

# SALOME 6.7.0

## Minor release announcement

July 2013



### GENERAL INFORMATION

CEA/DEN, EDF R&D and OPEN CASCADE are pleased to announce [SALOME](#) version [6.7.0](#). It is a minor release that contains the results of planned major and minor improvements and bug fixes against SALOME version 6.6.0 released in December 2012.

## Table of Contents

- **GENERAL INFORMATION .....1**
- **NEW FEATURES AND IMPROVEMENTS .....3**
  - PREREQUISITES .....3
  - LICENSE RESTRICTIONS .....4
  - MED MODULE CHANGE LOG .....5
    - Features* .....5
    - API modifications*.....5
    - Behavior modifications* .....5
    - Bugs correction* .....6
    - Performance improvements* .....6
  - YACSGEN CHANGES .....6
- **BUG CORRECTIONS.....7**
  - GEOM MODULE .....7
  - SMESH MODULE .....7
  - MED MODULE .....7
- **SUPPORTED LINUX DISTRIBUTIONS AND PRE-REQUISITES .....8**
- **HOW TO INSTALL AND BUILD SALOME .....11**
- **SALOME SYSTEM REQUIREMENTS .....11**
- **HOW TO GET THE VERSION AND PRE-REQUISITES .....11**
- **KNOWN PROBLEMS AND LIMITATIONS .....12**



## NEW FEATURES AND IMPROVEMENTS

### PREREQUISITES

SALOME 6.7.0 is based on the same pre-requisite products as previous version 6.6.0. The table below provides the full list of pre-requisite products used with SALOME 6.7.0.

Product	Version
Boost	1.49.0
Cgns	3.1.3-4
CMake	2.8.7
Docutils	0.8.1
Doxygen	1.8.0 <sup>1</sup>
Expat	2.0.1
Freeimage	3.14.1
freetype	2.4.10
Ftgl	2.1.3-rc5
gl2ps	1.3.5
Graphviz	2.28.0
HDF5	1.8.8
Homard	10.5
Intel® Threading Building Blocks	3.0
Jinja2	2.6
LAPACK	3.3.0
libBatch	1.6.0
Libxml2	2.7.8 <sup>2</sup>
Med	3.0.6
METIS	4.0
NETGEN	4.9.13
NumPy	1.5.1
omniORB	4.1.6
omniORBpy	3.6
omniNotify	2.1
Open CASCADE Technology	6.5.4
ParaView	3.14.0 <sup>3</sup>
Pygments	1.5

<sup>1</sup> Patched for SALOME (bugs in doc tree javascript)

<sup>2</sup> Patched for SALOME (bug for 64bits platforms)

<sup>3</sup> Patched (fix problems in build procedure with plug-ins and documentation)

Product	Version
PyQt	4.9.1 <sup>4</sup>
Python	2.6.6
QScintilla	2.6.1 <sup>5</sup>
Qt	4.6.3
Qwt	5.2.1
Scotch	5.1.11
Setuptools	0.6c11
SIP	4.13.2
Sphinx	1.1.3
SWIG	1.3.40 <sup>6</sup>
Tcl	8.5.8
Tk	8.5.8
TclX	8.4.0
VTK <sup>7</sup>	5.9.0
Xdata	0.9.6
Distene Blsurf <sup>8</sup>	MeshGems suite v1.0.1
Distene TetMesh-GHS3D <sup>9</sup>	MeshGems suite v1.0.1
Distene Hexotic <sup>10</sup>	1.0

For additional information about pre-requisite products and SALOME modules dependencies refer to the paragraph “[Supported Linux distributions and pre-requisites](#)” below.

#### LICENSE RESTRICTIONS

- Hereby we explicitly declare that PyQt and QScintilla (by Riverbank Computing Ltd) used by SALOME are distributed under the terms of GPL license, for more details please refer to:
  - <http://www.riverbankcomputing.com/software/pyqt/license>
  - <http://www.riverbankcomputing.com/software/qscintilla/license>

If you plan using SALOME for commercial usage please consider obtaining a commercial license for PyQt and/or QScintilla.

