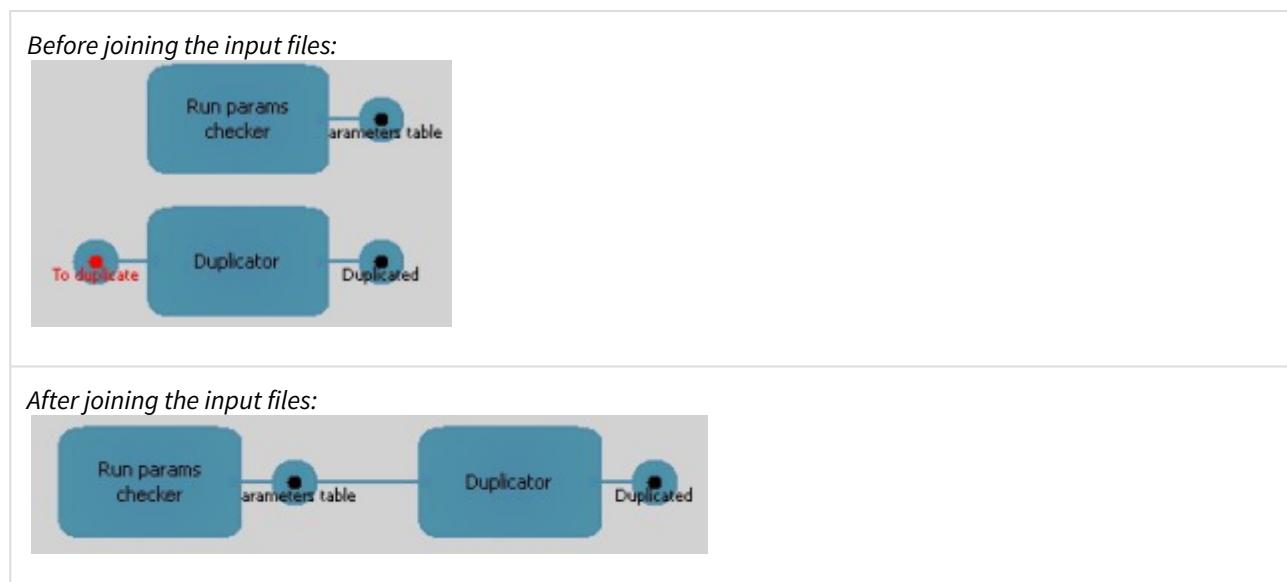
Organizing basic processes

To organize basic processes, you can join or split input/output files.

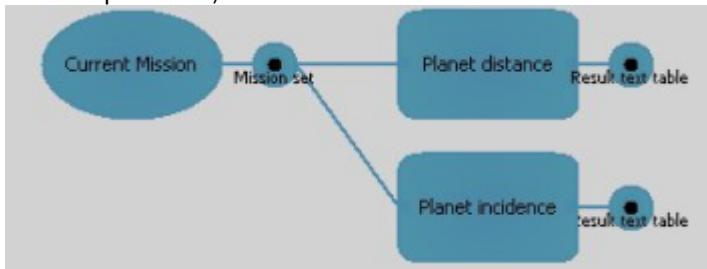
Joining files

You can join files in the following case:

When the output file of a basic process has to be used as the input file of another process (as shown in the example below)



When two basic processes have to use the same file as their input file, they are automatically joined (as shown in the example below)



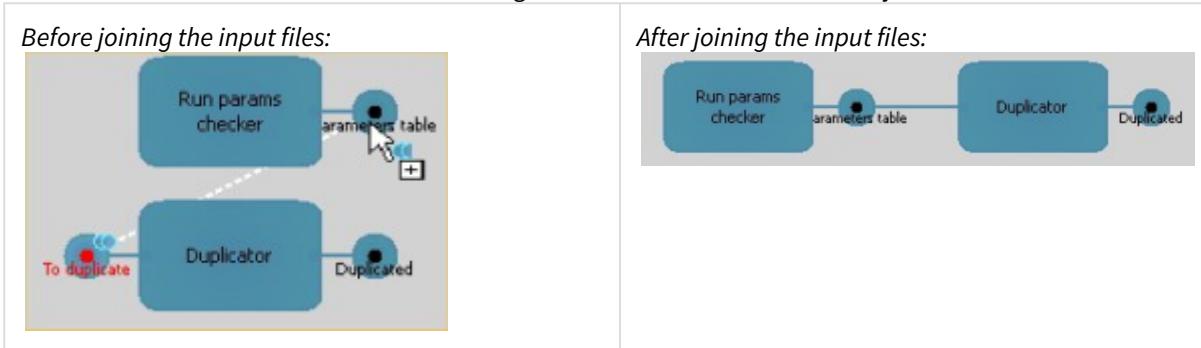
How to join files

To join two files:

1. Move the mouse to the input file.

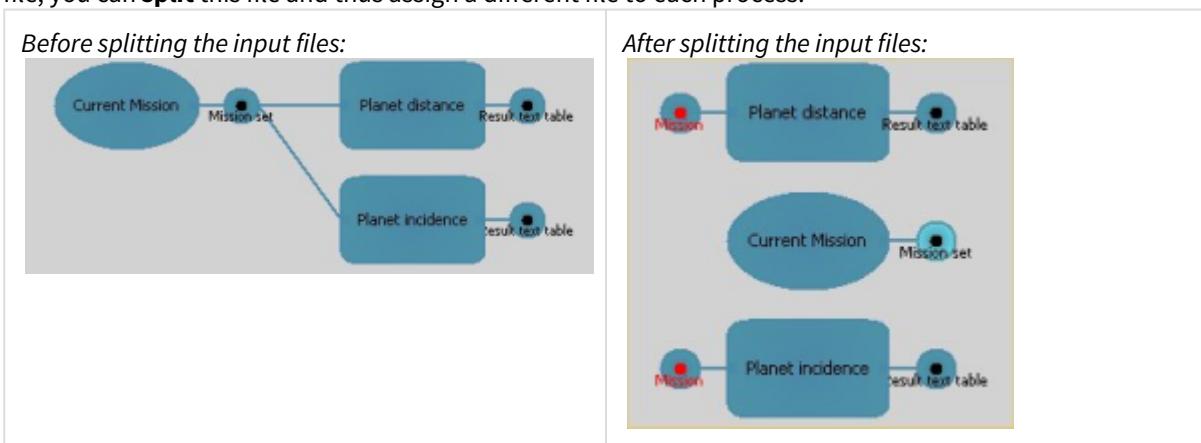


1. Click the joining icon
2. Hold down the left mouse button while moving the mouse toward the file to be joined.



Splitting files

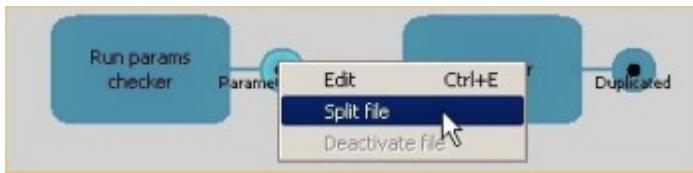
The split function is the reverse of the join function. When two basic processes initially have the same input file, you can **split** this file and thus assign a different file to each process.



How to split a file

To split a file using the right click menu:

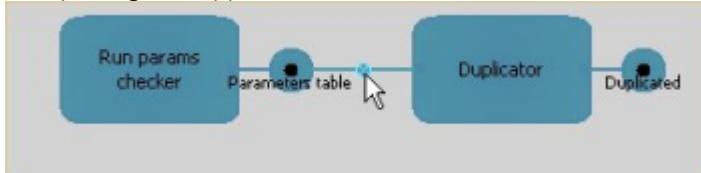
1. Move the mouse to the input file.
2. Using the right click menu, select the **Split file** option.



To split a file using the files joining link

1. Move the mouse to the files joining link.

The splitting icon appears.



1. Click  to break the link.

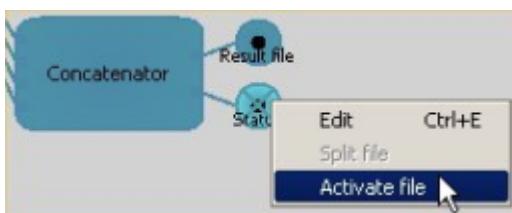
How to deactivate/activate a file

1. Move the mouse to the input/output file.
2. Using the right click menu, select the **Deactivate file/Activate file** option.

To deactivate a file



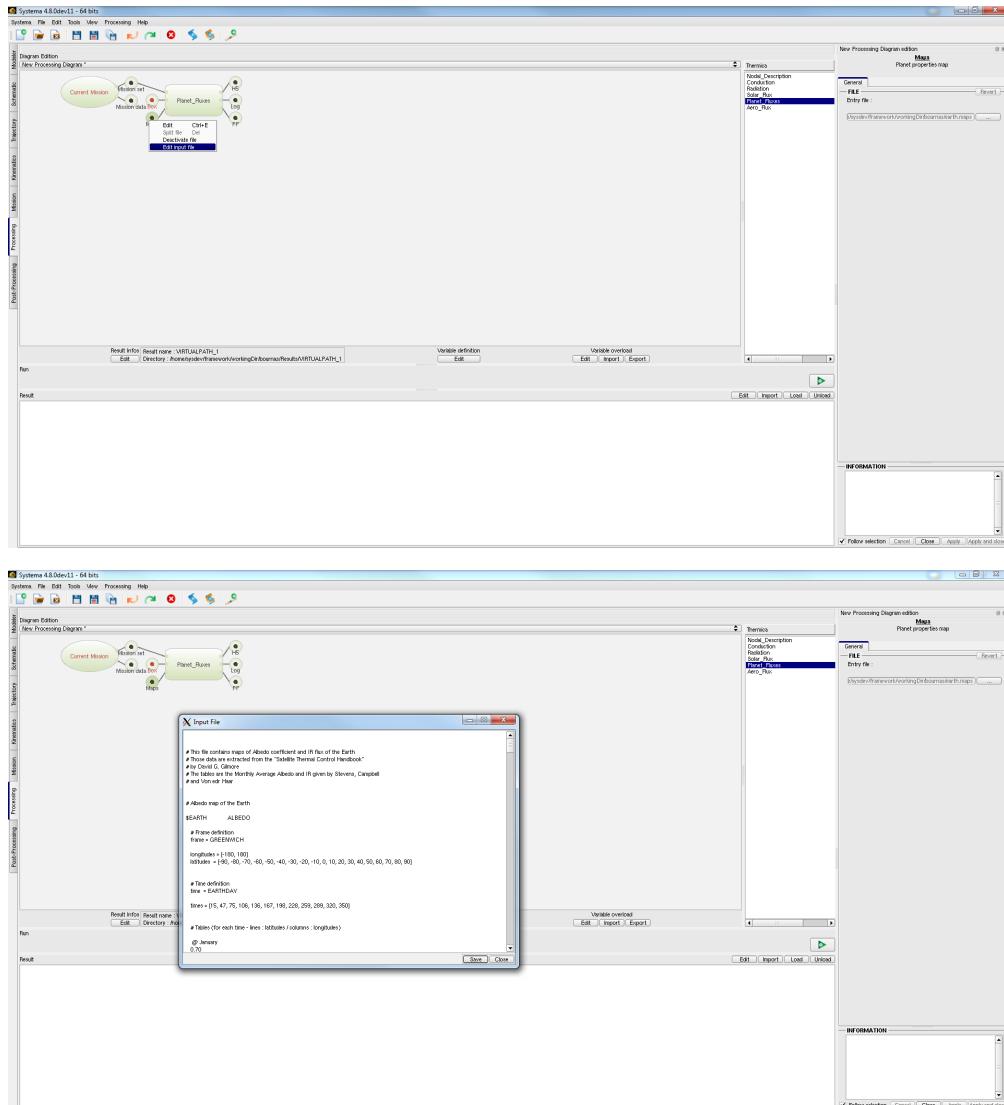
To activate a file



Edit an input file

Once you have set an input file, you can edit it directly in Systema by doing a right click on the input, and choosing the item "Edit input file".

It has to be a text file in order to be edited, and you have to activate it before the edition.



To save your modification, click on the save button of the editor.

⚠ A link to a file shall not be used as an input file.

Running a process and editing results

Principles

The general procedure for running a SYSTEMA process is as follows:

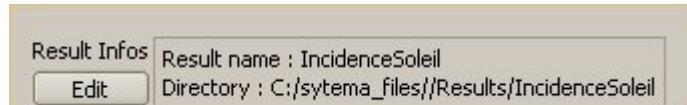
- **first step**, display the process to be run in the *Process* area;
- **second step**, define the result parameters in the *Run* area;
- **third step**, launch the process run in the *Run* area.

Define the result parameters

Prerequisite

The relevant process must be displayed in the Process area.

When a process is displayed in the *Process* area, the result name and directory are automatically displayed in the *Result Infos* zone of the *Run* area:



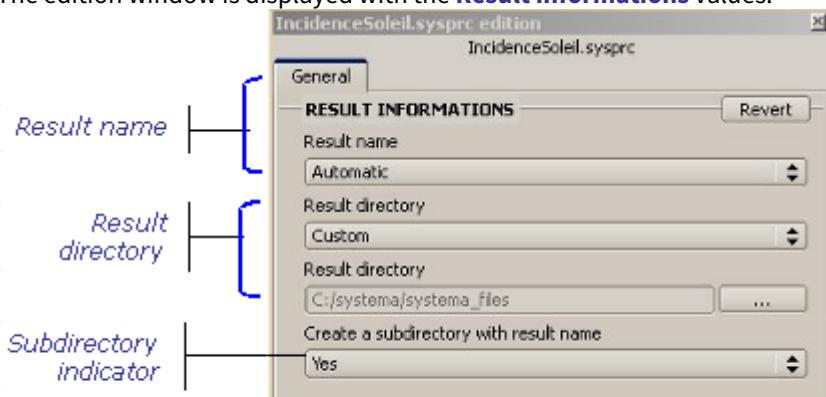
- **Result name** is the name of the ".sysres" file which is created. By default, its value is the same as the process name.
- **Directory** is the directory in which the result is created. By default, it is created in the "SYSTEMA_FILES/ Results" directory of the SYSTEMA environment with the process name.

The **Result name** and **Directory** values can be changed using the button.

How to define the result parameters

1. In the Result Infos area, click .

The edition window is displayed with the **Result informations** values.



2. If you wish to customize the result name, select "Custom" in the **Result name** field, then enter the required name in the **Result Name** field.

3. Select the required value in the **Result directory** field:

- select "Default result directory" to create the result files in the default directory (\$Home)
- select "Mission directory" to create the result files in the mission directory
- select "Processing directory" to create the result files in the processing directory
- select "Custom" to set manually the path to the result directory, select the required directory by clicking .

4. Select the required value in the **Create a subdirectory with result name** field:

- select "No", to create the result files directly in the **Result directory**;
- select "Yes", to create the result files directly in a new subdirectory of the **Result directory**, the name of which is **Result name**;
- select "Yes, with the date", to create the result files directly in a new subdirectory of the **Result directory**, the name of which is made of **Result name** followed by the run date.

5. Save the result parameters by clicking .

Define variables

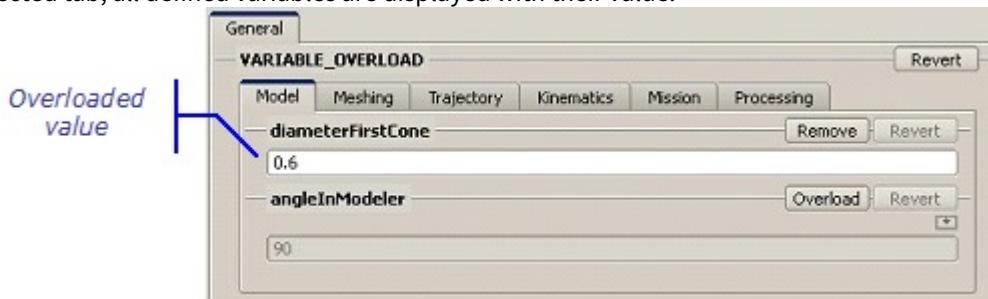
Cf [Working with variables](#)

Overload variables

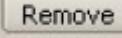
Overload variables using the edition window

In the *Variable overload* area, click  .

A *Variable Overload* area is displayed in an edition window with access to all different modules through tabs. In the selected tab, all defined variables are displayed with their value.



To overload a value, click  button and change the value.

To remove an overloaded value, click  button

To cancel the current action (overload or remove) for a variable, click  button

To validate the overloading of values, click **Apply**

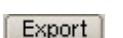
To cancel all current modifications, click **Cancel**

Overload variables importing a file

In the *Variable overload* area, click  .

Select the file to import. The file format is described in the batch command help section (process diagram execution mode).

Export the file with overload variables

In the *Variable overload* area, click  .

Select the path for the file to export. All the variables overloaded in the edit window are written in the file.

Run the current process

How to run the current process



1. In the *Run* area, click  .

The "Process run" window is displayed. It contains information about the run. By default, all processes are launched. To skip a process, unselect it in the "Process run" window.

2. Close the Process run window.

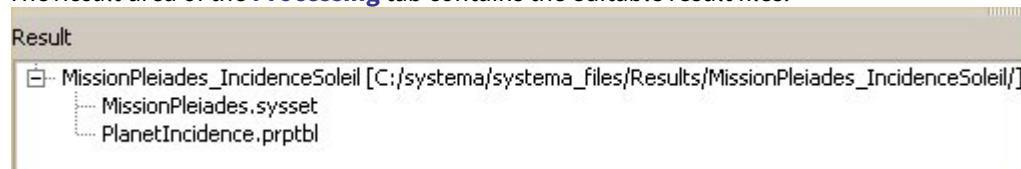
Result

The result files and directory are created in the SYSTEMA environment.

The following table shows the files which are created:

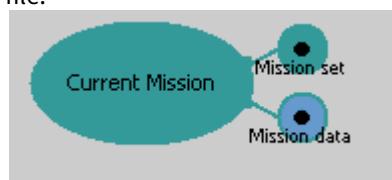
File	Content
log.txt	Information generated in the Process run window during the run execution <i>Editable file</i>
Mission.sysset file	Output file generated by a basic process and containing data on the current mission <i>File editable via the Processing tab</i>
"result name".sysres	Result data used by SYSTEMA when loading a result <i>Non-editable file</i>
"process name".prptbl	Result values (table of values) <i>File editable via the Processing tab</i>
"process name".sysprm	Run parameters <i>Editable file</i>
Mission.html file	Output file generated by a basic process and containing data on the current mission in an html format, it is the same format as the mission log in the Mission tab <i>File editable via the Processing tab</i>

The **Result** area of the **Processing** tab contains the editable result files:



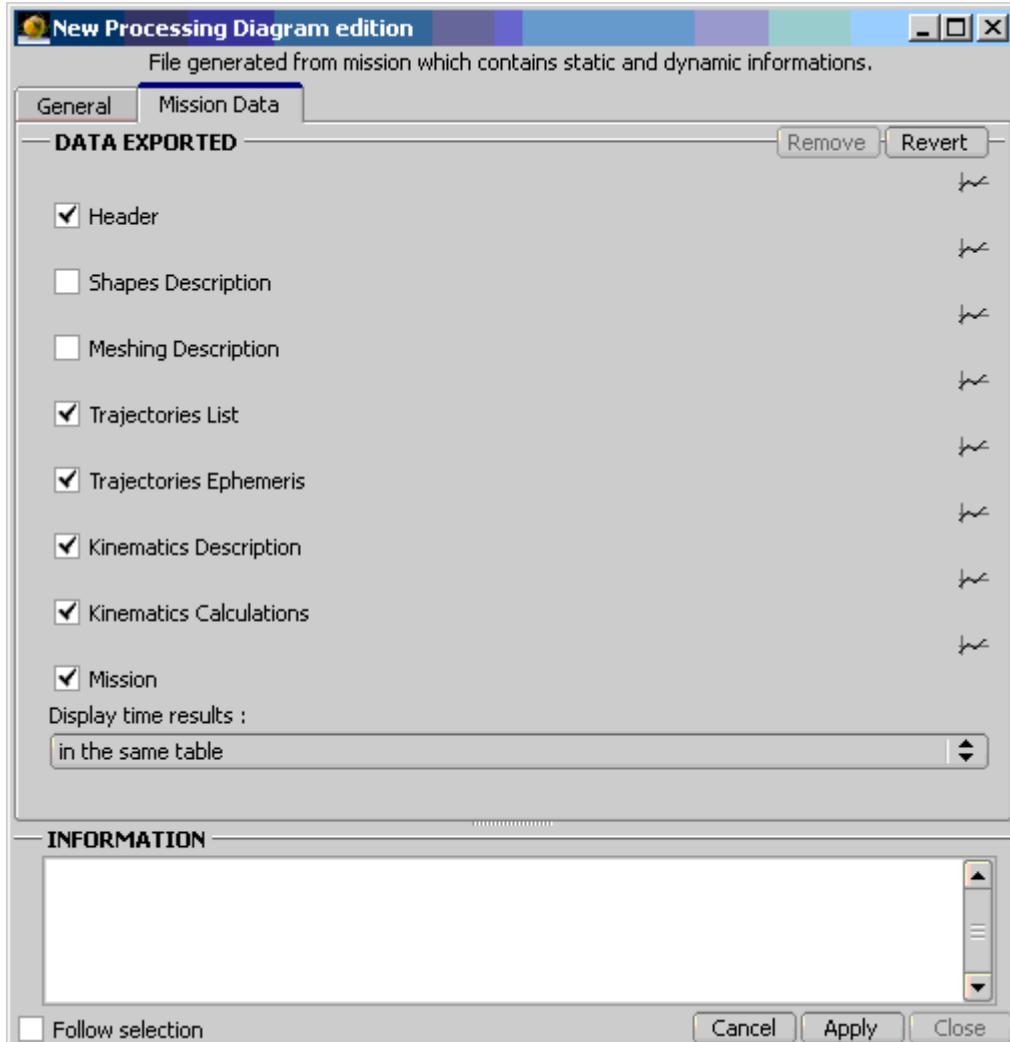
Export mission data

By default, the mission process generates 2 files, the **mission set** (.sysset) which is used by applications and the **mission data** (.html) which can be read by users. The mission data file can be deactivated if not used, it is not used by applications. The file is automatically deactivated if a meshing file is selected in the process instead of a mission file.



This file can be opened directly with excel in order to use calculated data.

This file is actually the same as the mission log generated in the mission tab. The only difference is that the mission data file can be configured to not export all data or to merge data in a single table. To configure data to export, edit the mission process and go to tab Mission Data (press the Add button if the window is empty).



Parameters description :

- Header (default value = true) : header of the html file with date of generation, sullary and list of files
- Shape description (default value = false) : list of shapes with type and building points
- Meshing description (default value = false) : meshing parameters of each shape
- Trajectories list (default value = true) : list of trajectories with there parameters
- Trajectories ephemeris (default value = true) : table with information about the position at each computation date (current arc, velocity, orbit angular momentum, ...)
- Kinematics description (default value = true) : list of kinematic bodies
- Kinematics calculation (default value = true) : table with information about the transformation beween kinematic bodies and misalignment
- Mission (default value = true) : table with information on transformation from a planet to the spacecraft and other data.
- Display time results (default value = "in the same table") : indicates if all table listed above should be merged in the same table, it is easier to use in excel that's way but it is harder to read in html (long lines).

Editing process results

From the *Result* area of the **Processing** tab, you can load and unload result data, and edit loaded results.

How to load run results

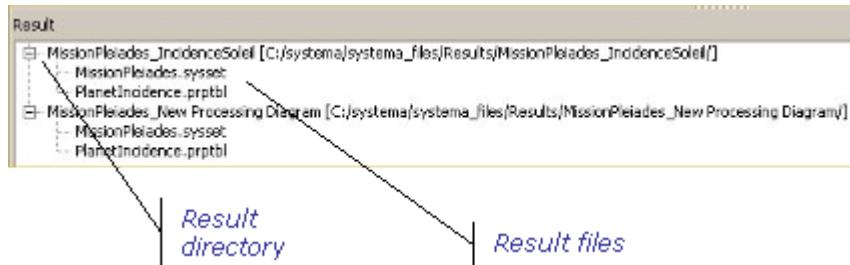
1. In the *Result* area, click  .

A dialog window prompts you to select the result file to be loaded.

2. Select the required ".sysres" file.

The editable result files are displayed in the *Result* area.

The example below shows the files which are displayed after selecting the "MissionPleiades_IncidenceSoleil.sysres" file:



How to edit run results

- In the *Result* area, double click the file to be edited.

or perform the following actions:

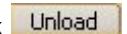
1. In the *Result* area, select the file to be edited.

2. Click  .

Result: The selected file is opened in the default text editor.

How to unload run results

1. In the *Result* area, select the result directory associated with the result to be unloaded.

2. Click  .

Result: The selected result directory and its files disappear from the *Result* area.

Using process results

Using process results in the Modeler tab

SYSTEMA can be used to view the result of processes applied to the meshes of a meshing.

The results can be displayed in the **Modeler** tab in two ways:

- in a viewport of a curve type (refer to the "Working with meshings" topic in the "Geometrical model management" chapter),
- in a 3D view, with a 3D representation of the model outlining the meshes.

How to read result data directly from file

SYSTEMA result data is stored in Hdf5 format (*.h5 files).

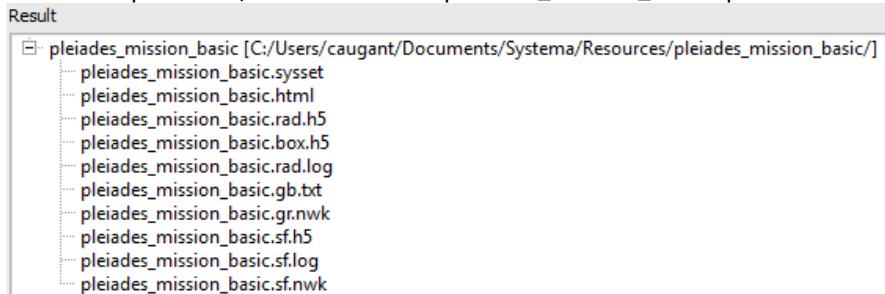
Hdf5 is a standard compressed binary format that is not directly readable.

Still, you can visualize the data using the HdfView software available for free at the following URL: <https://support.hdfgroup.org/products/java/release/download.html#bin>

How to display a result in a 3D view

1.If necessary, in the **Processing** tab, load the result of the process that you wish to display in the 3D view.

In the example below, the result of the "pleiades_mission_basic" process has been loaded:



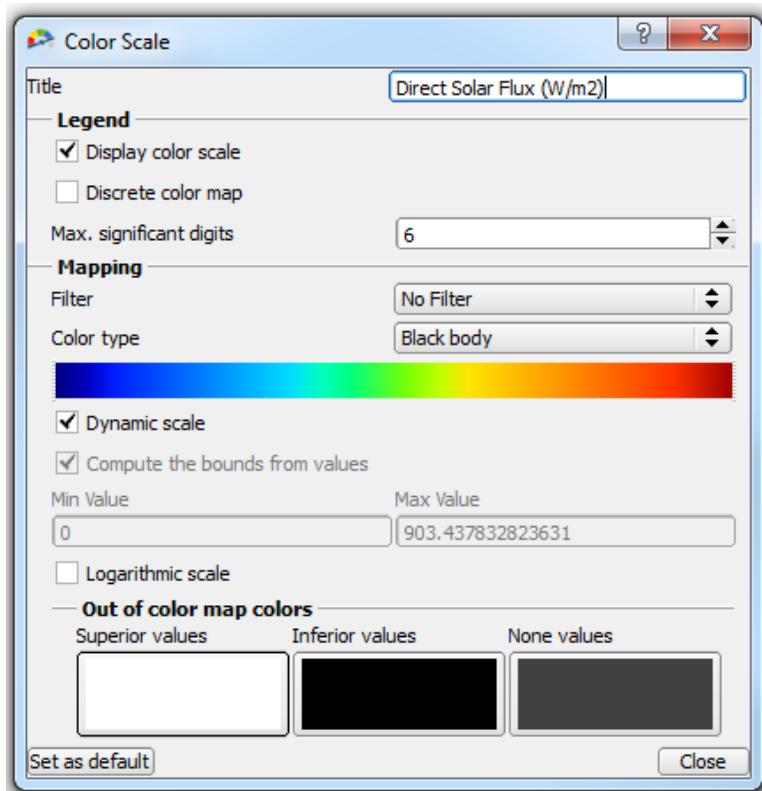
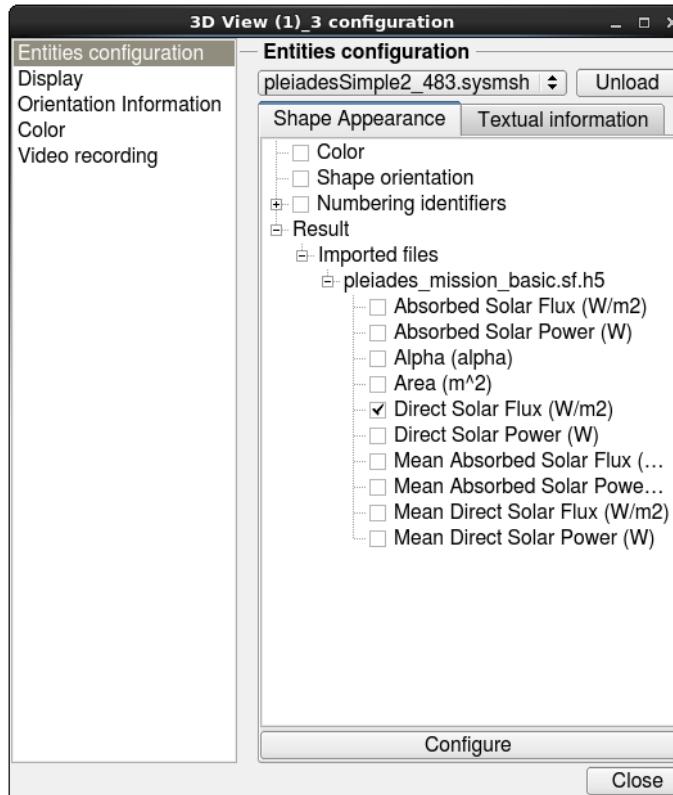
2.In a 3D view of the **Modeler** tab, load the meshing you wish to display a process result for.

3.From the 3D view, display the configuration window (In the right click menu, select Configuration window).

4.In the configuration window, select the result that you wish to display in the 3D view.

In the example below, a result set from the "pleiades_mission_basic" process is selected.

The values of the "Direct Solar Flux (W/m²)" property are associated with colors by means of a color scale (available from the "Configure" button)



Since Systema 4.9.1, several colors palette are available :



Wave length	
Viridis	
Black and white	

i Note : Viridis colors are robust to colorblindness

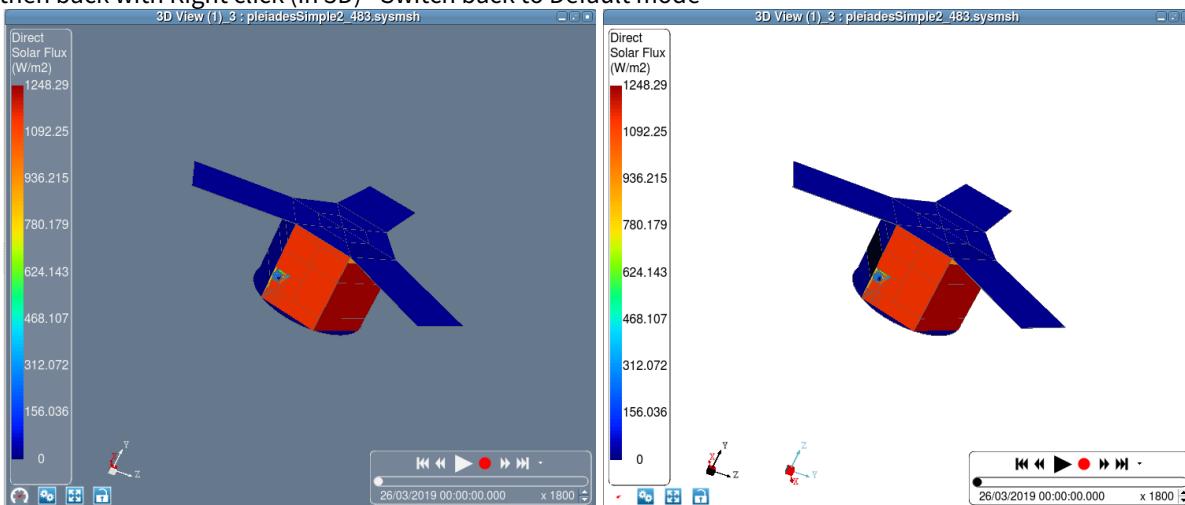
Also it is possible to define colors used for values out of a color scale. By default, the colors used are :

- black for values < to min
- white for values > to max
- grey if no value

These default values can be changed in the setting mechanism in "User Interface" settings, "3D" tab and "Results scale" section.

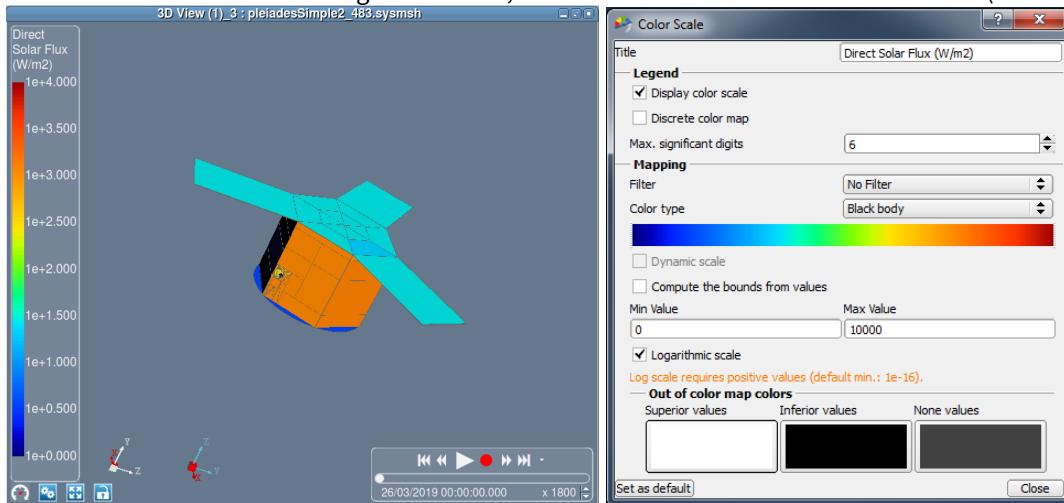
5.Close the configuration window (click **Close**).

Result: In the 3D view, the model meshing is displayed showing the result applied to the meshing color. The text and background colors can be changed quickly using Right click (in 3D)> Switch to Screenshot mode and then back with Right click (in 3D)> Switch back to Default mode



6.The result scale can be switched to logarithmic mode from the color scale window (for continuous scales only). In this mode, negative and zero values are always out of bounds (displayed in black).

When the minimum bound is negative or zero, the default minimum value in use is 1e-16 (it can be tuned manually).

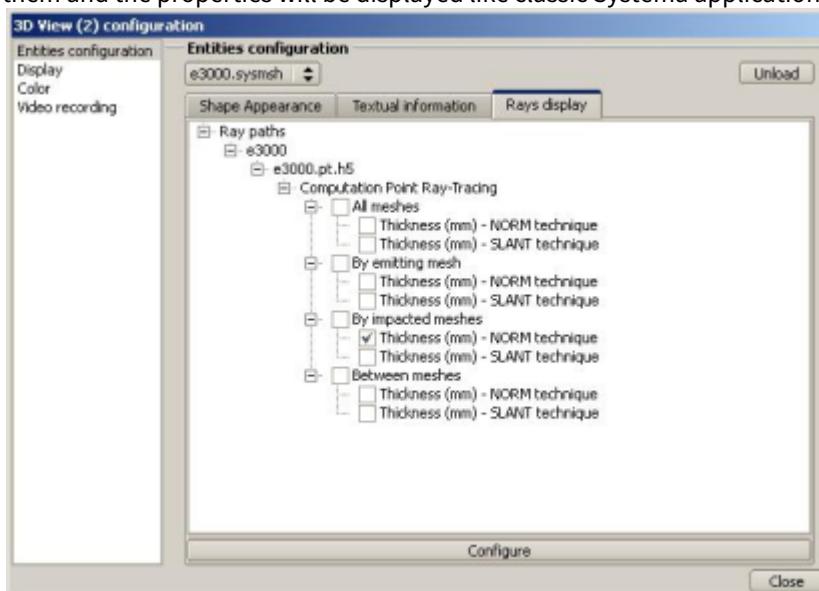


Options set in the color scale window can be saved for the current session of Systema by clicking on **Set as default**. In this case, these new default values override the common ones defined by the settings mechanism (**Max significant digits** and **Dynamic scale** options for instance).

