

# SALOME 5.1.1

## Public release announcement

July 2009

### GENERAL INFORMATION

CEA/DEN, EDF R&D and OPEN CASCADE are pleased to announce [SALOME](#) version [5.1.1](#). It is a public release that contains the results of planned major and minor improvements and bug fixes against SALOME version 4.1.5 released in May 2009.

SALOME version 5.1.1 includes the results of porting of SALOME GUI to the Qt series 4. The look-n-feel of SALOME desktop has been improved to take into account features provided by the Qt 4.

**NOTE** for the developers: the API of most of SALOME GUI functionality has been changed, thus the custom development based on SALOME series 4 will be incompatible with SALOME series 5. Such a code will require porting.

## Table of Contents

- **GENERAL INFORMATION .....1**
- **NEW FEATURES AND GENERAL CHANGES .....3**
  - PORTING TO QT 4 .....3
  - PREREQUISITES CHANGES (COMPARING WITH SALOME 4.1.5).....3
  - LICENSE RESTRICTIONS .....4
  - MAJOR CHANGES .....4
  - IMPROVEMENTS .....5
- **BUG CORRECTIONS .....19**
- **CASCADE 6.3 SERVICE PACK 5 BUG CORRECTIONS .....19**
- **SUPPORTED LINUX DISTRIBUTIONS AND PRE-REQUISITES .....20**
- **HOW TO INSTALL AND BUILD SALOME .....22**
- **SALOME SYSTEM REQUIREMENTS .....22**
- **HOW TO GET THE VERSION AND PRE-REQUISITES.....22**
- **KNOWN PROBLEMS AND LIMITATIONS .....23**

 **NEW FEATURES AND GENERAL CHANGES****PORTING TO QT 4**

The following SALOME modules have been ported to the Qt 4:

- Main modules:
  - **GUI** (IAPP)
  - **MED** (Med)
  - **GEOM** (Geometry)
  - **SMESH** (Mesh)
  - **VISU** (Post-Pro)
  - **YACS**
- Mesh plug-ins
  - **NETGENPLUGIN**
  - **GHS3DPLUGIN**
  - **BLSURFPLUGIN**
  - **GHS3DPRLPLUGIN**
  - **HexoticPLUGIN**
- Samples
  - **LIGHT**
  - **HELLO**
  - **PYHELLO**

The following SALOME modules do not depend on Qt and thus do not require porting:

- **KERNEL**
- **CALCULATOR**
- **PYCALCULATOR**
- **COMPOGUINENT**
- **RANDOMIZER**

The following SALOME modules have not been ported yet to the Qt 4:

- **SUPERV** (Supervision)
- **FILTER**
- **MULTIPR**

**PREREQUISITES CHANGES (COMPARING WITH SALOME 4.1.5)**

- Qt 4.4.3
- sip 4.7.7 (supports both Qt series 3 and 4)
- PyQt 4.4.3 (compatible with Qt 4.4.3)
- Qwt 5.1.1 (compatible with both Qt series 3 and 4)
- msg2qm tool is removed (obsolete)

**NOTE:** all other pre-requisites are the same as for SALOME release 4.1.5. For the full list of pre-requisites refer the table below.

## LICENSE RESTRICTIONS

- Hereby we explicitly declare that PyQt 4 toolkit (Riverbank Computing Ltd) is distributed under the terms of GPL license.

## MAJOR CHANGES

The Qt series 4 API is not compatible with the Qt series 3. In this concern, most of the SALOME GUI functionality has been changed to bring in concordance with Qt 4 API. Thus, the SALOME GUI API itself been re-implemented significantly. No any backward compatibility with Qt series 3 is provided.

In the same way, since Qt 4 series is not binary compatible with 3 series all the Qt-based applications should be re-compiled to work with Qt 4. This, surely, concerns SALOME also.

Naturally, the SALOME GUI series 5 is not compatible with SALOME GUI series 4. The custom development made for SALOME series 4 will require porting to be compatible with SALOME series 5.

Most of changes made in SALOME sources in frames of Qt 4 porting concern the SALOME GUI module and include:

- Re-implemented desktop, widgets, dialog boxes and utility classes
- Re-implemented menus, toolbars management
- Re-implemented and powered preferences management
- Improved splash screen window
- Fully re-implemented and powered Object browser
- Fully re-implemented data model classes
- Re-implemented SALOME GUI style
- Some sources packages are renamed (for example, PythonConsole package has been renamed to the PyConsole. Some classes and files are renamed also.
- Some classes are removed as obsolete and new classes are implemented.
- Packages inter-dependencies are changed (for example, Style package now depends on QtX)
- Changed internationalization system (ts-files instead of po-files)
- Improved resources management

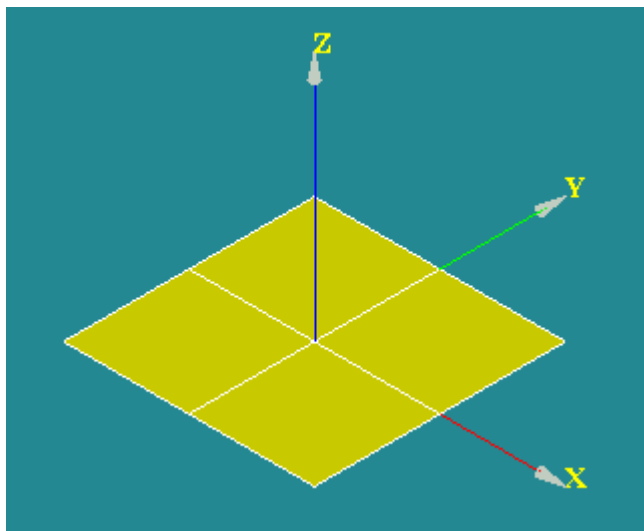
SALOME GU look-n-feel is changed in some concerns. Mainly all the improvements came from Qt 4 functionalities:

- Animated windows/toolbars docking
- Improved windows docking mechanism
- Improved painting system,
- ...

The changes in other SALOME modules are mainly caused by modifications of SALOME GUI module API.

## IMPROVEMENTS

## New primitives: Square Face



**Square Face** is a square plane of certain dimensions and orientation

To create a Square Face, select in the Main Menu **New Entity → Primitives → Face**.

It can be defined using two possible sets of parameters:

- at the origin of coordinates by the height, the width and the orientation axis;
- by the height, the width sizes and the normal to the center of the face.

It is also possible to create a **Square Face** from another face changing the values of height and width.

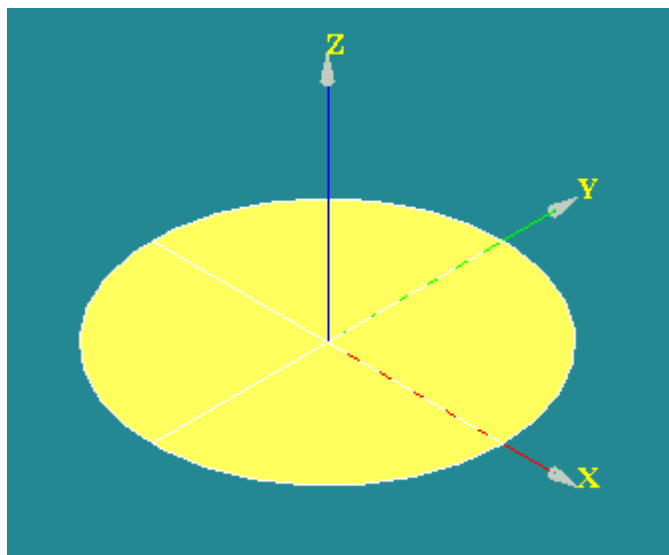
## New primitives: Disk

**Disk** is a round plane with a certain radius and orientation.

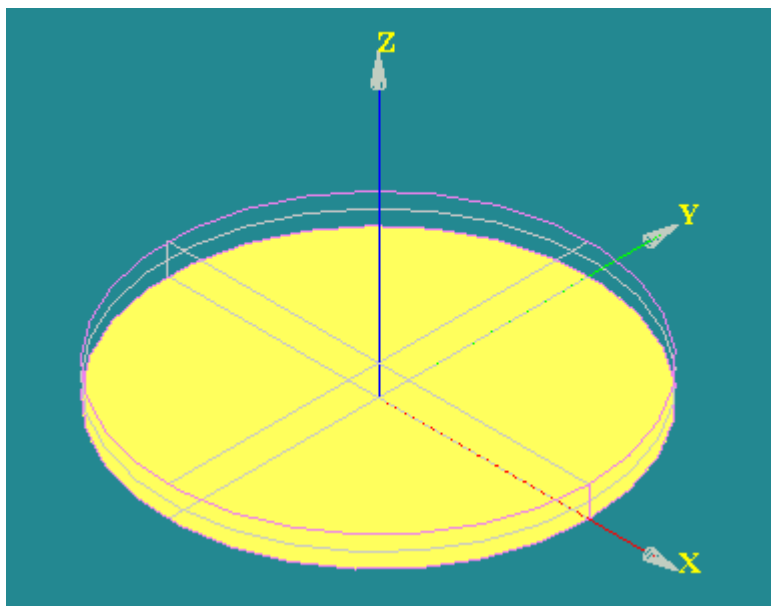
To create a **Disk**, select in the Main menu: **New Entity → Primitives → Disk**.