<sup>4</sup> Patched (fix issues of incompatibility with Qt 4.6.3)

<sup>5</sup> Patched (fix issues of incompatibility with Qt 4.6.3)

<sup>6</sup> SWIG 1.3.40 does not compile at latest versions of gcc (4.6); instead version 2.0.4 can be used

<sup>7</sup> Included to the ParaView distribution

<sup>8</sup> Commercial product, requires license for using in runtime

<sup>9</sup> Commercial product, requires license for using in runtime

<sup>10</sup> Commercial product, requires license for using in runtime

**MED MODULE CHANGE LOG****Features**

- `numpy` bindings without any copy from/to `DataArrayInt` and `DataArrayDouble`.
- `push/pop` in `DataArrayDouble/DataArrayInt`.
- `MEDCouplingUMesh::distanceToPoint`, `MEDCouplingUMesh::distanceToPoints` to know the distance and the location of entity of a mesh to a point.
- `CurveLinearMesh` mesh added (with IO) for non-cartesian structured meshes.
- IO on fields of integers32 (`MEDFileIntFieldITS`, `MEDFileIntFieldMultiITS`) without overhead.
- Integral computation on `GaussNodeElement` fields.
- In `MEDCouplingRemapper` interpolation `GaussPoint` → `GaussPoint`.
- `CaseReader/CaseWriter` (to/from ensight format files).
- `__pow__` in `DataArrayDouble` and `MEDCouplingFieldDouble`.
- `DataArrayDouble::minimalDistanceTo`.
- `DataArrayDouble::findClosestTupleId`.
- `MEDFileUMesh::addGroup`.
- `deepCpy` on `MEDFile*` objects.
- `MEDCouplingMesh::computeIsoBarycenterOfNodesPerCell`.
- `getHeapMemorySize` on all big objects to know the memory weight.
- I/O of `univname` in `MEDFileMesh`.
- `MEDFileMesh` class: `removeOrphanGroups`, `removeOrphanFamilies`, `removeEmptyGroups`, `computeAllFamilyIdsInUse`.

**API modifications**

- C++/Python API `MEDCouplingUMesh::zipConnectivityTraducer`.
- C++ API `MEDCouplingUMesh::findCommonCellsBase`.
- C++/Python `MEDCouplingFieldDouble::getWeightedAverageValue`.
- `MEDFileUMesh::setMeshAtLevelGen` suppressed (replaced by `MEDFileUMesh::setMeshAtLevel`).

**Behavior modifications**

- Family ids follow the MED file rules.
- `ONE_TIME` is used by default as time discretization instead of `NO_TIME` for `MEDCouplingFieldDouble`. So all methods generating such fields will have `ONE_TIME` as time discretization.
- Computation of center of mass of quadratic cells `SEG3`, `QUAD8`, `TRI6`, `QPOLYG` is OK now.
- Default `SplittingPolicy` for `HEXA8` is `PLANAR_FACE_5` for interpolators.
- `DataArrayInt.BuildUnion` `DataArrayInt.buildUnion`, `DataArrayInt.BuildIntersection` `DataArrayInt.buildIntersection` no more throw an exception for negative value.
- All chunks of data are allocated using `malloc` instead `new[]` to ease the exchange without copy with `numpy`.

### Bugs correction

- MEDCouplingUMesh::Intersect2DMeshes.
- MEDFileField\*.getFieldOnMeshAtLevel on partial fields on nodes.
- MEDCouplingPointSet::rotate in python bindings.
- MEDFileMesh::normalizeFamIdsMEDFile.
- MEDFileMesh::changeGroupName.
- MEDFileFields::changeMeshNames.
- MEDCouplingUMesh::orientCorrectlyPolyhedrons.
- MEDCouplingUMesh::ComputeSpreadZoneGradually.

### Performance improvements

- MEDCouplingUMesh::zipConnectivityTraducer.
- MEDCouplingUMesh::buildDescendingConnectivity.
- DataArrayInt::CheckAndPreparePermutation.
- MEDCouplingUMesh::renumberCells.

### YACSGEN CHANGES

- Added missing wrappings for MEDCoupling API.



## BUG CORRECTIONS

### GEOM MODULE

22237	<p><i>Summary:</i> Study dump produces a Python code containing syntax errors</p> <p>Fixed an error with wrong order of Python commands in the dump script, caused by incorrect processing of <code>GetVertexNearPoint</code> function.</p>
-------	---

### SMESH MODULE

22163	<p><i>Summary:</i> SMESH produces study dump with syntax errors</p> <p>The problem of producing an invalid dump script due to switching on/off of publishing of SMESH objects, so that published objects depend on not published ones, has been fixed.</p>
22238	<p><i>Summary:</i> Study dump produced by SMESH refers to undefined variable names</p> <p>A problem in Python dump for removed Mesh objects has been fixed, namely <code>aStudyBuilder.removeObjectWithChildren(SO)</code> is no more dumped if a Python command creating SO is commented or removed.</p>

### MED MODULE

22211	<p><i>Summary:</i> [CEA 810] Error "NULL field support" when opening sauv files with MEDLoader.</p> <p>A bug of splitting a SAUV field into several MED fields has been fixed.</p>
-------	--



## SUPPORTED LINUX DISTRIBUTIONS AND PRE-REQUISITES

**SALOME 6.7.0** supports Linux Debian Squeeze 6.0 64bits, Mandriva 2008 32bits and 64bits, Mandriva 2010 32bits and 64bits, CentOS 5.5 64 bits. **SALOME 6.7.0** version has been mainly tested with below listed pre-requisites on Mandriva 2010 32bits and Debian 6.0 Squeeze 64bits platforms.

