Interoperability for Space Environment Analysis Tools

First results and feedbacks

J.Forest[2]
B.Jeanty-Ruard[1]
A.Trouche[1]

J.Pervez[3]
K.Bunte[3]

F.Cipriani[4]
D.Rodgers[4]

(1) Artenum Toulouse
(2) Artenum Paris

(3) etamax

(4) ESA/ESTEC/TEC-EES

SPINE meeting, Toulouse, France
March, 2015

- **Multi-physics / multi-models simulations require to exchanges numerous and various data**
  - Geometrical models
  - Materials and numerical properties
  - Global settings
    - Environment
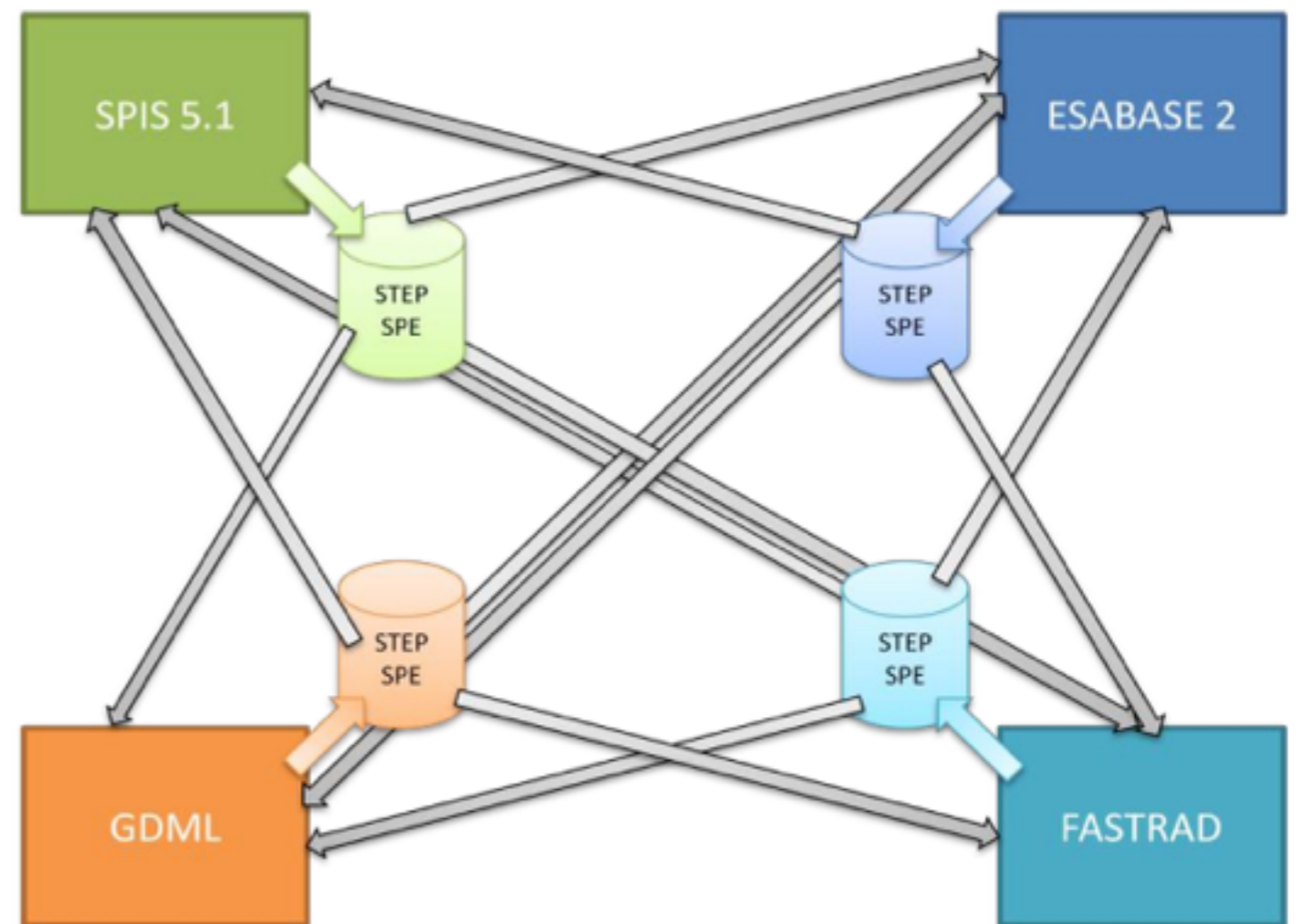   - Simulations
  - Results and outputs