It can be defined using three possible sets of parameters:

- at the origin of coordinates by the radius and the orientation;
- by the center point, the normal vector and the radius;
- by three points that lie on its boundary.



### New method for creation of extrusion

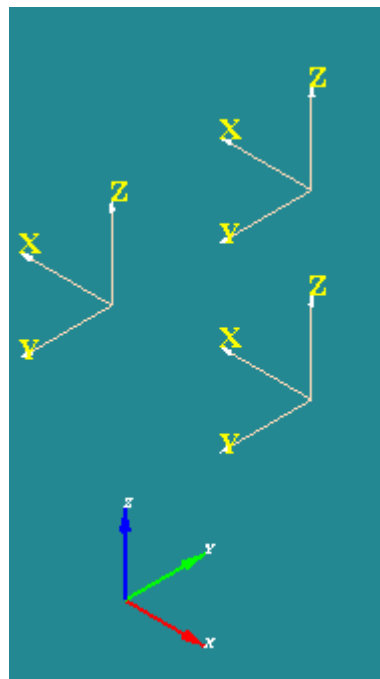


It has become possible to create an **Extrusion** by the base shape and the vector defined directly by **Dx**, **Dy**, **Dz** coordinates.

### Reference trihedron

A multi-colored **Reference Trihedron** with constant size and position has been added in the bottom-left corner of the OCC Viewer. It allows to distinguish the global LCS axis from the others and to keep it always visible

In the picture you can see the reference trihedron and several created local coordinate systems.

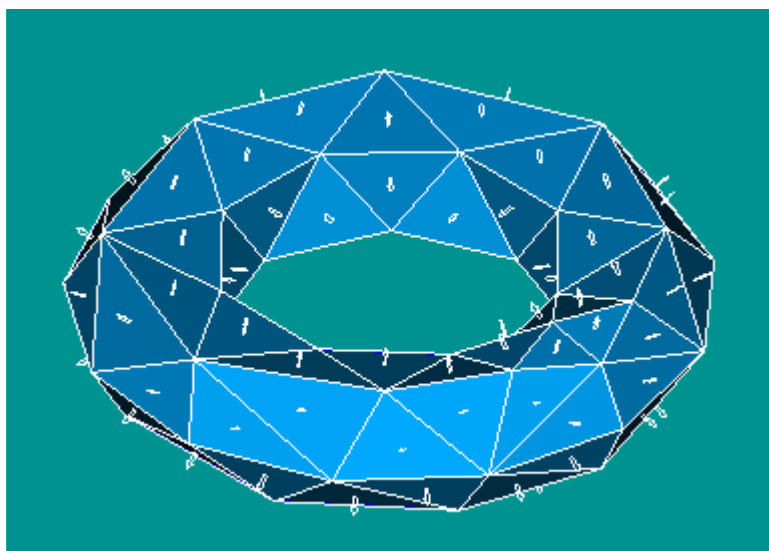


## Mesh computation option

New **Mesh computation** option has been added in **Mesh preferences**. It allows selecting the mode of notification about mesh computation results.

- **Never** - the dialog with results is not shown at all;
- **Errors only** - the dialog with results is shown if some errors have occurred during mesh computation;
- **Always** - the dialog with results is shown after each mesh computation. This is a default mode.

## Show mesh elements orientation



New **Show Orientation** functionality available from the pop-up on the selected object in the VTK viewer shows vectors of orientation of faces of the selected mesh.

It is possible to change the color, the size and the type of orientation vectors (2D planar or 3D) in **Preferences → Mesh → Mesh → Orientation of Faces**.

## Action groups

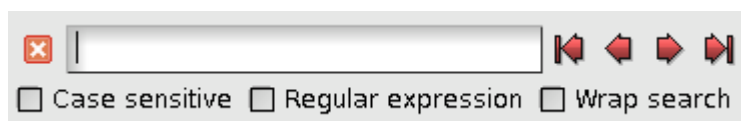
The functionality of using **Action Groups** in SalomePyQt Python interface has been restored in SALOME series 5x. The SalomePyQt Python interface exports the `QtActionGroup` class which behaves in the same way as it was done by the `QActionGroup` class in Qt series 3. The API of `QtActionGroup` class repeats the API of Qt 3's `QActionGroup` class. The function `SalomePyQt.createActionGroup()` now creates and returns the instance of the `QtActionGroup` class.

Thus, the porting expenses on the user side should be minimal (or even equal to zero, if the code never checks that the class name of the action group object is equal to "QActionGroup").

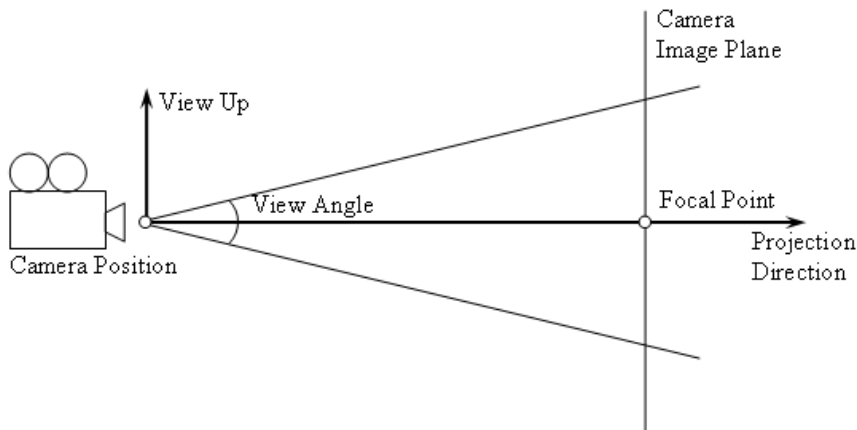
Testing of this functionality is possible via running of PYHELLO module which has been improved correspondingly to use **Action Groups** functionality. Its GUI creates and uses one `QtActionGroup` object in menu and toolbar.

## Find Tool

Object Browser has been improved with the **Find Tool**, which allows performing context search of the items.



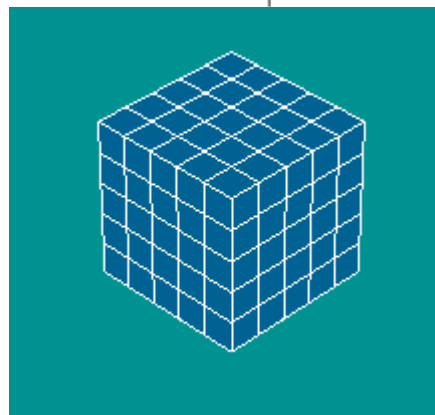
### Perspective view



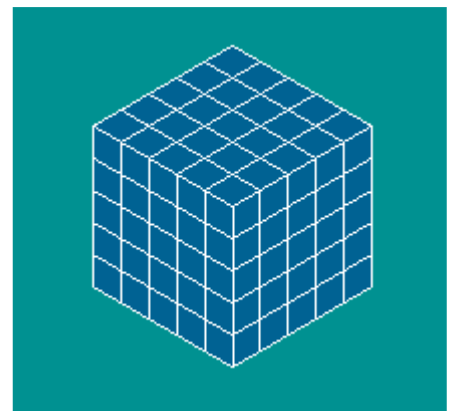
3D objects in **Orthogonal** projection sometimes look as if they were 2D. The possibility to use **Perspective** projection mode makes presentations look more realistic.