How to display some application properties on computation rays, if available

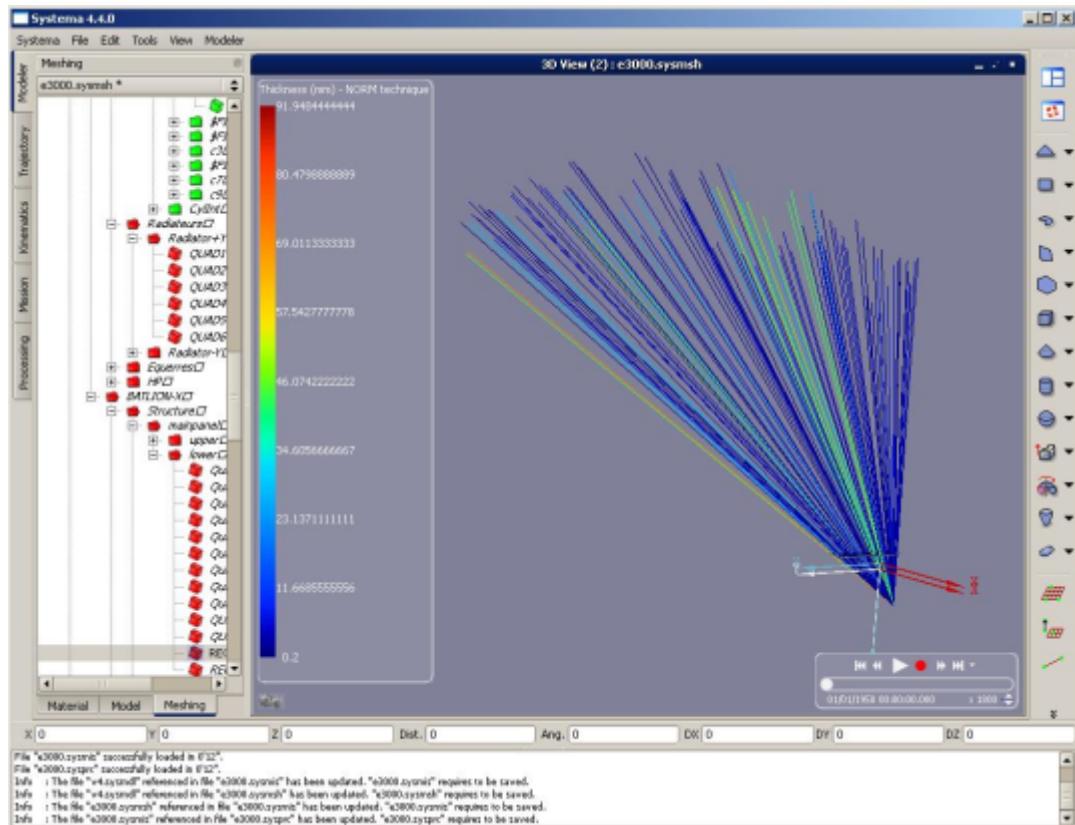
Some applications allow you to visualize the values of the properties that are computed on rays.

1. Open the meshing.
2. In the processing tab, load or compute results related with that meshing that contain rays data and computed rays properties. Refer to your application's documentation for further information.
3. In your meshing 3D view, select **Configuration – Configuration window – Entities configuration – Rays display**. As seen before, you have then four options and under each one, the set of displayable properties. Choose one of them and the properties will be displayed like classic Systema application results.



4. Clicking on the "Configure" button will open a color scale configuration windows similar as classic process results

one.



Using process results in the Mission tab

SYSTEMA can be used to view the result of processes on a model moving on a trajectory in the **Mission** tab.

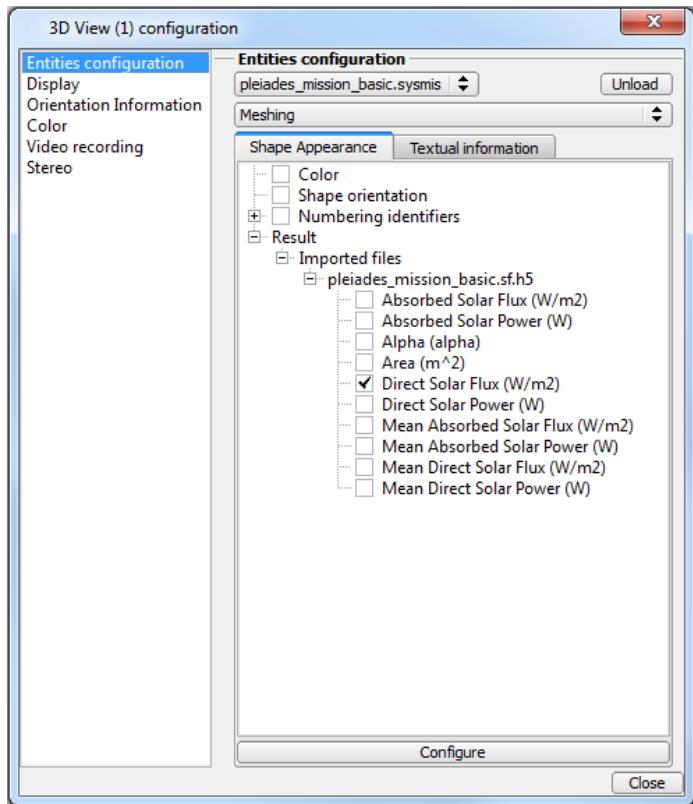
How to display a result on a model in the Mission tab

1. If necessary, in the **Processing** tab, load the result of the process that you wish to display in the 3D view.
2. If necessary, in the **Mission** tab, load the mission you wish to display a process result for.
3. In the **Mission** tab, display the configuration window (In the right click menu, select Configuration window).
4. In the configuration window, select the display parameters:

- the entity on which the result is to be displayed (**Display entity** field),
- the result that you wish to display (**Attribute**, **Result set** and **Property** fields),
- the color scale to be applied

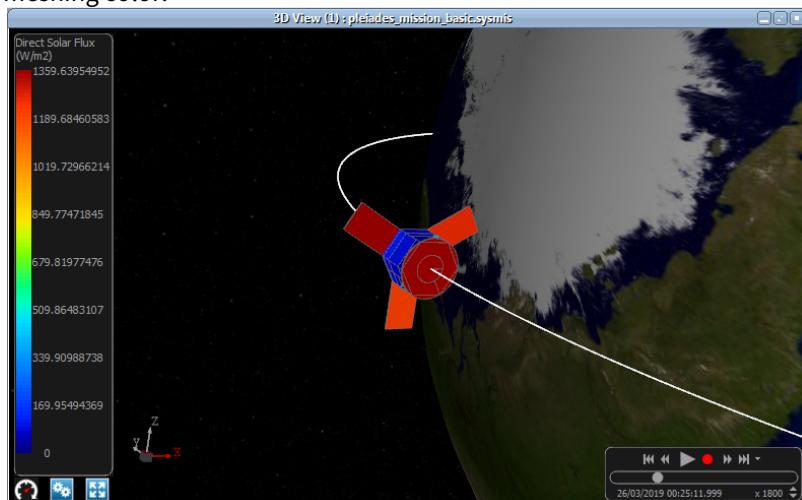
In the example below, the "Meshing" element is selected.

The "Direct Solar Flux (W/m²)" result is selected:



5. Close the configuration window (click **Close**).

Result: In the **Mission** tab, on the trajectory, the model meshing is displayed showing the result on the meshing color.



Post-processing management

Post-processing management

Overview

Introduction

This chapter describes how to use SYSTEMA in order to create, load, save, view and run post-processing functions. With SYSTEMA, you use the **Post-Processing** tab to work on results post-process. Everything described in this chapter takes place within the **Post-Processing** tab.

In this chapter

This chapter covers the following topics:

- Post-processing management
 - Overview
 - Introduction
 - In this chapter
 - The Post-Processing tab
 - What is the Post-Processing tab for?
 - The different parts of the Post-Processing tab
 - Example
 - The menu bar
 - The Post-Processing area
 - The Result Infos and Variables area
 - The Run area
 - The Result area
 - Post-Processes in SYSTEMA
 - What is a post-processing diagram?
 - The process in the Post-Processing area
 - Run parameters attributes
 - File attribute
 - File list attribute
 - Entity attribute
 - Working with post-processing
 - General
 - Creating a new post-processing
 - Opening an existing post-processing
 - Modifying a post-processing
 - Building a post-processing
 - How to build a post-processing
 - Adding standard post-processing boxes
 - How to add a standard post-processing box
 - To add a standard post-processing box from the Standard Boxes browser:
 - To add a standard post-processing box from the Post-Processing menu:
 - Result
 - Organizing standard post-processing boxes
 - Joining files
 - How to join files

- How to split a file
- How to deactivate/activate a file
- Definition of groups
- Examples
- Running a process and editing results
 - Principles
 - Define the result parameters
 - Define variables
 - Overload variables
 - Run the current process
 - How to run the current process
 - Result

The Post-Processing tab

What is the Post-Processing tab for?

You can use the **Post-Processing** tab of the SYSTEMA application to:

- open existing post-processing diagrams and modify them when necessary,
- create new post-processes by combining toolboxes,
- run post-processes on results.



*Note: You define post-processes in the **Post-Processing** tab. Once a post-processing has been defined, you can then apply it to results which have been loaded in the **Processing** tab or in the **Post-Processing** tab.*

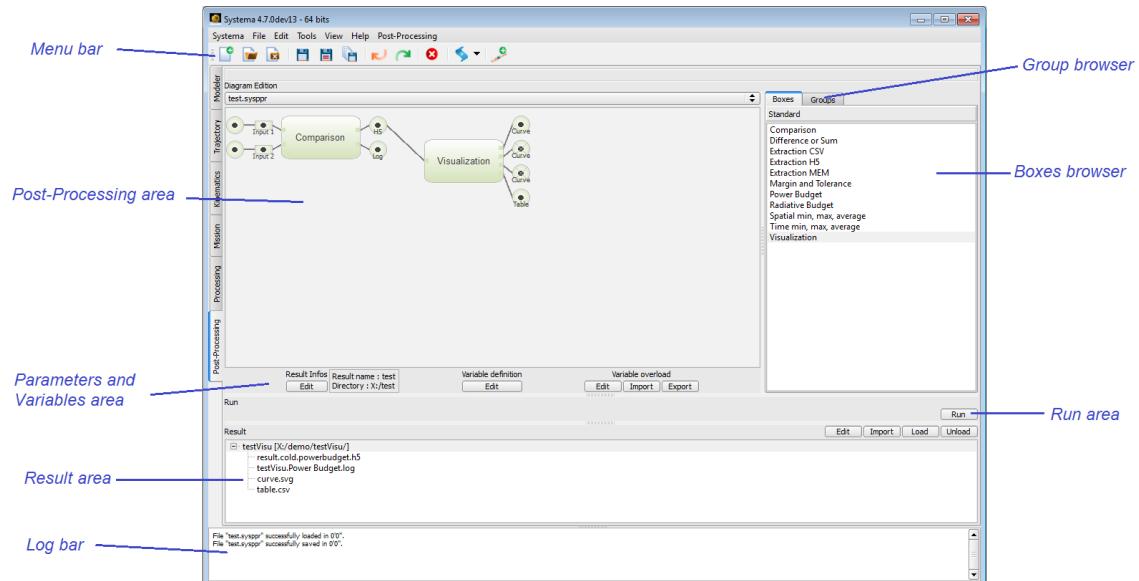
The different parts of the Post-Processing tab

The **Post-Processing** tab is divided into 4 main parts:

- the *Post-Process* area, containing the post-process description,
- the *Run* area, containing the elements necessary to run a process,
- the *Result* area, from which process results can be loaded and edited,

Example

The following diagram shows an example of the **Post-Processing** tab. The *Post-Process* area contains a post-process, the *Result* area contains two loaded post-process results:



The menu bar

The menu bar contains:

- The menus which are common to all SYSTEMA tabs: the **File**, **Edit**, **Tools**, **View** and **Help** menus.

For more information, refer to the "General User Interface" chapter

- One specific menu, the **Post-Processing** menu, which can be used to build the current post-processing.

The following table presents the options available from the **Post-Process** menu:

Option	Use
Import a post-processing box	Adds a box to the current post-processing with default attribute values <i>Available post-processing here are the same as those in the boxes browser displayed in the Post-Processing area</i>
Insert HDF5 file	Adds a HDF5 result file (with extension .h5) to the current post-processing
Insert HDF5 file list	Adds a HDF5 result file list (with extension .h5) to the current post-processing

The Post-Processing area

The Post-Processing area is divided into two parts:

- the left-hand part contains the *post-processing viewer* which is used to display, build and modify the diagram for the current post-processing;
- the right-hand part contains *two browsers* :
 - The *boxes browser* which displays the list of standard boxes that can be added to the current post-processing
 - The *groups browser* which defines the groups from a file

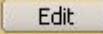
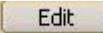
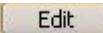
A zoom is available to increase the size of the diagram area.

The Result Infos and Variables area

The *Result Infos and Variables* area is used to:

- define the result file's name and directory,
- define variables usable in post-processing,
- overload values of all variables defined in other entities (modeler, meshing, trajectory, kinematics, mission) and in post-processing entity.

The following table presents the elements of the *Parameters and Variables* area:

Element	Use
Result infos area	Displays the result file name and directory <i>In this area, the names cannot be changed</i>
 Edit button (Result Infos)	Is used to define the result name file and directory
 Edit button (Variable definition)	Is used to define variables
 Edit button (Variable overload)	Is used to overload values of variables defined in all entities
 Import button (Variable overload)	Is used to import a file to overload several variables
 Export button (Variable overload)	Is used to export a file with all variable overloaded

The Run area

The *Run* area is used to:

- run the current post-processing on the selected results.

The following table presents the elements of the *Run* area:

Element	Use
 Run button	Applies the current post-processing to the selected results and runs

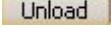
The Result area

The *Result* area is divided into two parts:

- the upper part contains buttons which are used to edit, import, load and unload result files;
- the lower part contains the *result browser* which displays the list of loaded results.

The following table presents the buttons of the *Result* area:

Button	Use
 Edit button	Is used to edit result files

 button	Is used to import any type of file <i>The corresponding files are then displayed in the result browser</i>
 button	Is used to load result by selecting a ".sysres" file <i>The corresponding files are then displayed in the result browser</i>
 button	Is used to unload a result which has been previously loaded in the <i>result browser</i>

Post-Processes in SYSTEMA

What is a post-processing diagram?

A SYSTEMA post-processing diagram is a sequence of boxes managing the input result files.

The process in the Post-Processing area

Each basic post-processing generally consists of:

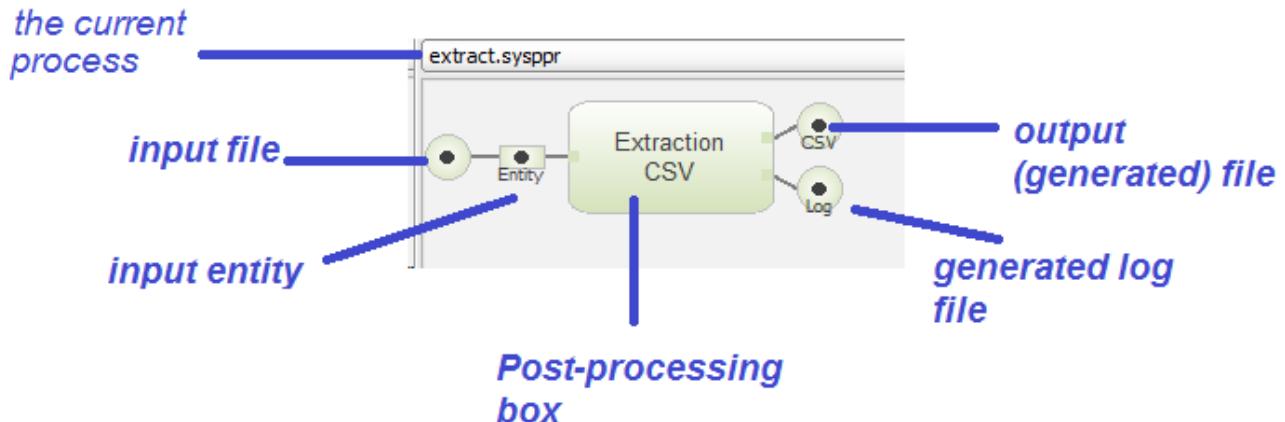
- input file(s), or file list
- a post-processing description,
- output file(s).

The color of the entry/output icon indicates whether there are any warnings (orange) or errors (red) associated with **the corresponding file**:

Icon	Meaning
	For file : The file does not exists or is not in the expected format For file list : The path is not defined
	The input entity is not defined
	The output file is not in the expected format
	The input file or the input file list is in the right format
	The input entity is well defined
	The output file is in the right format

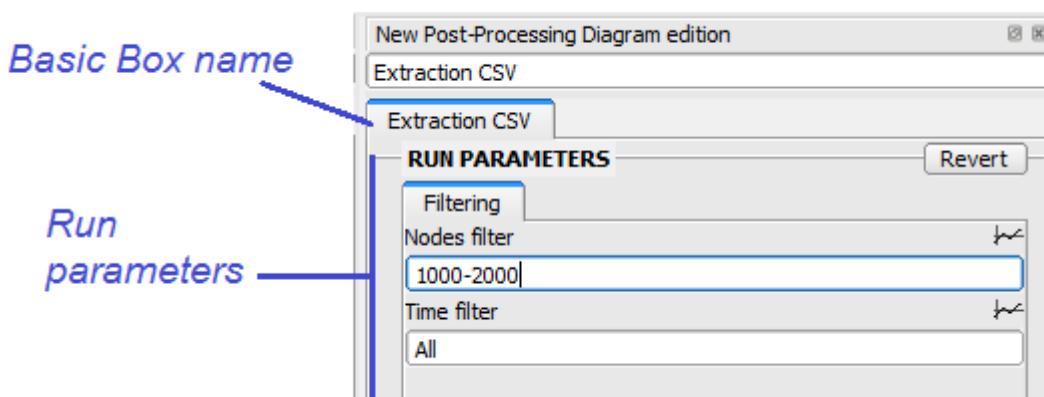
⚠ A link to a file shall not be used as an input file.

The following example shows a process in the *Process area*:

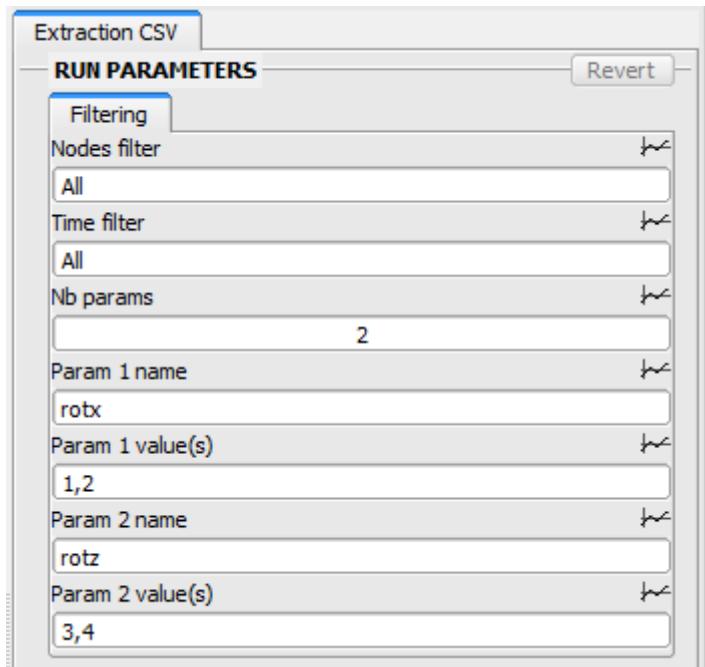


Run parameters attributes

Each Post-Processing box has **Run parameters** attributes depending on the purpose of the box. You modify them via the edition window, by double clicking the Post-Processing box:



Most of the post processing boxes propose Node or groups and Time filtering options, for the CSV Extraction box, a parametric extraction is also proposed :



Nodes filter option shall be :

- All
- A single node number (101 for example)
- A range of nodes number (101-110 for example)

Groups filter option shall be :

- All
 - A single group id (1 for example)
 - A range of group id (1-4 for example)
- See [Definition of groups](#) for more details

Time filter option shall be :

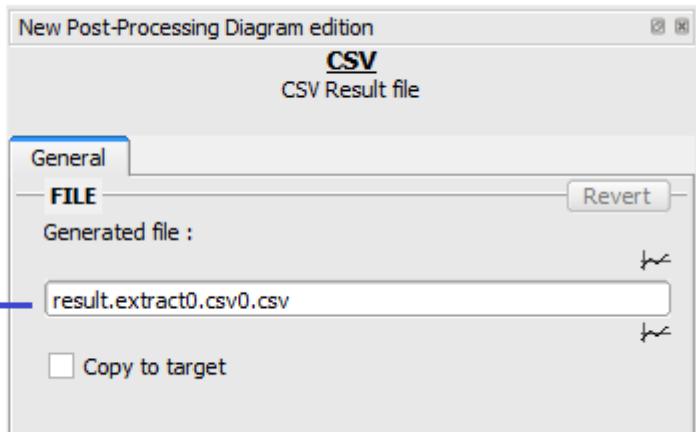
- All
- A single time value (523.123 for example)
- A range of time (0-1000 for example)

The unit of time is second.

! Syntax errors in parameters attributes are raised at runtime.

File attribute

The Input/Ouput files have one attribute: the **File** attribute. You can modify it via the edition window:

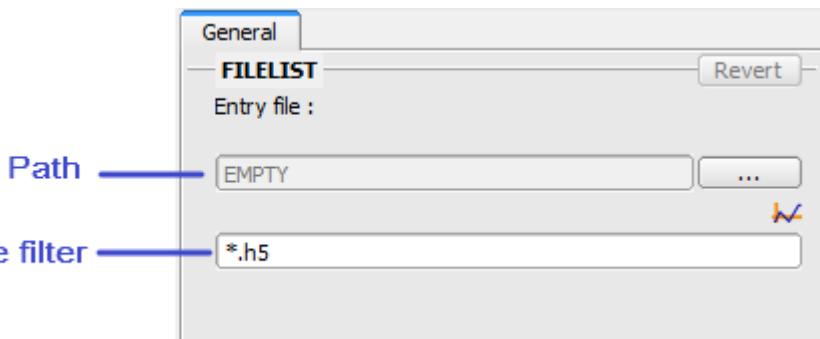


By default, the output files are generated, in the *Result* directory (defined in the *Run* area). This directory can be changed by ticking the **Copy to target** checkbox. The format of output file is defined by the box.

The input file has only the type HDF5.

File list attribute

The input file list has one attribute : the **FileList** attribute composed of a path and a filename filter. You can modify it via the edition window :



By default, the filename filter (*.h5) includes all HDF file contains in selected directory.

This filter can be defined by one or more regular expression(seperated by the character '|').

Example :

```
| Process*.h5 | *Temp.h5 | MyFile.h5
```

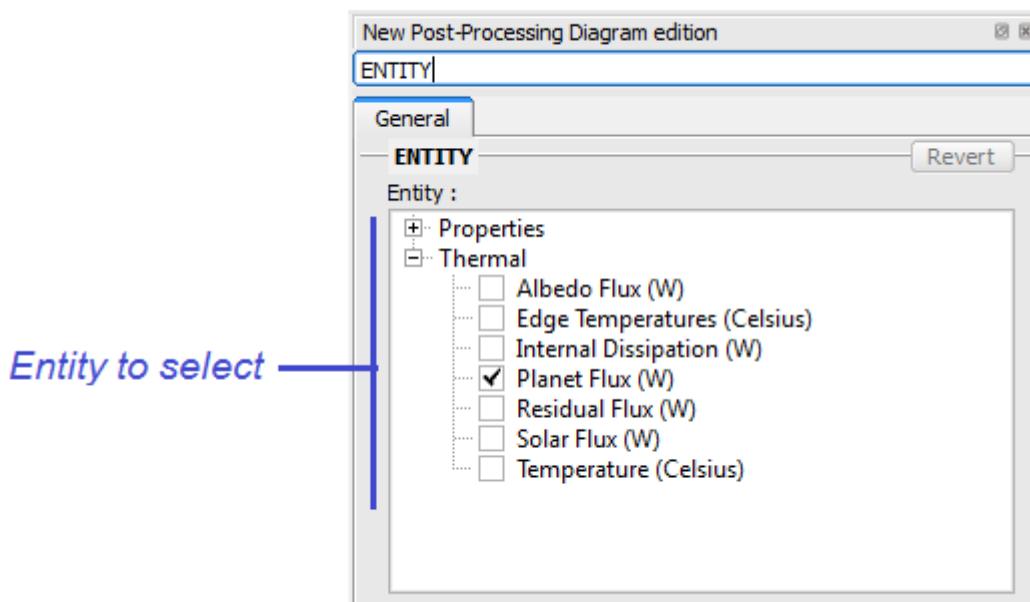
By default, the value of the path is "EMPTY". To define it, click on "..." and select a directory in the file selection dialog.

All files with name that matchs with the file name filter and that are located directly in the selected path (not in a recursive way) are used as input files.

- ✓ With a file list, you can use a result folder of a processing as an input of post-processing boxes. Thus when running the diagram, all result files (according to the file name filter) will be post-processed. To know how to define an entity without having an hdf5 file defined, please refer to [Entity attribute](#).

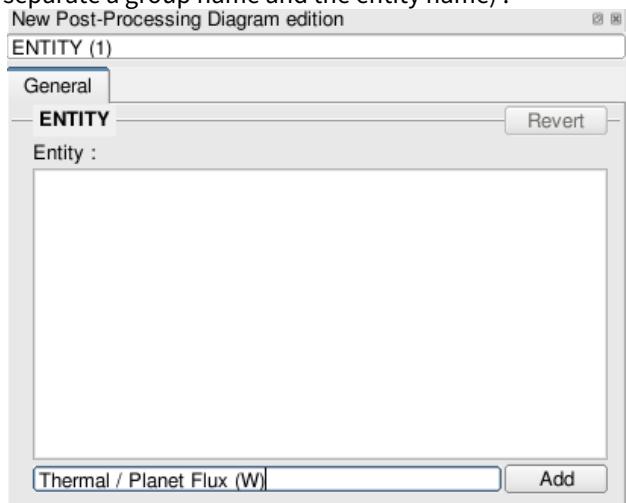
Entity attribute

The input entity have one attribute : the **Entity** attribute. The entity correspond to an entity of the input HDF5 file. You can modify it via the edition window :

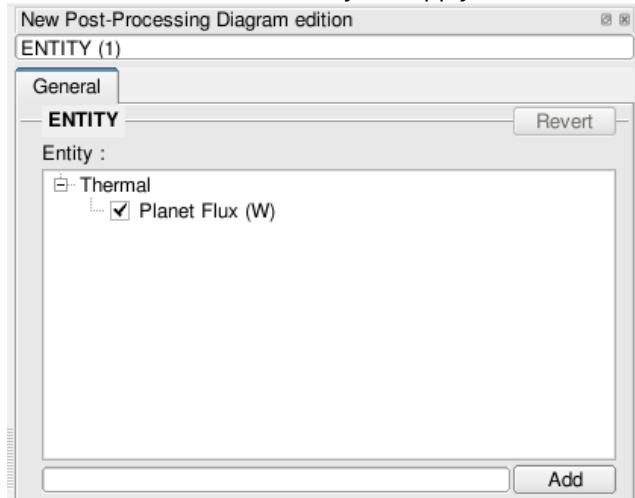


The entity tree is loaded when the entity is linked to an input file. If the input file does not exist or does not have the right format, the entity tree will be empty. In this case, it possible to add manually entities:

- In the area under the entity list, enter the entity name (with the exact case and if relevant with " / " to separate a group name and the entity name) :



- Click on "Add". Select the entity and apply the modifications :



Manual entities can be deleted in a row by selecting the main "Delete" command.

When defining the input HDF5 file, if a manual entity was defined and selected and if the entity name corresponds to an entity of the input file (exact match), the entity is selected. Other manual entities are no longer displayed.

If there is no entity selected, this input is in error. The displayed error message is "**The entity name is not set**". By default, nothing is selected in the entity tree.

Working with post-processing

General

In the **Post-Processing** tab, you can open an existing post-processing or create a new one.

When a post-processing is opened, you can:

- modify it,
- run it,
- consult the post-processing results.



*Note: The color of the entry/output icon indicates if there are any warnings (orange) or errors (red) associated with **the corresponding file or entity**.*

Creating a new post-processing

To create a new post-processing:

- In the **File** menu of the **Post-Processing** tab, select the **New** option.

Result: The *Post-Processing* area displays an empty editable space.

You can then build the diagram and save it.

Opening an existing post-processing

You can open a post-processing which has previously been saved within a ".sysppr" file.

To edit an existing post-processing:

- In the **File** menu of the **Post-Processing** tab, select the **Open** option and select the file containing the required post-process.

Result: The post-processing diagram is displayed in the *Post-processing* area.
You can then modify the post-processing and save your modifications.



Note: Since Systema 4.8.2, a warning is displayed if a processing built with an older version is opened:

Warning : The down compatibility with Processing files is not guaranteed for all installed applications

You should rebuild your Processing from zero to be safe.

Modifying a post-processing

You can modify a post-processing by :

- adding new basic boxes (refer to the "Building a post-processing" section),
- editing and modifying input/output files and entities, and post-process description.

Building a post-processing

How to build a post-processing

You build a post-processing by adding and organizing basic boxes as diagram.

Adding standard post-processing boxes

There are two different ways to add standard boxes:

- using the *Boxes* browser,
- or using the **Post-Processing** menu.

How to add a standard post-processing box

To add a standard post-processing box from the Standard Boxes browser:

- In the *Boxes* browser, double click the standard box you wish to add to the process diagram, or drag and drop it on the diagram part

To add a standard post-processing box from the Post-Processing menu:

- In the **Post-Processing** menu, select the **Import a post-processing box** option, then select Standard, and choose the box you want to add.

Result

The selected standard box with its associated input/output files are added with default values. Their attributes can then be modified by editing them.

Organizing standard post-processing boxes

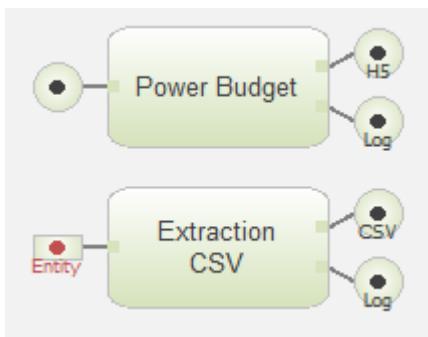
To organize standard box , you can join or split input/output files.

Joining files

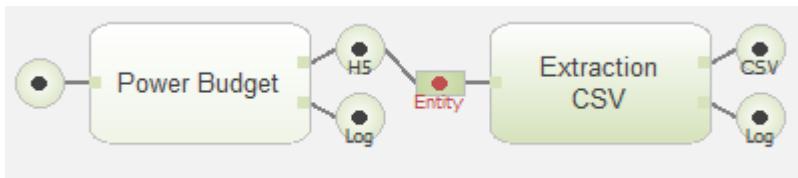
You can join files in the following case:

When the output file of a basic process has to be used as the input file of another process (as shown in the example below) :

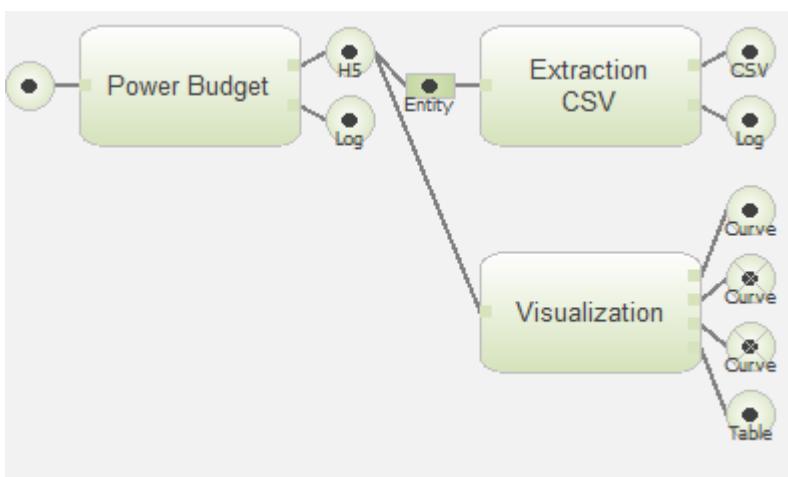
Before joining the input files:



After joining the input files:



When two standard boxes use the same file as an input file, these inputs are automatically joined (as shown in the example below):



How to join files

Cf [Process - How to join files](#)

How to split a file

Cf [Process - How to split files](#)

How to deactivate/activate a file

Cf [Process - How to deactivate/activate a file](#)

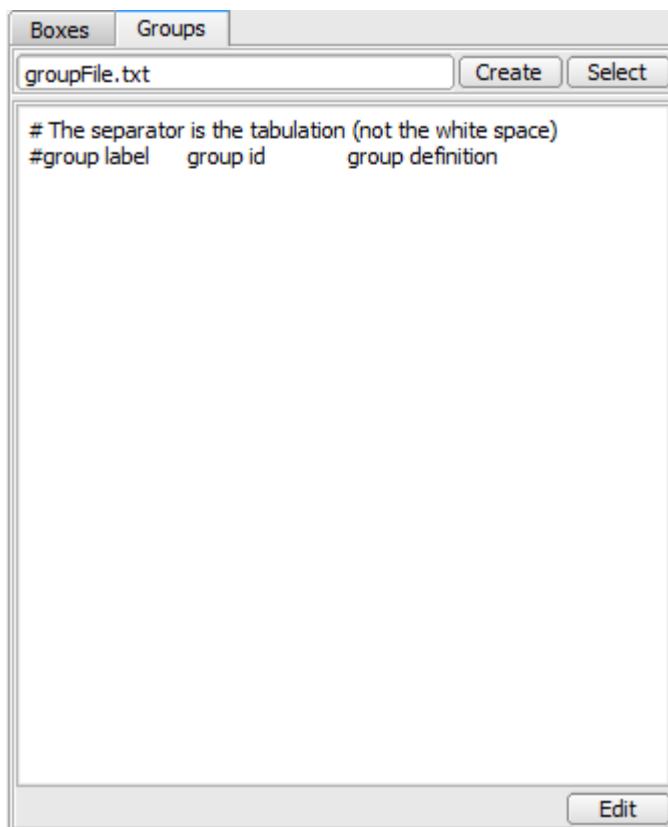
Definition of groups

In the group browser, you can create or import node groups.



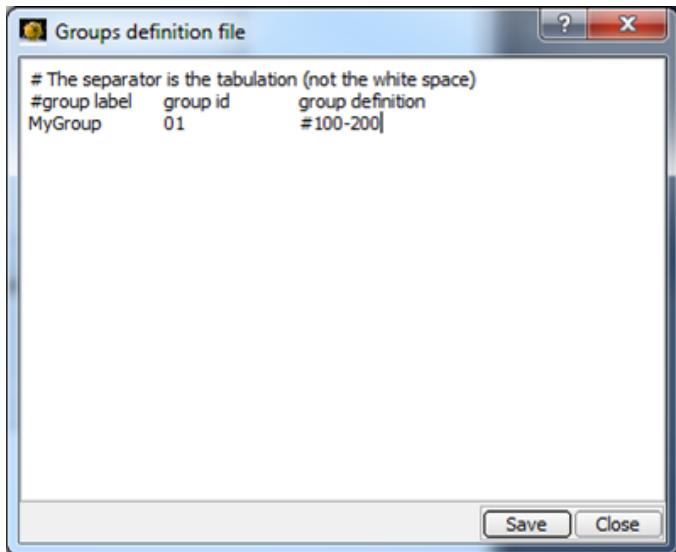
Groups are defined in a file. When you select the "Create" button, a file chooser opens and you can define the name of the group file you want to create.

An empty group file is created with the following header describing how to define the groups :



You can edit the file by clicking on the "Edit" button.

In the following example, the group named "MyGroup" with the id "01" has been created and defines the nodes from 100 to 200. During the run, Systema will resolve the group and work only on the nodes contained in this group.



The group definition is based on the ZNODES definition of Thermisol. The definition consists in one or more items separated with a comma ','. Each definition contains nodes or group ids.

The node definition shall be constructed as follow:

ALL: all the nodes from the input file.

#[node] : a single node

#[begin-end] : a range of nodes

#[submodel]: [node] : a single node from a sub model

#[submodel]: [begin-end] : a range of nodes from a sub model

[label] : a label or part of a label

[submodel]:[label] : a label or part of a label contained in a sub model

[submodel] : all nodes directly under the sub model (exact match)

A group may also contain groups or group content:

[group_id] : id of a group

[begin-end] : all groups in range of ids

[submodel]:[group_id] : id of a group from a sub model

[submodel]:[begin-end] : range of groups from a sub model

***[group_id]** : content of a single group

***[begin-end]** : content of a all groups in range

***[submodel]:[group_id]** : content of a single group from a sub model

***[submodel]:[begin-end]** : content of a all groups in range from a sub model

Examples

- one node : #100
- list of node : #100,#200
- range of nodes : #100-200
- multi-definition : #100,#200,#300-400

- all nodes : all
- list of groups : 10, SUB:2
- range of groups : 10-20, SUB:2-4
- content of group : *10, *SUB:2
- labels from the H5 file : PAN_01, WALL, SUB:my label -X

Labels can contain '+', '_', '-' and spaces.

If you want to use an existing group file, click on "Select".

Once the group file is defined, you can use the group id in the post-processing boxes with group filters.

Running a process and editing results

Principles

Cf [Process - Principles](#)

Define the result parameters

Cf [Process - Define the result parameters](#)

Define variables

Cf [Working with variables](#)

Overload variables

Cf [Process - Overload variables](#)

Run the current process

How to run the current process

1. In the *Run* area, click  .

The Post-Processing run window is displayed. It contains information about the run. By default, all steps are launched. To skip a process, unselect it in the "Process run" window.

 If the post-processing diagram contains a list of files as a box entry, this box and all these descendants will be executed as many times as the file list contains entries.

2. Close the Post-Processing run window.

Result

The result files and directory are created in the SYSTEMA environment.

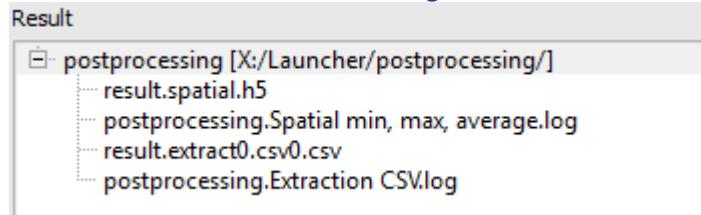
The following table shows the files which are created:

File	Content
*.log	Information generated in the Post-Processing run window during the run execution
*.csv, *.H5, *.txt ...	Output file generated by a post-processing and containing post-processing data

Output file(s) name can be set by variables :

Variable	Result
<code>\$(postpro_name)</code>	post-processing file name
<code>\$(boxname)</code>	name of the post-processing box
<code>\$(inputs_name)</code>	concatenation of input file names (separated by the character '_')

The **Result** area of the **Post-Processing** tab contains the editable result files:



Batch command

Batch command

Overview

Introduction

This chapter describes the batch commands usable in a command tool or in a command file.