- **Conversions are sources of mistakes, inconsistencies and losses of data**

- **Complexity of direct conversion, benefits of a pivot exchange protocol**
  - Direct conversions requiere n^2 converters
  - Use of a centre data structure reduce to $n+1$ converters

Plasma

Debris

SPIS 5.1

STEP SPE

ESABASE 2

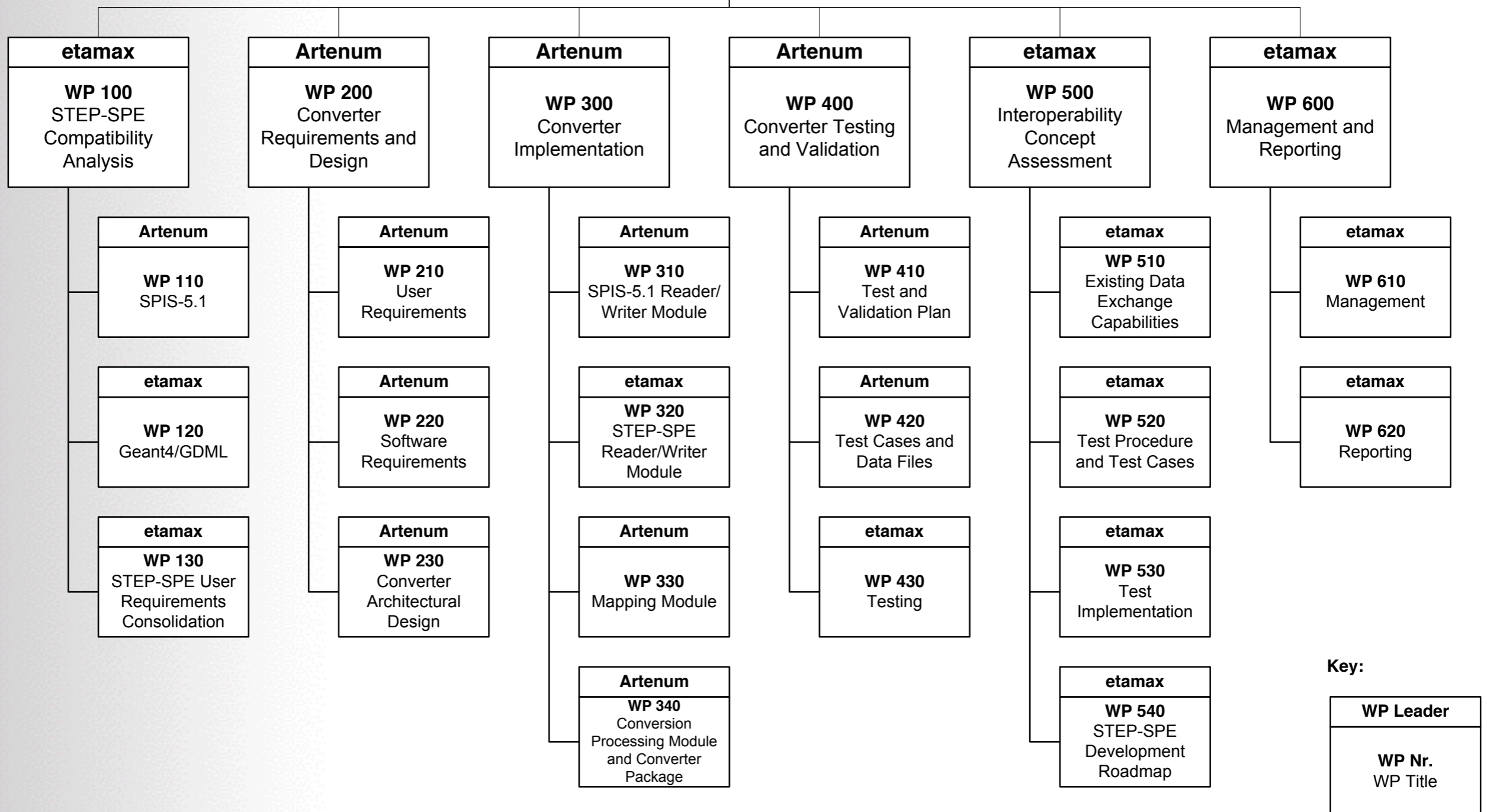STEP SPE

STEP SPE

GDML

STEP SPE

FASTRAD

Radiations

Radiations

- **ESA/ESTEC/TEC-EES contract (ref. AO/1-7798/14/NL/AK)**
  - Technical officer: F. Cippriani (TEC-EES)
- **Key objectives**
  - Evaluate the cross-compliance between STEP-SPE and the data structure of charging tools (SPIS) and radiations analysis tools using GDML (e.g. CIRSOS);
  - Design and develop a SPIS-to-STEP-SPE and STEP-SPE-to-SPIS converter;
  - Identify new thematics needs and further extension of the STEP-SPE protocol.
- **Make a generic exchange bridge for data models between SEATs**
- **European industrial consortium**
  - Prime contractor: **etamax**
    - Project management and leading
    - STEP-SPE expertise, cross-compliance analysis, requirement phase, validation phase, propose extensions of the protocol for both radiations and charging analysis;
  - Sub-contractor: **Artenum**
    - SPIS expertise, cross-compliance analysis, requirement phase;
    - Software design and development of the converter;
    - Contribution to propositions of protocole extension.

Project structure

AO/1-7798/14/NL/AK "Interoperability for Space Environment Analysis Tools"

Figure 4-1
Work Breakdown Structure

etamax	Artenum	Artenum	Artenum	etamax	etamax
WP 100 STEP-SPE Compatibility Analysis	**WP 200** Converter Requirements and Design	**WP 300** Converter Implementation	**WP 400** Converter Testing and Validation	**WP 500** Interoperability Concept Assessment	**WP 600** Management and Reporting

Artenum	Artenum	Artenum	Artenum	etamax	etamax
WP 110 SPIS-5.1	**WP 210** User Requirements	**WP 310** SPIS-5.1 Reader/Writer Module	**WP 410** Test and Validation Plan	**WP 510** Existing Data Exchange Capabilities	**WP 610** Management

etamax	Artenum	etamax	Artenum	etamax	etamax