Perspective and orthogonal modes can be switched on and off from the VTK viewer toolbar.

Additionally it has become possible to manually define the projection parameters: **Camera position, Focal point and Camera orientation** (View up).



Perspective view

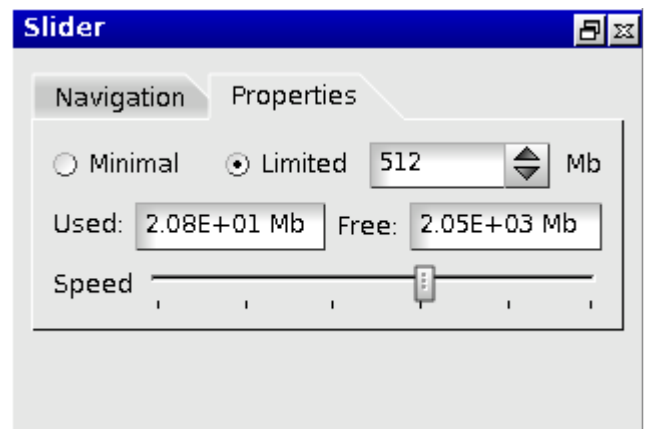
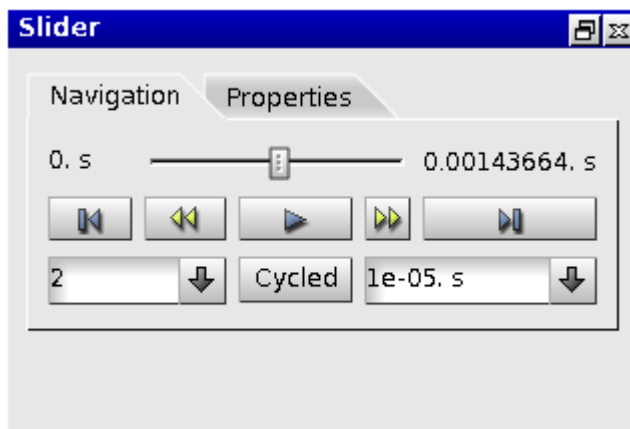


Orthogonal view

### Slider

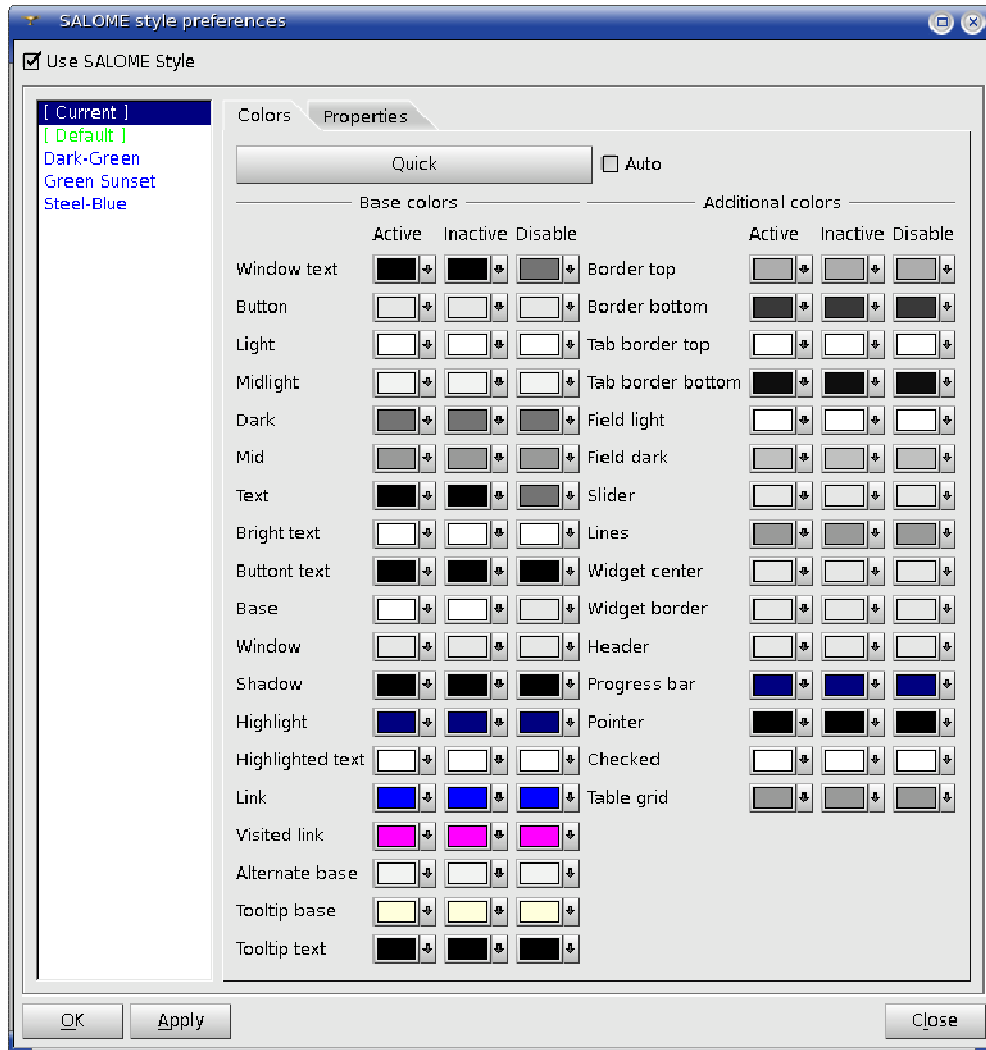
3D Cache system has been reworked and transformed into **Slider** functionality, which allows animating Post-Pro field presentations in the most simple and efficient way. The presentations compatible with this functionality can be synchronized (in terms of timestamps) and moved along the time axis to reproduce the corresponding physical process.

Slider also provides memory management possibilities, which allow the user to define a suitable compromise between the available computer resources and the navigation performance.





## Themes



### New Themes

functionality of the GUI module is used for customization of the look and feel of Salome.

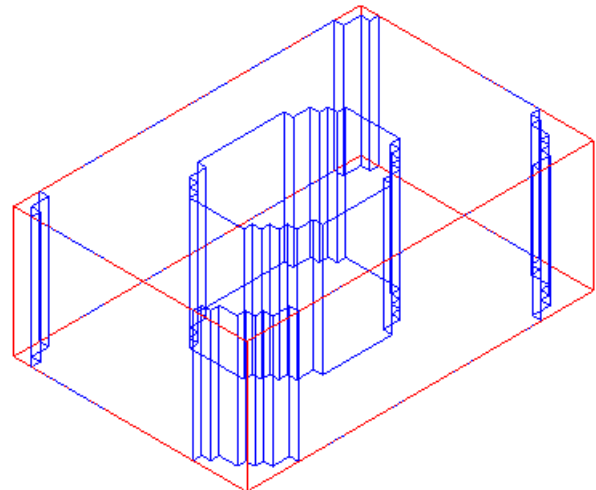
It is possible to choose a predefined color scheme or to edit a custom color scheme.

Other elements, such as menu fonts, widgets and handlers are also customizable.

## Feature Edges

This new Post-Pro module functionality highlights the contour of the object. To apply it, select the object in the VTK Viewer, right-click on it and choose in the context menu **Representation → Feature edges**.

It is possible to define the type of highlighted objects: feature, boundary, manifold or non-manifold edges.