In this chapter

This chapter covers the following topics:

- [Batch command](#)
 - [Overview](#)
 - [Introduction](#)
 - [In this chapter](#)
 - [Interactive mode with graphical user interface environment](#)
 - [Command line](#)
 - [Batch mode for Python script execution](#)
 - [Command line](#)
 - [Process diagram execution mode](#)
 - [Command line](#)
 - [Syntax for variables overloading](#)
 - [Batch mode for file conversion](#)
 - [Command line](#)
 - [Batch mode for archiving files](#)
 - [Command line](#)

Interactive mode with graphical user interface environment

Command line

SYSTEMA application can be launch with the pre-loading of an input file.

- **Systema -load <filePath> [-overload<overloadFilePath>]**

Parameter	Description
filePath	Name of the pre-loaded file

overloadFilePath	Name of the file containing values to overload variables, can only be used when loading a process diagram file. [<i>Optional</i>]
-------------------------	---

Batch mode for Python script execution

Command line

A Python script can be executed in command line.

- **Systema -script <filePath>**

Parameter	Description
filePath	Name of the python script file

Process diagram execution mode

Command line

A process diagram execution can be launched with no user interaction.

- **Systema -exec <filePath> [-process <process name>] [-overload<overloadFilePath>] [-resultPath <dirPath>] [-showProgress <choice>]**

Parameter	Description
filePath	Name of the process diagram
process name	Execute only the given box of the process. [<i>Optional</i>]
overloadFilePath	Name of the file containing values to overload variables [<i>Optional</i>]
dirPath	Name of the directory to store generated results [<i>Optional</i>]
choice	Choice to show or hide the execution progress (true / false) [<i>Optional</i>]

Syntax for variables overloading

The file is an ASCII file containing as many lines as necessary. The module is indentified by a name preceding by '\$' character and followed by the name of each variable to be overloaded and the overloading value. The variables names must not contain blanks characters. The module name is case insensitive.

- **.\$MODULE1**

- VariableName1 = Value
- VariableName2 = Value
- ...
- **.\$MODULE2**
 - VariableName1 = Value
 - VariableName2 = Value
 - ...
- ...

Module names	Description
Model	Name for the modeler module
Meshing	Name for the meshing module
Trajectory	Name for the trajectory module
Kinematics	Name for the kinematics module
Mission	Name for the mission module
Processing	Name for the processing module
Variable type	Syntax
Integer	42
Long	1458796542
Flag	YES or Y, TRUE or T, ON, NO or N, FALSE or F, OFF
Real	42.005
String (512 characters max)	"my_text" or 'my_text' or even my_text
Coordinates	(12.5,45.62,58.0) or (12.5;45.65;58.0) or even (12.5 45.65 58.0)

Batch mode for file conversion

Command line

A file conversion can be launched with no user interaction.

- **Systema --convert <sourceFilePath> -to <destinationFilePath> -type <module name> [-log <logFilePath>]**

Parameter	Description
sourcefilePath	Name of the source file to be converted
destinationFilePath	Name of the result file generated by the conversion The file extension must be given
Module name	Describes the type of the source file through the module name,
logFilePath	Name of the log file to be generated
Module names	Description
Model	Name for the modeler module
Meshing	Name for the meshing module
Trajectory	Name for the trajectory module
Kinematics	Name for the kinematics module
Mission	Name for the mission module
Processing	Name for the processing module

The module name is case insensitive.

Batch mode for archiving files

Command line

Archiving files can be launched with no user interaction.

- **Systema -archive <filePath> -dir <archiveDirPath>**

Parameter	Description
filePath	Name of the file to archive with its related files. It must be the absolute path
archiveDirPath	Name of the directory where files will be saved. It must be the absolute path. The directory must not exist.

Modeler technical annex

Modeler technical annex

Overview

Introduction

This annex describes all technical information about the shapes and the transformations applicable to the shapes in the axis system.

In this chapter

This chapter covers the following topics:

- Modeler technical annex
 - Overview
 - Introduction
 - In this chapter
 - The shapes
 - Antenna
 - Antenna with focal
 - Antenna with focus
 - Box
 - 4 faces box
 - 5 faces box
 - 6 faces box
 - Cone
 - 3 points cone defined by base center - height - radius
 - 4 points cone defined by 3 points on base - height
 - 4 points truncated cone defined by 3 points on base - 1 point on other base
 - General truncated cone
 - Cylinder
 - 3 points cylinder defined by base center - height - radius
 - 4 points cylinder defined by 3 points on base - height
 - 4 points truncated cylinder defined by 3 points on base - height
 - General truncated cylinder
 - Disc
 - 3 points disc
 - Default disc
 - 4 points truncated disc defined by 3 points on external border - 1 point for internal border
 - General truncated disc
 - Extrusion
 - Extruded polygon with holes
 - Extruded polygon
 - Parabola
 - Paraboloid 3 points
 - Truncated paraboloid
 - Polygon
 - Polygon with holes
 - Default polygon
 - Revolution

- Revolved polygon with holes
- Revolved polygon
- Simple shapes
 - Quadrangle
 - Rectangle
 - Triangle
- Sphere
 - 3 points sphere
 - 4 points sphere
 - 4 points truncated sphere defined by 3 points on upper border - 1 on downer
 - General truncated sphere
- Volumic box
 - Default volumic box
- Volumic revolved quadrangle
 - Default volumic revolved quadrangle
- Volumic extruded triangle
 - Default volumic extruded triangle
- Volumic extruded quadrangle
 - Default volumic extruded quadrangle
- Volumic tetrahedron
 - Default volumic tetrahedron
- Volumic cylinder
 - Default volumic cylinder
- Volumic cone
 - Default volumic cone
- Volumic sphere
 - Default volumic sphere
- The transformations
 - Translation
 - Rotation
 - Translations and rotations
 - Cosines matrix
 - One point translation
 - Three points transformation
 - XY or Z translationRotation around axis defined by two points
 - Rotation around axis defined by one point and one direction
 - XY or Z rotation
 - Position and quaternion
 - Direction transformation
 - Three rotations sequence

The shapes

In SYSTEMA, the shapes are organised by families (such as box, sphere ...) and by type in order to distinguish different formats inside a family (4 points sphere, truncated sphere ...).

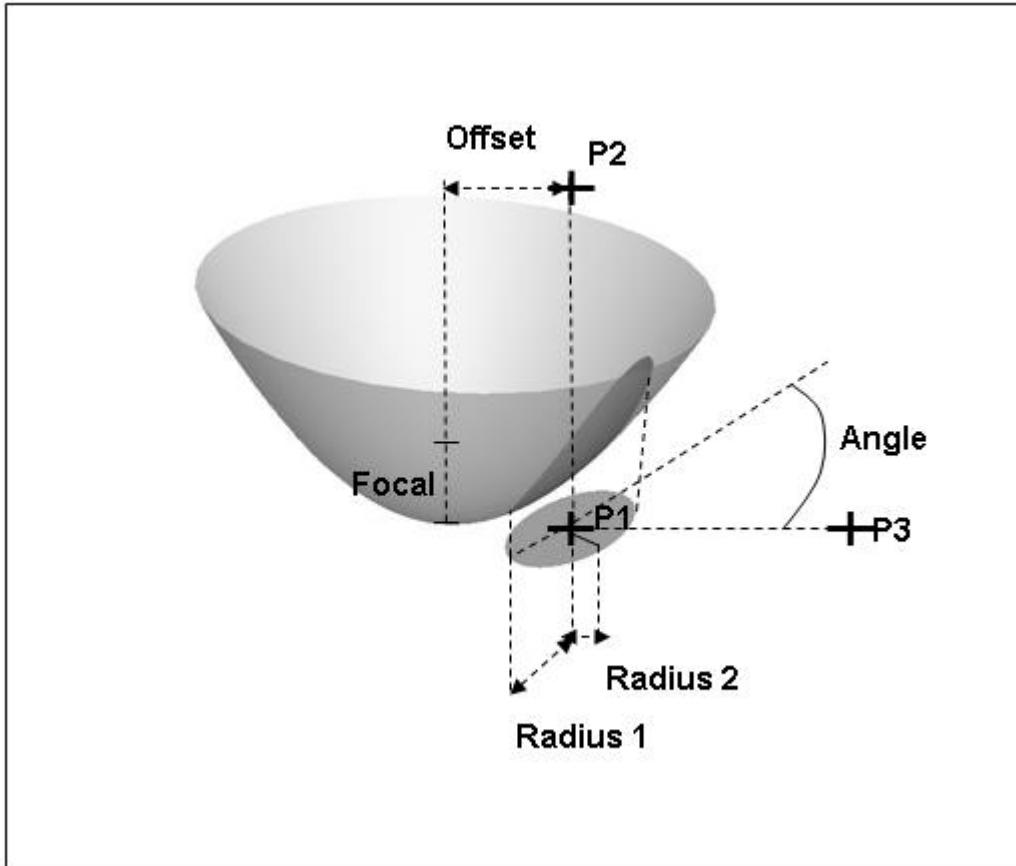
Shapes are built from points, vectors, lengths or angles. These elements can be defined in local axis system or in main axis system (root local axis system)

Antenna

This shape is the projection of an ellipse on a paraboloid along the paraboloid axis.

Antenna with focal

- Point 1 : center of the ellipse (P1)
- Point 2 : point defining with P1 the ellipse' normal (P2)
- Point 3 : point defining with P1 the ellipse' axis (P3)
- Focal : focal distance of the paraboloid
- Offset : offset between the paraboloid's axis and the ellipse' axis
- Radius 1 : semi major axis or semi minor axis of the ellipse
- Radius 2 : semi minor axis or semi major axis of the ellipse
- Angle : angle of the first ellipse' axis relative to P1P3



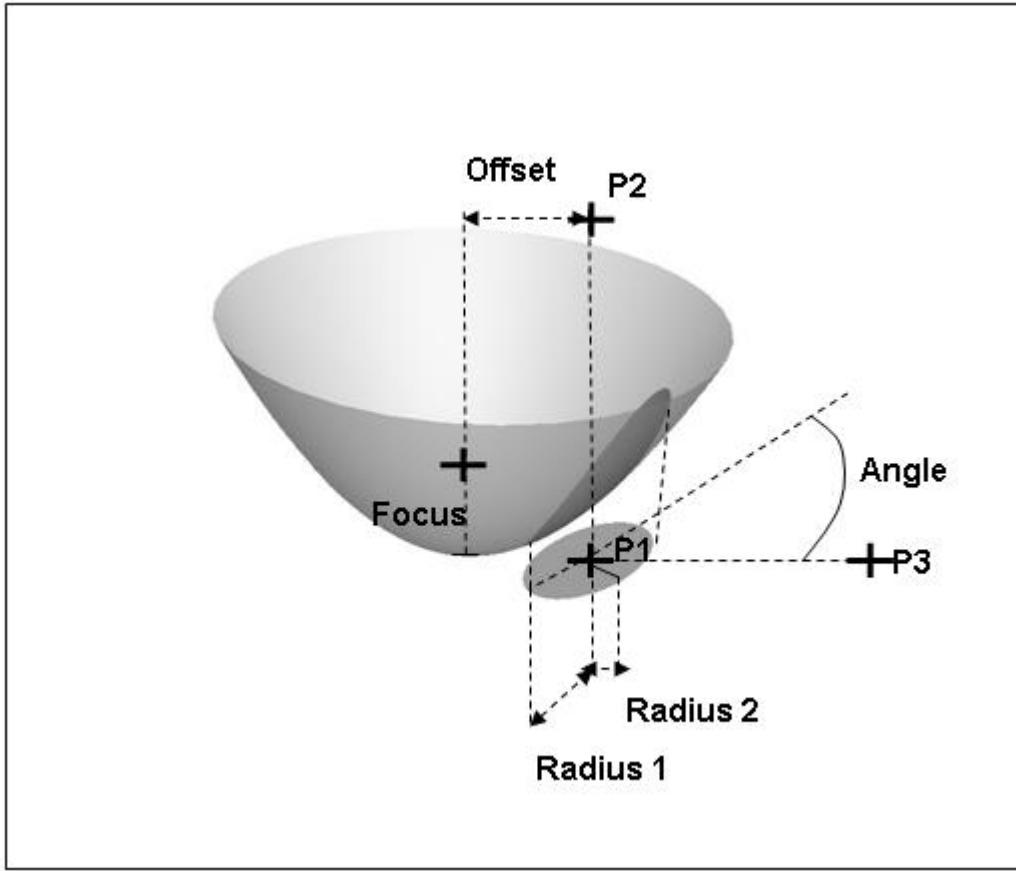
DEFINITION

This shape is the projection of an ellipse on a paraboloid along the paraboloid axis.

- Point 1 : center of the ellipse (P1)
- Point 2 : point defining with P1 the ellipse's normal (P2)
- Point 3 : point defining with P1 a reference axis to define the ellipse's axis (P3)
- Focal : focal distance of the paraboloid
- Offset : offset between the paraboloid's axis and the ellipse's axis
- Radius 1 : semi major axis or semi minor axis of the ellipse
- Radius 2 : semi minor axis or semi major axis of the ellipse
- Angle : angle of the first ellipse's axis relatively to P1P3

Antenna with focus

- Point 1 : center of the ellipse (P1)
- Point 2 : point defining with P1 the ellipse' normal (P2)
- Focus : focus point of the paraboloid
- Point 3 : point defining with P1 the ellipse' axis if P1, P2 and Focus are aligned (P3)
- Radius 1 : semi major axis or semi minor axis of the ellipse
- Radius 2 : semi minor axis or semi major axis of the ellipse
- Angle : angle of the first ellipse' axis relative to P1P3



DEFINITION

This shape is the projection of an ellipse on a paraboloid along the paraboloid axis.

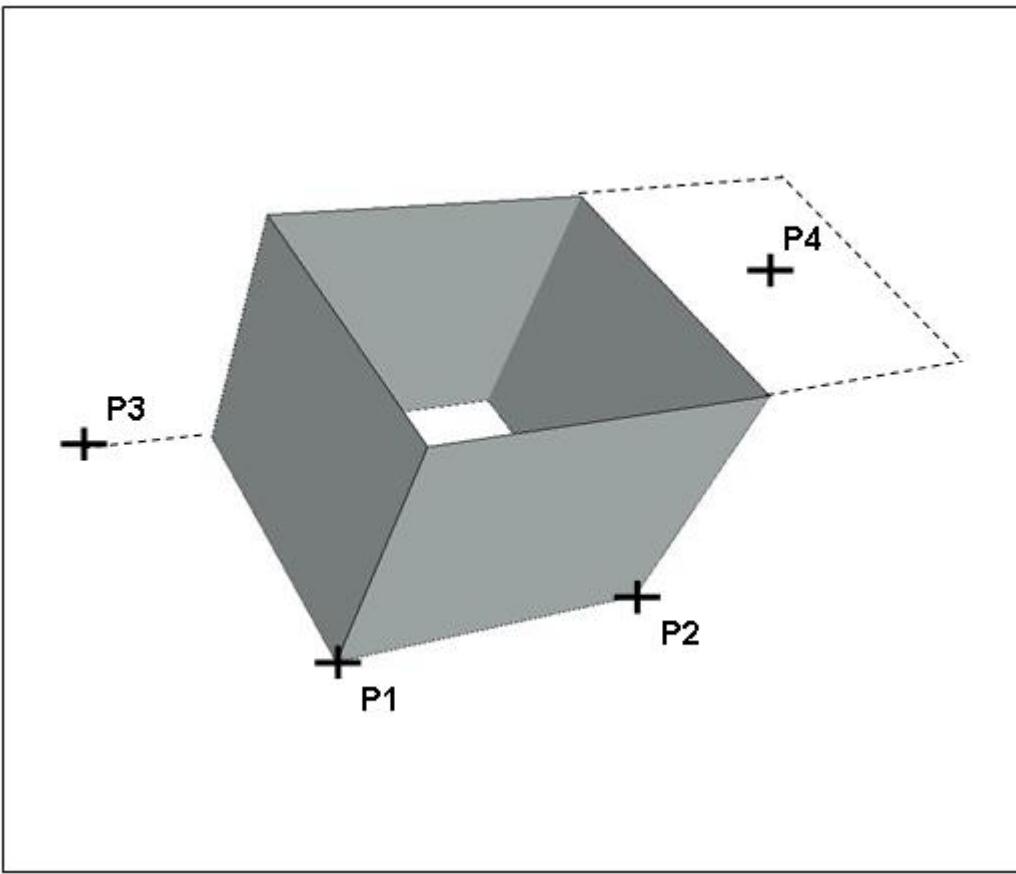
- Point 1 : center of the ellipse (P1)
- Point 2 : point defining with P1 the ellipse's normal (P2)
- Focus : focus point of the paraboloid
- Point 3 : if P1, P2 and Focus are aligned, point defining with P1 a reference axis to define the ellipse's axis, otherwise the reference axis is defined as orthogonal to P1P2 and belonging to P1P2Focus plane and P3 is ignored (P3)
- Radius 1 : semi major axis or semi minor axis of the ellipse
- Radius 2 : semi minor axis or semi major axis of the ellipse
- Angle : angle of the first ellipse's axis relatively to the reference axis

Box

4 faces box

This shape is a box with four faces, no faces in the P1P4 direction.

- Point 1 : origin point for the box (P1)
- Point 2 : point defining with P1 the first of edge the box (P2)
- Point 3 : point defining with P1 the second edge of the box (P3)
- Point 4 : point defining with P1 the third edge of the box (P4)



DEFINITION

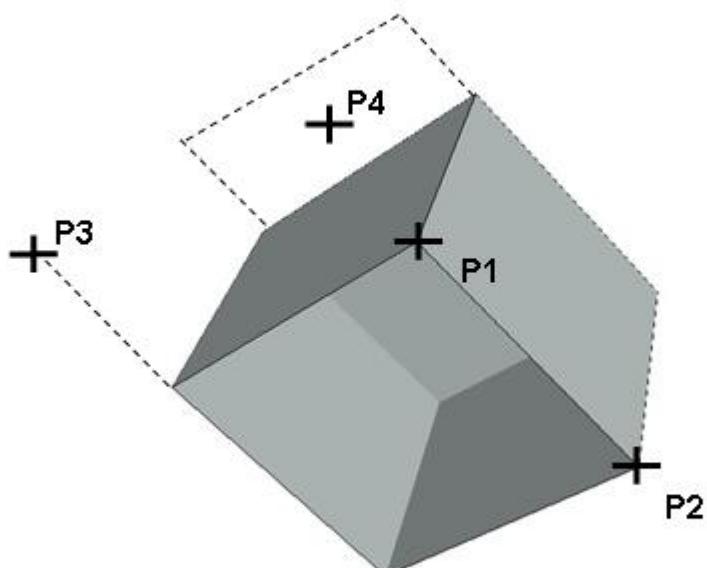
This shape is a box with four faces, no faces in the P1P4 direction.

- Point 1 : origin point for the box (P1)
- Point 2 : point defining with P1 the length and the first edge of the box (P2)
- Point 3 : point defining with P1 and P2 the plane of the second edge and the width of the box by orthogonal projection on the P1P2 axis (P3)
- Point 4 : point defining the height of the box by orthogonal projection on the P1P2P3 plane (P4)

5 faces box

This shape is a box with five faces, no face in the P1P4 direction on the negative side.

- Point 1 : origin point for the box (P1)
- Point 2 : point defining with P1 the first of edge the box (P2)
- Point 3 : point defining with P1 the second edge of the box (P3)
- Point 4 : point defining with P1 the third edge of the box (P4)



DEFINITION

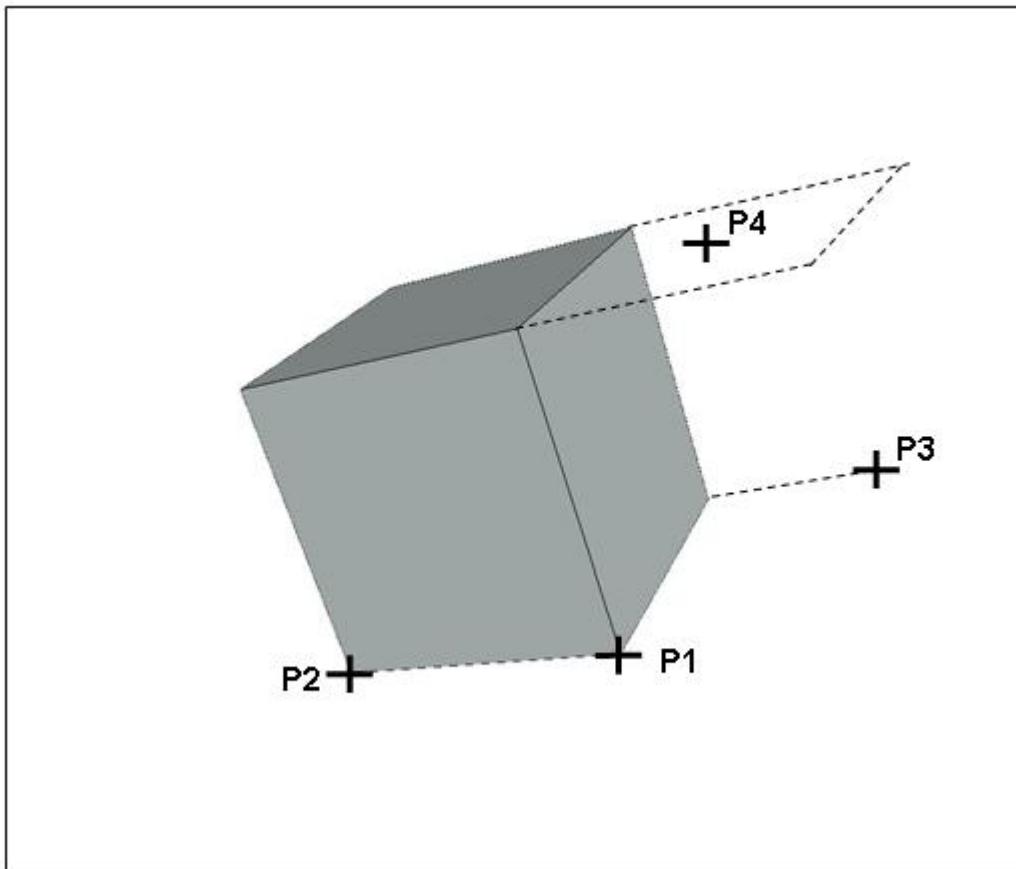
This shape is a box with five faces, no face in the P1P4 direction on the negative side.

- Point 1 : origin point for the box (P1)
- Point 2 : point defining with P1 the length and the first edge of the box (P2)
- Point 3 : point defining with P1 and P2 the plane of the second edge and the width of the box by orthogonal projection on the P1P2 axis (P3)
- Point 4 : point defining the height of the box by orthogonal projection on the P1P2P3 plane (P4)

6 faces box

This shape is a box with six faces.

- Point 1 : origin point for the box (P1)
- Point 2 : point defining with P1 the first of edge the box (P2)
- Point 3 : point defining with P1 the second edge of the box (P3)
- Point 4 : point defining with P1 the third edge of the box (P4)



DEFINITION

This shape is a box with six faces.

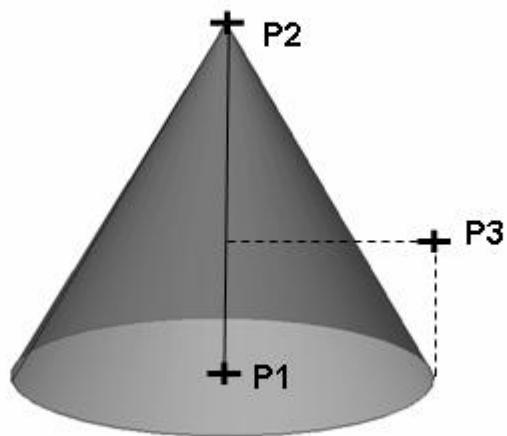
- Point 1 : origin point for the box (P1)
- Point 2 : point defining with P1 the length and the first edge of the box (P2)
- Point 3 : point defining with P1 and P2 the plane of the second edge and the width of the box by orthogonal projection on the P1P2 axis (P3)
- Point 4 : point defining the height of the box by orthogonal projection on the P1P2P3 plane (P4)

Cone

3 points cone defined by base center - height - radius

This shape is an entire cone.

- Point 1 : center of the cone's base (P1)
- Point 2 : point defining with P1 the cone's axis and height (P2)
- Point 3 : point defining the cone's radius by orthogonal projection on the cone's axis (P3)



DEFINITION

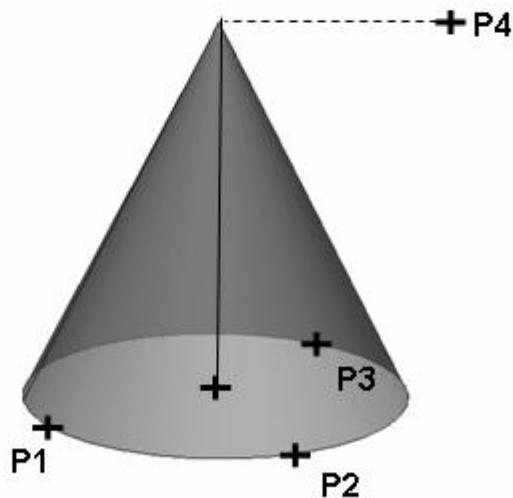
This shape is an entire cone.

- Point 1 : center of the cone's base (P1)
- Point 2 : point defining with P1 the cone's axis and height (P2)
- Point 3 : point defining the cone's radius by orthogonal projection on the cone's axis (P3)

4 points cone defined by 3 points on base - height

This shape is an entire cone.

- Point 1 : a point on the cone's base (P1)
- Point 2 : a point on the cone's base (P2)
- Point 3 : a point on the cone's base (P3)
- Point 4 : point defining the cone's height by orthogonal projection on the cone's axis (P4)



DEFINITION

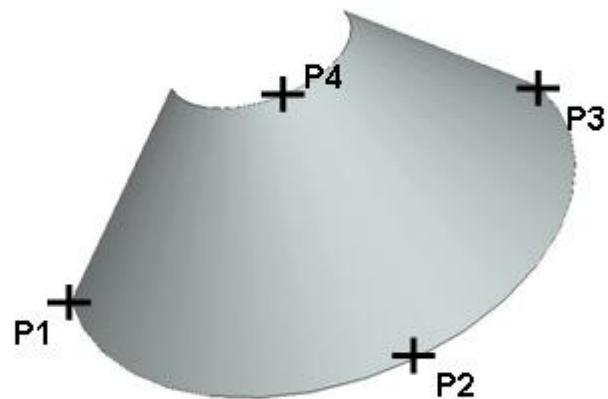
This shape is an entire cone.

- Point 1 : first point on the cone's base (P1)
- Point 2 : second point on the cone's base (P2)
- Point 3 : third point on the cone's base (P3)
- Point 4 : point defining the cone's height by orthogonal projection on the cone's axis (P4)

4 points truncated cone defined by 3 points on base - 1 point on other base

This shape is a truncated cone. The part of cone is defined between the two bases and between P1 and P3 with respect to the direction of the axis from base to other base.

- Point 1 : a point on the cone's base (P1)
- Point 2 : a point on the cone's base (P2)
- Point 3 : a point on the cone's base (P3)
- Point 4 : a point on the other base (P4)



DEFINITION

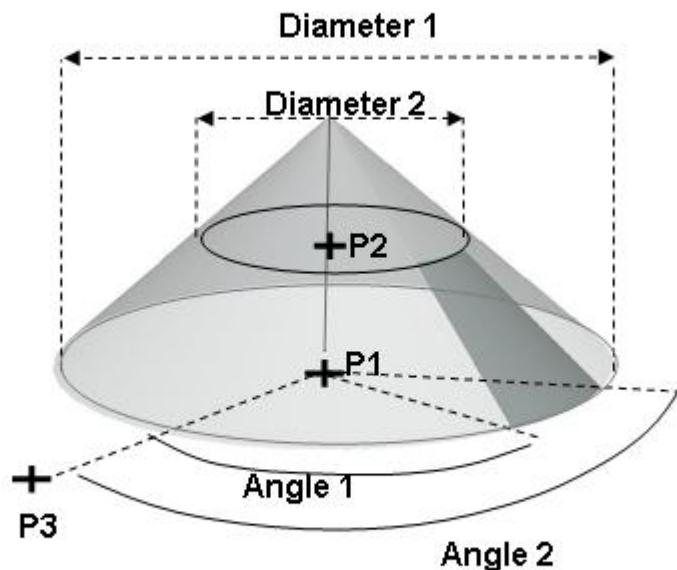
This shape is a truncated cone. The part of cone is defined between the two bases and between P1 and P3 with respect to the direction of the axis from base to other base.

- Point 1 : first point on the cone's base (P1)
- Point 2 : second point on the cone's base (P2)
- Point 3 : third point on the cone's base (P3)
- Point 4 : a point on the other base (P4)

General truncated cone

This shape is a truncated cone. The part of cone is defined between the two bases and between angle1 and angle2 with respect to the direction of the axis P1P2.

- Point 1 : center of the cone's base (P1)
- Point 2 : point defining with P1 the cone's axis and height (P2)
- Point 3 : point defining with P1 and P2 the plane of angular truncation (P3)
- Diameter 1 : diameter of the cone's base with P1 center
- Diameter 2 : diameter of the other base with P2 center
- Angle 1 : start angle of the cone relatively to the plane of angular truncation
- Angle 2 : stop angle of the cone relatively to the plane of angular truncation



DEFINITION

This shape is a truncated cone. The part of cone is defined between the two bases and between angle1 and angle2 with respect to the direction of the axis P1P2.

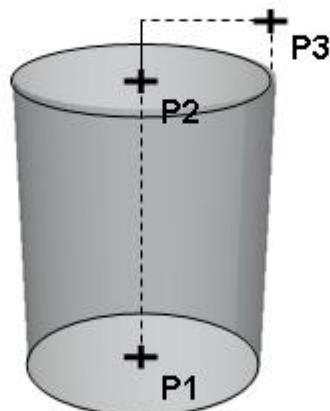
- Point 1 : center of the cone's base (P1)
- Point 2 : point defining with P1 the cone's axis and height (P2)
- Point 3 : point defining with P1 and P2 the plane of angular truncation (P3)
- Diameter 1 : diameter of the cone's base with P1 center
- Diameter 2 : diameter of the other base with P2 center
- Angle 1 : start angle of the cone relatively to the plane of angular truncation
- Angle 2 : stop angle of the cone relatively to the plane of angular truncation

Cylinder

3 points cylinder defined by base center - height - radius

This shape is an entire cylinder.

- Point 1 : center of the cylinder's base (P1)
- Point 2 : point defining with P1 the cylinder's axis and height (P2)
- Point 3 : point defining the cylinder's radius by orthogonal projection on the cylinder's axis (P3)



DEFINITION

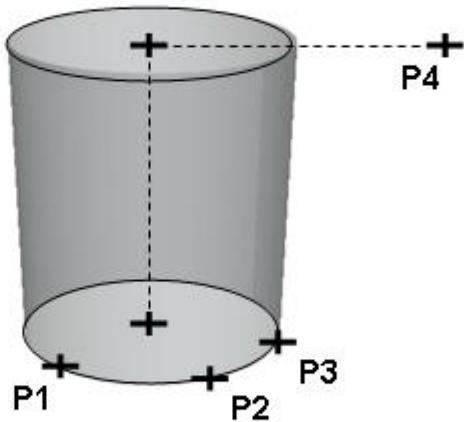
This shape is an entire cylinder.

- Point 1 : center of the cylinder's base (P1)
- Point 2 : point defining with P1 the cylinder's axis and height (P2)
- Point 3 : point defining the cylinder's radius by orthogonal projection on the cylinder's axis (P3)

4 points cylinder defined by 3 points on base - height

This shape is an entire cylinder.

- Point 1 : a point on the cylinder's base (P1)
- Point 2 : a point on the cylinder's base (P2)
- Point 3 : a point on the cylinder's base (P3)
- Point 4 : point defining the cylinder's height by orthogonal projection on the cylinder's axis (P4)



DEFINITION

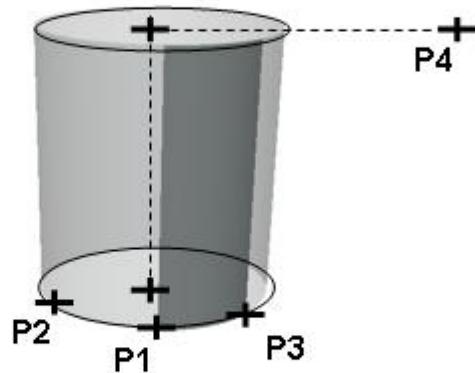
This shape is an entire cylinder.

- Point 1 : first point on the cylinder's base (P1)
- Point 2 : second point on the cylinder's base (P2)
- Point 3 : third point on the cylinder's base (P3)
- Point 4 : point defining the cylinder's height by orthogonal projection on the cylinder's axis (P4)

4 points truncated cylinder defined by 3 points on base - height

This shape is a truncated cylinder. The part of cylinder is defined between the two bases and between P1 and P3 with respect to the direction of the axis from base to other base.

- Point 1 : a point on the cylinder's base (P1)
- Point 2 : a point on the cylinder's base (P2)
- Point 3 : a point on the cylinder's base (P3)
- Point 4 : point defining the cylinder's height by orthogonal projection on the cylinder's axis (P4)



DEFINITION

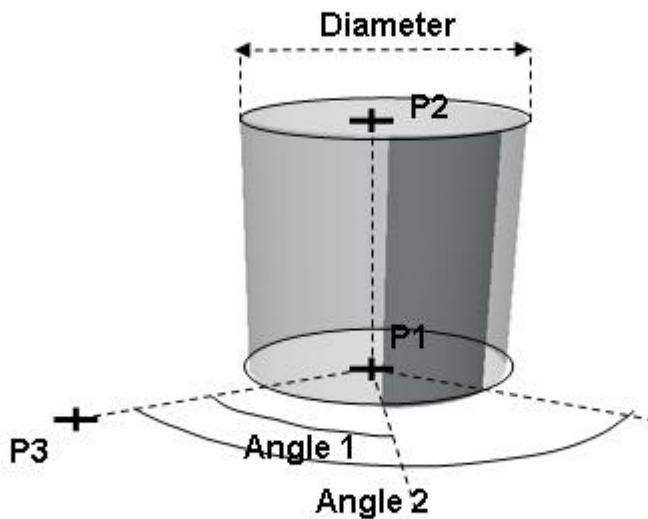
This shape is a truncated cylinder. The part of cylinder is defined between the two bases and between P1 and P3 with respect to the direction of the axis from base to other base.

- Point 1 : first point on the cylinder's base (P1)
- Point 2 : second point on the cylinder's base (P2)
- Point 3 : third point on the cylinder's base (P3)
- Point 4 : point defining the cylinder's height by orthogonal projection on the cylinder's axis (P4)

General truncated cylinder

This shape is a truncated cylinder. The part of cylinder is defined between the two bases and between angle1 and angle2 with respect to the direction of the axis P1P2.

- Point 1 : center of the cylinder's base (P1)
- Point 2 : point defining with P1 the cylinder's axis and height (P2)
- Point 3 : point defining with P1 and P2 the plane of angular truncation (P3)
- Diameter: diameter of the cylinder's base
- Angle 1 : start angle of the cylinder relatively to the plane of angular truncation
- Angle 2 : stop angle of the cylinder relatively to the plane of angular truncation



DEFINITION

This shape is a truncated cylinder. The part of cylinder is defined between the two bases and between angle1 and angle2 with respect to the direction of the axis P1P2.

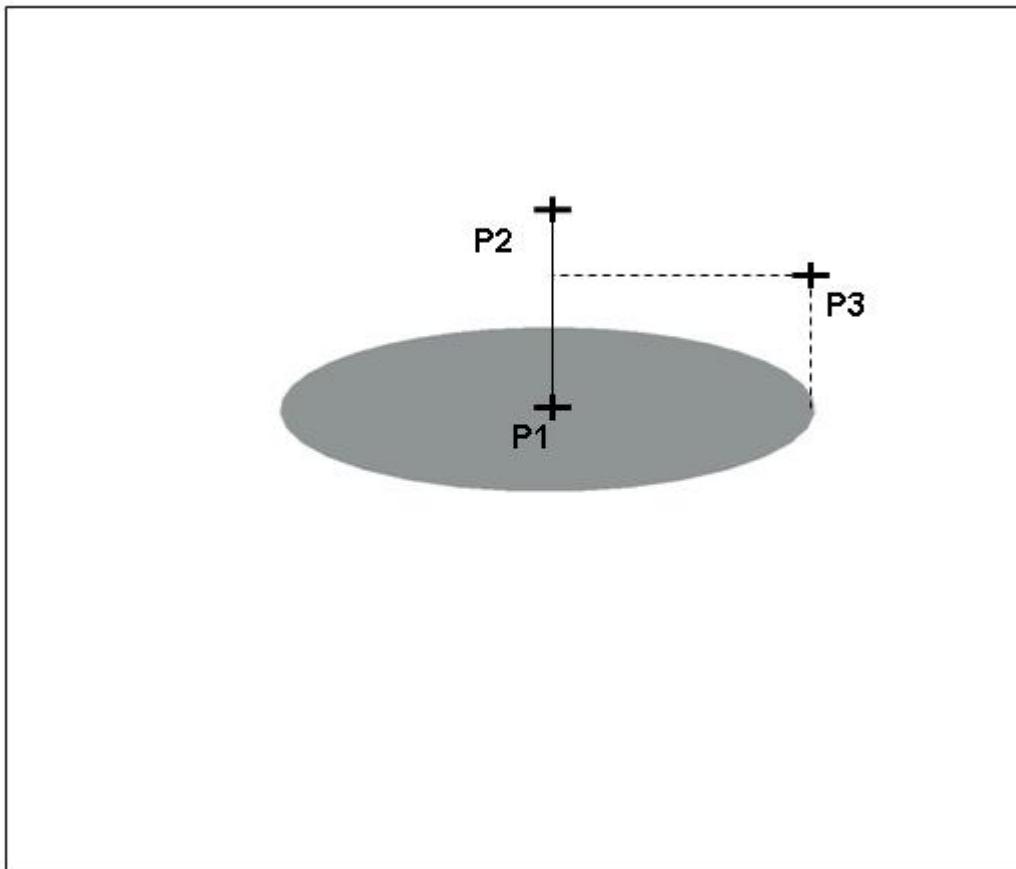
- Point 1 : center of the cylinder's base (P1)
- Point 2 : point defining with P1 the cylinder's axis and height (P2)
- Point 3 : point defining with P1 and P2 the plane of angular truncation (P3)
- Diameter: diameter of the cylinder's base
- Angle 1 : start angle of the cylinder relatively to the plane of angular truncation
- Angle 2 : stop angle of the cylinder relatively to the plane of angular truncation

Disc

3 points disc

This shape is an entire disc.