WP 120 Geant4/GDML	**WP 220** Software Requirements	**WP 320** STEP-SPE Reader/Writer Module	**WP 420** Test Cases and Data Files	**WP 520** Test Procedure and Test Cases	**WP 620** Reporting

etamax	Artenum	Artenum	etamax	etamax
WP 130 STEP-SPE User Requirements Consolidation	**WP 230** Converter Architectural Design	**WP 330** Mapping Module	**WP 430** Testing	**WP 530** Test Implementation

Artenum
WP 340 Conversion Processing Module and Converter Package

etamax
WP 540 STEP-SPE Development Roadmap

Key:

WP Leader
WP Nr. WP Title

What is STEP?

- **An ambitious aim, a complex task**

  - « ISO 10303 is an ISO standard for the computer-interpretable representation and exchange of product manufacturing information. Its official title is: Automation systems and integration — Product data representation and exchange. It is known informally as "STEP", which stands for "Standard for the Exchange of Product model data". ISO 10303 can represent 3D objects in Computer-aided design (CAD) and related information. » (thanks to Wikipedia);

  - Started in 1984 as a successor of IGES, SET and VDA-FS;

  - Aims to cover most of PLM issues: "STEP shall be based on one single, complete, implementation-independent Product Information Model, which shall be the Master Record of the integrated topical and application information models. »

- **Hierarchical structure based on the fundamental idea of « concept »**

What is STEP?

- **A data exchange protocol and not a file format only**
  - Describe not only the data structure, but also the data exchange scheme, their life cycle and their self-consistency.
  - Description done in EXPRESS language, which is a standard data modeling language for product data
  - Implementations exist into various languages, for instance:
    - Part 21 - STEP-File,
    - Part 22 - SDAI Standard data access interface,
    - Part 23 - C++ language binding,
    - Part 24 - C language binding,
    - Part 26 - Binary representation of EXPRESS-driven data using HDF5,
    - Part 27 - Java TM programming language binding,
    - Part 28 - STEP-XML XML representation for EXPRESS-driven data…).

STEP files are an ASCII implementation of such scheme and dedicated to be a normalized exchange file format.