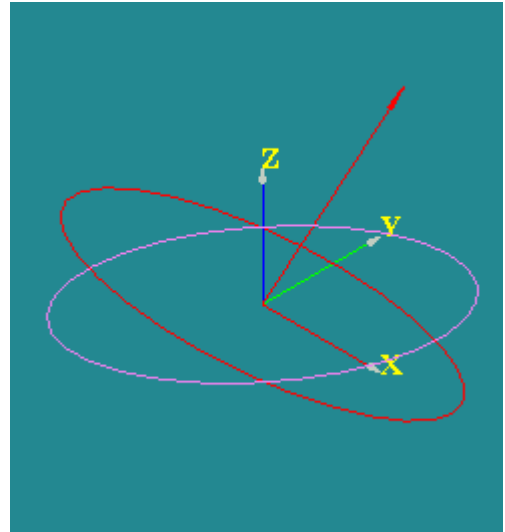


## Major Axis for ellipse creation

**Major Axis** vector parameter now can be used to define the direction of the major axis of an ellipse. By default, the major axis of an ellipse corresponds to the OX axis of the global coordinate system.

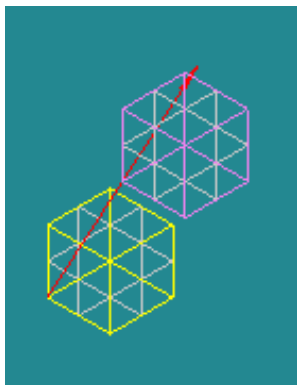
The **Major Axis** of an ellipse can be defined by various means:

- In the GUI Ellipse Construction dialog box in the input field "Major Axis" by selecting this field and an already constructed vector;
- In the TUI (geompyDC.py) using an additional parameter <theVecMaj> (none by default);
- In IDL interface: using the new function `GEOM_Object MakeEllipsevec(in GEOM_Object thePnt, in GEOM_Object theVec, in double theRMajor, in double theRMinor, GEOM_Object theVecMaj)`.



In the picture, the ellipse built using the default vector is red and the ellipse built using the custom vector is violet.

## New Modify the Location algorithm



The yellow box is translated along the red path. In this picture it is translated by half-length of the path.

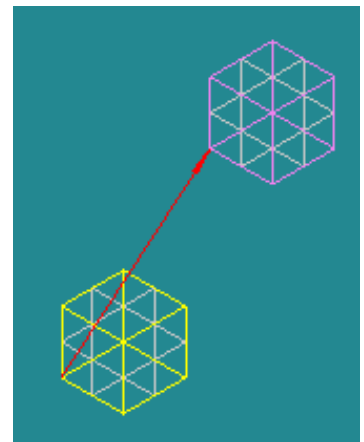
A new algorithm for **Modify the Location** operation has been implemented.

This algorithm modifies the location of an object using the **Path** object (Wire or Edge) and the **Distance** parameter (ranging from 0 to 1) defining how far the object will move along the path.

**Reverse** checkbox allows translating the object in the reversed direction.

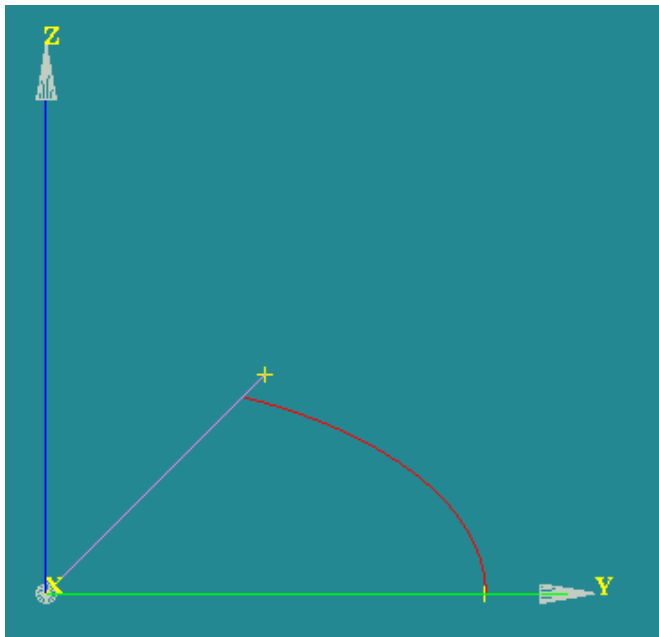
This operation is also available using the Python function:

```
PositionAlongPath(2GEOM_Obj
(aBaseObject, aPath), double
aDistance, bool aCopy).
```



And here it is translated by the full length.

## Arc of Ellipse



The method for construction of an **Arc of Ellipse** has been implemented.

For this, in the **Main Menu** select **New Entity** → **Basic** → **Arc** → **Third Radio Button**.

The **Arc** is built on the **Ellipse** that lies in the plane defined by the three points, with the center in the **Center Point**. The major radius of the **Ellipse** is defined by the distance between the **Center Point** and the most remote **Argument point**. The minor radius is defined by the shortest distance between the other **Argument point** and the major radius.

## Fillet 2D

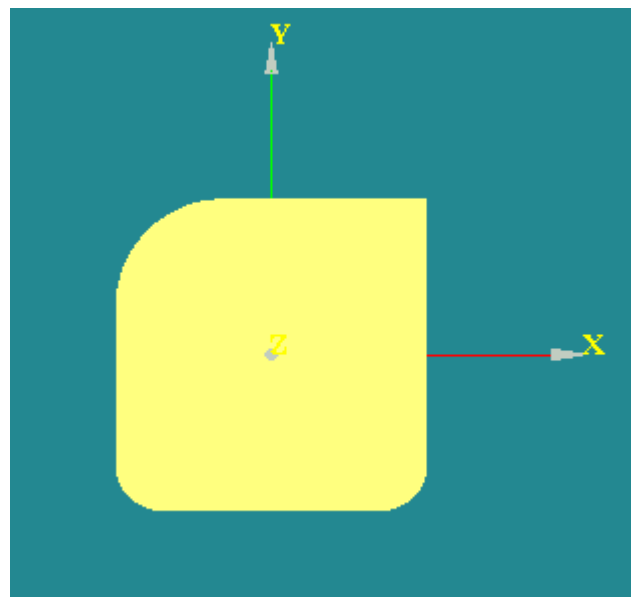
This new operation creates fillets on the corners of a **2D Planar Face**.

To produce a **Fillet 2D** in the **Main Menu** select **Operations** → **Transformation** → **Fillet 2D**

Define the **Planar Face** to create a fillet on, select the necessary vertices on this face in the OCC Viewer and define the **Radius** of the Fillet.

**Note:** This Operation Works for the **Planar 2D Faces** Only.

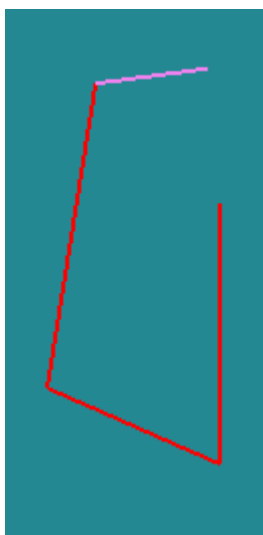
Alternatively, it is possible to use a **TUI Command**: `geompy.MakeFillet (Shape, R, ListVertexes)`



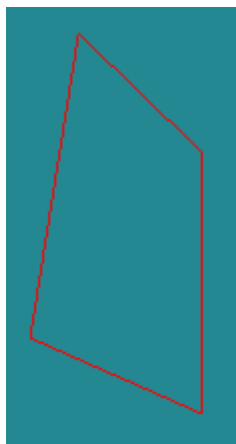
### 3D Sketcher

New 3D Sketcher is destined for the modeling of tubing lines with isometric sketches.