- Point 1 : center of the disc (P1)
- Point 2 : point defining with P1 the disc' normal (P2)
- Point 3 : point defining the disc' radius by orthogonal projection on the disc' axis (P3)



DEFINITION

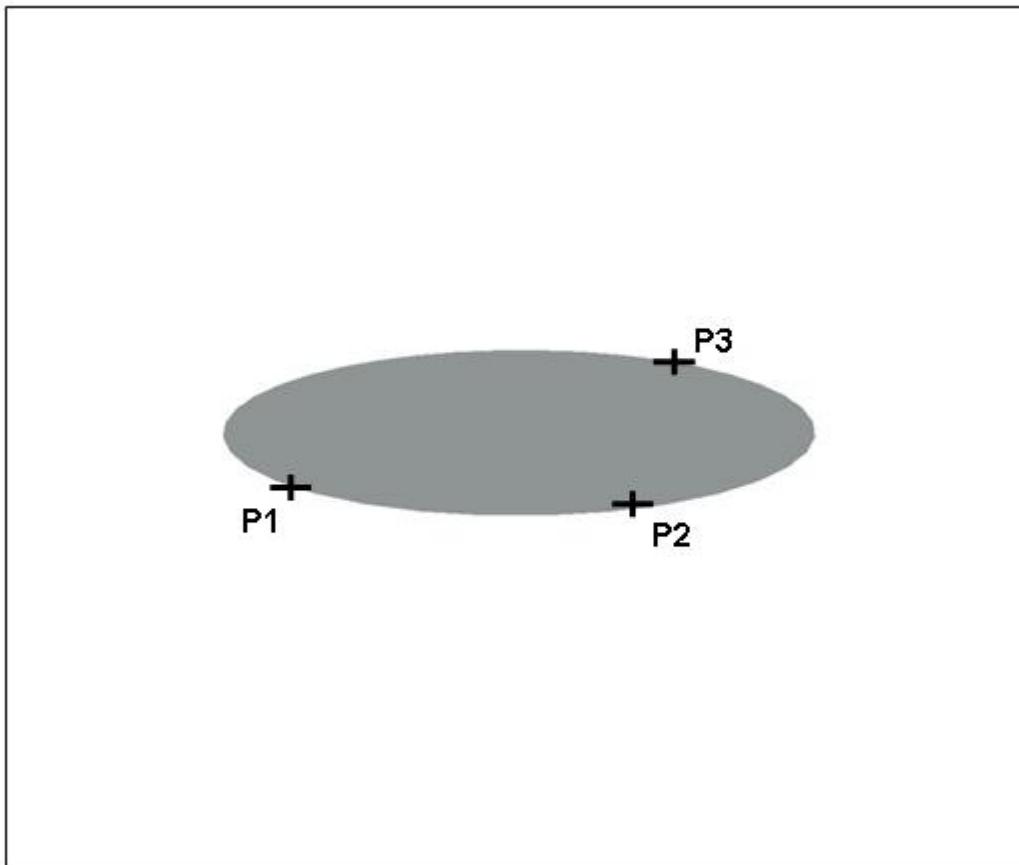
This shape is an entire disc.

- Point 1 : center of the disc (P1)
- Point 2 : point defining with P1 the disc's normal (P2)
- Point 3 : point defining the disc's radius by orthogonal projection on the disc's normal (P3)

Default disc

This shape is an entire disc.

- Point 1 : a point on the disc' outer border (P1)
- Point 2 : a point on the disc' outer border (P2)
- Point 3 : a point on the disc' outer border (P3)



DEFINITION

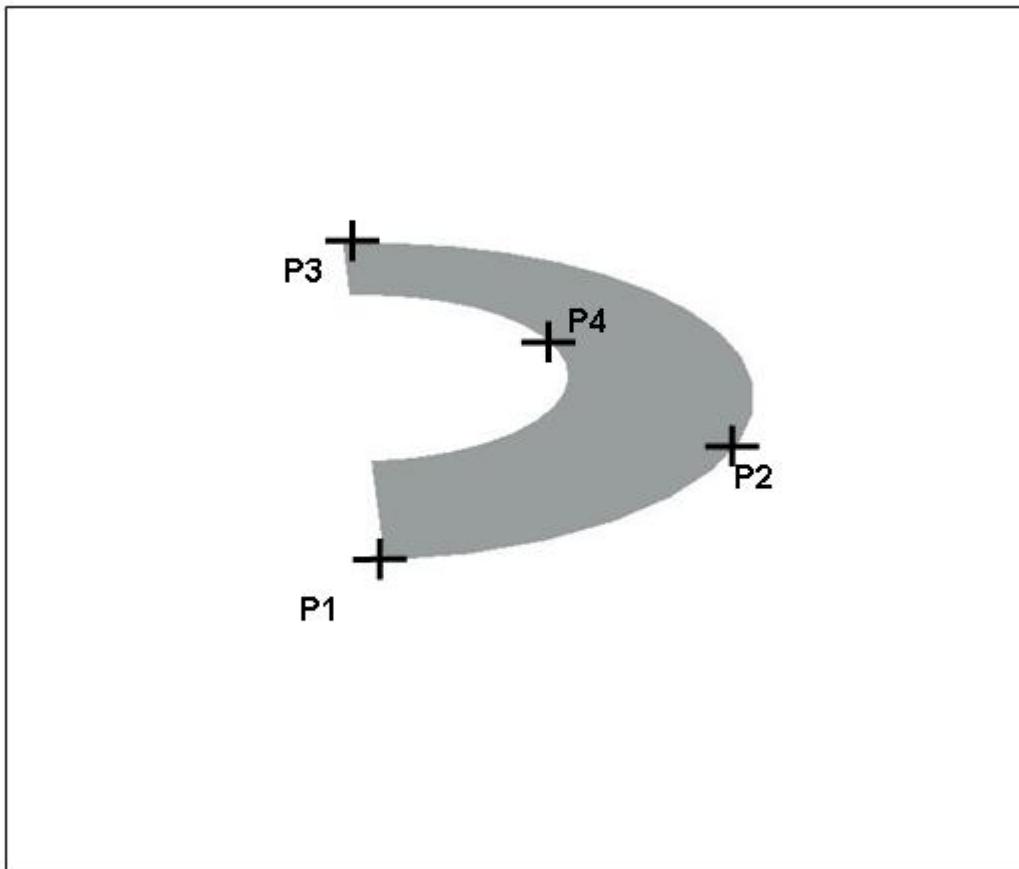
This shape is an entire disc.

- **Point 1 : first point on the disc' outer border (P1)**
- **Point 2 : second point on the disc' outer border (P2)**
- **Point 3 : third point on the disc' outer border (P3)**

4 points truncated disc defined by 3 points on external border - 1 point for internal border

This shape is a truncated disc. The part of the disc is defined between P1 and P3 including P2.

- Point 1 : a point on the disc' outer border (P1)
- Point 2 : a point on the disc' outer border (P2)
- Point 3 : a point on the disc' outer border (P3)
- Point 4 : a point on the disc' inner border (P4)



DEFINITION

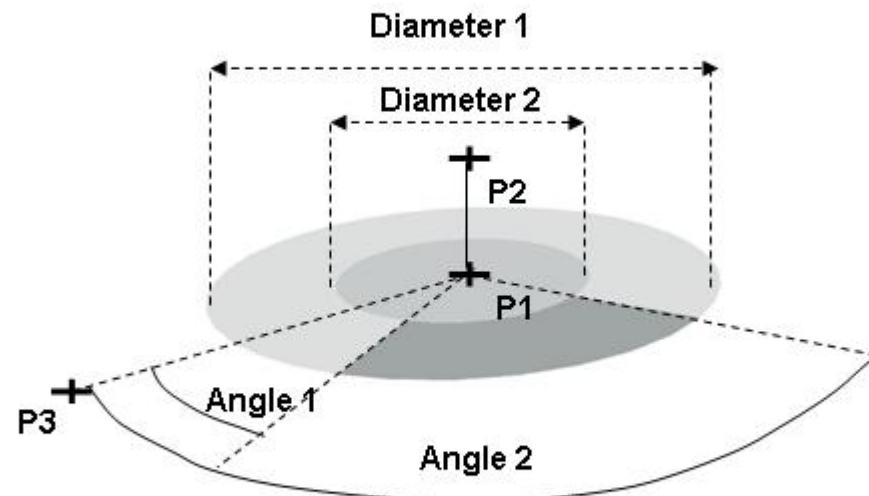
This shape is a truncated disc. The part of the disc is defined between P1 and P3 including P2.

- Point 1 : first point on the disc' outer border (P1)
- Point 2 : second point on the disc' outer border (P2)
- Point 3 : third point on the disc' outer border (P3)
- Point 4 : a point on the disc' inner border (P4)

General truncated disc

This shape is a truncated disc. The part of disc is defined between the inner border and the outer border and between angle1 and angle2 with respect to the direction of the axis P1P2.

- Point 1 : center of the disc (P1)
- Point 2 : point defining with P1 the disc' normal (P2)
- Point 3 : point defining with P1 and P2 the plane of angular truncation (P3)
- Diameter 1 : diameter of the disc' outer border
- Diameter 2 : diameter of the disc' inner border
- Angle 1 : start angle of the disc relatively to the plane of angular truncation
- Angle 2 : stop angle of the disc relatively to the plane of angular truncation



DEFINITION

This shape is a truncated disc. The part of disc is defined between the inner border and the outer border and between angle1 and angle2 with respect to the direction of the axis P1P2.

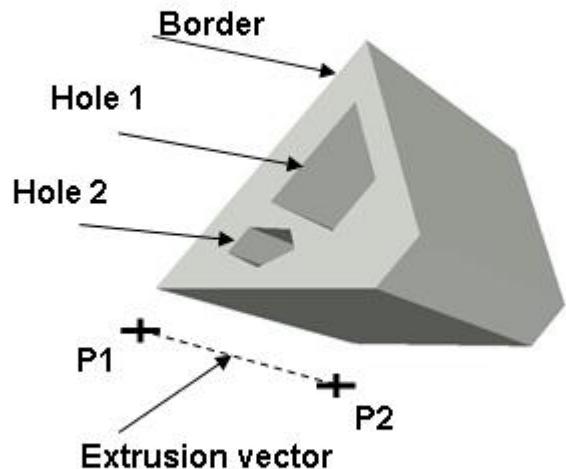
- Point 1 : center of the disc (P1)
- Point 2 : point defining with P1 the disc' normal (P2)
- Point 3 : point defining with P1 and P2 the plane of angular truncation (P3)
- Diameter 1 : diameter of the disc' outer border
- Diameter 2 : diameter of the disc' inner border
- Angle 1 : start angle of the disc relatively to the plane of angular truncation
- Angle 2 : stop angle of the disc relatively to the plane of angular truncation

Extrusion

Extruded polygon with holes

This shape is the extrusion along an axis of a plane and convex polygon with holes inside. The holes are plane and convex polygons and must not have any edge intersection. Each polygon is at least defined by three points.

- Border : external border of the polygon
 - Border points (at least 3) : points on the external border of the border
- Holes (optional) : external borders of the holes, no limitation for the number of holes
 - Holes points (at least 3) : points on the external border of the hole
- Point 1 : first point defining the axis of extrusion
- Point 2 : second point defining the axis of extrusion



DEFINITION

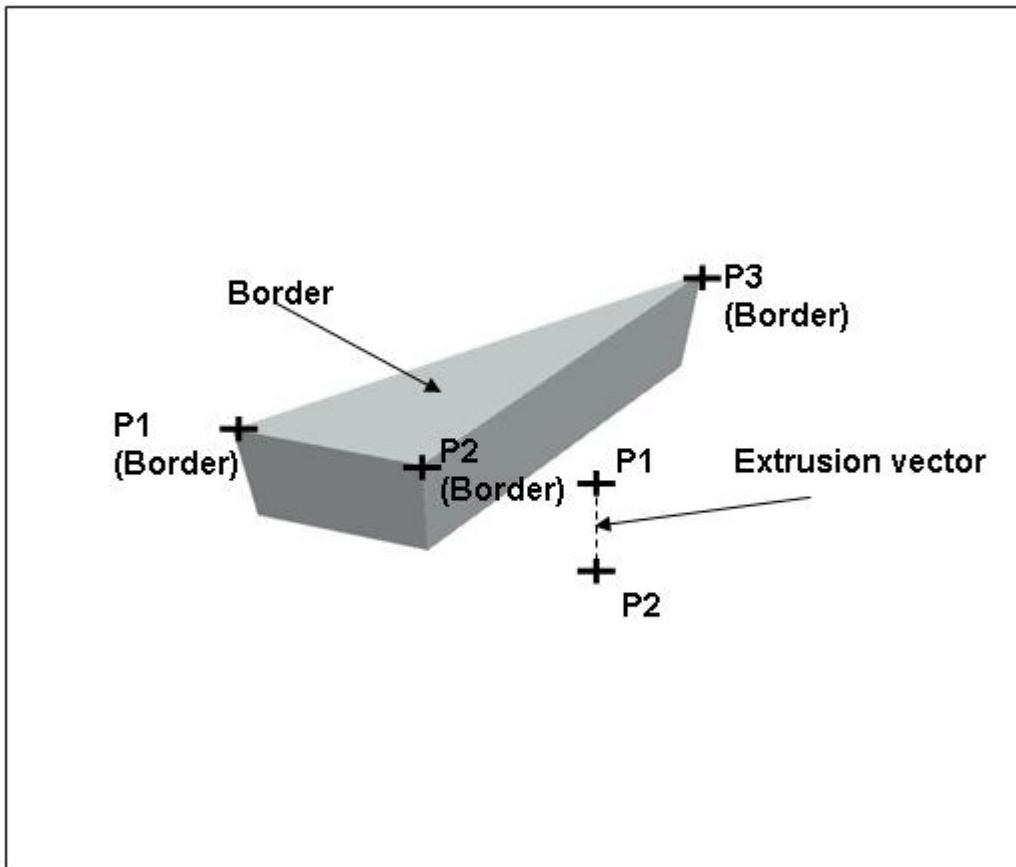
This shape is the extrusion along a vector of a plane and convex polygon with holes inside. The holes are plane and convex polygons and must not have any edge intersection. Each polygon is at least defined by three points.

- **Border** : external border of the polygon
 - **Border points** (at least 3) : points on the external border of the border
- **Holes (optional)** : external borders of the holes, no limitation for the number of holes
 - **Holes points** (at least 3) : points on the external border of the hole
- **Point 1** : first point defining the vector of extrusion
- **Point 2** : second point defining the vector of extrusion

Extruded polygon

This shape is the extrusion along an axis of a plane and convex polygon. The polygon is at least defined by three points.

- Polygon : external border of the polygon
 - Polygon points (at least 3) : points on the external border of the polygon
- Point 1 : first point defining the axis of extrusion
- Point 2 : second point defining the axis of extrusion



DEFINITION

This shape is the extrusion along a vector of a plane and convex polygon. The polygon is at least defined by three points.

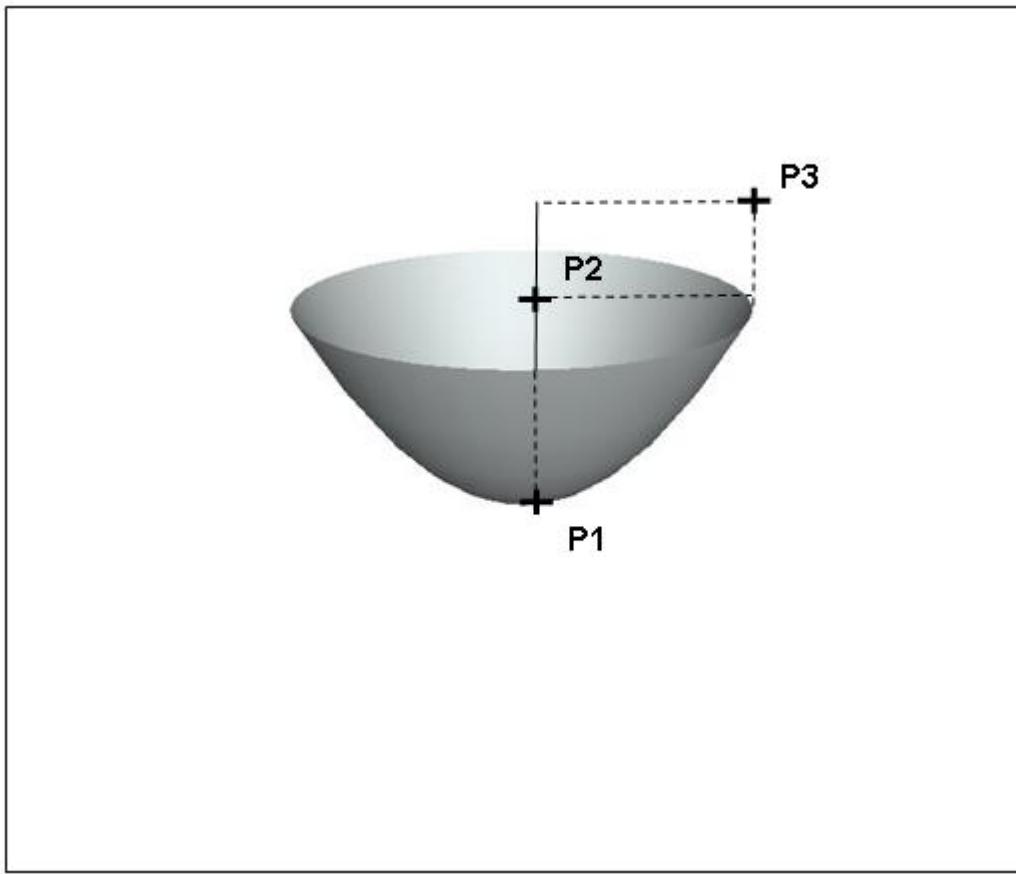
- **Polygon : external border of the polygon**
 - **Polygon points (at least 3) : points on the external border of the polygon**
- **Point 1 : first point defining the vector of extrusion**
- **Point 2 : second point defining the vector of extrusion**

Parabola

Paraboloid 3 points

This shape is an entire paraboloid.

- Point 1 : center of the paraboloid (P1)
- Point 2 : point defining with P1 the paraboloid's axis and height (P2)
- Point 3 : point defining the paraboloid's radius at P2 height by orthogonal projection on the paraboloid's axis (P3)



DEFINITION

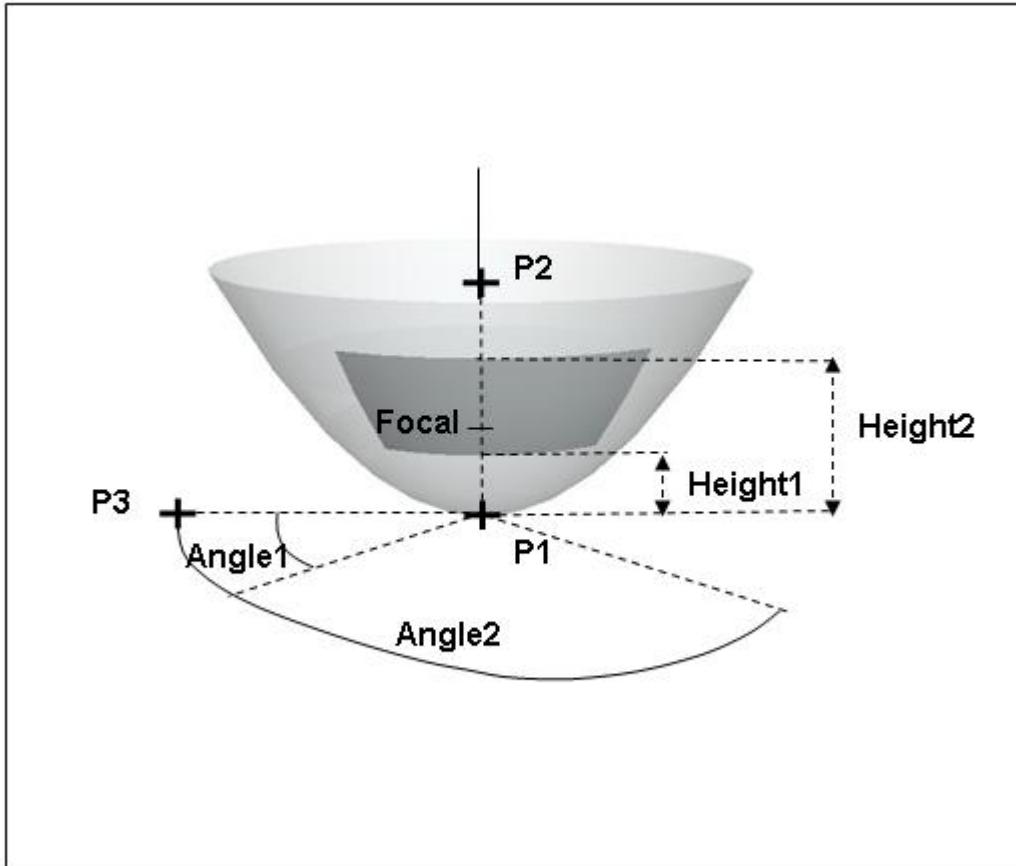
This shape is an entire paraboloid.

- Point 1 : center of the paraboloid (P1)
- Point 2 : point defining with P1 the paraboloid's axis and height (P2)
- Point 3 : point defining the paraboloid's radius at P2 height by orthogonal projection on the paraboloid's axis (P3)

Truncated paraboloid

This shape is a truncated paraboloid. The part of paraboloid is defined between the two heights and between angle1 and angle2 with respect to the direction of the axis P1P2.

- Point 1 : center of the paraboloid (P1)
- Point 2 : point defining with P1 the paraboloid's axis (P2)
- Point 3 : point defining with P1 and P2 the plane of angular truncation (P3)
- Focal : focal distance of the paraboloid
- Height 1 : start height of the paraboloid on the axis relatively to the center
- Height 2 : stop height of the paraboloid on the axis relatively to the center
- Angle 1 : start angle of the paraboloid relatively to the plane of angular truncation
- Angle 2 : stop angle of the paraboloid relatively to the plane of angular truncation



DEFINITION

This shape is a truncated paraboloid. The part of paraboloid is defined between the two heights and between angle1 and angle2 with respect to the direction of the axis P1P2.

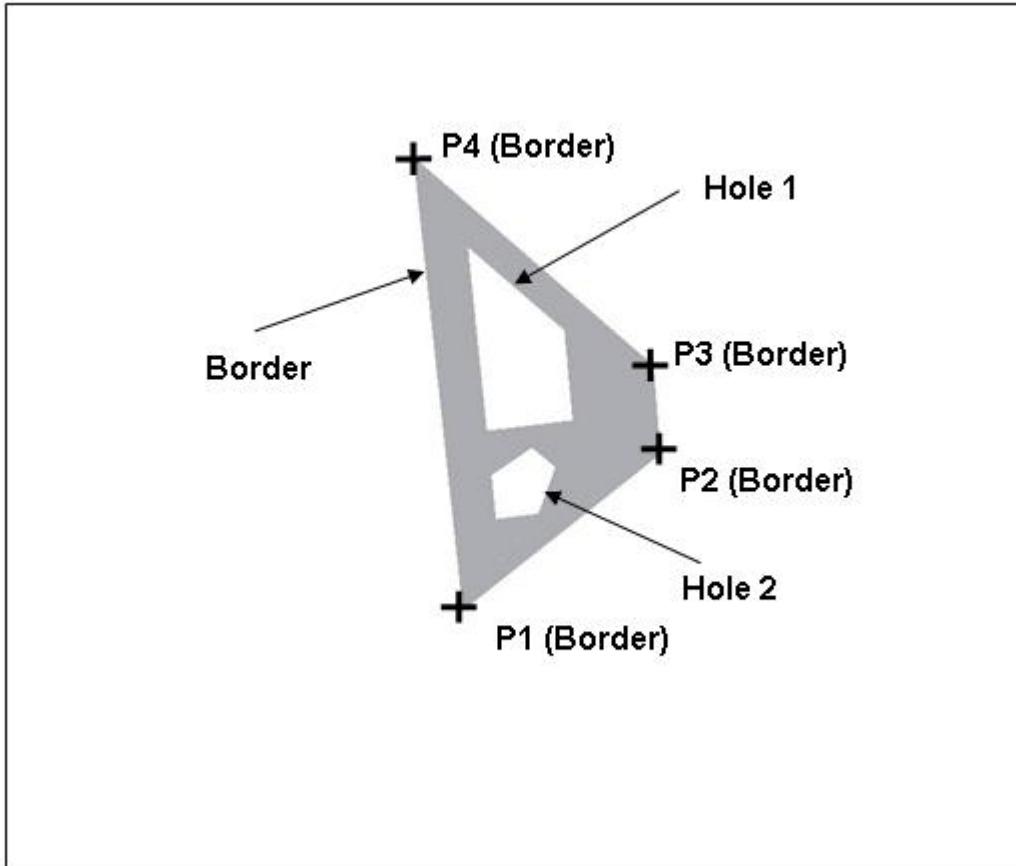
- Point 1 : center of the paraboloid (P1)
- Point 2 : point defining with P1 the paraboloid's axis (P2)
- Point 3 : point defining with P1 and P2 the plane of angular truncation (P3)
- Focal : focal distance of the paraboloid
- Height 1 : start height of the paraboloid on the axis relatively to the center
- Height 2 : stop height of the paraboloid on the axis relatively to the center
- Angle 1 : start angle of the paraboloid relatively to the plane of angular truncation
- Angle 2 : stop angle of the paraboloid relatively to the plane of angular truncation

Polygon

Polygon with holes

This shape is a plane and convex polygon with holes inside. The holes are plane and convex polygons and must not have any edge intersection. Each polygon is at least defined by three points.

- Border : external border of the polygon
 - Border points (at least 3) : points on the external border of the border
- Holes (optional) : external borders of the holes, no limitation for the number of holes
 - Hole points (at least 3) : points on the external border of the hole



DEFINITION

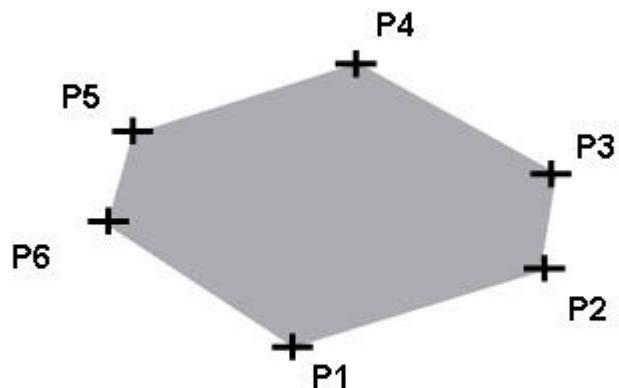
This shape is a plane and convex polygon with holes inside. The holes are plane and convex polygons and must not have any edge intersection. Each polygon is at least defined by three points.

- **Border** : external border of the polygon
 - **Border points (at least 3)** : points on the external border of the border
- **Holes (optional)** : external borders of the holes, no limitation for the number of holes
 - **Hole points (at least 3)** : points on the external border of the hole

Default polygon

This shape is a plane and convex polygon. The polygon is at least defined by three points.

- Points (at least 3) : points on the external border of the polygon



DEFINITION

This shape is a plane and convex polygon. The polygon is at least defined by three points.

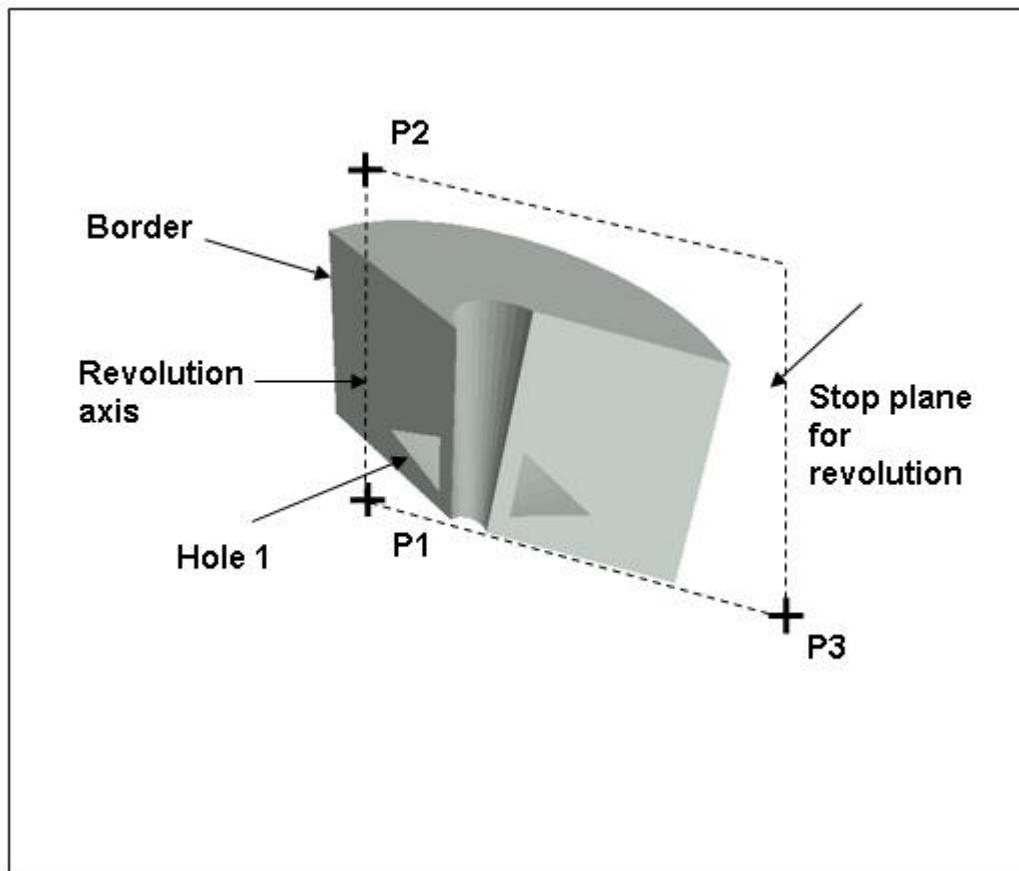
- **Points (at least 3) : points on the external border of the polygon**

Revolution

Revolved polygon with holes

This shape is the revolution around an axis of a plane and convex polygon with holes inside. The holes are plane and convex polygons and must not have any edge intersection. Each polygon is at least defined by three points.

- Border : external border of the polygon
 - Border points (at least 3) : points on the external border of the border
- Holes (optional) : external borders of the holes, no limitation for the number of holes
 - Holes point (at least 3) : points on the external border of the hole
- Point 1 : first point defining the axis of revolution (P1)
- Point 2 : second point defining the axis of revolution (P2)
- Point 3 : point defining with P1 and P2 the stop plane of the revolution (P3). If P3 equal P2 then the angle of revolution is 90°. If P3 equal P1 or if P1, P2 and P3 are aligned then the angle of revolution is 360°.



DEFINITION

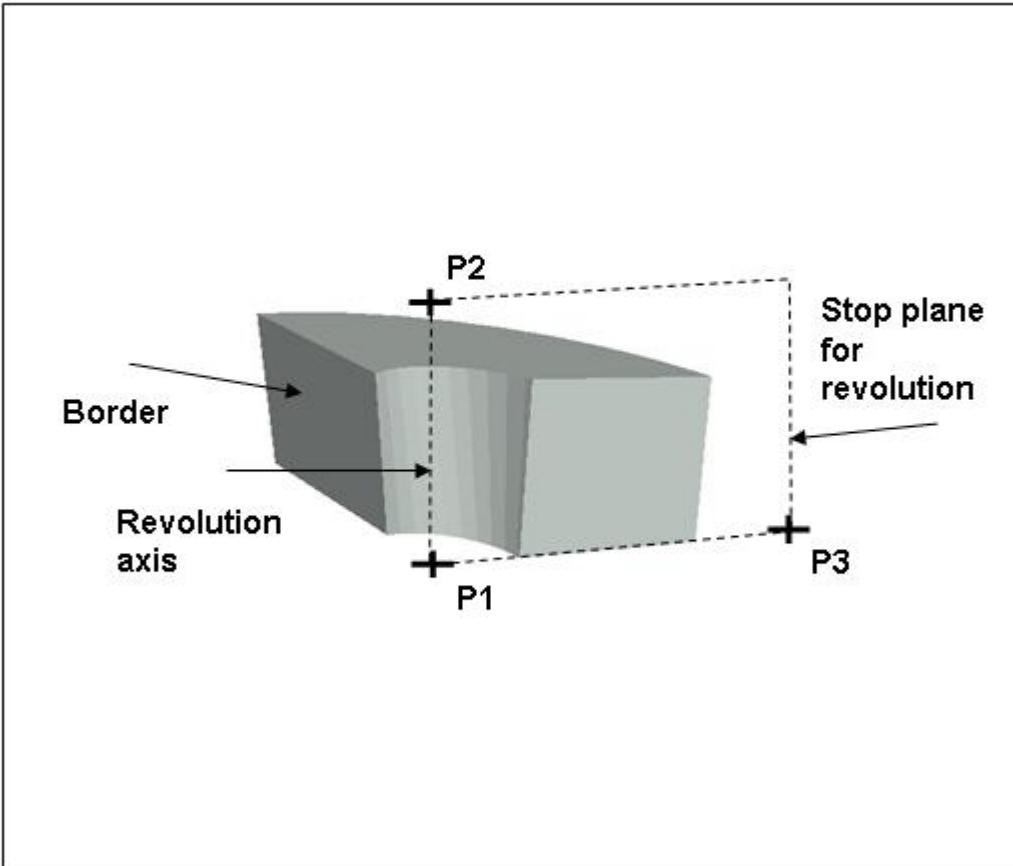
This shape is the revolution around an axis of a plane and convex polygon with holes inside. The holes are plane and convex polygons and must not have any edge intersection. Each polygon is at least defined by three points.

- **Border** : external border of the polygon
 - Border points (at least 3) : points on the external border of the border
- **Holes (optional)** : external borders of the holes, no limitation for the number of holes
 - Holes point (at least 3) : points on the external border of the hole
- **Point 1** : first point defining the axis of revolution (P1)
- **Point 2** : second point defining the axis of revolution (P2)
- **Point 3** : point defining with P1 and P2 the stop plane of the revolution (P3). If P3 equal P2 then the angle of revolution is 90°. If P3 equal P1 or if P1, P2 and P3 are aligned then the angle of revolution is 360°.

Revolved polygon

This shape is the revolution around an axis of a plane and convex polygon. The polygon is at least defined by three points.

- Polygon : external border of the polygon
 - Polygon points (at least 3) : points on the external border of the polygon
- Point 1 : first point defining the axis of revolution (P1)
- Point 2 : second point defining the axis of revolution (P2)
- Point 3 : point defining with P1 and P2 the stop plane of the revolution (P3). If P3 equal P2 then the angle of revolution is 90°. If P3 equal P1 or if P1, P2 and P3 are aligned then the angle of revolution is 360°.



DEFINITION

This shape is the revolution around an axis of a plane and convex polygon. The polygon is at least defined by three points.

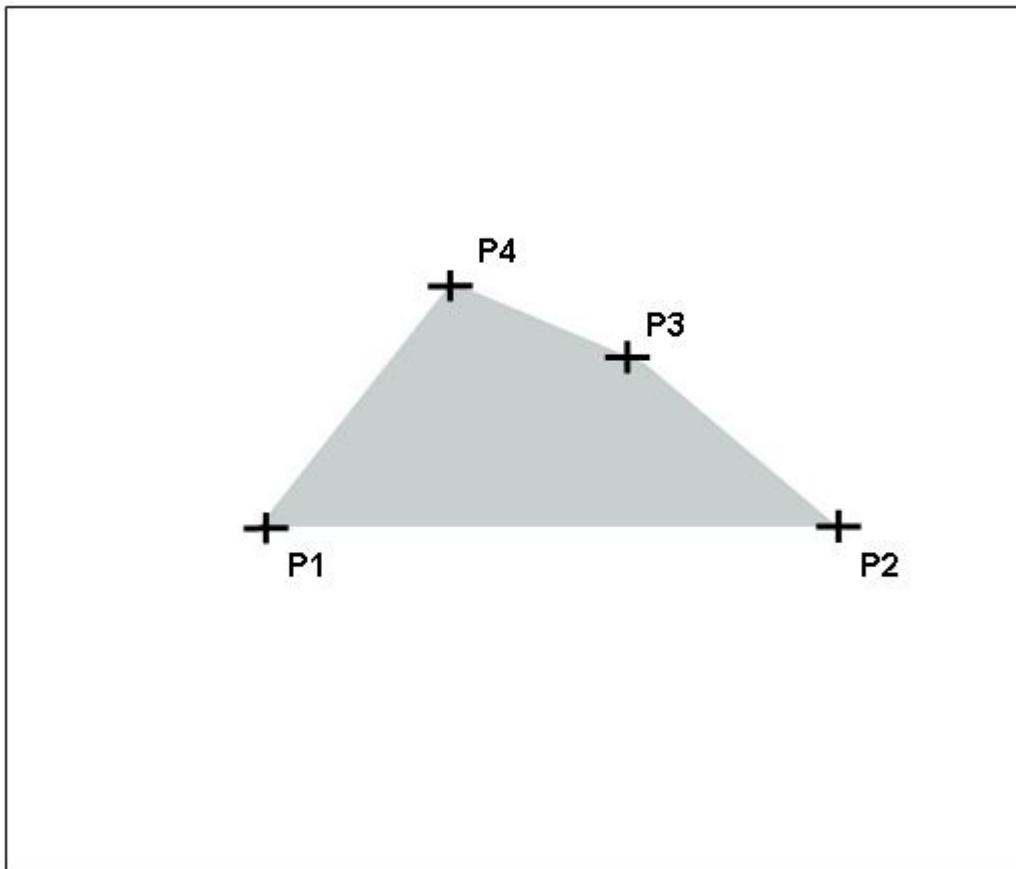
- **Polygon : external border of the polygon**
 - Polygon points (at least 3) : points on the external border of the polygon
- **Point 1 : first point defining the axis of revolution (P1)**
- **Point 2 : second point defining the axis of revolution 'P2'**
- **Point 3 : point defining with P1 and P2 the stop plane of the revolution (P3).** If P3 equal P2 then the angle of revolution is 90°. If P3 equal P1 or if P1, P2 and P3 are aligned then the angle of revolution is 360°.