- **Complex approach split into smaller parts developed, balloted and approved separately**
  - Modular architecture, the STEP Application modules define common building blocks to create modular Application Protocols (AP):
    - AP203 configuration controlled 3D design of mechanical parts and assemblies, B-Rep based description. Typically used by industrial CAD tools like CATIA, AutoCAD, Gmsh, OpenCascad…
    - AP214 adds colors, layers, geometric dimensioning and tolerance, and design intent to AP203AP214 is considered an extension of AP203.
    - AP210 electronic assembly interconnect and packaging design
    - AP233 system engineering and design
    - AP239 product life cycle support,….
- **If all sub-set share the same protocol, they do not address the same things.**

This means that this is not because you are able to read/understand one specification that you will able to do it for another one.

What is STEP-SPE?

- **STEP-SPE is one of such sub-specification**
  - Extensions of AP 232 - Exchange file specification
  - Dedicated to the data exchange in space-environment analysis, including:
    - Geometrical description;
    - Material properties;
    - Mission settings, environment characterization;
    - Fields …
  - Already used in debris analysis (i.e. esabase2) and aims to constitutes the reference and central exchange protocol in space-environment analysis.
  - Based on:
    - STEP-TAS for the geometry, based on a CSG description (inherited from thermal analysis tools used at ESA)
    - STEP-NRF for the material properties and fields;
    - Include extra layer for space environment aspects.
- **But, we remind that STEP-SPE is a specific specification distinct from other ones used in other industrial tools:**

> STEP AP203/214, as generated by mots of industrial CAD tools, and STEP-SPE differs in numerous points and required different readers/writers.