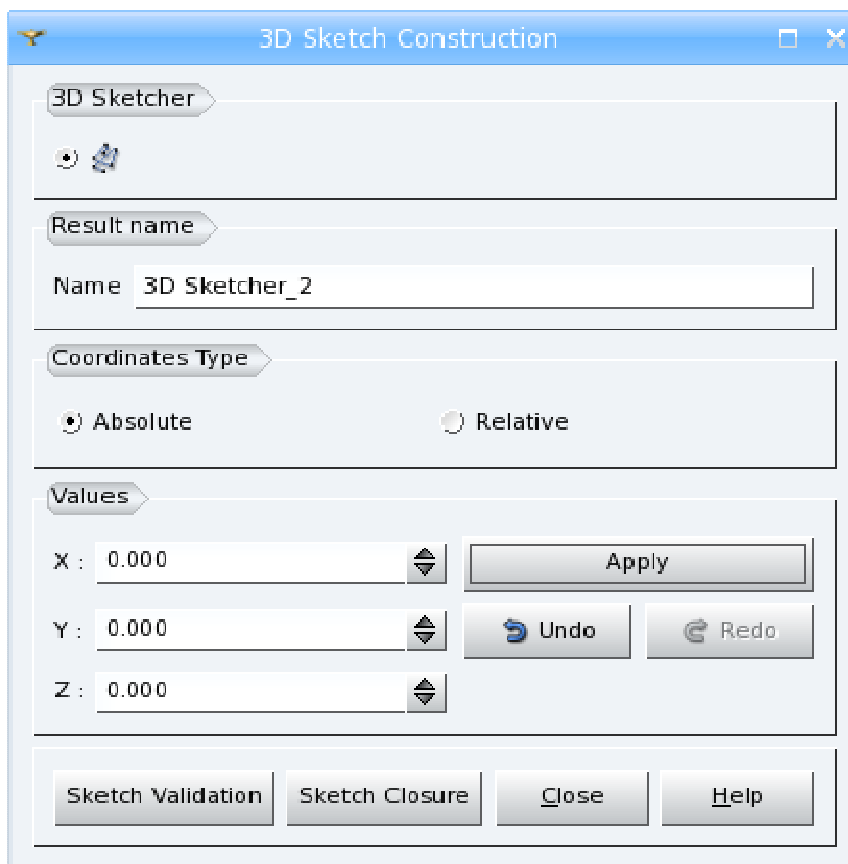
To activate this functionality, select in the main menu **New Entity** → **3D Sketch**.



In this picture the applied segments of the sketch are shown in red, the currently built one in violet.



At any time the sketch can be closed with Sketch Closure button



The position of each point can be defined by **Absolute coordinates X, Y, Z** or by **Relative coordinates DX, DY, DZ** with respect to the previous **Applied** point. The type of coordinates can be selected by the **Coordinates Type** radio buttons.

To add the point in the list of points and to proceed with the definition of the next point, click **Apply** button. **Undo and Redo** buttons, respectively, remove or restore the last point in the list.

**Sketch Validation** button applies the wire, built by the user, "as is". **Sketch Closure** closes the Sketch by a straight line from the start to the end point and applies it.

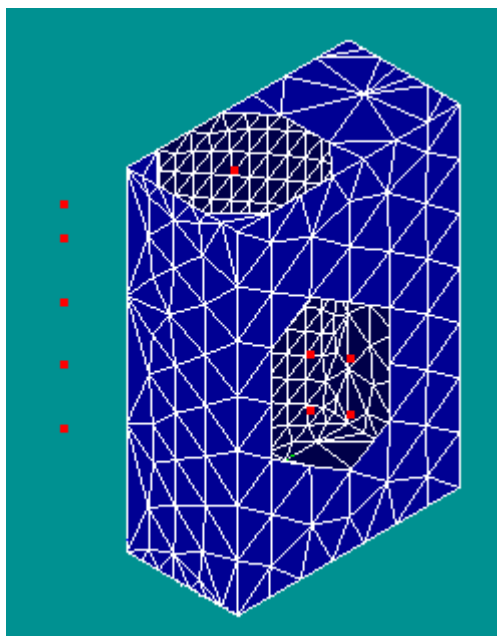
## New Tooltips in Geometry module

An improved tooltip has been implemented for GEOM module. In comparison with the previous implementation, the new tooltip:

- gives the name of the module to which the object belongs;
- does not display the name of the object (which is already known because it is displayed under the mouse cursor) and the entry (listed in the entry column in the object browser);
- adds a contextual attribute, for example the type of the object if such a concept exists for every object. For GEOM, it looks like "GEOM Module, <kindofshape>". The object type information is equal to the result of `geompy.KindOfShape()` and **Kind of Shape** field contents in main menu **Measures → What is** dialog box.

The tooltip structure for all other modules remains the same: Object <name>, module <module name>, ID=<entry>.

## New Quality Controls: Free Nodes and Free Faces

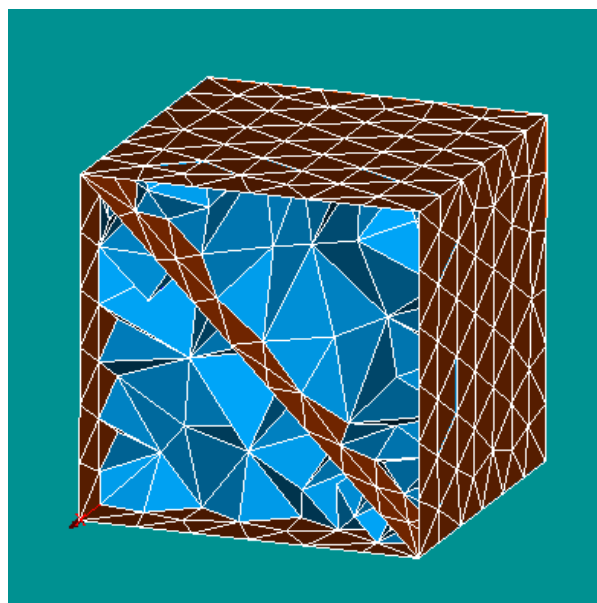


New quality control **Free Nodes** is available from main menu via **Controls → Free Nodes**.

This mesh quality control highlights the nodes which are not connected to any mesh element.

Also, "Free nodes" available as a filter and can be used in "Filter for Nodes" dialog (it concerns mesh group creation) or in the filter library in a usual way.

In the picture the free nodes are highlighted with red.



New quality control **Free Faces** is available from main menu via **Controls → Free Faces**.

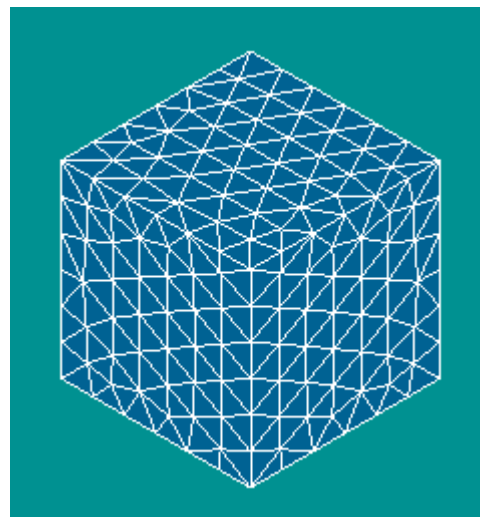
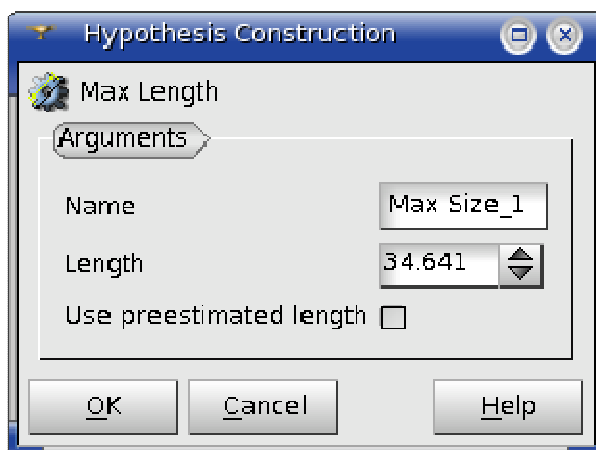
This mesh quality control highlights the faces connected to less than two mesh volume elements. The free faces are shown with the color different from the color of shared faces.

In the picture some volume mesh elements have been removed, as a result some faces became free.

## Max Size Hypothesis

**Max Size** hypothesis allows splitting geometrical edges into segments no longer than the given length. Definition of this hypothesis consists of setting the maximal allowed **length** of segments.