Simple shapes

Quadrangle

This shape is plane and convex quadrangle

- Point 1 : first point of the quadrangle
- Point 2 : second point of the quadrangle
- Point 3 : third point of the quadrangle
- Point 4 : last point of the quadrangle



DEFINITION

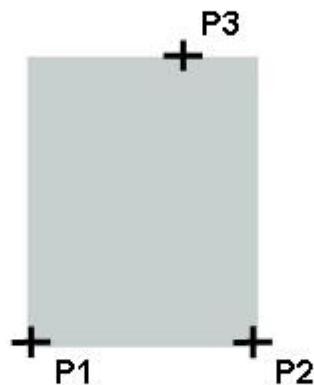
This shape is plane and convex quadrangle

- **Point 1 : first point of the quadrangle**
- **Point 2 : second point of the quadrangle**
- **Point 3 : third point of the quadrangle**
- **Point 4 : last point of the quadrangle**

Rectangle

This shape is a rectangle

- Point 1 : first point of the rectangle (P1)
- Point 2 : second point of the rectangle defining with P1 the first edge (P2)
- Point 3 : point defining the height of the rectangle by orthogonal projection on the first edge (P3)



DEFINITION

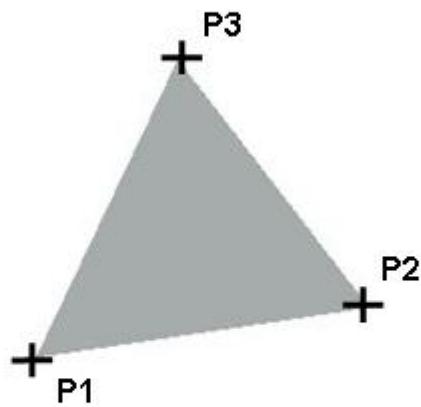
This shape is a rectangle

- **Point 1 : first point of the rectangle (P1)**
- **Point 2 : second point of the rectangle defining with P1 the first edge (P2)**
- **Point 3 : point defining the height of the rectangle by orthogonal projection on the first edge (P3)**

Triangle

This shape is a triangle

- Point 1 : first point of the triangle (P1)
- Point 2 : second point of the triangle (P2)
- Point 3 : third point of the triangle (P3)



DEFINITION

This shape is a triangle

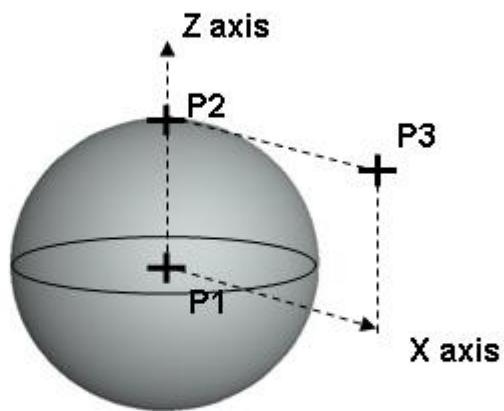
- **Point 1 : first point of the triangle (P1)**
- **Point 2 : second point of the triangle (P2)**
- **Point 3 : third point of the triangle (P3)**

Sphere

3 points sphere

This shape is an entire sphere.

- Point 1 : center of the sphere (P1)
- Point 2 : point defining with P1 the radius of the sphere and the Z axis (P2)
- Point 3 : point defining with P1 and P2 the plane of the X axis (P3)



DEFINITION

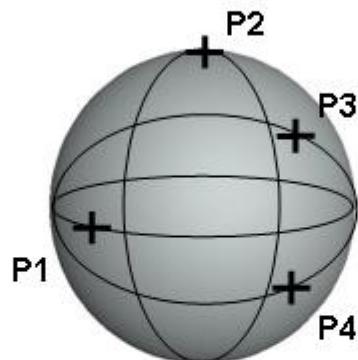
This shape is an entire sphere.

- **Point 1 : center of the sphere (P1)**
- **Point 2 : point defining with P1 the radius of the sphere and the Z axis (P2)**
- **Point 3 : point defining with P1 and P2 the plane of the X axis (P3)**

4 points sphere

This shape is an entire sphere.

- Point 1 : first point of the sphere (P1)
- Point 2 : second point of the sphere (P2)
- Point 3 : third point of the sphere (P3)
- Point 4 : fourth point of the sphere (P4)



DEFINITION

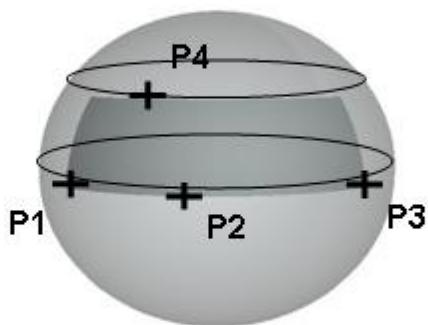
This shape is an entire sphere.

- **Point 1 : first point of the sphere (P1)**
- **Point 2 : second point of the sphere (P2)**
- **Point 3 : third point of the sphere (P3)**
- **Point 4 : fourth point of the sphere (P4)**

4 points truncated sphere defined by 3 points on upper border - 1 on downer

This shape is a truncated sphere. The part of sphere is defined between the two borders and between P1 and P3 including P2.

- Point 1 : first point of the sphere on the upper border (P1)
- Point 2 : second point of the sphere on the upper border (P2)
- Point 3 : third point of the sphere on the upper border (P3)
- Point 4 : fourth point of the sphere on the downer border (P4)



DEFINITION

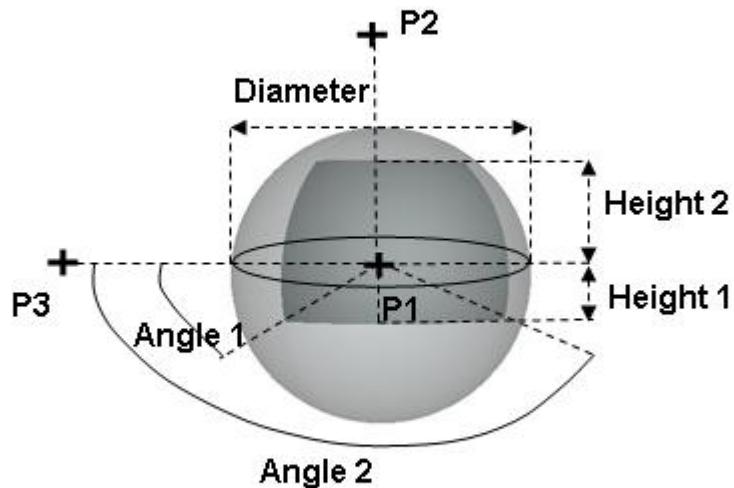
This shape is a truncated sphere. The part of sphere is defined between the two borders and between P1 and P3 including P2.

- Point 1 : first point of the sphere on the upper border (P1)
- Point 2 : second point of the sphere on the upper border (P2)
- Point 3 : third point of the sphere on the upper border (P3)
- Point 4 : fourth point of the sphere on the downer border (P4)

General truncated sphere

This shape is a truncated sphere. The part of sphere is defined between the two bases and between angle1 and angle2 with respect to the direction of the axis P1P2.

- Point 1 : center of the sphere (P1)
- Point 2 : point defining with P1 the sphere's axis to define the bases (P2)
- Point 3 : point defining with P1 and P2 the plane of angular truncation (P3)
- Diameter : diameter of the sphere
- Height 1 : defining with P1 the height of the first sphere's base
- Height 2 : defining with P1 the height of the second sphere's base
- Angle 1 : start angle of the sphere relatively to the plane of angular truncation
- Angle 2 : stop angle of the sphere relatively to the plane of angular truncation



DEFINITION

This shape is a truncated sphere. The part of sphere is defined between the two bases and between angle1 and angle2 with respect to the direction of the axis P1P2.

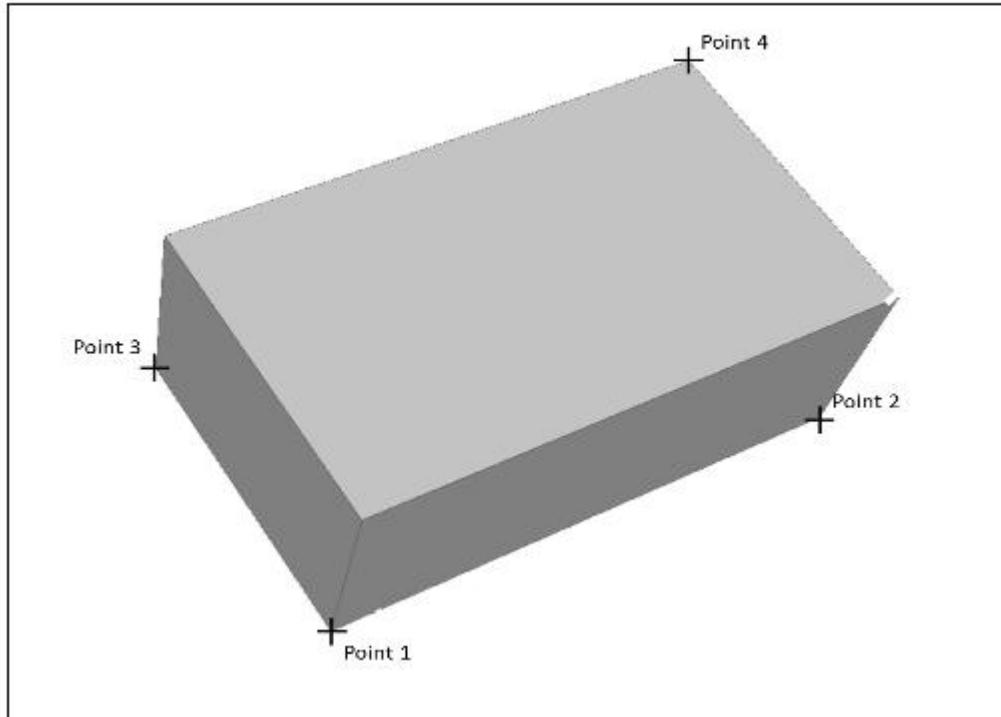
- Point 1 : center of the sphere (P1)
- Point 2 : point defining with P1 the sphere's axis to define the bases (P2)
- Point 3 : point defining with P1 and P2 the plane of angular truncation (P3)
- Diameter : diameter of the sphere
- Height 1 : height of the first sphere's base relatively to P1
- Height 2 : height of the second sphere's base relatively to P1
- Angle 1 : start angle of the sphere relatively to the plane of angular truncation
- Angle 2 : stop angle of the sphere relatively to the plane of angular truncation

Volumic box

Default volumic box

This shape is a closed box, i.e. with six faces.

- Point 1 : origin point for the box (P1)
- Point 2 : point defining with P1 the first of edge the box (P2)
- Point 3 : point defining with P1 the second edge of the box (P3)
- Point 4 : point defining with P1 the third edge of the box (P4)



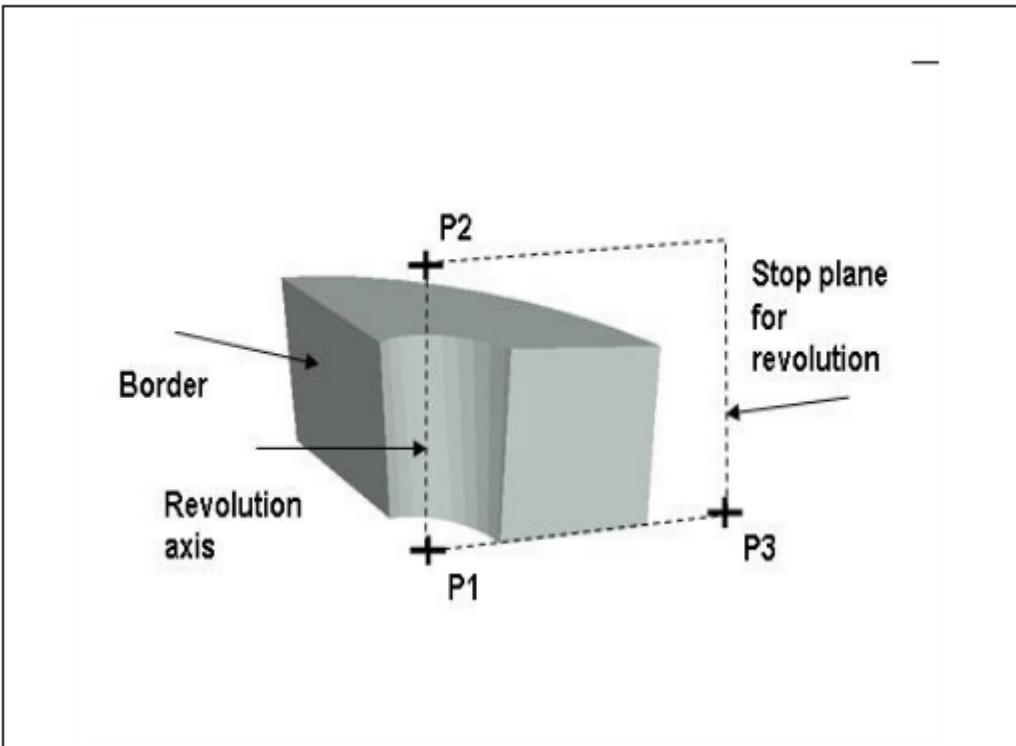
Definition

This shape is a closed box, i.e. with six faces.

- **Point 1 : origin point for the box (P1)**
- **Point 2 : point defining with P1 the first of edge the box (P2)**
- **Point 3 : point defining with P1 the second edge of the box (P3)**
- **Point 4 : point defining with P1 the third edge of the box (P4)**

Volumic revolved quadrangle

Default volumic revolved quadrangle



Definition

This shape is the revolution of a quadrangle around an axis.

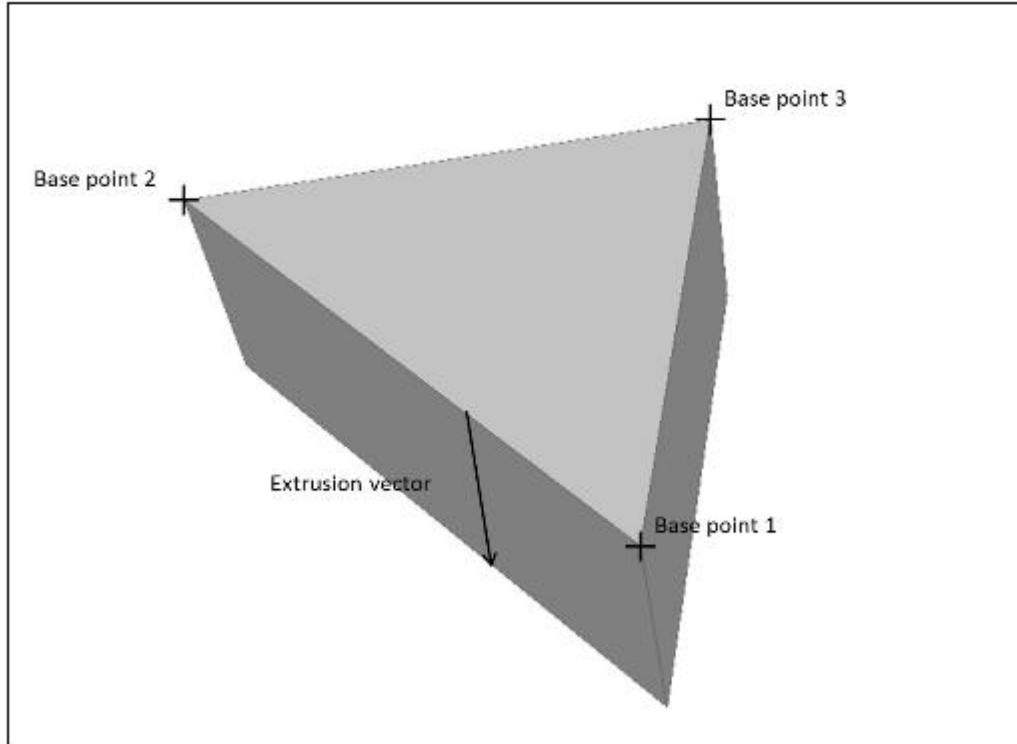
- **Base point 1** : first point of base quadrangle
- **Base point 2** : second point of base quadrangle
- **Base point 3** : third point of base quadrangle
- **Base point 4** : fourth point of base quadrangle
- **P1** : first point defining the axis of revolution
- **P2** : second point defining the axis of revolution
- **P3** : point defining with P1 and P2 the stop plane of the revolution. If P3 equal P2 then the angle of revolution is 90°. If P3 equal P1 or if P1, P2 and P3 are aligned, then the angle of revolution is 360°.

Volumic extruded triangle

Default volumic extruded triangle

This shape is an extruded triangle defined by a base triangle and an extrusion vector.

- Base point 1: first point of base triangle
- Base point 2: second point of base triangle
- Base point 3: third point of base triangle
- Extrusion vector point 1: first point of the extrusion vector
- Extrusion vector point 2: second point of the extrusion vector



Definition

This shape is an extruded triangle defined by a base triangle and an extrusion vector.

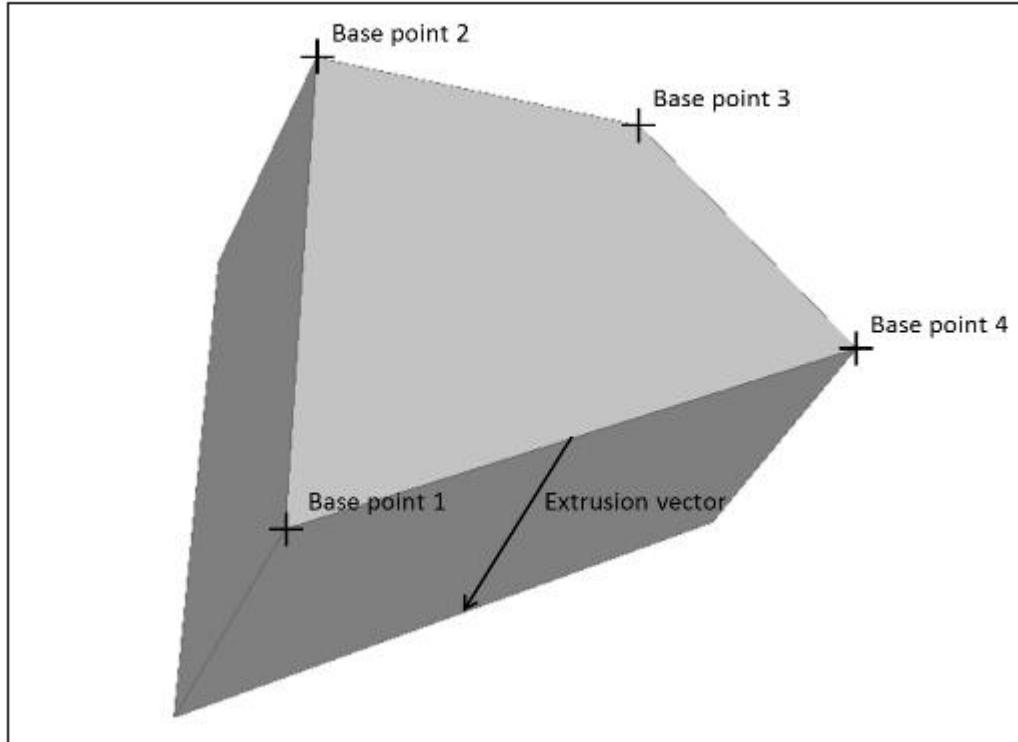
- **Base point 1: first point of base triangle**
- **Base point 2: second point of base triangle**
- **Base point 3: third point of base triangle**
- **Extrusion vector point 1: first point of the extrusion vector**
- **Extrusion vector point 2: second point of the extrusion vector**

Volumic extruded quadrangle

Default volumic extruded quadrangle

This shape is an extruded quadrangle defined by a base quadrangle and an extrusion vector.

- Base point 1 : first point of base quadrangle
- Base point 2 : second point of base quadrangle
- Base point 3 : third point of base quadrangle
- Base point 4 : fourth point of base quadrangle
- Extrusion vector point 1 : first point of the extrusion vector
- Extrusion vector point 2 : second point of the extrusion vector



Definition

This shape is an extruded quadrangle defined by a base quadrangle and an extrusion vector.

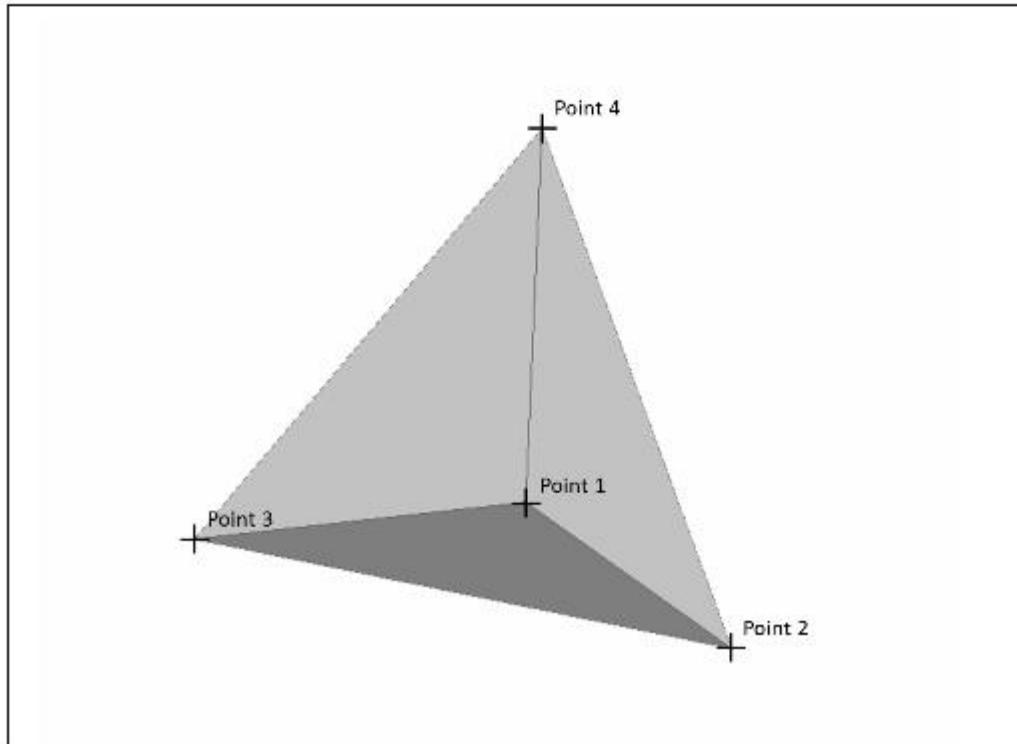
- **Base point 1** : first point of base quadrangle
- **Base point 2** : second point of base quadrangle
- **Base point 3** : third point of base quadrangle
- **Base point 4** : fourth point of base quadrangle
- **Extrusion vector point 1** : first point of the extrusion vector
- **Extrusion vector point 2** : second point of the extrusion vector

Volumic tetrahedron

Default volumic tetrahedron

This shape is a tetrahedron.

- Point 1: first point of the tetrahedron base.
- Point 2: second point of the tetrahedron base.
- Point 3: third point of the tetrahedron base.
- Point 4: summit of the tetrahedron.



Definition

This shape is a tetrahedron.

- **Point 1:** first point of the tetrahedron base.
- **Point 2:** second point of the tetrahedron base.
- **Point 3:** third point of the tetrahedron base.
- **Point 4:** summit of the tetrahedron.

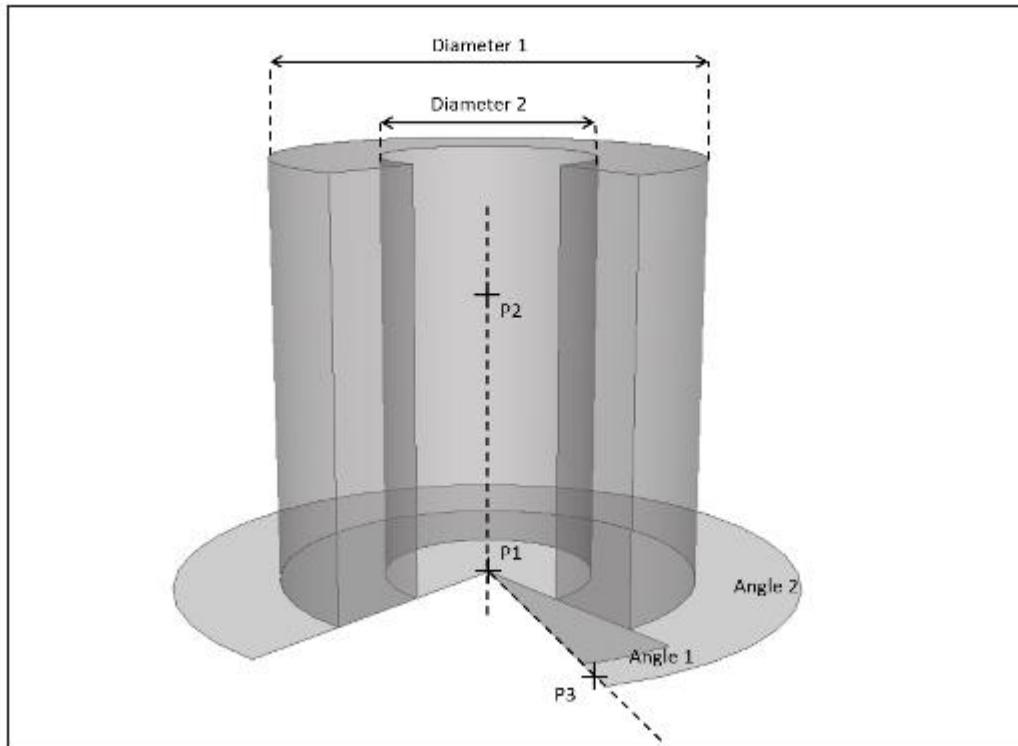
Volumic cylinder

Default volumic cylinder

This shape is a cylinder (*external face*) cut by another cylinder (*internal face*) and an angular cut.

- Point 1: center of the base disc (P1)

- Point 2: point defining with P1 the base disc normal.
- Point 3: point defining with P1 and P2 the plane of angular truncation.
- Diameter 1: diameter of the base disc's outer border.
- Diameter 2: diameter of the base disc's inner border.
- Angle 1: start angle of the cylinder relatively to the plane of angular truncation.
- Angle 2: stop angle of the cylinder relatively to the plane of angular truncation.
- Height: cylinder's height.



Definition

This shape is a cylinder (external face) cut by another cylinder (internal face) and an angular cut.

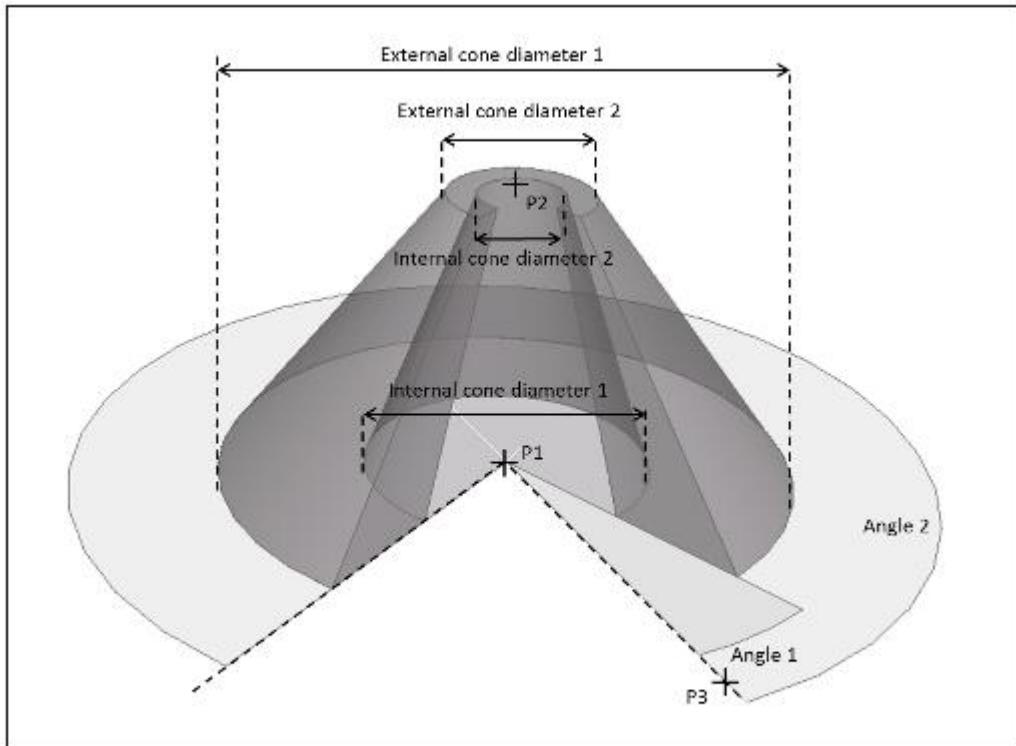
- Point 1: center of the base disc (P1)
- Point 2: point defining with P1 the base disc normal.
- Point 3: point defining with P1 and P2 the plane of angular truncation.
- Diameter 1: diameter of the base disc's outer border.
- Diameter 2: diameter of the base disc's inner border.
- Angle 1: start angle of the cylinder relatively to the plane of angular truncation.
- Angle 2: stop angle of the cylinder relatively to the plane of angular truncation.
- Height: cylinder's height.

Volumic cone

Default volumic cone

This shape is a cone (*external face*) cut by another cone (*internal face*), an angular cut, and with a truncated summit.

- Point 1: center of the cone base face.
- Point 2: point defining with P1 the cone axis and height. It is also the center of the top face.
- Point 3: point defining with P1 and P2 the plane of angular truncation.
- External cone diameter 1: diameter of the extern cone base face.
- External cone diameter 2; diameter of the extern cone top face.
- Internal cone diameter 1; diameter of the intern cone base face.
- Internal cone diameter 2: diameter of the intern cone top face.
- Angle 1; start angle of the cone relatively to the plane of angular truncation.
- Angle 2: stop angle of the cone relatively to the plane of angular truncation.



Definition

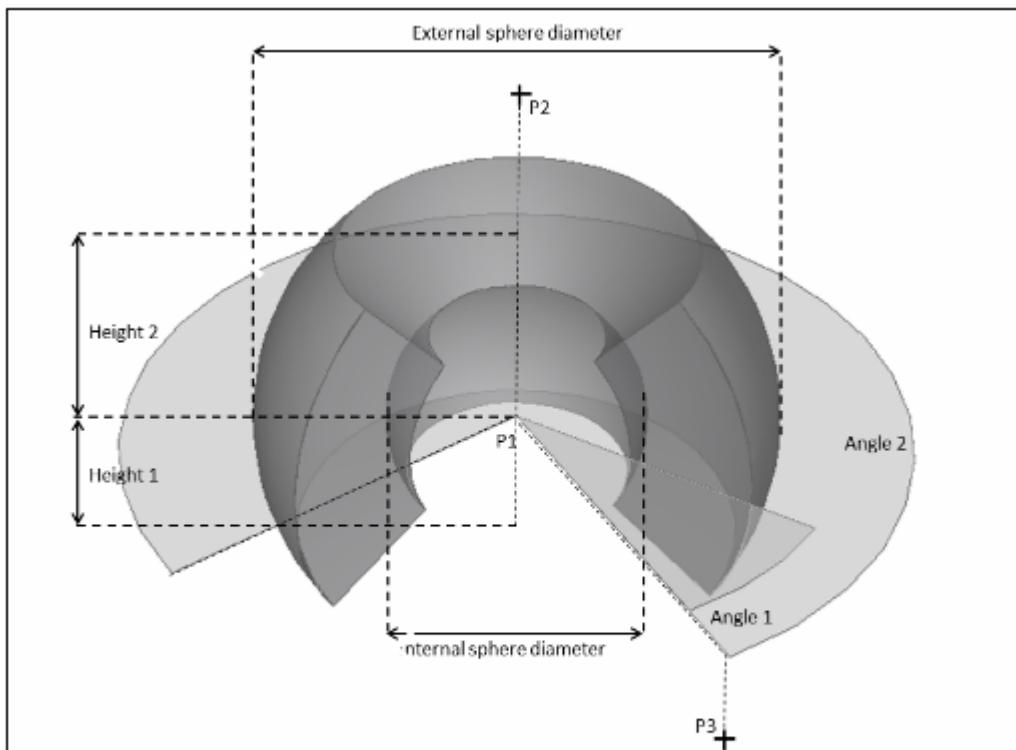
This shape is a cone (external face) cut by another cone (internal face), an angular cut, and with a truncated summit.

- Point 1: center of the cone base face.
- Point 2: point defining with P1 the cone axis and height. It is also the center of the top face.
- Point 3: point defining with P1 and P2 the plane of angular truncation.
- External cone diameter 1: diameter of the external cone base face.
- External cone diameter 2: diameter of the external cone top face.
- Internal cone diameter 1; diameter of the internal cone base face.
- Internal cone diameter 2: diameter of the internal cone top face.
- Angle 1; start angle of the cone relatively to the plane of angular truncation.
- Angle 2: stop angle of the cone relatively to the plane of angular truncation.

Volumic sphere

Default volumic sphere

- This shape is a truncated sphere (external face) cut by another sphere (internal face). The part of sphere is defined between the two bases and between angle1 and angle2 with respect to the direction of axis P1P2
- Point 1: center of the sphere. (P1)
- Point 2: point defining with P1 the sphere's axis to define the bases. (P2)
- Point 3: point defining with P1 and P2 the plane of angular truncation.
- External sphere diameter : diameter of the extern sphere.
- Internal sphere diameter : diameter of the intern sphere.
- Angle 1 : start angle of the sphere relatively to the plane of angular truncation.
- Angle 2 : stop angle of the sphere relatively to the plane of angular truncation.



Definition

This shape is a truncated sphere (external face) cut by another sphere (internal face). The part of sphere is defined between the two bases and between angle1 and angle2 with respect to the direction of axis P1P2

- **Point 1:** center of the sphere. (P1)
- **Point 2:** point defining with P1 the sphere's axis to define the bases. (P2)
- **Point 3:** point defining with P1 and P2 the plane of angular truncation.
- **External sphere diameter :** diameter of the extern sphere.
- **Internal sphere diameter :** diameter of the intern sphere.
- **Angle 1 :** start angle of the sphere relatively to the plane of angular truncation.
- **Angle 2 :** stop angle of the sphere relatively to the plane of angular truncation.

The transformations

A transformation is always relative and defines the position of the current object or shape with reference to the parent object. In SYSTEMA, a transformation is defined through translations and rotations (no homothety can be defined).

Translation

This transformation is defined as a translation along X or Y or Z axis of a given distance:

- Translation direction : choice among X, Y, Z for the translation direction
- Distance : distance of the translation

Rotation

This transformation is defined as a rotation around X or Y or Z axis of a given angle:

- Rotation axis: choice among X, Y, Z for the rotation axis
- Angle: angle of the rotation

Translations and rotations

This transformation is the result of an unlimited number of ordered translations and rotations defined by the user.

Each transformation is expressed relatively to the new axis system resulting of the previous transformation

Each translation is defined as a translation along X or Y or Z axis of a given distance

- Translation direction : choice among X,Y,Z
- Distance : value of the translation

Each rotation is defined as a rotation around X or Y or Z axis of a given angle

- Rotation axis : choice among X,Y,Z
- Angle : value of the rotation

Cosines matrix

This transformation is defined by a translation and a rotation of the local system axis expressed through an origin and three vectors. The matrix must be orthogonal direct.

- Origin : coordinates of the axis origin after the transformation
- X direction : coordinates of the axis X direction after the transformation
- Y direction : coordinates of the axis Y direction after the transformation
- Z direction : coordinates of the axis Z direction after the transformation

One point translation

This transformation is a translation defined by two points

- Point 1 : coordinates of the point before translation
- Transform 1 : coordinates of the point after translation

Three points transformation

This transformation is defined by six points. The first transformation defines a translation of the local system axis. The resulting transformation is a translation followed by a rotation and try as closed as possible to transform the

axis system defined by P1, P2 and P3 points in the expected axis system defined by Transform1, Transform2 and Transform3 points.

- Point 1 : coordinates of first point before the transformation
- Transform 1 : coordinates of first point after the transformation
- Point 2 : coordinates of second point before the transformation
- Transform 2 : coordinates of second point after the transformation
- Point 3 : coordinates of third point before the transformation
- Transform 3 : coordinates of third point after the transformation

X Y or Z translationRotation around axis defined by two points

This transformation is a translation rotation of a given angle along around X or Y or Zan axis defined by two points.of a given distance

- Translation direction : choice among X,Y,Z
- Point 1 : coordinates of a first axis point
- Point 2 : coordinates of a second axis point
- Distance Angle : value of the rotationvalue of the translation

Rotation around axis defined by one point and one direction

This transformation is a rotation of a given angle around an axis defined by a point and a direction.