A STEP-SPE file

ASCII and (easily) human readable

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('Model built with SPE_ARM library.'), '');
FILE_NAME('','2015-07-02T15:28:10',(),(), '', '', '');
FILE_SCHEMA(('SPE_ARM'));
ENDSEC;
DATA;
#1=NRF_ORGANIZATION('ESA_ESTEC','European Space Agency / European Space Research and Technology Centre',' ');
#2=NRF_ORGANIZATIONAL_ADDRESS('D/TEC-MCV','1','Keplerlaan','P.O. Box 299','Noordwijk','ZH','2201 AZ',
'The Netherlands',$,$,'step-spe@thermal.esa.int',$,(#1),' ');
#3=NRF_MATERIAL_CLASS('debris_analysis_material','ESABASE DEBRIS (Material) properties.',$,(),
('debris_shielding_type','shield_thickness','shield_mass_density','backup_shield_thickness','backup_shield_mass_density',
'outer_backup_shield_distance'));
#4=NRF_SCALAR_BASIC_QUANTITY_TYPE('one','1','eneric dimensionless unit quantity (see ISO 31-0)',0.0,0.0,0.0,0.0,0.0,0.0,0.0);
#5=NRF_SCALAR_BASIC_QUANTITY_TYPE('length','L','Length (SI base quantity, see ISO 31-0)',1.0,0.0,0.0,0.0,0.0,0.0,0.0);
#6=NRF_SCALAR_BASIC_QUANTITY_TYPE('mass','m','Mass (SI base quantity, see ISO 31-0)',0.0,1.0,0.0,0.0,0.0,0.0,0.0);
#8=NRF_SCALAR_BASIC_QUANTITY_TYPE('temperature','T','Thermodynamic temperature (SI base quantity, see ISO 31-0)',0.0,0.0,0.0,0.0,1.0,0.0,0.0);
#9=NRF_SCALAR_BASIC_QUANTITY_TYPE('plane_angle','alpha','Plane angle (SI base quantity, see ISO 31-0)',0.0,0.0,0.0,0.0,0.0,0.0,0.0);
#10=NRF_SCALAR_BASIC_QUANTITY_TYPE('electric_potential','V','Electric potential (SI base quantity, see ISO 31-0)',2.0,1.0,-3.0,-1.0,0.0,0.0,0.0);
(…)
#17=NRF_SCALAR_BASIC_QUANTITY_TYPE('thickness','thick','thickness of elements.',1.0,0.0,0.0,0.0,0.0,0.0,0.0);
#18=NRF_SCALAR_BASIC_QUANTITY_TYPE('shielding_type','ST','Type of shielding.',0.0,0.0,0.0,0.0,0.0,0.0,0.0);
#20=NRF_CONTEXT_DEPENDENT_UNIT('dimensionless','-',#5);
#21=NRF_CONTEXT_DEPENDENT_UNIT('cube centimetres per atom','10^24.cm^3/atom',#19);
#22=NRF_SI_UNIT(*,*,#5,'','metre');
#23=NRF_SI_UNIT(*,*,#6,'kilo','gram');
(…)
#92=NRF_ROOT('root','rootName','',#93,'http://mechanical-engineering.esa.int/thermal/tools/step-spe/v5.3/spe_arm.exp',('http://mechanical-
engineering.esa.int/thermal/tools/step-spe/v5.3/dictionary-2006-04-18/spe_arm_dictionary.stp'),$,$,$,$,(),$,$,$,$,$,$,$,(#94),());
#93=NRF_ORGANIZATIONAL_PROJECT('orgProjectName','orgProjectDescription',());
#94=TAS_MESHED_GEOMETRIC_MODEL('GeomModel','GeomModel','',$,$,'',(),(),(),(),(),(),(),(),(),(),$,(),$,#95,());
#95=TAS_COMPOUND_MESHED_GEOMETRIC_ITEM('OBJECT_0','Title','',$,$,#96,(#98));
#96=TAS_AXIS_TRANSFORMATION_SEQUENCE((#97));
#97=TAS_TRANSLATION(0.0,0.0,0.0,#72);
#98=TAS_COMPOUND_MESHED_GEOMETRIC_ITEM('OBJECT_1','BOX','',$,$,$,(#125));
#125=TAS_COMPOUND_MESHED_GEOMETRIC_ITEM('OBJECT_1_3','CYLINDER','',$,$,#132,(#126));
#126=SPE_MESHED_PRIMITIVE_BOUNDED_SOLID('4','','',$,$,$,#127,$,$,$,$,$,$,$,$,$,(),());
#127=SPE_SOLID_CYLINDER(#126,#128,#129,#130,#131,$,$,$,$);
#128=TAS_3D_CARTESIAN_POINT('P1',0.0,0.0,0.0,#72);
#129=TAS_3D_CARTESIAN_POINT('P2',0.0,0.0,0.6,#72);
#130=TAS_3D_CARTESIAN_POINT('P3',1.0,0.0,-3.0,#72);
#131=NRF_REAL_QUANTITY_VALUE_EXPRESSION(#72,0.125,$,$,$);
#132=TAS_AXIS_TRANSFORMATION_SEQUENCE((#133));
#133=TAS_TRANSLATION(0.2,0.2,0.1,#72);
ENDSEC;
END-ISO-10303-21;
```

- **The Spacecraft Plasma Interaction Software**
  - Simulation software to model spacecraft-plasma interactions, including:
    - 3D and dynamical modelling of the surrounding plasma sheath;
    - Particles and current collections;
    - Surface effects, secondary emissions, and active sources
    - Internal electrical balance.
    - Numerical kernel, SPIS-NUM, an electrostatic 3D unstructured electrostatic Particle-In-Cell plasma model ;
  - Unstructured mesh and supporting complex and realistic geometries;
  - Include a complete Integrated Modelling Environment (IME), SPIS-UI:
    - Pre-processing (CAD, meshing, IBCs settings, simulation settings…)
    - Simulation control and monitoring;
    - Data-mining and post-processing.
  - An open-source project, initiated in 2001 by ESA, with support of several national agencies (CNES)
  - About 10 major releases and more than 8 500 downloads since 2003
- **SPINE**
  - http://dev.spis.org
  - More than 900 registered members