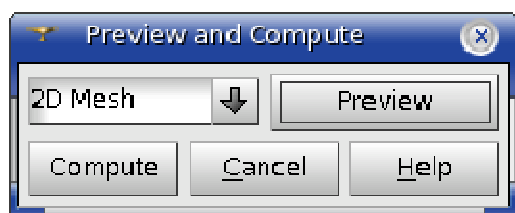
**Use pre-estimated length** check box allows calculating default **length** automatically basing on the size of the geometrical object, namely as a diagonal of its bounding box divided by ten. The divider can be changed via **Ratio Bounding Box Diagonal / Max Size** preference parameter. **Use pre-estimated length** check box is enabled only if the geometrical object has been selected before the definition of the hypothesis.



**Warning!** **Max Length** hypothesis is now used by **Automatic Tetrahedralization** algorithm instead of **Automatic Length** hypothesis.

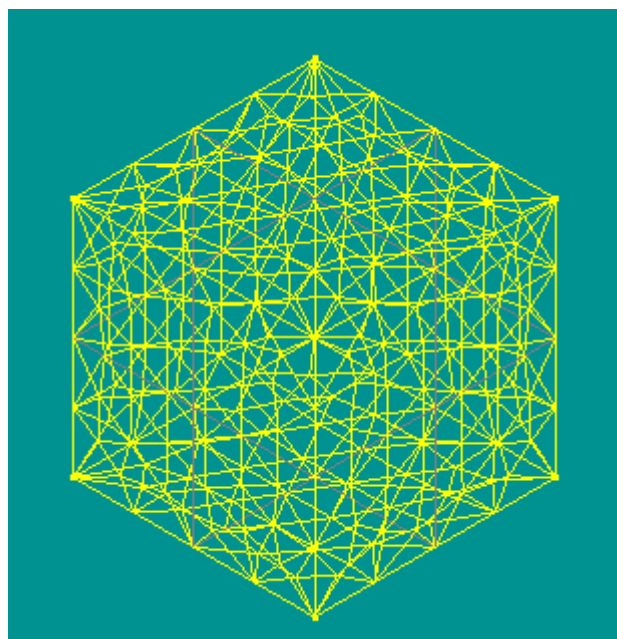
## Mesh Preview

**Mesh Preview** functionality has been added in Mesh module. It allows computing the 2D or 3D mesh step-by-step, seeing the intermediary results (1D and 2D elements).



It can be activated from the context menu on a non-computed mesh in the Object Browser.

**1D Mesh** preview shows nodes computed on geometry edges. **2D Mesh** preview shows edge mesh elements computed on geometry faces. **Compute** button finishes the computation of the whole mesh.

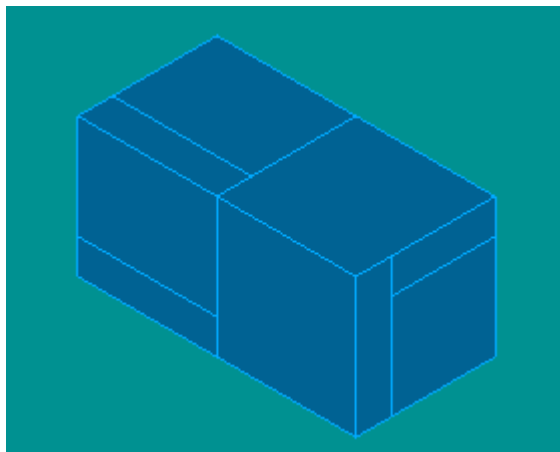


2D Preview of a meshed box.

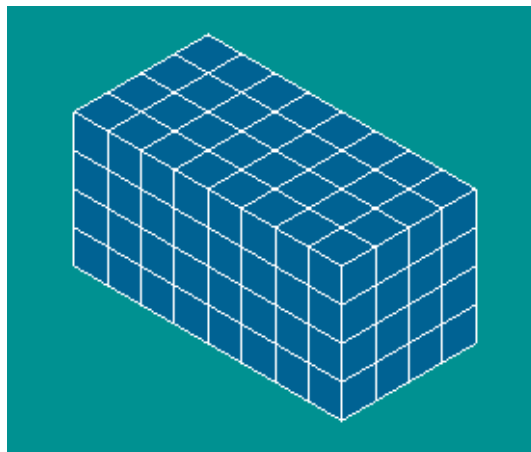
## Meshing of boxes with composite sides

The new functionality for meshing of boxes with composite sides, which uses **Hexahedron (i,j,k)** meshing algorithm has been introduced.

For this, 1D sub-meshes on opposite faces should contain the same number of segments.

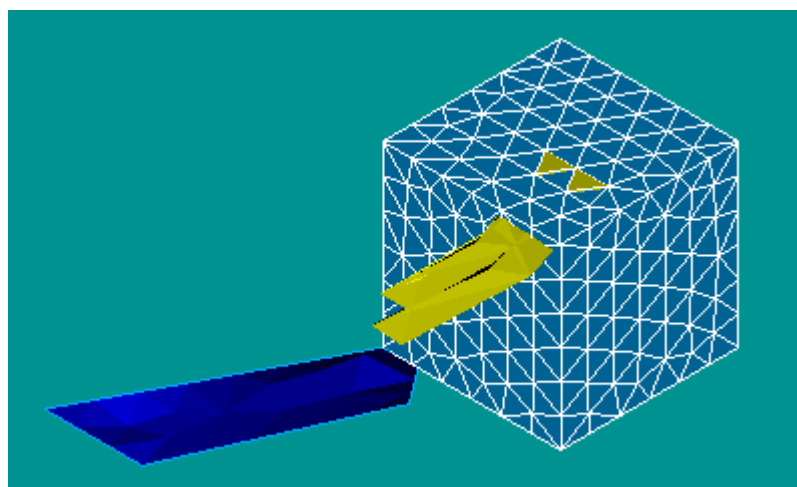


The initial box with composite edges.



Mesh on the box

## Extrusion and revolution with normal to a face



The vector for extrusion and revolution of mesh elements can be now also defined as the normal to another mesh face, selected in the viewer.

In the image: the rotated mesh cells and the results of rotation are yellow and the vector face is deep blue.

## Doubling of nodes

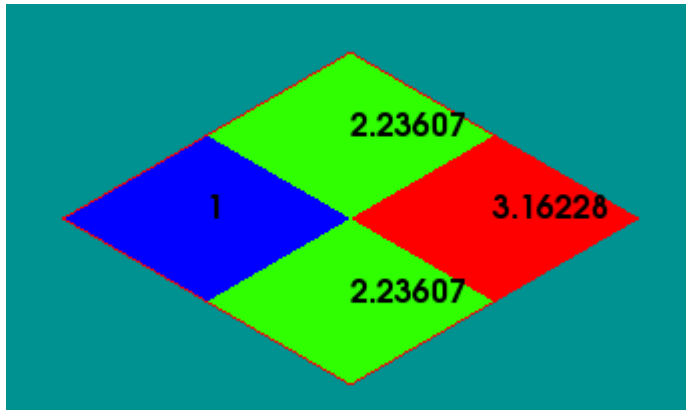
It has become possible to create a hole in the mesh by doubling the nodes of the selected elements using the methods of SMESH\_MeshEditor, which splits each node of the selected element into two coinciding nodes.

To perform this operation, specify a set of source nodes, which should be doubled, and a set of elements, which obtain new nodes instead of the source nodes. If the set of elements is not defined, the nodes are doubled but they are not assigned to elements.

The nodes can be specified by:

- one node – `def DoubleNode(self, theNodeId, theModifiedElems;`
- a list of nodes – `def DoubleNodes(self, theNodes, theModifiedElems);`
- a group of nodes – `def DoubleNodeGroup(self, theNodes, theModifiedElems);` or
- a list of groups of nodes – `def DoubleNodeGroups(self, theNodes, theModifiedElems)`

### Display Field Values

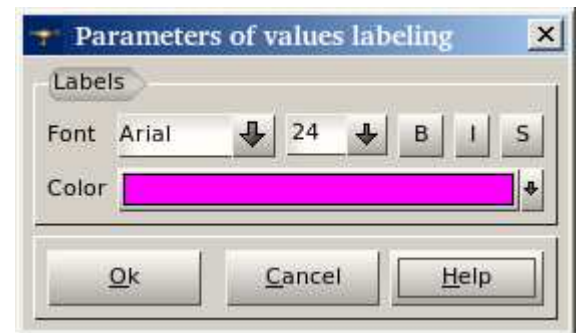


**Values Labeling** allows displaying values applied to the cells or nodes of a 3D presentation.

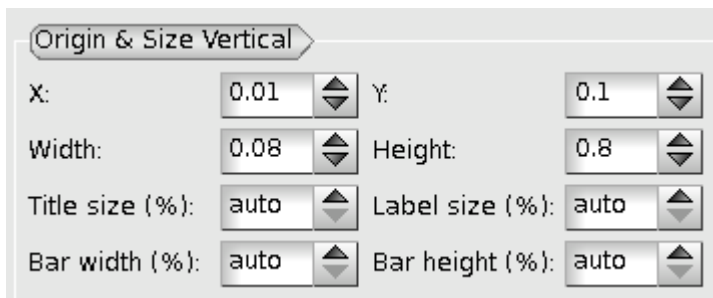
This visualization option can be selected by right-clicking on the presentation in the 3D viewer and choosing **Values labeling** command in the pop-up menu

**Labeling Parameters** that can be called from the same menu allow changing font and color of labels for the current presentation.