- Point 1 : coordinates of a first axis point
- Direction : coordinates of the direction of the rotation axis
- Angle : value of the rotation

X Y or Z rotation

This transformation is a rotation around X or Y or Z axis of a given angle

- Rotation axis : choice among X,Y,Z
- Angle : value of the rotation

Position and quaternion

This transformation is defined by a translation followed by a rotation expressed by a position and a quaternion. The translation is defined by the transformation of the local axis origin into the new position. The quaternion is normalized by SYSTEMA.

- Position : coordinates of a point
- Quaternion : coordinates (W,X,Y,Z) of the quaternion

Direction transformation

This transformation is defined by a initial axis system transformed in a resulting axis system. The resulting axis system is computed from the two transformed vectors with respect of the first one and as close as possible to the second one.

- Direction 1 : first direction among X,Y,Z
- Transformed 1 : coordinates of the first direction after transformation
- Direction 2 : second direction among X,Y,Z
- Transformed 2 : expected coordinates of the second direction after transformation

Three rotations sequence

This transformation is also known as "Euler angles".

This transformation is defined by roll, pitch and lace angles with choice of the rotation sequence between "Z-Y-X Yaw-Pitch-Roll" and "X-Y-Z Roll-Pitch-Law".

- Rotation sequence : choice between "Z-Y-X Yaw-Pitch-Roll" and "X-Y-Z Roll-Pitch-Law".
- Roll (Phi) : roll angle
- Pitch (Theta) : pitch angle
- Lace (Psi) : lace angle

Trajectory technical annex

Trajectory technical annex

Overview

Introduction

This annex describes all technical information about trajectory.

In this chapter

This chapter covers the following topics:

- [Trajectory technical annex](#)
 - [Overview](#)
 - [Introduction](#)
 - [In this chapter](#)
 - [The celestial bodies](#)
 - [The celestial bodies constants](#)
 - [The values of celestial bodies constants](#)
 - [Legacy custom celestials](#)
 - [Ephemerides](#)
 - [Frames](#)
 - [Veis1950 / Gamma50](#)
 - [J2000 / EME2000](#)
 - [Planet inertial](#)
 - [Planet rotational](#)
 - [ICRF](#)
 - [Propagation](#)
 - [Time scale management:](#)
 - [Eclipses detection](#)

The celestial bodies

In SYSTEMA application, the main celestial bodies of the Solar System are modelled :

- The height planets (Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus, Neptune),
- The Sun,
- The Moon,
- Custom moons (Phobos_2024, Europa_2030, Callisto_2031, Ganymede_2032)

The celestial bodies constants

Several constants have been defined for each celestial body:

- the equatorial radius named Re
- the J2 term

- the planet attraction constant named μ
- the flatness coefficient named Flat (null for a spherical body)
- the semi major axis for a planet stationary orbit named a_geo
- the rotational velocity of the celestial body named vitRot
- the inclination of the rotational axe with regard to the normal of the planet orbit plan named i_axe
- the semi major axis of the celestial body (with regard to the central body) named sma
- the inclination i_0 of the planet orbit with regard to the ecliptic plan at the first order ; in fact the inclination of the planets are modeled as following : $i = i_0 + i_1 \cdot T + i_2 \cdot T^2 + i_3 \cdot T^3$ with T the number of centuries since 1900,
- the time reference used for ephemeris calculation (source: <http://ssd.jpl.nasa.gov/?horizons>)

The values of celestial bodies constants

Body	Equatorial / Mean radius (km)	J2	μ (km³.s⁻²)	Flattening	a_geo (km)
Sun	696000.0 / -	0.	132712440042	0.	/
Mercury	2439.7 / -	0.000060	22031.78	0.	242758.0
Venus	6052.0 / 6051.8	0.000027	324858.592	0.0	1536473.0
Earth	6378.144 / 6371.0	0.00108265	398600.435	0.00335	42164.17
Moon	1738.0 / 1737.4	0.002027	4902.8	0.0012	/
Mars	3394.2 / 3389.5	0.001964	42828.37	0.00589	20427.505
Jupiter	71492.0 / 69911.0	0.014736	126712764.8	0.06487	160043.723
Saturn	60268.0 / 58232.0	0.016298	37940585.2	0.09796	239519.553
Uranus	25559.0 / 25362.0	0.00334343	5794548.6	0.02293	82715.4006
Neptune	25269.0 / 24622.0	0.003411	6836527.1	0.01708	90079.8897
Phobos_2024*	11.105 / -	0.10053	0.0007254	0.	/
Ganymede_2032*	2631.2 / -	0.000127	9887.83	0.	12820.5384
Callisto_2031*	2403.0 / -	0.0000327	7179.29	0.	/
Europa_2030*	1565.0 / -	0.0004355	3202.72	0.	/
Body	vitRot (deg/s)	i_axe (deg)	sma (astro unit)	i0 (deg)	Time ref (date)

Sun	/	/	/	/	/
Mercury	0.0000711	179.99	0.387	7.002881	/
Venus	-0.0000171	177.36	0.7233	3.393631	/
Earth	0.00417807	23.44497222	1	0	/
Moon	/	/	/	/	/
Mars	0.0040613	25.2	1.524	1.850333333333 3333	/
Jupiter	0.010075567	3.13	5.203	1.308736	/
Saturn	0.00952381	26.72	9.530	2.492519	/
Uranus	0.005802708	97.86	19.185	0.772464	/
Neptune	0.005555556	29.56	30.087	1.779242	/
Phobos_2024*	0.01306165	0.	0.0000627	1.082	01/07/2024
Ganymede_2032*	0.010075567	0.	0,007155	0.195	10/12/2032
Callisto_2031*	0.000249665	0.	0,0012587	0.281	01/04/2031
Europa_2030*	0.001173111	0.	0,00448589	0.464	01/10/2030

(*) Celestial defined with ephemeris at a reference date (_XXXX). The validity of its trajectory can only be ensured close to this reference date.

Legacy custom celestials

The legacy custom celestials introduced in Systema-4.8.3 (Earth_2020, Earth_2030, Earth_2040 and Mars_2030) were made obsolete by the 4.9.0 updates.

The import of trajectories containing these celestials will lead to a conversion to built-in celestials (ex: Earth_2020 → Earth, Mars_2030 → Mars).

Ephemerides

Arc's orbit are computated thanks to the Orekit library. This library relies on some external data for physical models. Typical data are the Earth Orientation Parameters and the leap seconds history, both being provided by the IERS or the planetary ephemerides provided by JPL or IMCCE.

Default configuration of Orekit in Systema uses JPL DE 430 ephemerides from 1550 to 2650, IERS Earth orientation parameters from 1973 (both IAU-1980 and IAU-2000), UTC-TAI history from 1972, Marshall Solar Activity Future Estimation from 1999, the Eigen 06S gravity field and the FES 2004 ocean tides model.

Frames

Veis1950 / Gamma50

The default Earth inertial reference frame is Veis1950 as defined in Orekit (<https://www.orekit.org>). This frame is similar to the Gamma50 frame definition used by Systema prior to 4.9.

An Earth inertial frame is geocentric and is linked to stars positions at a particular date. It is defined by :

- the origin O which is the Earth barycenter
- the Z axis which is the mean pole axis at the date t (with positive direction toward North)
- the X axis. The intersection of the orbit plan and the equatorial plan at the date t makes the vernal axis. This axis intersect the equator in 2 nodes, the vernal point y_t is the ascending one. The X axis is defined by the line $O - y_t$.
- the Y axis complete the frame in order to have a direct frame.

The Gamma50 frame is the inertial frame for the date $t = 01/01/1950$ at 00h00m00s.

J2000 / EME2000

The J2000 frame (which is also known as EME2000) is defined using the mean equinox at epoch J2000.0, i.e. 2000-01-01T12:00:00 in Terrestrial Time. This frame is inertial and Earth centered.

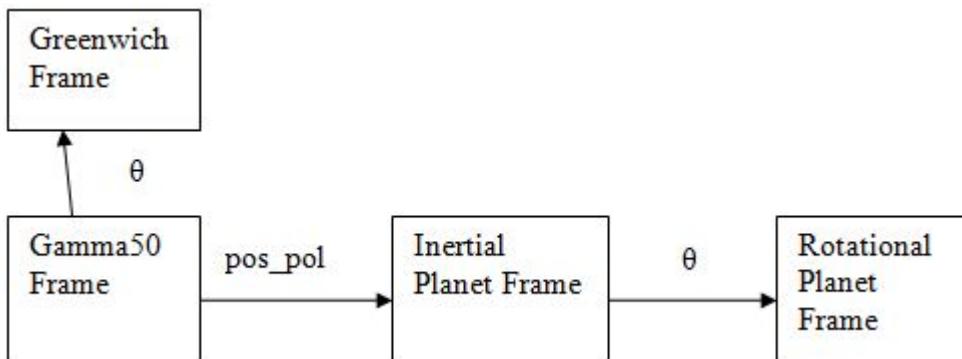
Planet inertial

All celestial bodies are linked to their own body-centered inertial frame, just as the Earth is linked to Veis1950 and J2000. Through Orekit, Systema provides inertial body-centered frames for all the main solar system celestial bodies. The orientations of these frames are compliant with IAU poles and prime meridians definitions. For the Earth, the inertial frame is Veis1950.

Planet rotational

For all the main solar system celestial bodies, Systema also provides thanks to Orekit rotational body-centered frames for all the main solar system celestial bodies.

The rotational frame for the Earth is the Greenwich frame. It is obtained from the inertial one by a rotation θ around pole axis. For the other planets, the rotational frame is obtained by a rotation θ around pole axis from its own inertial frame as described in the following figure.



ICRF

In addition to body-centered frames, Systema supports the International Celestial Reference Frame. The ICRF axes are defined as the inertial axes associated with a general relativity frame centered at the solar system barycenter.

Propagation

Systema propagation model is a custom propagator based on the Orekit Keplerian analytic propagator (see <https://www.orekit.org/> for more details). To be retro-compatible with Systema 4.8, the custom model in Systema 4.9 considers the J2 potential zonal coefficient.

- ⓘ For information, before Systema 4.9, here is the description of how the propagators were defined :

Default propagator (All arcs except geosynchronous) (Deprecated till Systema 4.9.0)

Only the perturbations due to the J2 terms are implemented. Consequently, only the argument of periapsis, the ascending node longitude and the anomaly are perturbed. The applied perturbations are

defined by these equations (with μ_p planet attraction constant, R_p planet radius and J_2 J2 terms of the planet):

$$\frac{d\Omega}{dt} = -\frac{3}{2}\sqrt{\frac{\mu_p}{a^3}} \left(\frac{R_p}{a}\right)^2 J_2 \cdot \frac{\cos(i)}{(1-e^2)^2}$$

$$\frac{d\omega}{dt} = \frac{3}{4}\sqrt{\frac{\mu_p}{a^3}} \left(\frac{R_p}{a}\right)^2 J_2 \cdot \frac{4-5\sin^2(i)}{(1-e^2)^2}$$

$$\frac{d\omega}{dt} = -\frac{3}{4}\sqrt{\frac{\mu_p}{a^3}} \left(\frac{R_p}{a}\right)^2 J_2 \cdot \frac{1-5\cos^2(i)}{(1-e^2)^2}$$

equivalent to

$$\frac{dM}{dt} = \sqrt{\frac{\mu_p}{a^3}} + \frac{3}{4}\sqrt{\frac{\mu_p}{a^3}} \left(\frac{R_p}{a}\right)^2 J_2 \cdot \frac{2-3\sin^2(i)}{(1-e^2)^2}$$

$$\frac{dM}{dt} = \sqrt{\frac{\mu_p}{a^3}} + \frac{3}{4}\sqrt{\frac{\mu_p}{a^3}} \left(\frac{R_p}{a}\right)^2 J_2 \cdot \frac{3\cos^2(i)-1}{(1-e^2)^2}$$

equivalent to

One of important parameters for the propagator is the elapsed time: in fact, if values of variation ($\frac{d\Omega}{dt}$, $\frac{d\omega}{dt}$, $\frac{dM}{dt}$) are different between 2 propagators, the position will differ more and more with the time. For arcs of type "time position velocity", the elapsed time is the time between the current time and the closest defined point. For other arcs, it is the time between the current time and the reference time.

Geosynchronous arc propagator (Deprecated till Systema 4.9.0)

The geosynchronous arcs are particular; in fact, the propagator on this type of orbits make drift a lot the orbit but users want satellites relatively fixed from the planet so the position is calculated by another way : velocity is function of the planet rotational velocity (V_{rot}) and satellite longitude (lon) leads to the position in the rotating frame (a is the half major axis) as described in the following equations. Then, position and velocity are transformed in the planet inertial frame.

$$XYZ = [a \cdot \cos(lon); a \cdot \sin(lon); 0]$$

$$V = [(V_{rot} \cdot a) \sin(lon); (V_{rot} \cdot a) \cos(lon); 0]$$

Notice that in Systema, half major axis for geosynchronous orbit is defined as a constant for each of the height planets.

Time scale management:

The time scale reference used by Orekit in Systema is Universal Time Coordinate.

UTC is mainly related to TAI, but some step adjustments are introduced from time to time to keep take into account Earth rotation irregularities and to prevent the *legal* time from drifting with respect to day and night. These adjustments require introduction of leap seconds, which means some days are not 86400 seconds long (see https://en.wikipedia.org/wiki/Leap_second)

Announced leap seconds to date:

Year	Jun 30	Dec 31
1972	+1	+1
1973	0	+1
1974	0	+1
1975	0	+1
1976	0	+1
1977	0	+1
1978	0	+1
1979	0	+1
1980	0	0
1981	+1	0
1982	+1	0
1983	+1	0
1984	0	0
1985	+1	0
1986	0	0
1987	0	+1
1988	0	0
1989	0	+1
1990	0	+1

Year	Jun 30	Dec 31
1991	0	0
1992	+1	0
1993	+1	0
1994	+1	0
1995	0	+1
1996	0	0
1997	+1	0
1998	0	+1
1999	0	0
2000	0	0
2001	0	0
2002	0	0
2003	0	0
2004	0	0
2005	0	+1
2006	0	0
2007	0	0
2008	0	+1
2009	0	0
2010	0	0
2011	0	0
2012	+1	0

Year	Jun 30	Dec 31
2013	0	0
2014	0	0
2015	+1	0
2016	0	+1
2017	0	0
2018	0	0
2019	0	0
2020	0	0
Total	11	16

Eclipses detection

Eclipses detection (entering and leaving penumbra and umbra) is done by Orekit. The detection considers the flattening of the occulting bodies.

Because of this, with the use of ephemerides files, and with the integration of the leap seconds from the time scale management, eclipses event dates and durations in Systema 4.9 can be slightly different from Systema 4.8.

Kinematics technical annex

Kinematics technical annex

Overview

Introduction

This annex describes all technical information about the types of kinematic bodies and kinematic laws.

In this chapter

This chapter covers the following topics:

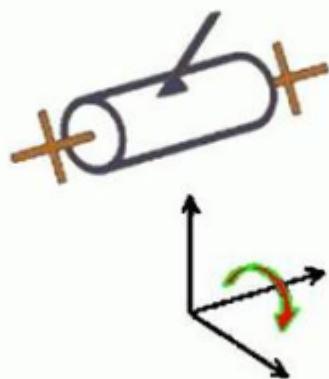
- Kinematics technical annex
 - Overview
 - Introduction
 - In this chapter
 - The kinematic bodies
 - Pivot connection
 - Slot ball joint
 - Ball joint
 - The kinematic laws
 - One pointing vector law
 - Ecliptic north
 - Orbit angular momentum
 - Orbit planet reference
 - Orbit planet reference north direction
 - Orbit planet reference vernal direction
 - Orbit velocity vector
 - Planet
 - Sun
 - 2 pointing vectors law
 - Linear attitude laws
 - Linear attitude
 - 3 attitudes laws
 - Rotating laws
 - Regular spin around axis
 - Fast spin around axis
 - Transformation defined in a file
 - Another satellite trajectory
 - Fixed attitude law
 - Fixed position
 - Fixed last position
 - X direction of Gamma50
 - Y direction of Gamma50
 - Z direction of Gamma50
 - X direction of J2000
 - Y direction of J2000
 - Z direction of J2000
 - X direction of ICRF

- Y direction of ICRF
- Z direction of ICRF
- Solve the pointing law
 - Pivot connection and one pointing vector law
 - Slot ball joint and one pointing vector law
 - Ball joint and one pointing vector law
 - Pivot connection and linear attitude law
 - Slot ball joint and linear attitude law
 - Ball joint and linear attitude law

The kinematic bodies

Three types of kinematic bodies are available in Systema.

Pivot connection



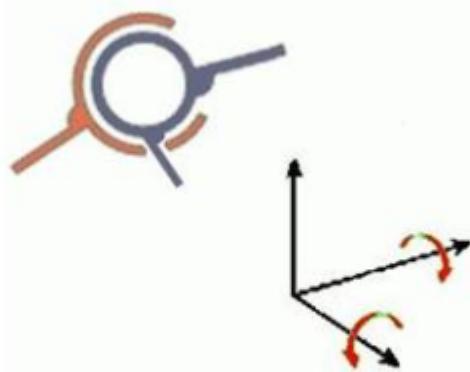
DEFINITION

This kinematic body has one degree of freedom in rotation.

- Axis 1 : choice of the rotation axis among X,Y,Z. A restriction can be set between a minimum angle and a maximum angle.

Compute the rotation matrix from the requested pointing direction (according to the pointing law) and the actual pointing vector with respect of the rotation axis and restriction. The result is as close as possible to the requirement.

Slot ball joint



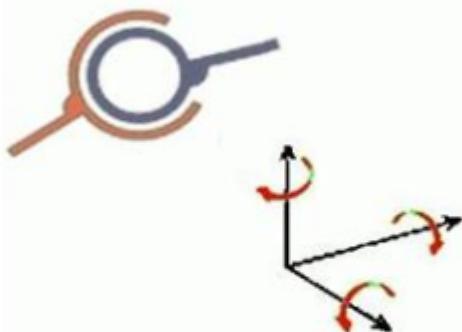
DEFINITION

This kinematic body has two degrees of freedom in rotation.

- First rotation : choice of the rotation axis among X,Y,Z. A restriction can be set between a minimum angle and a maximum angle.
- Second rotation : choice of the rotation axis among X,Y,Z. Must be distinct from the first rotation axis. A restriction can be set between a minimum angle and a maximum angle.

Compute the rotation matrix from the requested pointing direction (according to the pointing law) and the actual pointing vector with respect of the rotation axis order and restrictions. The result is as close as possible to the requirement.

Ball joint



DEFINITION

This kinematic body has three degrees of freedom in rotation.

- First rotation : choice of the rotation axis among X,Y,Z. A restriction can be set between a minimum angle and a maximum angle.
- Second rotation : choice of the rotation axis among X,Y,Z. Must be distinct from the first rotation axis. A restriction can be set between a minimum angle and a maximum angle.
- Third rotation : choice of the rotation axis among X,Y,Z. Must be distinct from the first and second rotation axis. A restriction can be set between a minimum angle and a maximum angle.

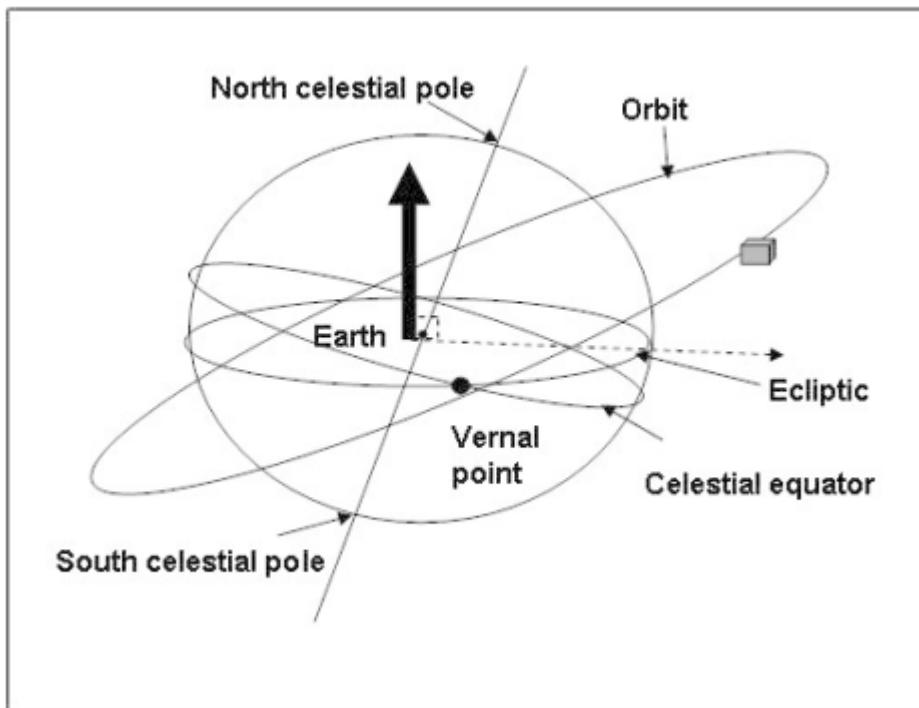
Compute the rotation matrix from the requested pointing direction (according to the pointing law) and the actual pointing vector with respect of the rotation axis order and restrictions. The result is as close as possible to the requirement.

The kinematic laws

A transformation is always relative and defines the position of the current object or shape with reference to the parent object. In SYSTEMA, a transformation is defined through translations and rotations (no homothety can be defined).

One pointing vector law

Ecliptic north



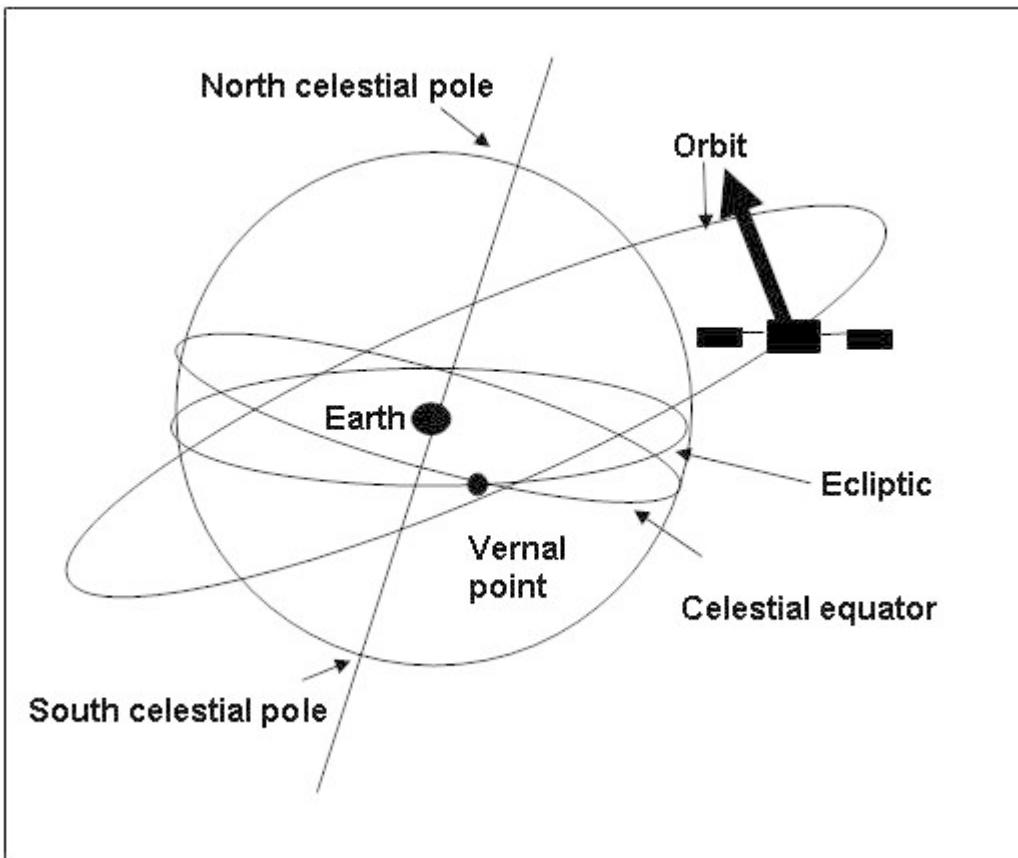
DEFINITION

This law is defined with a pointing direction and the associated pointing vector.

The pointing vector is an axis of the kinematic body and the pointing direction is defined as the ecliptic north direction.

- Pointing vector : axis of the kinematic body defined through its coordinates.

Orbit angular momentum



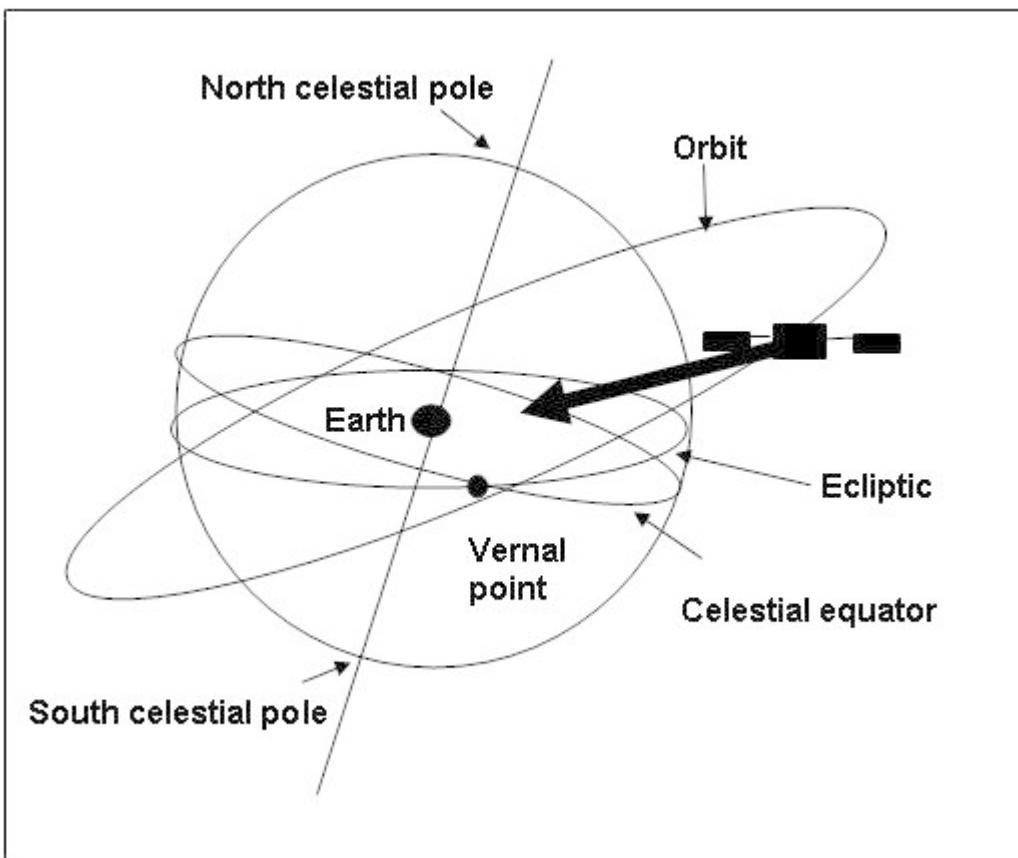
DEFINITION

This law is defined with a pointing direction and the associated pointing vector.

The pointing vector is an axis of the kinematic body and the pointing direction is defined as the orbit angular momentum.

- Pointing vector : axis of the kinematic body defined through its coordinates.

Orbit planet reference



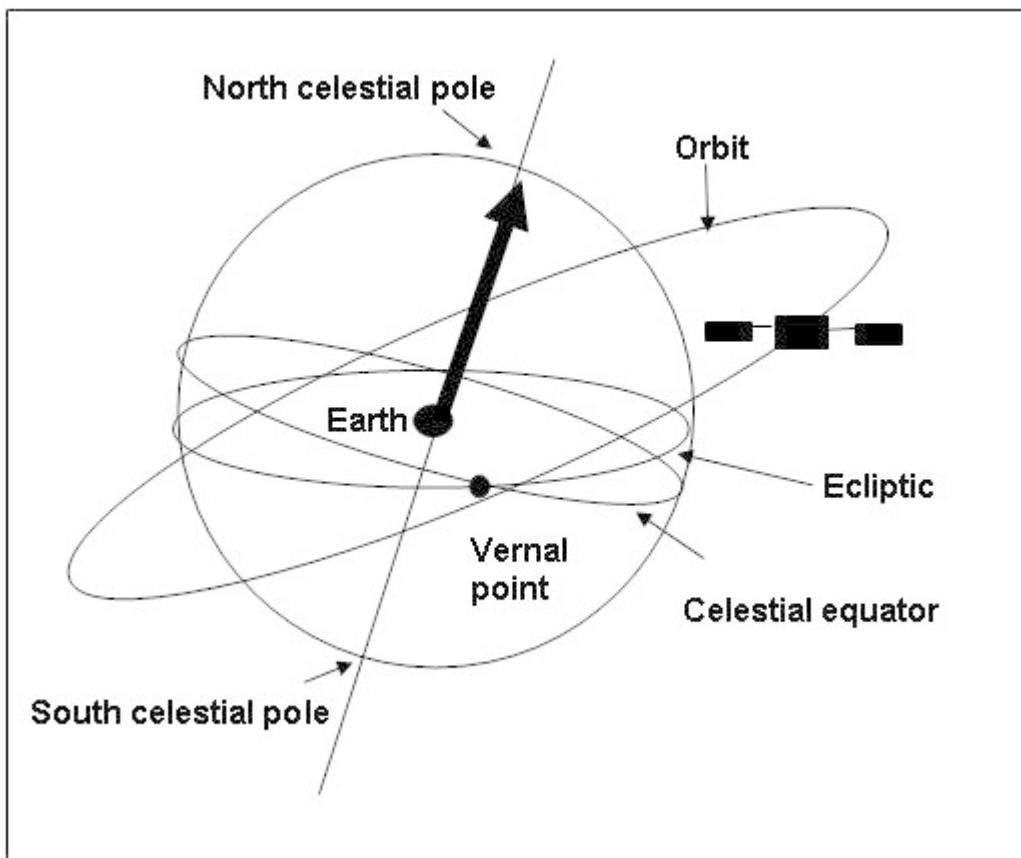
DEFINITION

This law is defined with a pointing direction and the associated pointing vector.

The pointing vector is an axis of the kinematic body and the pointing direction is defined as pointing to the planet center or opposite to the planet center.

- Pointing vector : axis of the kinematic body defined through its coordinates.
- OrbitCenter or anti-OrbitCenter : pointing to the planet center or opposite to the planet center.

Orbit planet reference north direction

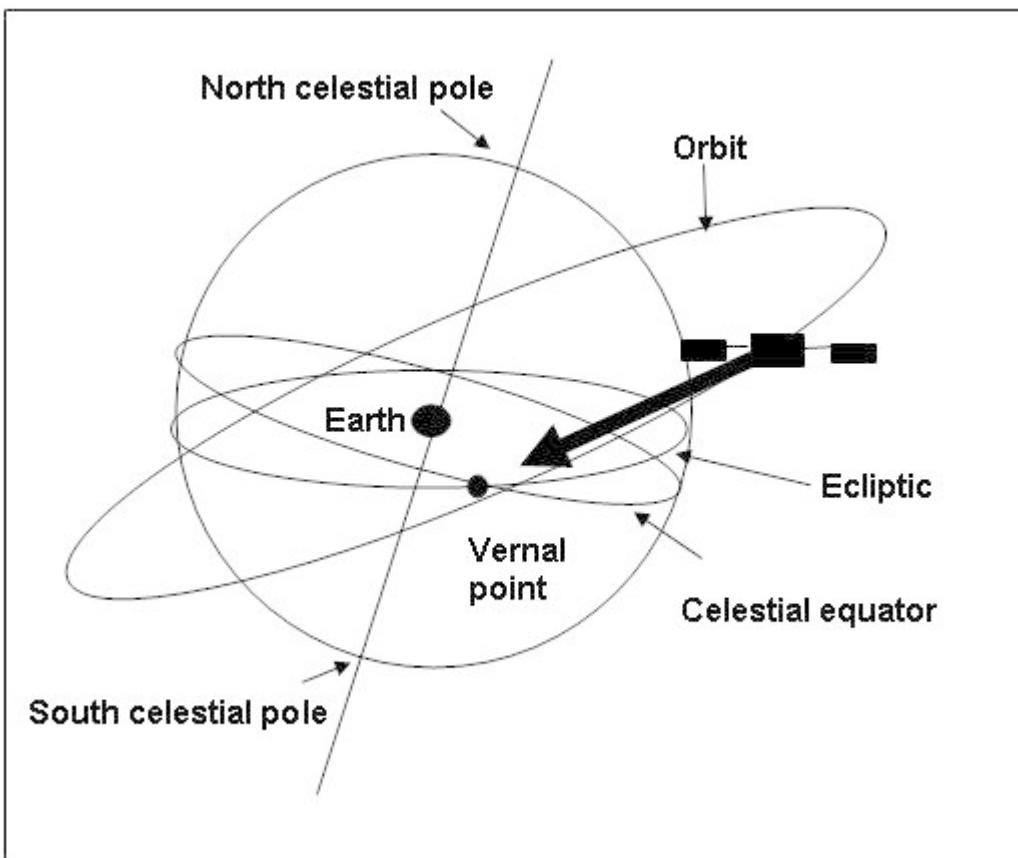
**DEFINITION**

This law is defined with a pointing direction and the associated pointing vector.

The pointing vector is an axis of the kinematic body and the pointing direction is defined as the north direction of the orbit planet reference.

- Pointing vector : axis of the kinematic body defined through its coordinates.

Orbit planet reference vernal direction



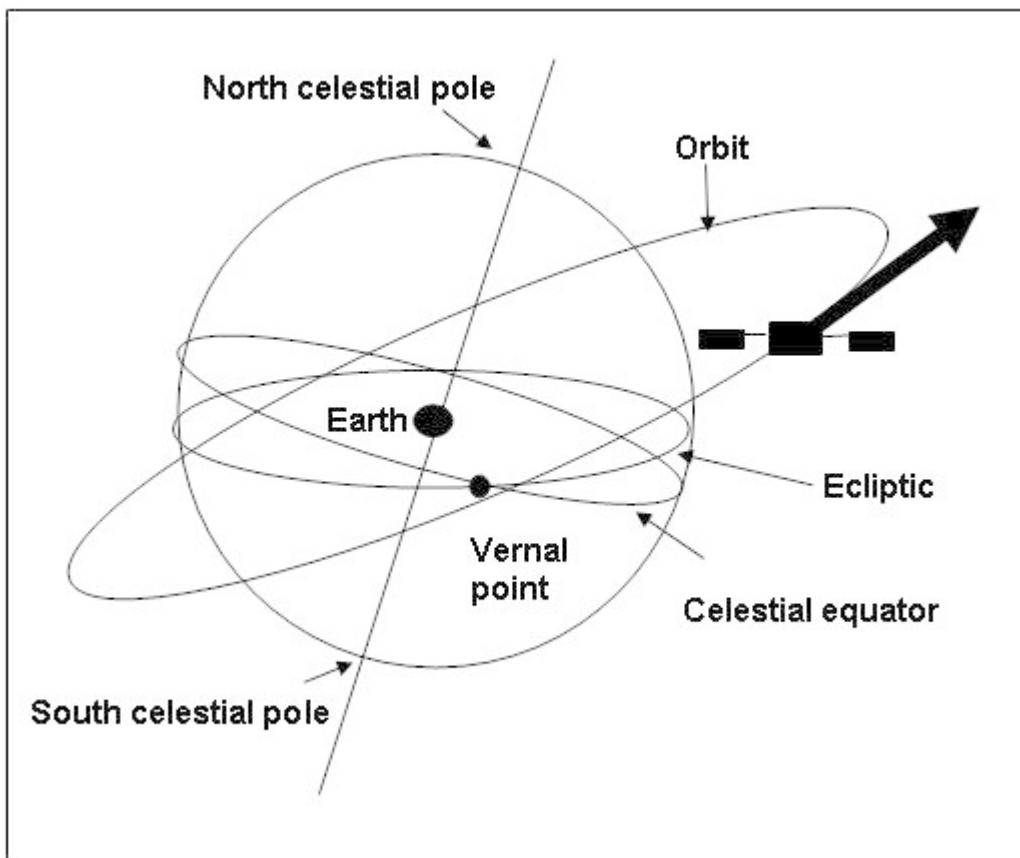
DEFINITION

This law is defined with a pointing direction and the associated pointing vector.

The pointing vector is an axis of the kinematic body and the pointing direction is defined as the vernal direction of the orbit planet reference.

- Pointing vector : axis of the kinematic body defined through its coordinates.