```
ExampleOfNewProject.spis5        ----> Root directory
├── debug_simu.txt

├── DefaultStudy                 ----> Study directory
│   ├── Preprocessing            ----> Preprocessing directory for the study
│   │   ├── Datafields
│   │   │   ├── BackGroundDens.nc ----> Example of pre-processing DataField
(...)
│   │   │   └── XyzS.nc
│   │   ├── ElectricalCircuit    ----> Internal circuit settings directory
│   │   │   └── circuit.txt
│   │   ├── Geometry             ----> Geometrical models directory
│   │   │   ├── GeometricalSystem ----> Studied geometrical models
│   │   │   │   └── GeoCase.geo
│   │   │   └── Instruments      ----> Virtual instruments models
│   │   ├── Groups               ----> Groups and properties settings dir.
│   │   │   └── groups.xml
│   │   ├── Mesh                 ----> Mesh directory
│   │   │   ├── GeometricalSystem
│   │   │   │   └── GeoCase.msh
│   │   │   └── Instruments
│   │   └── Properties           ----> Properties directory
│   │
│   └── Simulations              ----> Simulation directory
│       └── Run1                 ----> Run directory
│           ├── GlobalParameters ----> Global parameters directory
│           │   └── globalParameters.xml
│           ├── NumKernel        ----> Spis-Num specific directory
│           │   ├── Input
│           │   │   └── circuit.txt
│           │   └── Output
│           │       ├── collectedCurrents.txt
│           │       ├── emittedCurrents.txt
(...)
│           │       └── SpisNum.log
│           │
│           ├── OutputFolder     ----> Output data directory
│           │   ├── DataFieldExtracted
│           │   │   ├── Collected_current_versus_time.msh
│           │   │   ├── plasma_elec_field_at_t_=_2.002453E-5_s.nc
(...)
│           │   │   └── surfFlagSC-FACEMask.nc
│           │   ├── DataFieldMonitored
│           │   │   └── Energy_of_elec1_mesh.msh
│           │   │
│           │   └── Export       ----> Exported data
│           │       ├── final_plama_potential.vtk
│           │       └── screenShot1.png
│
└── model.xml
```

- **A whole simulation definition and results**
  - From SPIS's point of view, its simulation project in its whole
- **Uses as a conversion bridges between SEATs (not used as persistent scheme for SPIS)**
- **However under the current limits of the STEP-SPE protocol, main focus on:**
  - Geometrical model
  - Groups
  - Project self-consistency
  - Conversion of non-tailored data
- **Bi-directional**