**SALOME 6.7.0** comes with the same prerequisites versions on all supported platforms (with some exceptions). 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.

Product	Version	GUI (IAPP)	KERNEL	GEOM	SMESH	VISU	MED	YACS	PARAVIS	HOMARD	HEXABLOCK	NETGENPLUGIN	GHS3DPLUGIN	GHS3DPRPLUGIN	BLSURFPLUGIN	HexoticPLUGIN	HEXABLOCKPLUGIN
gcc*	4.2**	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
autoconf*	2.59**	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
GNU make*	3.80**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Cmake	2.8.7								X								
Python	2.6.6	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Qt	4.6.3	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X
Sip	4.13.2	X			X												
PyQt	4.9.1	X			X												
Boost	1.49.0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Swig	1.3.40	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X
OCCT	6.5.4	X		X	X	X	X		X	X	X	X	X	X	X	X	X
Qwt	5.2.1	X			X												
QScintilla	2.6.1							X									
OmniORB	4.1.6	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
OmniORBpy	3.6	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
omniNotify	2.1		X														
Hdf5	1.8.8	X	X	X	X	X	X		X	X		X	X	X	X	X	X
Med	3.0.6				X	X	X		X	X		X		X			
Vtk	5.8.0	X		X	X	X	X		X		X	X	X	X	X	X	X
numpy	1.5.1		X														
lapack	3.3.0		X														
graphviz	2.28.0	X	X	X	X	X	X	X				X	X	X	X	X	
Doxygen	1.8.0	X	X	X	X	X	X	X				X	X	X	X	X	X
NETGEN	4.9.13											X					
docutils	0.8.1	X	X	X	X	X	X	X		X	X	X	X	X	X	X	
metis	4.0						X										
scotch	5.1.11						X										
libxml2	2.7.8	X	X		X		X	X	X		X		X				
Distene MeshGems	1.0.1												X	X	X		
Hexotic	1.0														X	X	
sphinx	1.1.3		X	X	X			X		X	X						
expat	2.0.1							X									
libBatch	1.6.0		X														
cgns	3.1.3				X												
ParaView	3.14.0								X								
Homard	10.5									X							

\*) Not included into SALOME Installation procedure

\*\*\*) Minimal required version



Product	Version	RANDOMIZER	SIERPINSKY	PYCALCULATOR	COMPONENT	CALCULATOR	HELLO	PYHELLO	LIGHT	PYLIGHT	ATOMIC	ATOMGEN	ATOMSOLV	HXX2SALOME	YACSGEN	JOBMANAGER
gcc*	4.2**	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
autoconf*	2.59**	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
GNU make*	3.80**	X	X	X	X	X	X	X	X	X	X	X	X	X		X
Python	2.6.6	X	X	X	X	X	X	X	X	X	X	X	X		X	X
Qt	4.6.3		X			X	X	X	X		X	X	X	X		X
Sip	4.13.2											X				
PyQt	4.9.1								X		X					
Boost	1.49.0		X			X	X						X			X
Swig	1.3.40		X		X	X										
OCCT	6.5.4		X		X	X	X		X		X		X			
Qwt	5.2.1															
OmniORB	4.1.6	X	X	X	X	X	X	X				X	X			X
OmniORBpy	3.6	X	X	X	X	X	X	X				X	X			X
Hdf5	1.8.8		X		X	X			X		X					
Med	3.0.6		X	X	X	X										
Vtk	5.8.0		X						X	X	X		X			
graphviz	2.28.0	X	X	X	X		X	X			X					
Doxygen	1.8.0	X	X	X	X		X	X			X					
sphinx	1.1.3															X

\*) Not included into SALOME Installation procedure

\*\*) Minimal required version

The following products are not used in SALOME directly; they are only required to build other pre-requisite products.

Product	Version	Required by	Comment
tcl	8.5.8	Open CASCADE Technology, VTK	Optional
tk	8.5.8	Open CASCADE Technology, VTK	Optional
tclX	8.4.0	Open CASCADE Technology	Optional
jinja	2.6	Sphinx	
pygments	1.5	Sphinx	
setuptools	0.6c11	Sphinx	
freetype	2.4.10	Open CASCADE Technology	
ftgl	2.1.3	Open CASCADE Technology	
freeimage	3.14.1	Open CASCADE Technology	Optional
gl2ps	1.3.5	Open CASCADE Technology	Optional
Intel TBB	3.0	Open CASCADE Technology	Optional
xdata	0.9.6		Can be used to create 3 <sup>rd</sup> - party SALOME modules

**NOTE:** For some platforms SALOME uses prerequisites with patches (to fix different problems, like it is done in RPM) and defines specific configuration/compilation options. If you compile products without the Install Wizard we strongly recommend you to check configuration/compilation options using shell scripts located in config\_files folder of the SALOME Installation Wizard.