Orbit velocity vector

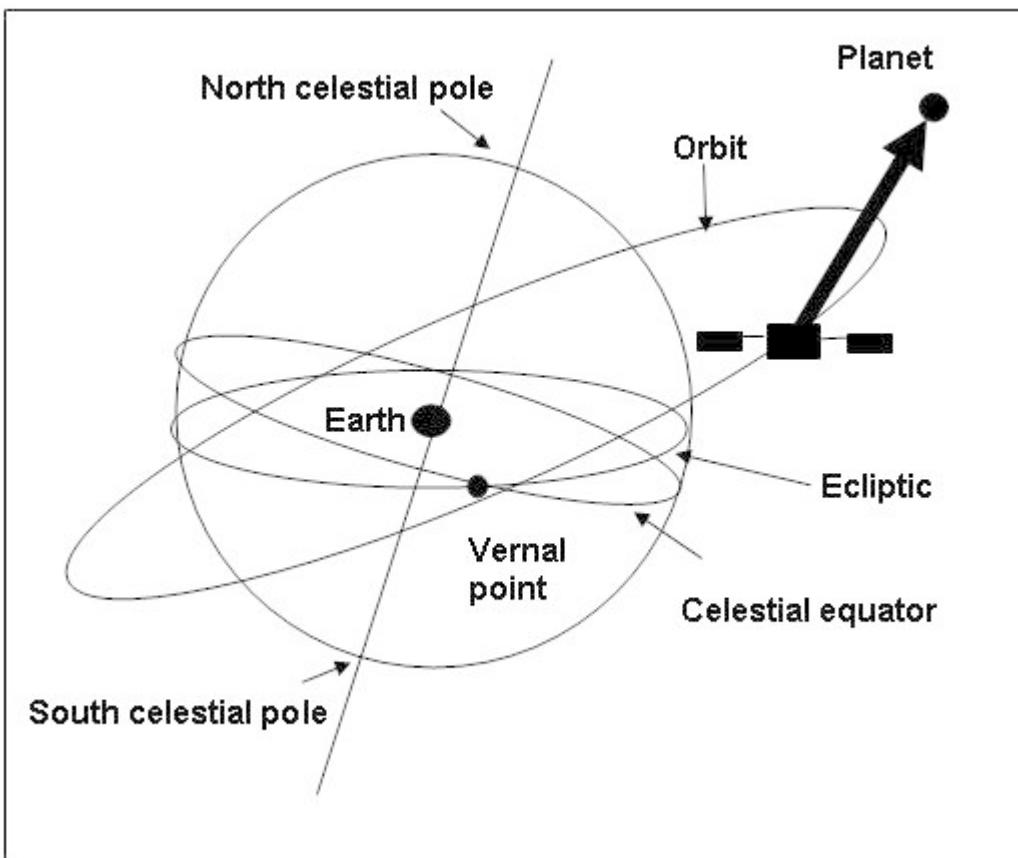


DEFINITION

This law is defined with a pointing direction and the associated pointing vector.
The pointing vector is an axis of the kinematic body and the pointing direction is defined as the orbit velocity vector.

- Pointing vector : axis of the kinematic body defined through its coordinates.

Planet



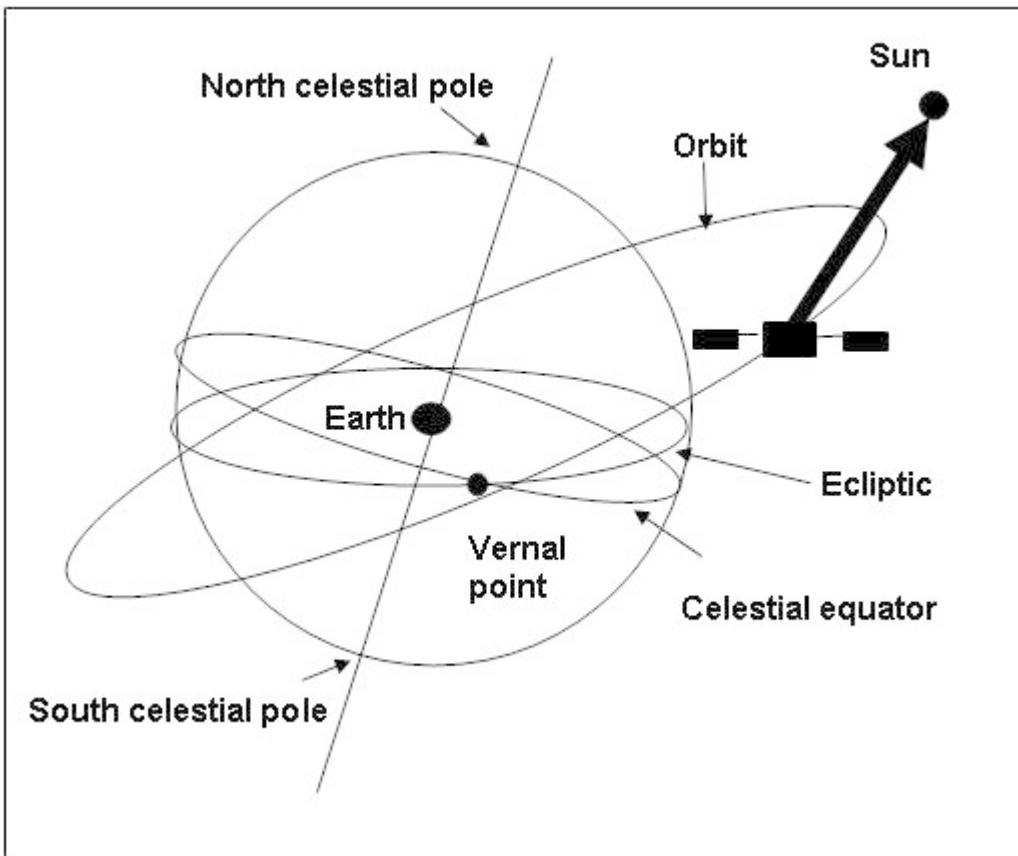
DEFINITION

This law is defined with a pointing direction and the associated pointing vector.

The pointing vector is an axis of the kinematic body and the pointing direction is defined as a pointed planet.

- Pointing vector : axis of the kinematic body defined through its coordinates.
- Pointed planet : choice among a list of planets.

Sun



DEFINITION

This law is defined with a pointing direction and the associated pointing vector.

The pointing vector is an axis of the kinematic body and the pointing direction is defined as the Sun.

- Pointing vector : axis of the kinematic body defined through its coordinates.

2 pointing vectors law

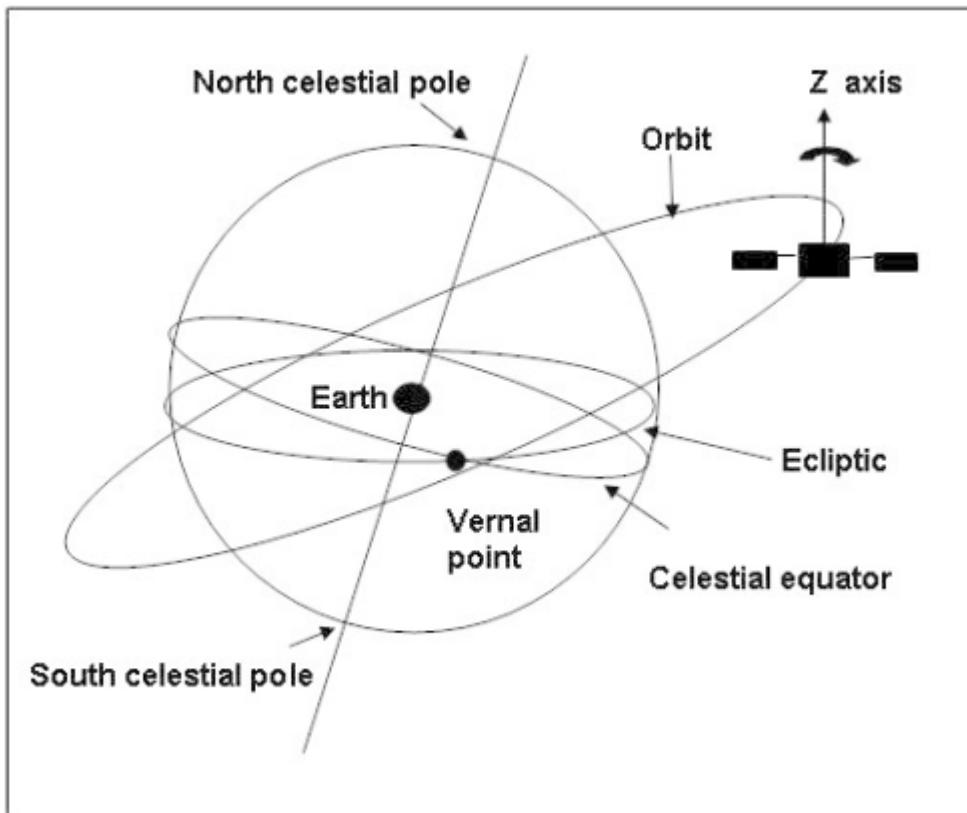
This law is defined as the combination of two laws with a pointing direction and the associated pointing vector for each.

The first law is solved with a free connection, and the second law with a rotation around the first vector.

- Each pointing vector is an axis of the kinematic body and the pointing direction is defined through the choice of a law among the list of available laws.
- Pointing vector : axis of the kinematic body defined through its coordinates

Linear attitude laws

Linear attitude

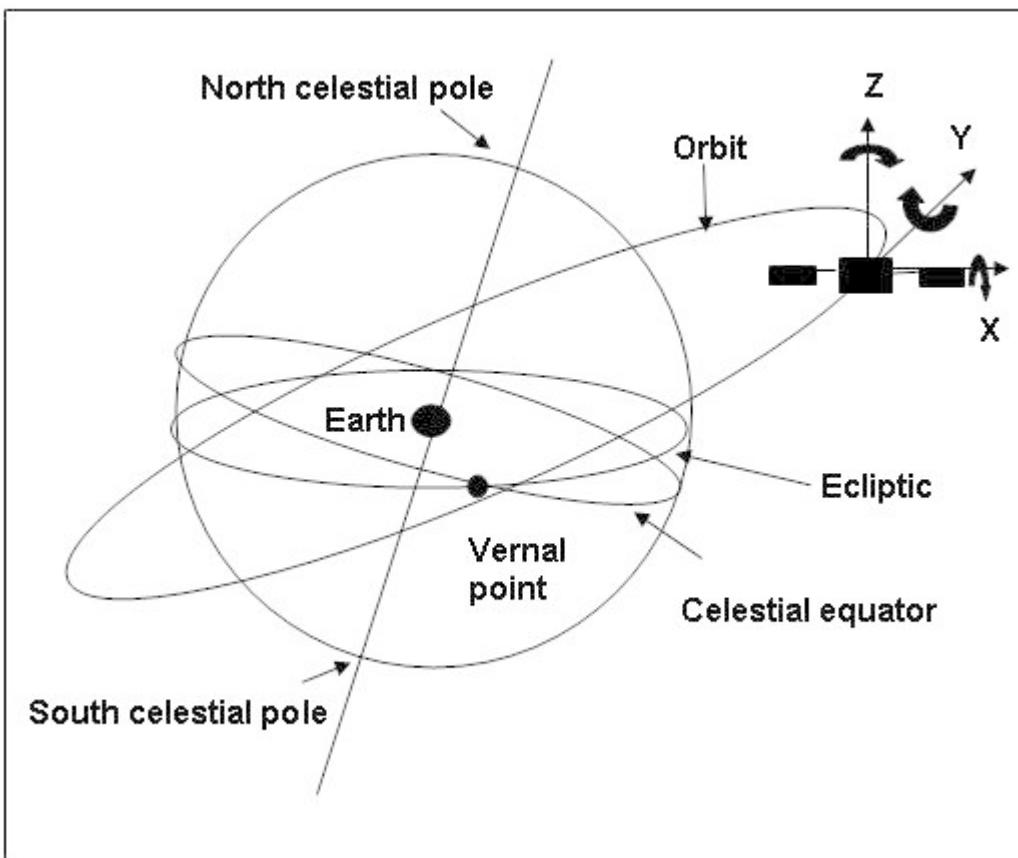


DEFINITION

This law is defined as an attitude relatively to a specific axis of the kinematic body. The attitude angle around the specific axis is linearly interpolated according to the selected variable (time, true anomaly or longitude) so that 'angle = A1 * factor + A0'.

- Axis : choice of the axis of the kinematic body among X,Y,Z.
- Unit : choice of the unit among time, true anomaly and longitude.
- A1 : first coefficient of the interpolation in °/unit.
- A0 : second coefficient of the interpolation in °.

3 attitudes laws



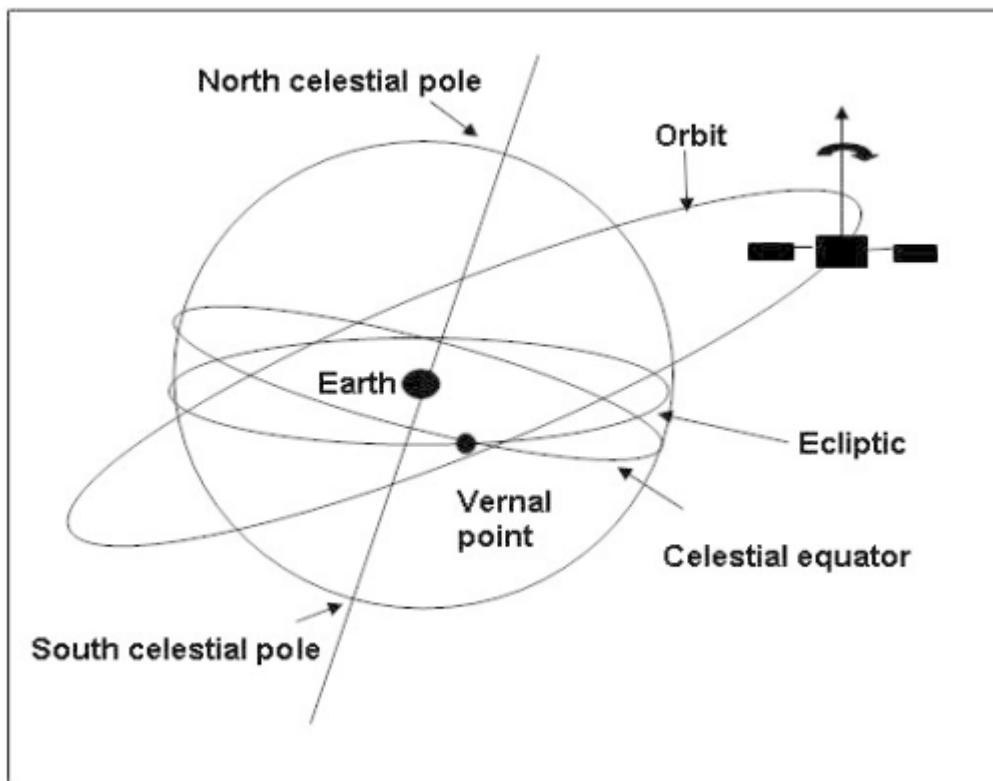
DEFINITION

This law is defined as the combination of three linear attitude laws. Each linear attitude law is defined as an attitude relatively to a specific axis of the kinematic body. The attitude angle around the specific axis is linearly interpolated according to the selected variable (time, true anomaly or longitude) so that 'angle = A1 * factor + A0'.

- Axis : choice of the axis of the kinematic body among X,Y,Z.
- Unit : choice of the unit among time, true anomaly and longitude.
- A1 : first coefficient of the interpolation in °/unit.
- A0 : second coefficient of the interpolation in °.

Rotating laws

Regular spin around axis

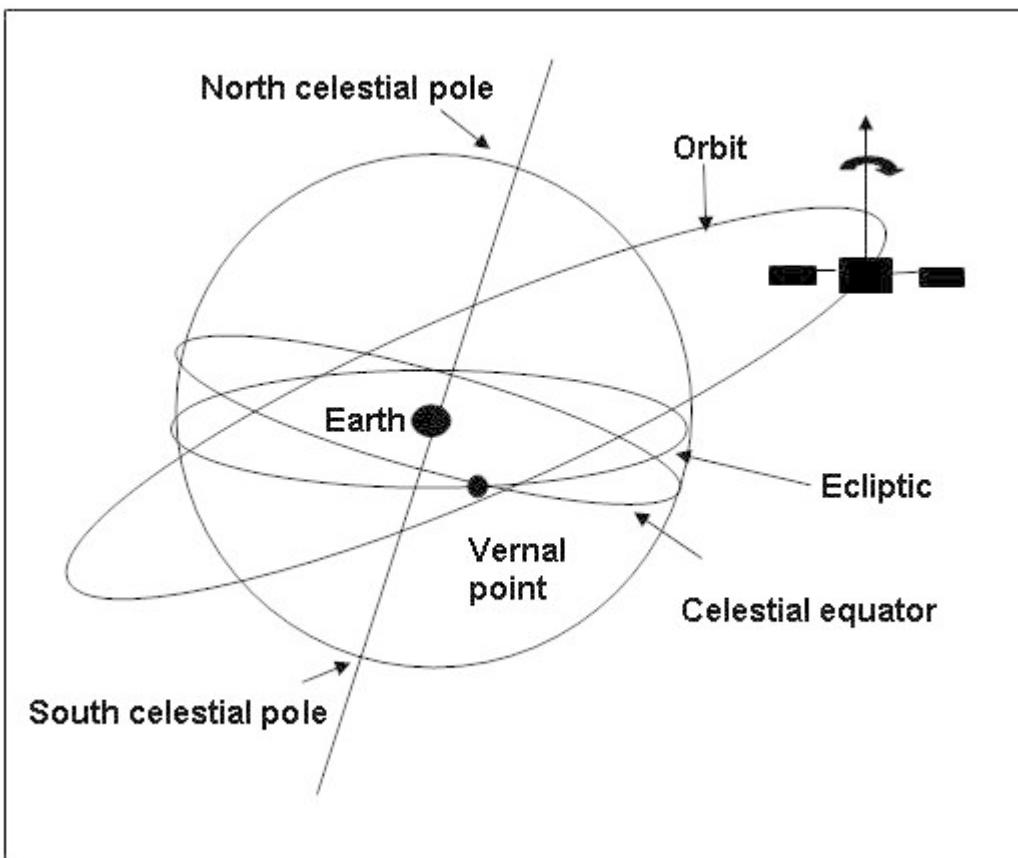


DEFINITION

This law is defined as a rotation given in revolutions per hour around a specific axis of the kinematic body.

- Rotation axis : rotation axis of the kinematic body defined through its coordinates.
- Rotation speed : speed of the rotation in revolutions per hour.

Fast spin around axis

**DEFINITION**

This law is defined as a rotation around a specific axis of the kinematic body with a given number of computations.

- **Rotation axis :** rotation axis of the kinematic body defined through its coordinates.
- **Number of computation (N) :** the step angle for rotational matrix computation is $360^\circ/N$.

Transformation defined in a file

This law is defined by an input file containing informations on the transformation to apply on the kinematic body at each time entry.

Please refer to [Kinematics management](#) for the list of supported files.

Another satellite trajectory

This law is defined by a pointing vector and a trajectory file (systrj). The referenced trajectory must be opened to be taken into account.

Fixed attitude law

Fixed position

This law is defined as a fixed given position.

Fixed last position

This law is defined as a fixed position which is the last position of the previous phase.

X direction of Gamma50

The X direction of the Gamma50 inertial frame on 1 January 1950.

Y direction of Gamma50

The Y direction of the Gamma50 inertial frame on 1 January 1950.

Z direction of Gamma50

The Z direction of the Gamma50 inertial frame on 1 January 1950.

X direction of J2000

The X direction of the J2000 inertial frame on 1 January 2000.

Y direction of J2000

The Y direction of the J2000 inertial frame on 1 January 2000.

Z direction of J2000

The Z direction of the J2000 inertial frame on 1 January 2000.

X direction of ICRF

The X direction of the ICRF (International Celestial Reference Frame. The ICRF axes are defined as the inertial axes associated with a general relativity frame centered at the solar system barycenter)

Y direction of ICRF

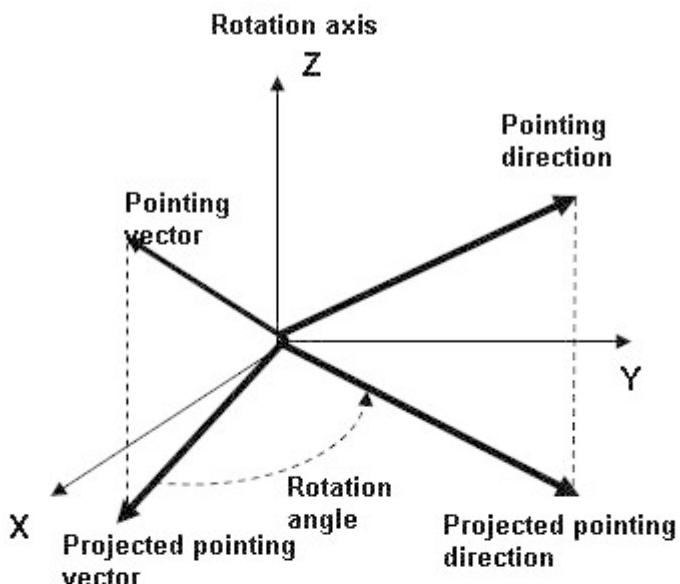
The Y direction of the ICRF (International Celestial Reference Frame. The ICRF axes are defined as the inertial axes associated with a general relativity frame centered at the solar system barycenter)

Z direction of ICRF

The Z direction of the ICRF (International Celestial Reference Frame. The ICRF axes are defined as the inertial axes associated with a general relativity frame centered at the solar system barycenter)

Solve the pointing law

Pivot connection and one pointing vector law



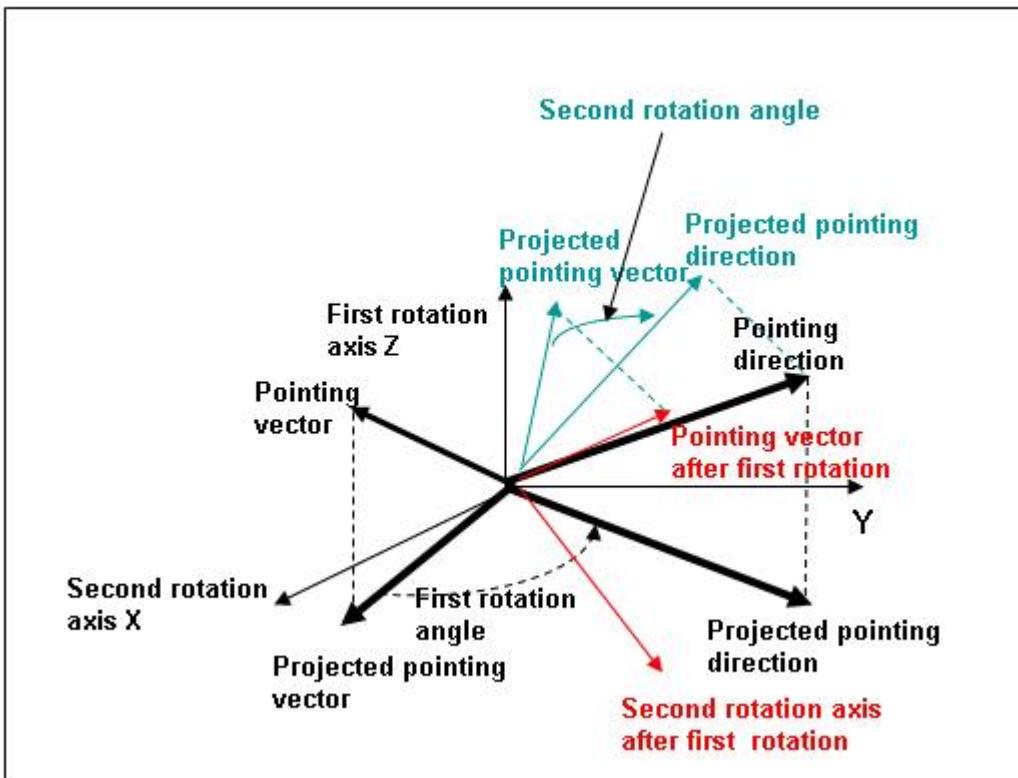
DEFINITION

This kinematic body has one degree of freedom in rotation.

Compute the rotation matrix from the requested pointing direction (according to the pointing law) and the actual pointing vector with respect of the rotation axis and restriction. The result is as close as possible to the requirement.

The actual pointing vector and the requested pointing direction are projected on the plane perpendicular to the rotation axis and the resulting transformation is a rotation around the rotational axis of the angle between the two projected vectors, limited by the restriction if necessary.

Slot ball joint and one pointing vector law

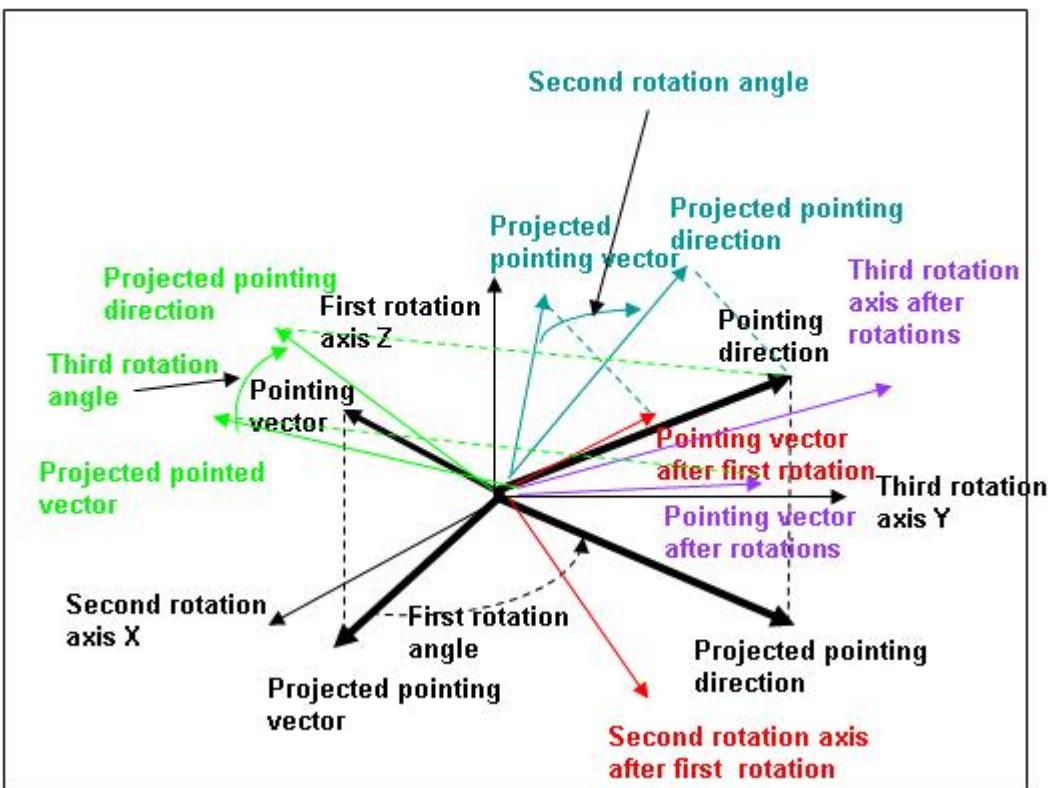


DEFINITION

This kinematic body has two degrees of freedom in rotation.

The actual pointing vector and the requested pointing direction are projected on the plane perpendicular to the first rotation axis and the resulting transformation is a rotation around the rotational axis of the angle between the two projected vectors, limited by the restriction if necessary. The first rotation is applied to the second rotation axis and to the pointing vector, and then this new pointing vector and the pointing direction are projected in the orthogonal plane of the new second rotation axis. The second resulting transformation is a rotation of the angle between the two projected vectors limited by the restriction. The resulting transformation is the combination of the two rotations.

Ball joint and one pointing vector law



DEFINITION

This kinematic body has three degrees of freedom in rotation. The actual pointing vector and the requested pointing direction are projected on the plane perpendicular to the first rotation axis and the resulting transformation is a rotation around the rotational axis of the angle between the two projected vectors, limited by the restriction if necessary. The first rotation is applied to the second rotation axis and to the pointing vector, and then this new pointing vector and the pointing direction are projected in the orthogonal plane of the new second rotation axis. The second resulting transformation is a rotation of the angle between the two projected vectors limited by the restriction. The first and second rotations are applied to the third rotation axis and to the pointing vector, and then this new pointing vector and the pointing direction are projected in the orthogonal plane of the new third rotation axis. The third resulting transformation is a rotation of the angle between the two projected vectors limited by the restriction. The resulting transformation is the combination of the three rotations.

Pivot connection and linear attitude law

The resulting transformation is a rotation of the computed angle limited by the restriction around the axis of the kinematic body if the axis of the linear attitude law and the axis of the kinematic body are identical, otherwise no transformation is done.

Slot ball joint and linear attitude law

The resulting transformation is a rotation of the computed angle limited by the restriction around the first or the second axis of the kinematic body if the axis of the linear attitude law is identical to one of the two axis of the kinematic body, otherwise no transformation is done.

Ball joint and linear attitude law

The resulting transformation is a rotation of the computed angle limited by the restriction around the first or the second or the third axis of the kinematic body

Post-Processing technical annex

Post-Processing technical annex

Overview

Introduction

This annex describes all technical information about the post-processing boxes and the development of new ones.

In this chapter

This chapter covers the following topics:

- Post-Processing technical annex
 - Overview
 - Introduction
 - In this chapter
 - The Post-Processing boxes
 - Comparison box
 - Difference or Sum box
 - Extraction CSV box
 - Extraction H5 box
 - Extraction XLSX box
 - Extraction MEM box
 - Margin and Tolerance box
 - Power Budget box
 - Radiative Budget box
 - Spatial min, max, average box
 - Time min, max, average box
 - Visualization box
 - How to create your toolbox

The Post-Processing boxes

Comparison box



The comparison box takes 2 input files and compares the values of each input entities to generate an H5 file.

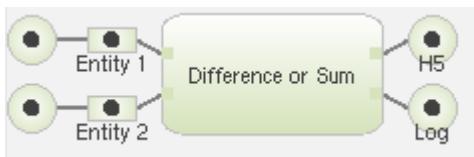
If one or more entries is a file list, the comparison box will combine one by one the files contains in these entries and generate as many outputs files as input.

Filters can be defined :

- the nodes filter : can not use group definition, only interval of nodes
- the time filter : time must be specified in the same unit as the input file

Different kind of comparison can be used : the absolute differences or the relative ones. In case of temporal comparison, common times mut be found otherwise an error is raised.

Difference or Sum box



The difference or sum box takes 2 input files or filelist and makes the difference or the sum for each input entity values to generate an H5 file.

If one or more entries is a file list, the Difference of Sum box will combine one by one the files contains in these entries and generate as many outputs files as input.

Filters can be defined :

- the nodes filter : can not use group definition, only interval of nodes
- the time filter : time must be specified in the same unit as the input file

In case of operations on entities depending on time, common times mut be found otherwise an error is raised.

Extraction CSV box



The Extraction CSV box takes 1 or more input files or file list and extracts the data into one or more CSV file (one for each input entity), according to the following filters:

- the nodes filter : can not use group definition, only interval of nodes
- the time filter : time must be specified in the same unit as the input file

⚠ If an entity depends on N parameters, these parameters shall be specified in the params filter section. The maximum number of parameters supported is 6. The name of parameters are case sensitive.

To extract all the values of a string parameter, please use "All" in the parameter values filter. For numerical parameters, values shall be specified (separated by comma).

For example, if an entity is computed for :

- times : 0, 1, 2
- one parameter named "Material" with ALU, DEFAULT values

The extraction for all the times and all values of the "Material" parameters will give a csv files with the following columns

T=0, Material=ALU	T=1, Material=ALU	T=2, Material=ALU	T=0, Material=DEFAU LT	T=1, Material=DEFAU LT	T=2, Material=DEFAU LT

...
-----	-----	-----	-----	-----	-----

If one or more entries is a file list, the Extraction CSV box will combine one by one the files contains in these entries and generate as many outputs files as input.

Extraction H5 box



The Extraction H5 box takes 1 or more input files and extracts the data into one or more H5 file (one for each input entity), according to the following filters:

- the nodes filter : can not use group definition, only interval of nodes
- the time filter : time must be specified in the same unit as the input file

If one or more entries is a file list, the Extraction H5 box will combine one by one the files contains in these entries and generate as many outputs files as input.

Extraction XLSX box

The Extraction XLSX box takes 1 or more input files and extracts the data into one single xlsx file (one worksheet for each input entity), according to the following filters:

- the nodes filter : can not use group definition, only interval of nodes
- the time filter : time must be specified in the same unit as the input file

Extraction MEM box



The Extraction MEM box takes 1 or more input files and extracts the data into one or more MEM file (one for each input entity), according to the following filters:

- the nodes filter : can not use group definition, only interval of nodes
- the time filter : time must be specified in the same unit as the input file

If one or more entries is a file list, the Extraction MEM box will combine one by one the files contains in these entries and generate as many outputs files as input.

The Extraction MEM box options are :

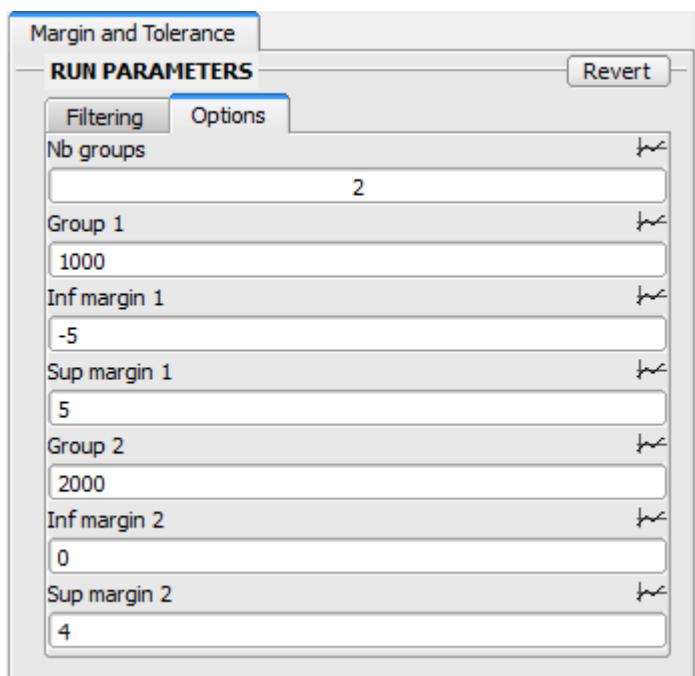
- Originator : This value will be used for the ORIGINATOR MEM flag. By default, this value is Systema
- Object Name : This value will be used for the OBJECT_NAME MEM flag. By default, this value is empty
- Object Id : This value will be used for the OBJECT_ID MEM flag. By default, this value is empty
- Space Node Number : If specified, the space node defined by this number will be ignored when extracting data. By default, this value is 99999999.

Margin and Tolerance box



The Margin and Tolerance box requires 1 or more input files and extracts the entity result into one H5 file. This box applies margin on result of nodes group.

You should define the inferior and superior margin for each group.

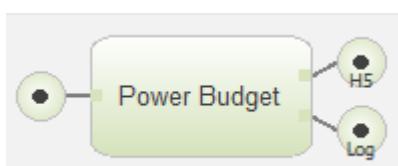


Filters can be defined :

- the nodes filter : can not use group definition, only interval of nodes
- the time filter : time must be specified in the same unit as the input file

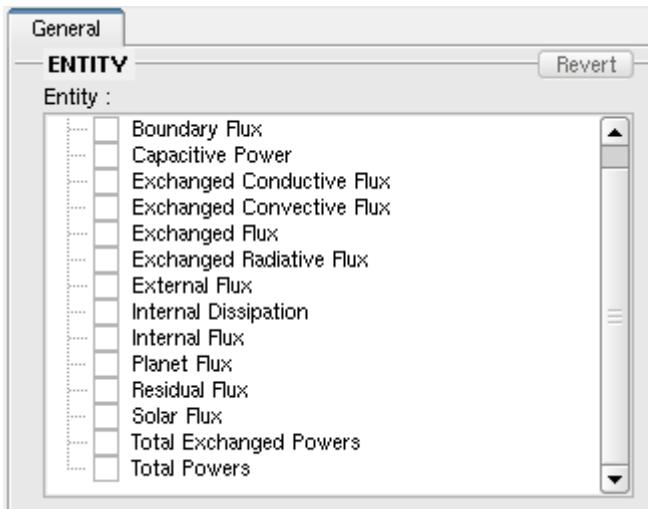
If one or more entries is a file list, the Margin and Tolerance box will combine one by one the files contains in these entries and generate as many outputs files as input.

Power Budget box



The Power Budget box takes 1 input file and calculates different fluxes and powers into a H5 file. If one or more entries is a file list, the Power Budget box will combine one by one the files contains in these entries and generate as many outputs files as input.

The following figure displays the entities available into this generated H5 file :



Filters can be defined :

- the groups filter : use group definition, not interval of nodes
- the time filter : time must be specified in the same unit as the input file

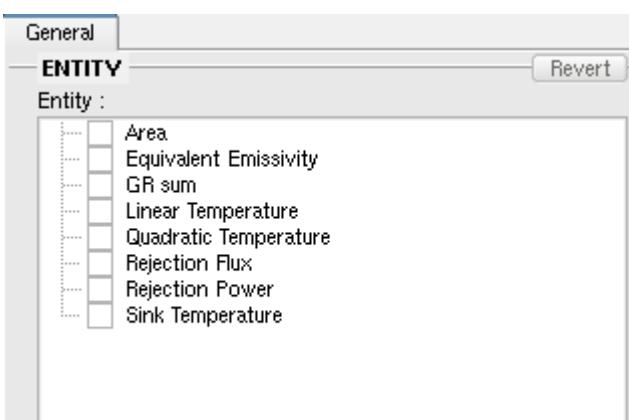
⚠ At least one group must be defined (see [Post-processing management](#) for more details about groups definitions).

Radiative Budget box



The Radiative Budget box takes 1 input file and calculates new entities into a H5 file.

The following figure displays the entities available into this generated H5 file :



Filters can be defined :

- the groups filter : use group definition, not interval of nodes
- the time filter : time must be specified in the same unit as the input file

- ⚠** At least one group must be defined (see [Post-processing management](#) for more details about groups definitions).

If one or more entries is a file list, the Radiative Budget box will combine one by one the files contains in these entries and generate as many outputs files as input.

Spatial min, max, average box



The Spatial min, max, average box will extract to a H5 file the minimum, the maximum or the average from the selected entity contained in the node groups, on a given time range. The output entities will be named with the following syntax : 'SpatialMin_ ' + selected_entity + find_of_average (all times, all elements,).

The Spatial min, max, average box takes 1 or more input files and extracts the data into one H5 file, according to the following filters:

- the groups filter : use group definition, not interval of nodes.
- the time filter : time must be specified in the same unit as the input file

- ⚠** At least one group must be defined (see [Post-processing management](#) for more details about groups definitions).

If one or more entries is a file list, the Spatial min, max, average box will combine one by one the files contains in these entries and generate as many outputs files as input.

- ⚠** If a Spatial min, max, average box takes several entities as input, these entities must come from the same H5 file.
If the average type is "Weighted arithmetic" or "Weighted arithmetic", the average entity must be defined and must be available in the H5 input file.