SPIS project (.spis5) ⇄ **STEP-SPE file**

Cross-compliance analysis phase
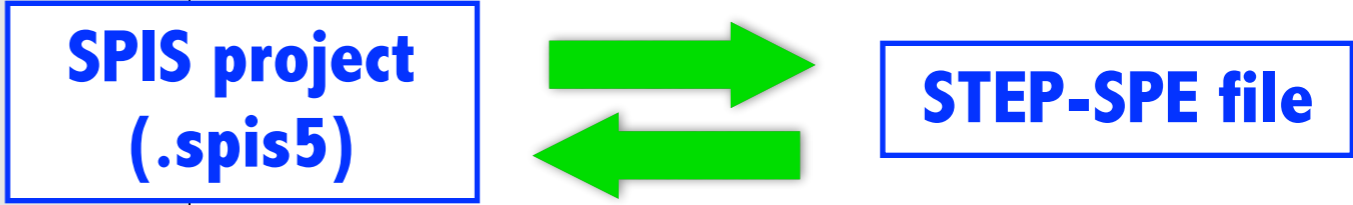
- **An intense effort**

- **Review of all (or almost) concepts supported by STEP-SPE and identification of equivalent data node in SPIS**

- **Large compliance analysis matrix,**

- **Leading to the fact that concepts mainly supported are geometrical ones for now**

- **No direct support or no relevance w.r.p tailored data/needs of SPIS for:**

  - Global parameters

  - Mesh and data-fields

  - Circuit definition

- **Done on the basis capabilities of the protocol**
- **Identification of uses cases/exchange with the industry**
- **Main general requirements, contexts of uses and usage scenarios**
  - UR-GR-001: Uses as a conversion bridges between SEATs
  - UR-GR-002: Uses as a normalised and perennial persistency scheme for SPIS
  - UR-GR-003: Standalone application
  - UR-GR-004: Bidirectional conversion
  - UR-GR-005: Backwards compatibility
  - UR-GR-006: Usage scenario 0 – Exchange of the geometry only
  - UR-GR-007: Usage scenario 1 - STEP-SPE file generated by another tool (e.g. ESABASE2) and imported into SPIS
  - UR-GR-008: Usage scenario 2: STEP-SPE file generated by SPIS and imported by an-other tool
  - UR-GR-009: Usage scenario 3 (i.e. CIRSOS case): STEP-SPE file generated by SPIS and imported by an-other tool plus import of outputs fields issued from another SEATs
  - UR-GR-007: Usage scenario 4: full exchange of input and output data of various SEATs
- **UR phase finalised for the current implementation, but the use of the converter by the community should help to identify new needs and requirements for future evolutions.**

Your feedbacks are precious and very welcome.

- **The developed converter is:**

  - Local and in command line (i.e. head-less) standalone application;
  - Running independently on tailored applications (e.g. SPIS, esabase2);
  - Only concepts clearly supported (cf. compliance analysis) implemented;
  - Only non-tailored concepts or concept non specific to a thematic supported
  - Identification and minimization of losses of information and/or artifacts; generation (e.g. creation of artificial extra information, over characterization)
  - Respect of the STEP-SPE protocol
  - No storing of raw nor tailored nor native data

  - Obtain and operational and robust conversion loop (STEP-to-SPIS, SPIS-to-STEP, SPIS-to-STEP-to-SPIS, STEP-to-SPIS-to-STEP)