SALOME 6.7.0 depends on 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.

Product	Compilation and Development		Execution		Remarks
	Mandatory	Optional	Mandatory	Optional	
gcc	X		X		
Automake	X				Except for PARAVIS
Autoconf	X				Except for PARAVIS
libtool	X				Except for PARAVIS
GNU make	X				
cmake	X				For PARAVIS only
Python	X		X		
Qt	X		X		
sip	X				
PyQt	X		X		
Boost	X		X		
Swig	X				
OCCT	X		X		
Qwt	X		X		
QScintilla		X		X	For YACS only Required only if used at compilation step
omniORB	X		X		
omniORBpy	X				
omniNotify	X		X		
Hdf	X		X		
Med	X		X		
Vtk	X		X		
numpy + lapack		X			
graphviz	X		X		In run-time required for YACS only
Doxygen	X				
NETGEN	X		X		For NETGENPLUGIN only
docutils		X			
cppunit		X			Used for unitary testing
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	Required only if used at compilation step
scotch		X		X	Required only if used at compilation step
libxml2	X		X		
Blsurf*	X	X	X	X	mandatory for BLSURF plugin, optional for Hexotic plugin
TetMesh-GHS3D*	X		X		For GHS3D and GHS3DPRL plugins only
Hexotic			X		For HexoticPLUGIN mesh only
sphinx		X			
expat	X		X		For YACS only
libBatch		X		X	Required only if used at compilation step
ParaView	X		X		For PARAVIS module only
Homard			X		For HOMARD module only
cgns		X		X	For SMESH only Required only if used at compilation step
freetype		X		X	Required only if used when building OCCT
ftgl		X		X	Required only if used when building OCCT
freeimage		X		X	Required only if used when building OCCT
gl2ps		X		X	Required only if used when building OCCT
Intel TBB		X		X	Required only if used when building OCCT

\*) Included to the Distene MeshGems suite v1.0.1 package



## HOW TO INSTALL AND BUILD SALOME

- **Linux**

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.sh -h`" to see all available options of this script.

- **Windows**

For Windows platform SALOME is distributed in form of ZIP archive. To install SALOME on Windows, unpack the archive and follow instructions listed in the enclosed README file.



## 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 6.7.0** pre-compiled binaries for Linux Mandriva 2008 (32bits and 64bits), Mandriva 2010 (32bits and 64bits), Debian 6.0 Squeeze 64bits, CentOS 5.5 64bits can be retrieved from the <ftp://ftp.opencascade.com> repository or from the SALOME web site <http://www.salome-platform.org>.

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

SALOME Installation procedure for Linux includes a patch for **NETGEN** which is placed inside NETGENPLUGIN modules sources. This patch is used for all platforms to fix several bugs of NETGEN. During the compilation on NETGEN from sources by the SALOME Installation Wizard, the patch is applied automatically to the standard NETGEN distribution. You can download NETGEN 4.9.13 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 (for example, <http://qt.nokia.com> for Qt). Note that for some of pre-requisite products SALOME Installation procedure also includes patches that fix the problems detected by SALOME.



## KNOWN PROBLEMS AND LIMITATIONS

- The following modules have not been migrated to Qt series 4 and thus are not included into SALOME 6.7.0 release: FILTER, SUPERV, MULTIPR. These modules are considered obsolete and not supported anymore.
- Application crash might occur on the data publication in the study if both data server and CPP container are running in the standalone mode.
- On some platforms the default font settings used in SALOME might cause bad application look-n-feel. This problem can be solved by changing the font settings with `qtconfig` utility included into the distribution of Qt 4.
- 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 using the following commands:
 

```
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.

- When generating a 2D mesh with “Maximum Area” hypothesis used, MEFISTO algorithm can produce cells with maximum area larger than specified by the hypothesis.
- For the current moment, because of the ParaView application architecture limitations, PARAVIS module has the following known limitations:
  - PARAVIS is a “singleton” module: that means that it can be used within one study only. As soon as the user activates the PARAVIS in a study, this module becomes unavailable in other studies.
  - PARAVIS module works unstably using a remote connection; when SALOME is running on a remote computer, activation of PARAVIS module can sometimes lead to the application hang-up.
  - PARAVIS module compilation can fail on 64bits platforms when building ParaMEDCorba plugin (due to crash of kwProcessXML tool during generation of the plugin’s documentation). In such a case it is necessary to unset the VTK\_AUTOLOAD\_PATH environment variable and restart compilation, for example:

```
[bash%] unset VTK_AUTOLOAD_PATH
```