Time min, max, average box



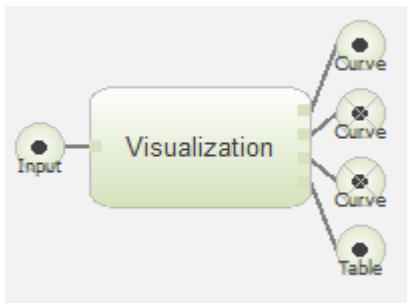
The Time min, max, average box will extract to a H5 file the minimum, the maximum or the average of selected entity over times.

The Time min, max, average box takes 1 or more input files and extracts the data into one H5 file, according to the following filters:

- the nodes filter : can not use group definition, only interval of nodes
- the time filter : time must be specified in the same unit as the input file

If one or more entries is a file list, the Time min, max, average box will combine one by one the files contains in these entries and generate as many outputs files as input.

Visualization box

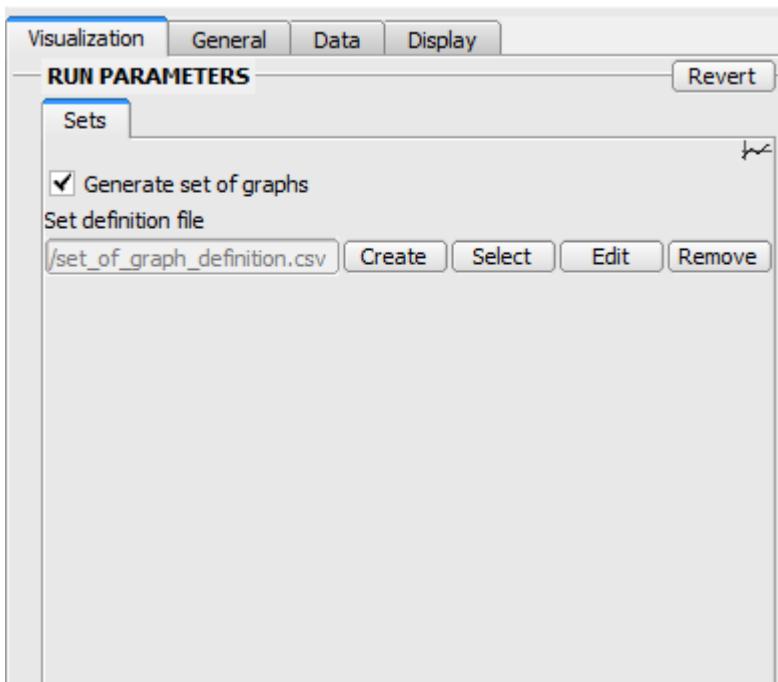


The visualization box requires 1 or more input files and generates tables and curves with different extensions: CSV for the tables, and .scg, .ps, .jpg, .bmp for the curves.

The Visualization box is not compatible with file list.

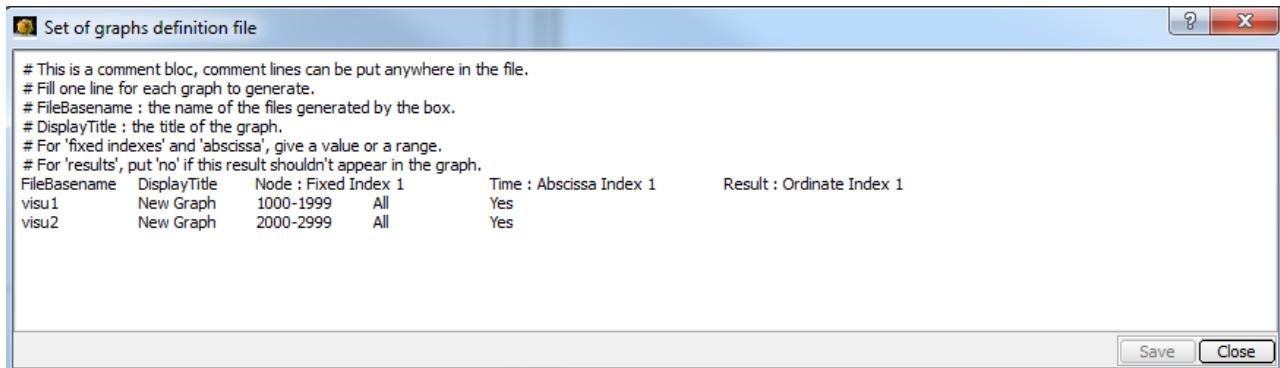
The configuration of this box is the same as the configuration of 2D Curve View described in [Displaying the result of a process in tables and curves](#).

An option is available to generate a set of graphs.



You have to create a set of graph definition file, or select an existing one. This is a csv file you can edit directly in Systema.

When you create a new one, a header is automatically added to explain how to complete the file. You must instanciate the visualization box parameters (fixed indexes, abscissa and ordinates) before using this option, because the definition file will be automatically filled.



Each line of this file represent a graph to generate. Each column is separated by a tabulation.

The first column is the name of the generated file. The first filename will be "visu1" and the second file generated will have the name "visu2"

The second column is the title of the graph. The two generated files will have the same title "New Graph".

The third column represents the range of the first Fixed index. The first index is a node range. For the first file it will be defined with the nodes 1000 to 1999 and for the second file : 2000-2999

The forth column is the abscissa. Here it is the time and the graph will display all the time for both generated files.

The last column is the result. "Yes" corresponds to the approval of generation on this result. "No" remove this result of the generated graph.

Examples of definition files :

- To generate a graph with result using parameter : Fixed indexes are nodes and parameter, the abscissa is the time and the ordinate is the result.

The first graph will display one curve for the node 3, the first parameter (Fixed index 1) has the value 1 and the second parameter (Fixed index 3) has the value 1.

The second graph will display 6 curves for the node 3 :

Parameter 1 = -1 and Parameter 2 = 1

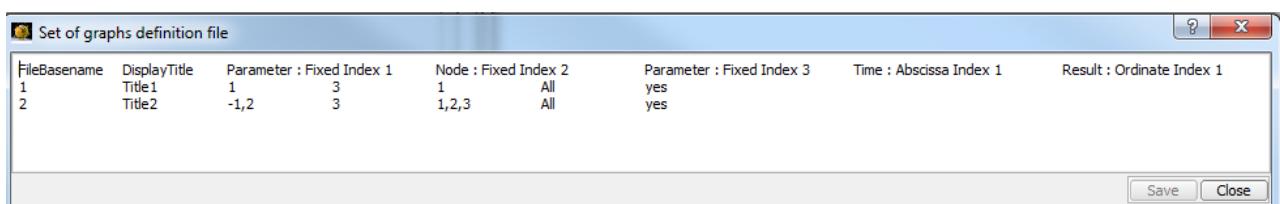
Parameter 1 = -1 and Parameter 2 = 2

Parameter 1 = -1 and Parameter 2 = 3

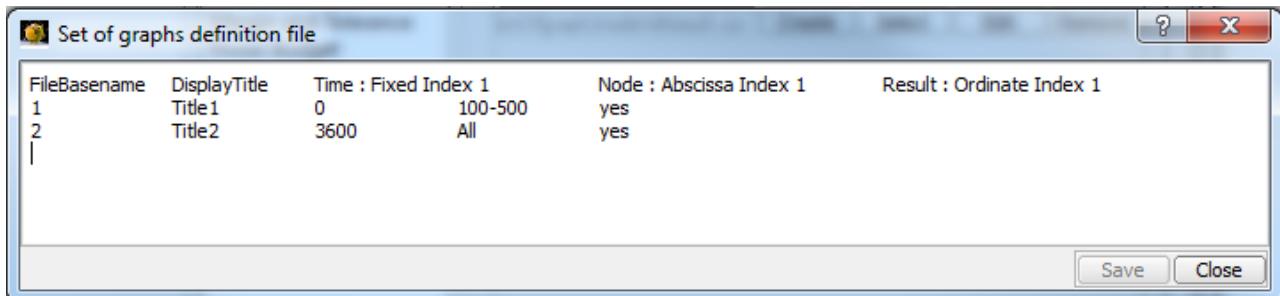
Parameter 1 = 2 and Parameter 2 = 1

Parameter 1 = 2 and Parameter 2 = 2

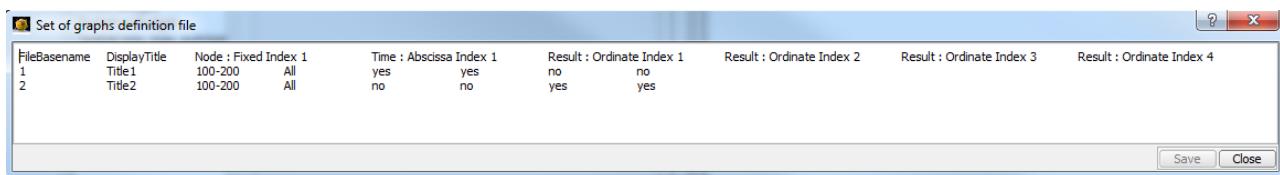
Parameter 1 = 2 and Parameter 2 = 3



- To generate a graph Nodes vs Result : the node is in abscissa and result in ordinate for a fixed time.



- To generate multicasue graphs: the first graph will display the first and second results defined in the visualization box and the second graph will display the third and forth result



How to create your toolbox

Post-Processing toolboxes are built with python scripts (except for the Visualization box).

In the installation directory of Systema, there is a directory `python/src/sysPostProcessing/` where all the files you need to make your own toolbox are available. This directory is composed of the following directoorries:

- Components**: contains the definition of the node and group of nodes (`element.py`), the entity (`entity.py`), the time (`timedefinition.py`), and a result (`dataproxy.py`)
- Generic**: contains the definition of constants (`constants.py`), log manager (`logmanager.py`) and mathematical operations (`mathkernel.py`).
- IOContainer**: provides the definition of each kind of file management thanks to file container: CSV, MEM and HDF5 file creation, HDF5 file reading.
- postProBoxes**: contains all the box definitions.

For each box, 2 files are required:

- an XML file to describe the box (**.sysppb** file). You can define the kind of input files and output files, the filters (node, group of nodes, time) and other free parameters. This definition is used to build the graphical interface of the box.
- a python script to define the actions of the box (**.py**) like the load of input file, the management and calculation on result, the generation of the output file.

In this directory, two files are also useful : `postProBox.py` and `postProBoxParameters.py`. All the boxes uses methods defined in these files but it is possible to surcharged them in your box.

Two file templates are available in this directory:

- `templatebox.sysppb`

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!-- @copyright: Airbus DS 2014 -->
<SYML CONTENTS="POSTPROBOX_DEFINITION" SYSML_VERSION="1.2">
  <!-- NAME is the name of the box ;
      EXE is the python script associated to the sysppb ;
      MAIN_CLASS is the main class of the associated python script ;
```

```

MAIN_METHOD is the method of the associated python script which will be called
first-->
<POSTPROBOX_DEF NAME="Name of the box" EXE="templatebox.py" MAIN_CLASS="TemplatePos
tProBox" MAIN_METHOD="process">
  <DESCRIPTION></DESCRIPTION>
  <!-- MULTI_INOUT_DEF defines more than one input in the box ;
      if only one entity is necessary use ENTITY_DEF (example in comparebox.sysppb)
      if a file is necessary as input use FILE_DEF (example in
radiativebudgetbox.sysppb)
      ROLE_TITLE_IN is the title of the input displayed in the edition window of
the box
      ROLE_DESC_IN is the description of the input displayed in the edition window
of the box
      ROLE_TITLE_OUT is the title of the output displayed in the edition window of
the box: CSV, MEM ...
      ROLE_DESC_OUT is the description of the output displayed in the edition
window of the box
      MIN_NB is the minimum number of input/output
      MAX_NB is the maximum number of input/output
      KEY_OUT is the key of the output : csv, h5, mem
      TYPE_OUT is the type of the output : CSV, HDF5, MEM
      FORMAT_OUT is the format of the output : CSV, H5, TXT
      BASENAME_OUT is the basename displayed in the edition window of the ouput
-->
<MULTI_INOUT_DEF ROLE_TITLE_IN="Entity" ROLE_DESC_IN="input entity" ROLE_TITLE_OUT="
CSV" ROLE_DESC_OUT="CSV Result file"
  MIN_NB="1" MAX_NB="2" KEY_OUT="csv" TYPE_OUT="CSV" FORMAT_OUT="CSV" BASENAME_OUT="
generated.output.filename" >
</MULTI_INOUT_DEF>

<!-- FILE_DEF define a file as input or output
    ID is the kind of file : LOGfile or HDF5file
    INOUT defines if the file is an input or an output of the box
    KEY is the key: log, h5
    TYPE is the type of the file : CSV, HDF5, MEM
    FORMAT is the format of the output : CSV, H5, TXT, LOG
    ROLE_TITLE is the title of the file displayed in the edition window of the
file
    ROLE_DESC_IN is the description of the file displayed in the edition window
of the file
-->
<FILE_DEF ID="LOGfile"  INOUT="OUTPUT" KEY="log" TYPE="TXT" FORMAT="LOG" TOOL="">
  ROLE_TITLE="Log" ROLE_DESC="Log file">
    <!-- ELEM allows to define for each file the name of the file and its type -->
    <ELEM ID="Default_Path" TYPE="STRING" STR="name.of.the.file.log"/>
</FILE_DEF>

<!-- RUN_PARAMETERS_DEF defines the parameters displayed in the edition window of
the box -->
<RUN_PARAMETERS_DEF>
  <!-- PARAM_CATEGORY_DEF defines a tab named "Filtering" in the RUN PARAMETERS
part of the edition window -->
  <PARAM_CATEGORY_DEF NAME="Filtering">
    <!-- PARAMETER_DEF defines a string field with the label "Nodes filter" and
the default value is "All"
        ID is the value used also in the Python script to get the value of this
string
-->
    <PARAMETER_DEF ID="nodes_filter" NAME="Nodes filter" TYPE="STRING" UNIT="">

```

```

<ELEM ID="Default_Value" TYPE="STRING" STR="All"/>
</PARAMETER_DEF>
<!-- PARAMETER_DEF defines a string field with the label "Time filter" and
the default value is "All"--&gt;
&lt;PARAMETER_DEF ID="times" NAME="Time filter" TYPE="STRING" UNIT=""&gt;
    &lt;ELEM ID="Default_Value" TYPE="STRING" STR="All"/&gt;
&lt;/PARAMETER_DEF&gt;
&lt;/PARAM_CATEGORY_DEF&gt;
<!-- PARAM_CATEGORY_DEF defines a tab named "Options" in the RUN PARAMETERS
part of the edition window
It is possible to define the options you want in this part.
--&gt;
&lt;PARAM_CATEGORY_DEF NAME="Options"&gt;
    &lt;PARAMETER_DEF ID="Myoption" NAME="Name of my option" TYPE="STRING" UNIT=""&gt;
        &lt;ELEM ID="Default_Value" TYPE="STRING" STR="default value of my option"/&gt;
    &lt;/PARAMETER_DEF&gt;
&lt;/PARAM_CATEGORY_DEF&gt;
&lt;/RUN_PARAMETERS_DEF&gt;
&lt;/POSTPROBOX_DEF&gt;
&lt;/SYSML&gt;</pre>

```

- templatebox.py

This template box extracts all the entity values into a CSV file, and display in the run window the defined option. You can adapt these templates to make your own toolbox.

Each time a modification is done on one of these 2 files, Systema must be relaunched to take into account the modifications.

File formats annex

File formats annex

- File formats annex
 - Internal File Formats
 - External File Formats
 - STEP-TAS
 - STEP
 - Bulk import/export
 - Import
 - Export
 - Customization
 - Groups definition
 - Materials definition
 - Temperatures
 - Coating
 - Trajectory import/export
 - Kinematic import/export
 - CNES import/export
 - OEM
 - AEM
 - Time references
 - Gamma50
 - J2000
 - Modified julian
 - Epoch time

Internal File Formats

Systema has resort to different file that are usable only inside Systema.

- **Model:**
 - .sysmdl: SYSTEMA Model file
 - .sysml: outdated SYSTEMA Model file
 - .wrl: VRML97 simplified format using only quadrangles and triangles(easy to open, even for big models)
 - .SYSBAS: SYSTEMA V2 format (human readable)
 - .SYSEXP: SYSTEMA V3 format (ASCII but less readable than SYSBAS)
 - .sysmdlts: SYSTEMA Model file - Old version 4.5.x
 - .yaml: Thermica ascii format
 - .yml: Thermica ascii format
- **Material:**
 - .sysmtr: material file
 - .sysmtrlt: SYSTEMA Material file - Old version 4.5.x
- **Meshing:**
 - .sysmsh: SYSTEMA Meshing file
 - .sysmshlt: SYSTEMA Meshing file - Old version 4.5.x
 - .SYSBAS: SYSTEMA V2 format (human readable)
 - .SYSEXP: SYSTEMA V3 format (ASCII but less readable than SYSBAS)
 - .bdlblk: Bulk format (using only triangles and quadrangles)
 - .yaml: Thermica ascii format
 - .yml: Thermica ascii format
- **Schematic:**
 - .syschm: SYSTEMA Schematic file

- **Trajectory:**
 - .systrj: SYSTEMA Trajectory file
 - .systrjts: SYSTEMA Trajectory file - Old version 4.5.x
- **Kinematics:**
 - .syskin: SYSTEMA Kinematics file
 - .syskinlts: SYSTEMA Kinematics file - Old version 4.5.x
- **Mission:**
 - .sysmis: SYSTEMA Mission file
 - sysmislt: SYSTEMA Mission file - Old version 4.5.x
- **Processing:**
 - .sysprc: SYSTEMA Processing file
 - .sysprclts: SYSTEMA Processing file - Old version 4.5.x
- **Post-Processing**
 - .sysppr: SYSTEMA Post-Processing file
 - .sysvisu: SYSTEMA format for 2D graph saving
- **Applications**
 - .sysapp: element definitions
 - .sysppb: post-processing box definition
- **Result**
 - .h5: Hdf5 result file using the Systema Data Storage (SDS) formalism
Readable using external free software HdfView: <https://support.hdfgroup.org/products/java/release/download.html#bin>
 - .prptbl: ascii format for results to be displayed in the 3D

External File Formats

Systema-specific data can be imported/exported from/into external file formats used by other software.

Exchanged format	Category	CATI A	ESAT AN	Thermal Desktop	Nastran	Systema V3	Blender	IDM-CIC	STK
.stp/.enf: CAD	Model	X	X	X	X				
.bdf/.blk: Nastran	Model, Meshing				X				
.stp/.step: STEP-TAS	Model, Meshing		X					X	
.wrl: VRML97	Model	X					X		
.trjinp: TRJINP	Trajectory					X			
.oem: CIC ephemeris	Trajectory							X	
.stk: ephemeris	Trajectory								X
.stk: attitude (quaternion only)	Kinematics								X

Exchanged format	Category	CATIA	ESATAN	Thermal Desktop	Nastran	Systema V3	Blender	IDM-CIC	STK
.aem: CIC spacecraft attitude (quaternion only)	Kinematics							X	
.mem: CIC moving body attitude (quaternion only)	Kinematics							X	

STEP-TAS

For more details about model conversions, please refer to :

Model Conversions Guidelines

Systema vs Step-Tas

Thermisol vs Esatan

This document provides guidelines for the model conversions between Systema and other thermal formats and software. It is available on demand at engineering.software@airbus.com.

STEP

The STEP protocols supported by Systema are :

- AP203
- AP214

The tessellation parameters (chord = 1mm and angle = 60deg) used at Step import can be customized using [environment variables](#).

Bulk import/export

Import

The import is available for model and meshing with the following filter: *.bdf, *.blk (NASTRAN files).

The generated model in Systema is organized as such:

- Model
 - Default Properties (object)
 - shape1 (type=FACETEDSHAPE)
 - ...
 - shapeN (type=FACETEDSHAPE)

Export

The bulk export is available for meshing only with the following filter: *.bdf, *.blk (NASTRAN files). It creates GRID, CTRIA3 and CQUAD4.

The data is exported in free format, using a comma separator.

Customization

Import/Export can be customized by setting environment variables. Please refer to [General](#) for more details.

Groups definition

PSHELL (or PSOLID) represent a group of shapes (like CTRIA3, CQUAD4, etc). Shapes are referencing a PSHELL PID. The PSHELL reference a material using the MID1 field. It is used in Systema to import thickness (as a property) only, and to create subgroups (main groups are defined with materials). Systema do not export PSOLID.

Materials definition

Systema supports both MAT1 (structural) and MAT4 (thermal) fields for materials import. Main groups are defined by material at import. Systema do not export materials for now.

Temperatures

The temperatures are not handled by Systema at bdf interface.

Coating

The coating is not handled by Systema at bdf interface.

Trajectory import/export

The trajectory module can import Time-Position-Speed ASCII files. Please see [Trajectory management](#) for more details about the supported formats.

All trajectory in Systema can be exported as ASCII files from the menu File -> Export Points

Kinematic import/export

Kinematic law can be imported from files containing external data files indicating attitude or transformation of the object over time (CSV, STK, or text files). Please see [Kinematics management](#) for more details about the supported formats.

CNES import/export

Since 4.7.1, it is possible to import and export the trajectory and the kinematic in OEM and AEM format

OEM

OEM files are managed by Systema in the Trajectory tab, through a Time-Position-Velocity import process. OEM trajectory (Orbit Ephemeris Message) is defined by ASCII files and describes the orbit via the position and the velocity of the satellite. It is based on the reference format CCSDS. The CIC exchange protocol limits the use of some parameters (see the description below) :

CNES CIC based OEM ephemeride (*.txt) file

The file contains a header, a metadata block and a data block

See following example:

```
CIC_OEM_VERS = 2.0
CREATEION_DATE = 2015-03-30T17:01:25
ORIGINATOR = CNES
```

META_START

```
COMMENT Ephemeride file example
OBJECT_NAME = CubeSat
OBJECT_ID = CubeSat
CENTER_NAME = EARTH
REF_FRAME = EME2000
TIME_SYSTEM = UTC
```

META_STOP

COMMENT Date in modified julian days followed by seconds, positions (x, y z)
in km and optional velocity values

```
55276 0.0 4264.085921 -832.254441 5618.208465
55276 30.24 4421.338968 -916.904119 5481.941484
55276 59.616 4574.176227 -1000.637990 5340.185449
```

AEM

AEM and MEM files are managed by Systema in the Kinematics tab, with the kinematic law "Transformations defined in a file". AEM kinematic (Attitude Ephemeris Message) describes the satellite attitude. MEM file (Mission Ephemeris Message) describes the rotation of satellite moving bodies. They are defined by ASCII files and based on the reference format CCSDS. The CIC exchange protocol limits the use of some parameters (see the descriptions below).

The order of the quaternions is QW, QX, QY, QZ.

AEM file description:

CNES CIC based AEM attitude (*.txt) file

See following example:

```
CIC_AEM_VERS = 1.0
CREATEION_DATE = 2015-03-30T17:12:25
ORIGINATOR = CNES
```

META_START

```
COMMENT Attitude AEM file example
OBJECT_NAME = CubeSat
OBJECT_ID = CubeSat
REF_FRAME_A = EME2000
REF_FRAME_B = SC_BODY_1
ATTITUDE_DIR = A
TIME_SYSTEM = UTC
ATTITUDE_TYPE = QUATERNION
```

META_STOP

```
COMMENT Date in modified julian days followed by seconds and quaternion (QW
Q1 Q2 Q3).
```

```
55276 14730.336 0.210085 0.240801 0.025170 0.947231
55276 14759.712 0.210409 0.255787 0.022011 0.943301
55276 14789.952 0.210681 0.270709 0.018847 0.939135
```

MEM file description:

CNES CIC based MEM attitude(*.txt) file

See following example:

CIC_MEM_VERS = 1.0

CREATEION_DATE = 2015-03-30T17:17:25

ORIGINATOR = CNES

META_START

COMMENT Rotation MEM file example

OBJECT_NAME = CubeSat

OBJECT_ID = CubeSat

USER_DEFINED_PROTOCOL = CIC

USER_DEFINED_CONTENT = ROTATION_ANGLE_SA_

ATTITUDE_DIR = A

TIME_SYSTEM = UTC

ATTITUDE_TYPE = QUATERNION

META_STOP

COMMENT Date in modified julian days followed by seconds and angle in degrees.

55275 85516.3872 -180

55276 607.9104 -90

55276 2099.4336 0

The time reference used by those file is 1858/11/17@00:00:00.

To convert to Gamma50 epoch, you need to withdraw 33282.0 days and to convert the second file column from seconds to day fraction.

In order to import those kinematics file (AEM/MEM) properly into Systema, you need to set the relevant reference axis at the parent level of your kinematics. See [Kinematics technical annex](#) for more details.

Time references

Systema supports several time reference to define dates :

Gamma50

Number of days since 01/01/1950 at 00:00:00. For instance : 26449.5 == 01 June 2022 12:00:00.000

J2000

Number of days since 01/01/2000 at 12:00:00.. For instance : 8187 == 01 June 2022 12:00:00.000

Modified julian

Number of days since 17 november 1858 at 00:00:00.. For instance : 58731.5 == 01 June 2022 12:00:00.000. Used in OEM, AEM and MEM files

Epoch time

Number of seconde since 01 january 1970 at 00:00:00. For instance : 120.0 == 01 January 1970 00:02:00.000. Used for STK files if ScenarioEpoch is not defined.

Administration annex

Administration annex

Overview

Introduction

This annex provides instructions on how to install and configure SYSTEMA V4 on Linux and Windows systems. This document assumes that you have a working knowledge of the applicable operating environments.

Please feel free to contact us if you need further information about SYSTEMA. To get how to contact us, please visit <http://www.systema.airbusdefenceandspace.com/contact-us.html>.

In this chapter

This chapter covers the following topics:

- [Administration annex](#)
 - [Overview](#)
 - [Introduction](#)
 - [In this chapter](#)
 - [Installing SYSTEMA](#)
 - [Download products](#)
 - [Installation step-by-step](#)
 - [System requirements](#)
 - [SYSTEMA Licensing](#)
 - [Introduction](#)
 - [Configuration](#)
 - [Troubleshooting](#)
 - [SYSTEMA environment](#)

Installing SYSTEMA

Download products

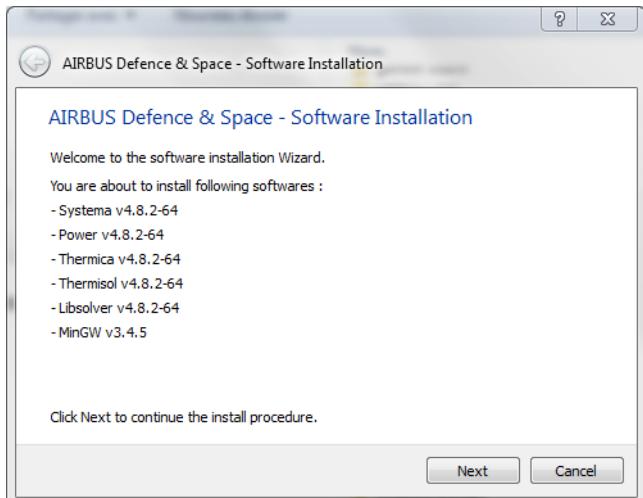
All releases of SYSTEMA are available on the SYSTEMA website <http://www.systema.airbusdefenceandspace.com/>.

To download it, please:

- Sign in in the **Support** section (if needed, request an account on this page)
- Click **Download** in the Support page.
- In the Download menu, select the relevant version of SYSTEMA (for instance Thermica -> V4 -> Thermica X.Y.Z) and select **Click to download**. Windows and Linux installers are available.

Installation step-by-step

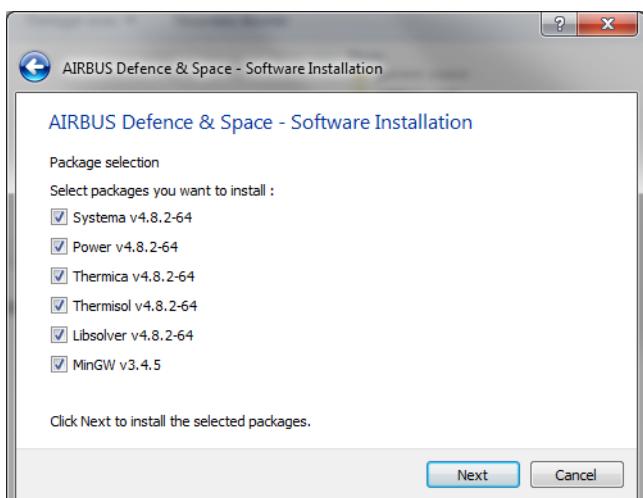
- Start the Systema installer



- Review the Software License Agreement

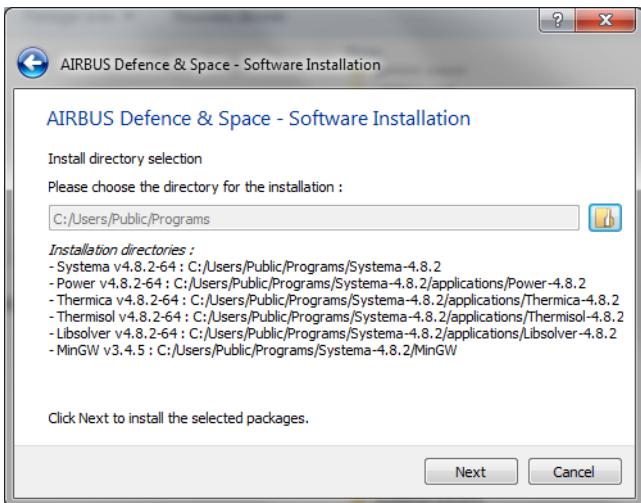


- Specify the packages to install



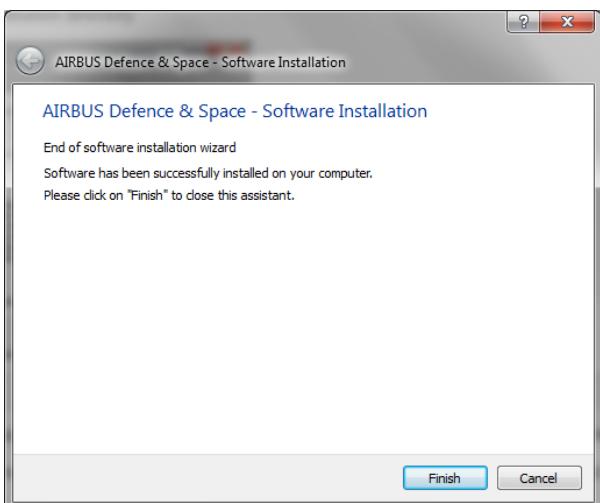
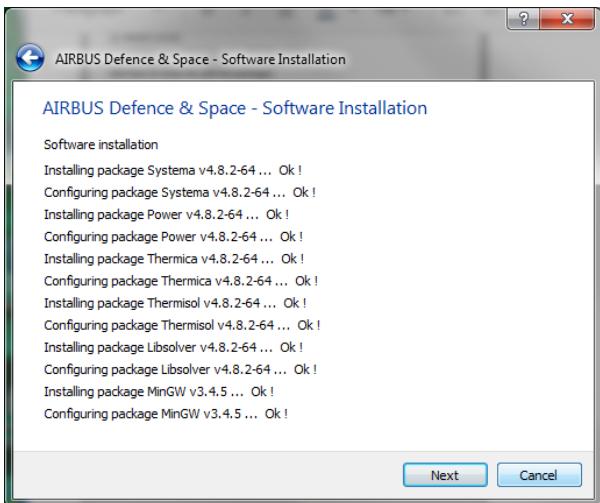
⚠ It is highly recommended to keep the default selection.

- Select the installation directory



⚠ Please avoid the spaces in the Systema folder names, especially in the path where Systema is installed.

- Launch the installation



- ⚠** Any user of Systema shall have sufficient access right to access and execute any files located in the installation folder.

System requirements

Systema is fully supported on the following OS:

- Windows 7
- Windows 10
- Linux Red Hat 6.6

Only 64 bits versions of Systema are available.

To be able to load SYSBAS files on Linux 64bit the following packages have to be installed:

- expat.i686
- fontconfig.i686
- freetype.i686
- glibc.i686
- libICE.i686
- libSM.i686
- libX11.i686
- libXau.i686
- libXcursor.i686
- libXdamage.i686
- libXext.i686
- libXfixes.i686
- libXi.i686
- libXinerama.i686
- libXmu.i686
- libXrandr.i686
- libXrender.i686
- libXt.i686
- libXxf86vm.i686
- libdrm.i686
- libgcc.i686
- libselinux.i686
- libstdc++.i686
- libuuid.i686
- libxcb.i686
- mesa-dri-drivers.i686
- mesa-libGL.i686
- elfutils-libelf.i686
- mesa-dri-filesystem.i686
- mesa-private-llvm.i686
- mesa-libGLU.i686

SYSTEMA Licensing

Introduction

SYSTEMA requires a valid SYSCOD V5 license to be fully operational. Two licensing mode are support by SYSTEMA :

- The “local license” system is used to provide authorization to execute applications without using any floating licenses (authentication through a license server). The benefit of this system is that a license server

is not required. All information to make applications work are defined in a license file. This file's name must be suffixed by .moslf to be recognized as a license file by the applications using SYSCOD V5.

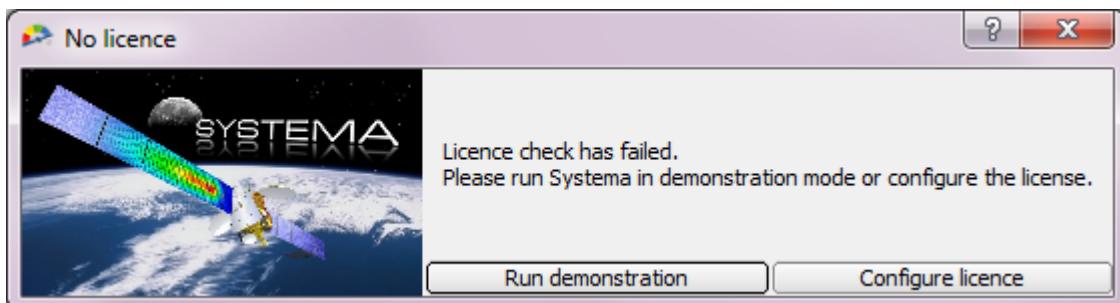
- The "license server" system is used to provide the authorization to execute applications on any computers to the network. The benefit of this system is that only one license file is required, containing all needed information to make the application to work. This file's name must be suffixed by .moslf to be recognized as a license file by the applications using SYSCOD V5.

SYSCOD V5 is available on Windows and Linux (see § [Download Systema](#)). For more detail about this license manager, please refer to "SYSCOD V5 - User manual" (available on <http://www.systema.airbusdefenceandspace.com/support/download/licence-server/v4.html>).

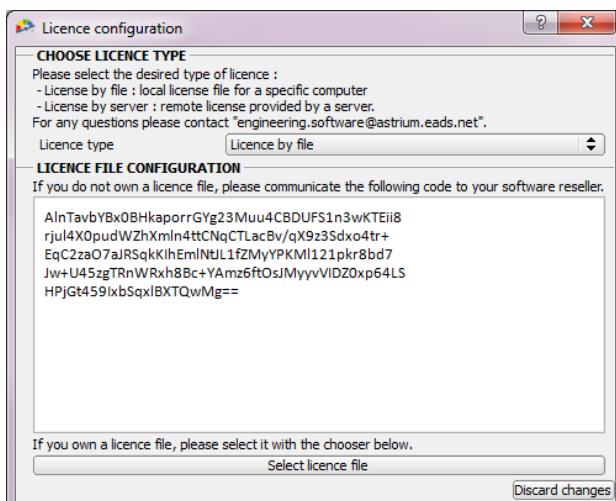
Configuration

Once downloaded and installed, to get a local license please :

- Launched SYSTEMA
- Choose **Configure licence** from the popup window.



- A new window will open displaying a crypted key. Please copy paste this key and send it to our technical support team (see [Introduction](#)) so that we will generate a license file in turn.



For server licenses, please contact the technical support team.

Troubleshooting

Issue	Solution
When using Systema through a ssh connexion, the GUI is often refreshed.	<p>Set QT_GRAPHICSSYSTEM = native</p> <p>3 graphics backend are supported by Qt4(.5+) : <i>native, raster,opengl</i>.</p> <p><i>native</i> must be used when using a SSH X-forward mode whereas the other ones must be used for a local rendering</p>
All 3D views of Systema are empty	Please check that the OpenGL rendering is allowed by your system
<p>3D problem through SSH Fowarding :</p> <p>Running OpenGL bases applications on a remote computer through a ssh tunnel with X11 fowarding might not be successful and result a X error.</p> <p>The OpenGL base rendering application may also complain that it is unable to create a GL context.</p>	<p>Verify that Direct rendering is enabled. You can check this by running</p> <pre>glxinfo grep rendering</pre> <p>The Nvidia libGL (at least by default) does not allow indirect GLX calls. Preventing the application that uses OpenGL to be run on the remote computer through the secure shell with X11 fowarding and authentification</p> <p>Make sure that both the local and the remote machines are using mesa's libGL.</p> <p>One way to figure out which libGL implementation the system is currently using is by running</p> <pre>ldconfig -p grep libGL.so</pre> <p>Installing mesa library over Nvidia driver could mess up the libGL used by the host.</p> <p>A quick fix is copying the libGL.so of /usr/lib/x86_64-linux-gnu/mesa from hother host. Then, once ssh to the host, set LD_LIBRARY_PATH as follow :</p> <pre>export LD_LIBRARY_PATH=/usr/lib/x86_64-linux-gnu/mesa:\${LD_LIBRARY_PATH}</pre> <p>The the current session will use mesa's libGL.so instead of the Nvidia one</p>
Some applicative modules report a crash	Check that there is no restriction on all Systema folders for reading and executing.
"Unable to generate mission file" raised when running a processing linked to a schematic file	Check that no error about unknown components were raised when opening the schematic file (see log area). if it is the case, rebuild the schematic in a new file.

SYSTEMA environment

Please refer to [SYSTEMA environment](#) paragraph.