Default **labeling parameters** can be specified in the Post-pro Preferences



### Scalar Bar Parameters



Some of Scalar Bar parameters have been changed/

Parameters **Title width (%)** and **Title height (%)** have been replaced with one parameter **Title size (%)**, in % of the size of the viewer area.

Parameters **Labels width (%)** and **Labels height (%)** have been replaced with one parameter **Labels size (%)**, in % of the size of the viewer area.

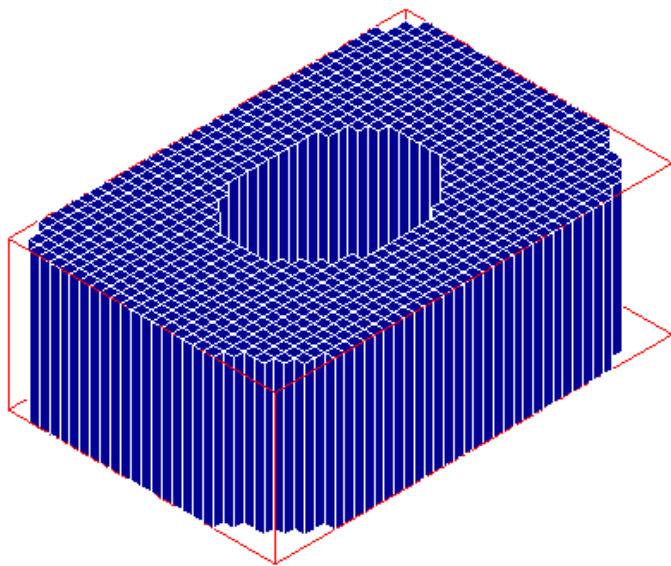
The label **Dimensions** of **Scalar bar properties** dialog is replaced with **Dimensions (in % of the size of view)**



The **Precision** of labels is now indicated as the integer value instead of the string value



## Surface Frame Mode



New **Surface Frame** mode has been introduced for visualization of all presentations except for Vectors, Cut lines and Stream lines.

This mode is similar to **Surface** mode, but it additionally highlights edges between cells.

## Other Improvements

- SMESH filters `BelongToGeom` and `LyingOnGeom` have been improved to work not only with sub-shapes.
- A special case has been introduced for building 3D meshes on shells: if the shell is not closed, building of a 3D mesh is disabled in Mesh GUI (the corresponding tab in the mesh construction dialog is not accessible), and in Mesh python interface (an attempt to add a 3D hypothesis to an opened shell will give an error).
- A curve now can be displayed in the Plot 2D browser using Show command in the object browser even if a PLOT2D scene has not been created. In this case, it will be created automatically and the selected curve will be displayed.
- The function `convertToPoly input_med_file output_med_file mesh_name` that translates a MED file in another MED file containing only poly-lines has been implemented in MED module.
- The source files for KERNEL and GUI modules have been corrected to allow compilation of the KERNEL module with cmake tool. Additionally, a set of scripts enabling multi-platform compilation of GUI module with the cmake tool has been integrated. These scripts generate cmake projects by conversion of the existing `Makefile.am` files. Cmake tool can be run to build SALOME GUI module on Linux and Windows platforms
- The configuration script checking Qt 4 availability has been improved to support native Qt 4 installation. If the Qt 4 is not found in the environment and not specified by `--with-qt` option it is searched in `/usr/lib/qt4` or `/usr/local/lib/qt4` folders.
- "Add Point on Edge" functionality now provides a preview of the created object.
- The message "Unable to create solid from unclosed shape" is now shown at the attempt to Create Solid from the unclosed object.
- The first available constructor is now selected by default for creation of: vector, working plane, box, cone, sphere, cylinder and torus.
- The item groups in the toolbar are now saved and restored correctly after module activation/deactivation.

- "Save/Restore GUI State" functionality has been implemented for the following parameters of GEOM objects in the OCC and the VTK viewers: color, transparency, display mode and number of isolines; and of SMESH objects in the VTK viewer: display mode, displayed entities, colors and sizes of entities, transparency and clipping.
- The behavior of Geometry GUI dialogs has been changed: now the selection focus is automatically switched to the next selection field when the current field is filled. The current selection field is marked with white background, the others are gray, and the selection button near the current field is set down.
- New **Edit Group as Standalone** operation allows converting an existing Group on geometry into a group of mesh elements. The resulting group loses reference to geometry and contains only the list of element IDs.
- New **Create Group of Underlying Entities** operation allows creating several groups of entities from the existing groups of superior dimension instead of only two groups (for example, groups of nodes from a group of faces).
- It has become possible to create a sub-mesh directly by right clicking on a Mesh and selecting **Create sub-Mesh** from the context menu.
- The eps image format is now supported by the Plot2d view. Dump to postscript format from VTK viewer has been enabled
- The IDL interface for the modifying of the **Shrink** parameter of the presentations in Post-Pro scripts has been implemented.
- Now the list of timestamps is not reset to the first timestamp after change of the type of Scalar Field in the **Deformed Shape and Scalar Map** dialog box.
- It has become possible to Hide or Show the **Scalar Bar** of the separate Post-Pro presentations.
- Python script generation mechanism has been improved significantly to produce shorter and more readable scripts.
- The build procedure for the SALOME modules which support an option to install GUI libraries has been modified so that availability of the GUI module is automatically detected.

## **BUG CORRECTIONS**

SALOME version 5.1.1 also provides a lot of bug fixes. Please contact OPEN CASCADE for full list of bugs fixed in version 5.1.1.

## **CASCADE 6.3 SERVICE PACK 5 BUG CORRECTIONS**

*Summary:* Convert Quadratic and Group on GEOM - OCC20329

Changes:

The problem of memory leaks made by OCC during partition has been fixed in OCC bug 20329.

*Summary:* Partition with a plane shows incoherent result - OCC20546

Changes:

Processing of edges on planes has been improved to prevent return value (True) when the basic surface is a plane.

*Summary:* Scale transform DXDYDZ of a sphere should give an ellipsoid - OCC20373

Changes

Problem of Shading on a b-spline sphere has been fixed.

## SUPPORTED LINUX DISTRIBUTIONS AND PRE-REQUISITES

**SALOME 5.1.1** supports Debian 3.1 Sarge, Debian 4.0 Etch 32bit and 64bit, Mandriva 2006 32bit and 64bit, Mandriva 2008 32bit and 64bit.

**SALOME 5.1.1** version has been mainly tested with the following pre-requisite list on Mandriva 2008 32bit and Debian 4.0 Etch 64bit platforms.

**SALOME 5.1.1** comes with the same prerequisites versions on all supported platforms. The table below lists the versions of the products used by SALOME platform. Other versions of the products can also work but it is not guaranteed.

**NOTE:** For some platforms Salome uses prerequisites with patches like in RPM and defines specific keys. So if you compile products without Install Wizard we strongly recommend you to check compilation keys using shell files located in `config_files` folder of the Installation Procedure.

	Version	GUI (IAPP)	KERNEL	GEOM	SMESH	VISU	MED	HexoticPLUGIN	GHS3DPRLPLUGIN	RANDOMIZER	SIERPINSKY	BLSURFPLUGIN	GHS3DPLUGIN	PYCALCULATOR	COMPONENT	NETGENPLUGIN	LIGHT	CALCULATOR	HELLO	PYHELLO	YACS	xdata	HXX2SALOME
gcc*	3.3.5**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
automake*	1.9**	X	X	X	X	X	X			X	X		X	X	X	X	X	X	X	X	X	X	X
autoconf*	2.59**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
libtool*	1.5.6**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
GNU make*	3.80**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Python	2.4.4	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
Qt	4.4.3	X		X	X	X	X	X	X		X	X	X		X	X	X	X	X	X	X	X	X
Sip	4.7.7	X													X								
PyQt	4.4.3	X													X							X	
Boost	1.34.1	X	X	X	X	X	X	X	X		X	X	X			X		X	X				
Swig	1.3.31	X	X	X	X	X	X	X	X		X	X	X		X	X		X			X		
OpenCASCADE Technology	6.3 sp5	X		X	X	X	X	X	X		X	X	X		X	X	X	X	X		X		
Qwt	5.1.1	X													X								
QScintilla	2.3																				X		
OmniORB	4.1.3																						
OmniORBpy	3.3	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	
omniNotify	2.1																						
Hdf5	1.6.4	X	X	X	X	X	X	X	X		X	X	X		X	X	X	X					
Med	2.3.5				X	X	X		X		X			X	X	X		X					
Vtk	5.0.0	X		X	X	X	X	X	X		X	X	X		X	X	X					X	
Numeric	24.2		X																				
Graphviz	2.16.1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X				X		
Doxygen	1.5.6	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X				X		
NETGEN	4.5															X							
docutils	0.4.0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X				X		
metis	4.0						X																
scotch	4.0						X																
libxml2	2.6.27	X	X				X															X	
elementtree	1.2.6																					X	
cElementTree	1.0.5																					X	
blsurf	2.7.12											X											
TetMesh-GHS3D	4.0								X				X										
TclTk	8.4.14																						
Sphinx	0.5																				X		

\*) Not included into SALOME Installation procedure

\*\*\*) Minimal required version

SALOME 5.1.1 depends of a number of products for run time execution, others are necessary only for compilation or generation of development documentation (like doxygen for example). Below there is a list of mandatory and optional products.

### Software Requirements

	Compilation and Development		Execution		Remarks
	Mandatory	Optional	Mandatory	Optional	
gcc	X		X		
Automake	X				
Autoconf	X				
libtool	X				
GNU make	X				
Tcltk					for OCCT compilation from source files only
Python	X		X		
Qt	X		X		
sip	X				
PyQt	X		X		
Boost	X		X		
Swig	X				
OpenCASCADE Technology	X		X		
Qwt	X		X		
QScintilla		X		X	
OmniORB	X		X		
Hdf	X		X		
Med	X		X		
Vtk	X		X		
Numeric		X			
Graphviz		X			
Doxygen		X			
NETGEN	X				for NETGENPLUGIN mesh plug-in only
docutils		X			
cppunit		X			
mpi		X		X	required only if used at compilation step
openpbs		X		X	required only if used at compilation step
Lsf		X		X	required only if used at compilation step
metis	X		X		
scotch	X		X		
libxml2	X				
elementtree	X				
cElementTree	X				
blsurf	X		X		for BLSURFPLUGIN mesh plug-in only
TetMesh-GHS3D			X		for GHS3DPLUGIN mesh plug-in only
Sphinx				X	for YACS documentation generation only

## HOW TO INSTALL AND BUILD SALOME

Please follow README file from Installation Wizard for processing correctly installation of SALOME and all prerequisites.

If you would like to compile SALOME from scratch, please use `build.csh` or `build.sh` script delivered with the Installation Wizard. Call "`build.csh -h`" to see available options of this script.

## SALOME SYSTEM REQUIREMENTS

### Minimal Configuration:

- Processor: Pentium IV.
- 512 Mb RAM.
- Hard Drive Space: 1.5 Gb.
- Video card 64mb.
- CD/DVD

### Optimal Configuration:

- Processor: Dual Core.
- 2 Gb RAM.
- Hard Drive Space: 5Gb.
- 2Gb Swap.
- Video card 128mb.
- CD/DVD

## HOW TO GET THE VERSION AND PRE-REQUISITES

**SALOME 5.1.1** pre-compiled binaries for Linux Debian 3.1 Sarge, Mandriva 2006 (32bit and 64bit), Mandriva 2008 (32bit and 64bit), Debian 4.0 Etch (32bit and 64bit) can be retrieved from the <http://www.salome-platform.org> site.

Besides it is possible to install Debian 3.1 Sarge Salome universal binaries on the platforms that are not supported officially.

The SALOME Installation procedure includes SALOME modules sources, and it is possible to build sources from scratch using `build.sh` or `build.sch` script coming with installation procedure.

There are two patches on **NETGEN** which are placed inside NETGENPLUGIN modules sources. The first patch file is used for all 32 bit platforms; the second patch file is an addition to the first one and should be applied only for 64bit platforms.

During the compilation on NETGEN from sources by SALOME Installation Wizard, the patches are applied automatically to the standard NETGEN distribution. You can download NETGEN 4.5 from its official site using the following link: <http://www.hpfem.jku.at/netgen>.

All other pre-requisites can be obtained either from your Linux distribution (please be sure to use a compatible version) or from the distributors of these pre-requisites ([www.trolltech.com](http://www.trolltech.com) for Qt for example).

## KNOWN PROBLEMS AND LIMITATIONS

- The following modules have not been migrated to Qt series 4 and thus are not included into SALOME 5.1.1 release: MULTIPR, FILTER and SUPERV.
- The store/restore GUI state functionality is not fully supported in SALOME version 5.1.1. There are some bugs with this functionality which are planned to be fixed in the future releases of SALOME platform.
- The following limitations refer to BLSURF plug-in:
  - Mesh contains inverted elements, if it is based on a shape, consisting of more than one face (box, cone, torus...) and if the option "Allow Quadrangles (Test)" has been checked before computation.
  - SIGFPE exception is raised after trying to compute a mesh based on a box with "Patch independent" option checked.
  - It has been found out that BLSURF algorithm can't be used as a local algorithm (on sub-meshes) and as a provider of low-level mesh for some 3D algorithms because BLSURF mesher (and, consequently, the plug-in) does not provide information on node parameters on edges (U) and faces (U, V). For example, the following combinations are impossible:
    - global MEFISTO or Quadrangle(mapping) + local BLSURF;
    - BLSURF + Projection 2D from faces meshed by BLSURF;
    - local BLSURF + Extrusion 3D.
- Sometimes regression test bases give unstable results; in this case the testing should be restarted.
- A native VTK can be used only after manual recompilation with the GL2PS component.
- NETGEN 1D-2D and 1D-2D-3D algorithm do not require definition of 2D and 1D algorithms and hypotheses for both mesh and sub-mesh. 2D and 1D algorithms and hypotheses defined with NETGEN 1D-2D or 1D-2D-3D algorithm will be ignored during calculation.
- SALOME supports reading of documents from earlier versions but the documents created in the new version may not open in earlier ones.
- If SALOME modules are not installed in a single folder, SALOME may not work in the CSH shell since the environment variables are too long by default. In this case, it is suggested to use SH or to install all modules in the same folder.
- During the compilation of OCT 6.x by makefiles on a station with NVIDIA video card you can experience problems because the installation procedure of NVIDIA video driver removes library libGL.so included in package libMesaGL from directory /usr/X11R6/lib and places this library libGL.so in directory /usr/lib. However, libtool expects to find the library in directory /usr/X11R6/lib, which causes compilation failure (See /usr/X11R6/lib/libGLU.la). We suggest making symbolic links in that case: "ln -s /usr/lib/libGL.so /usr/X11R6/lib/libGL.so ln -s /usr/lib/libGL.la /usr/X11R6/lib/libGL.la".
- VISU module does not support timestamps defined on the same field but on different meshes
- Stream lines presentation can not be built on some MED fields due to limitations in VTK.
- MEFISTO algorithm sometimes produces different results on different platforms.
- In some cases the number of triangles generated by MEFISTO may be different at each attempt of building the mesh.