- **Firstly focusing on the conversion of geometrical concepts and related geometrical operations**

- **Java and multi-modules Maven project (i.e. see SPIS's rules)**
- **Converter based on several external and pre-existing components**
  - SPIS's data model and project's reader/writer
  - Existing open-source Artenum components (e.g. Keridwen, Penelope, Frida…)

    > Please see,
    > http://dev.artenum.com

  - Full Java based software components
    - Maven driven compilation and dependencies tracking
    - Clearly identified (Maven artefact)
  - Existing etamax STEP-SPE-SDK
    - C/C++ components

      > For STEP-SPE SDK registration, please see
      > http://esabase2.net/step-spe/step-spe-registration/

    - Provided by etamax
    - Accessible through a JNI wrapping
    - Repackaging for a better handling and use

STEP-SPE protocol definition

EXPRESS language source codes

Generation of readers/writers implementations

Code generator

C++ readers/writers

C++ STEP-SPE SDK

Developments done in the frame of the present activity

Java Wrapping

Swig and JNI

Java wrapping of readers/writers

Java wrapping of STEP-SPE SDK

Integration as external components in the Interop converters

STEP-SPE / SPIS Interop converter

- **All main functional blocs developed and operational, including:**
  - Converter factory
  - Global readers/writers on both side
  - Central controller, UI (cmd line) and properties handler
  - Most of converters for key geometrical concepts
- **Thanks to Keridwen's templates CSG-to-BRep decomposition operational for most key geometrical concepts**
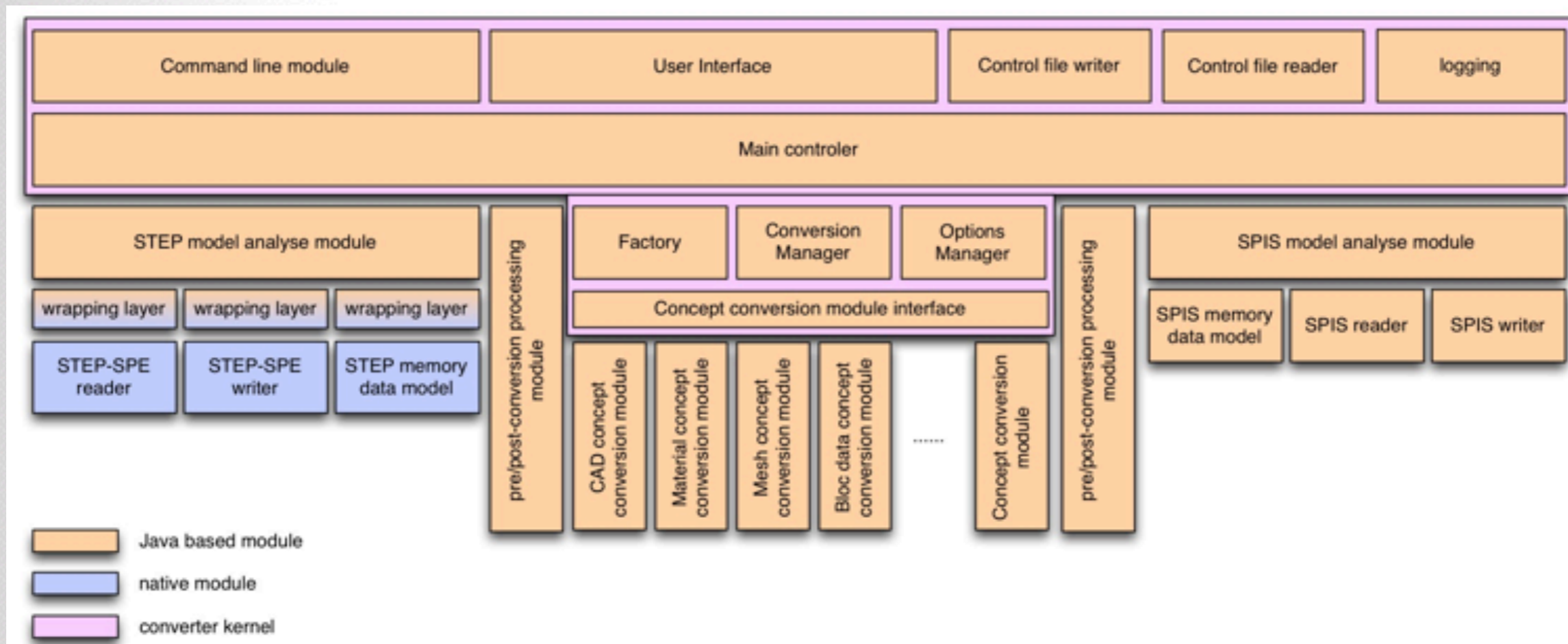
- **Major issues**
  - Losses of information (e.g. CSG sphere converted into a set of ruled surfaces)
  - Non unicity of the conversions schemes
  - Impossibility to converted B-Rep to CSG in a unique, robust and exact scheme

CSG
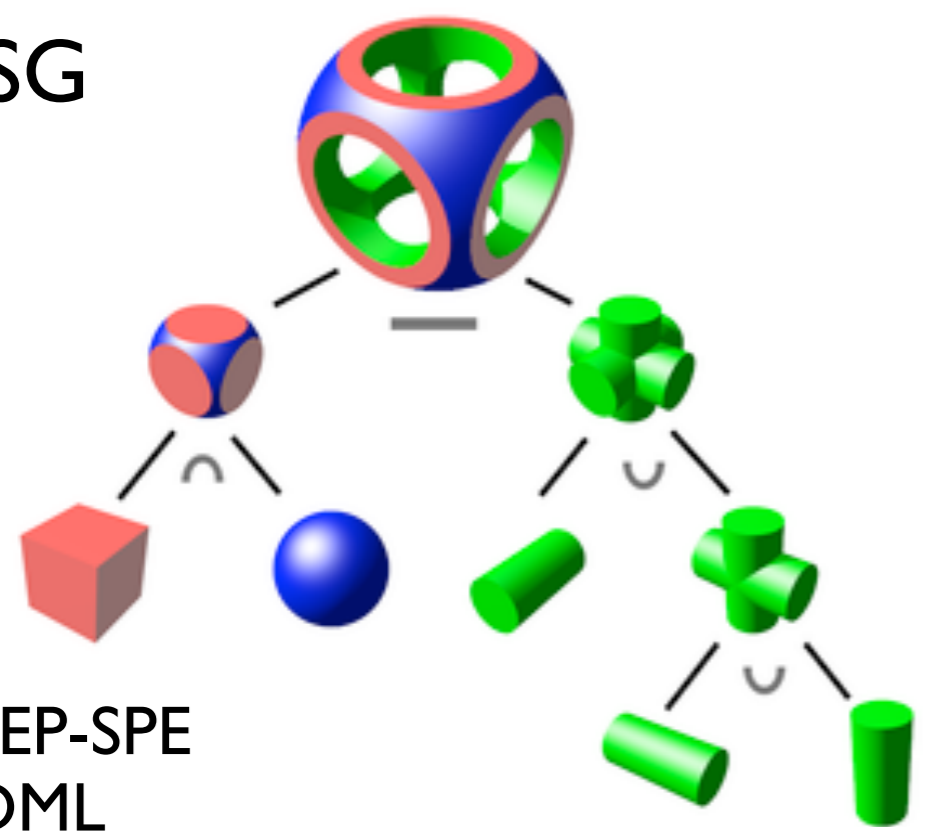
B-Rep

STEP-SPE
GDML
…

STEP-AP 203/214

SPIS (Gmsh geo)

Catia SolidWorks AutoCAD

OpenCascad / Salomé   …

ARTENUM, PARIS Science & Groupware

etamax ñ
member of the ckc group

- **Based on a CSG-to-Brep decomposition**
  - ·Use the template based CAD conversion capability of Keridwen CAD;
  - ·Conversion of the exact geometry without meshing nor aliazing errors;
  - · Most of STEP-SPE geometrical concepts supported;
  - · Support of geometrical operations (i.e. translation, rotations…);
  - · Support of physical flags (for further properties attribution);
  - · No Boolean operations supported yet.
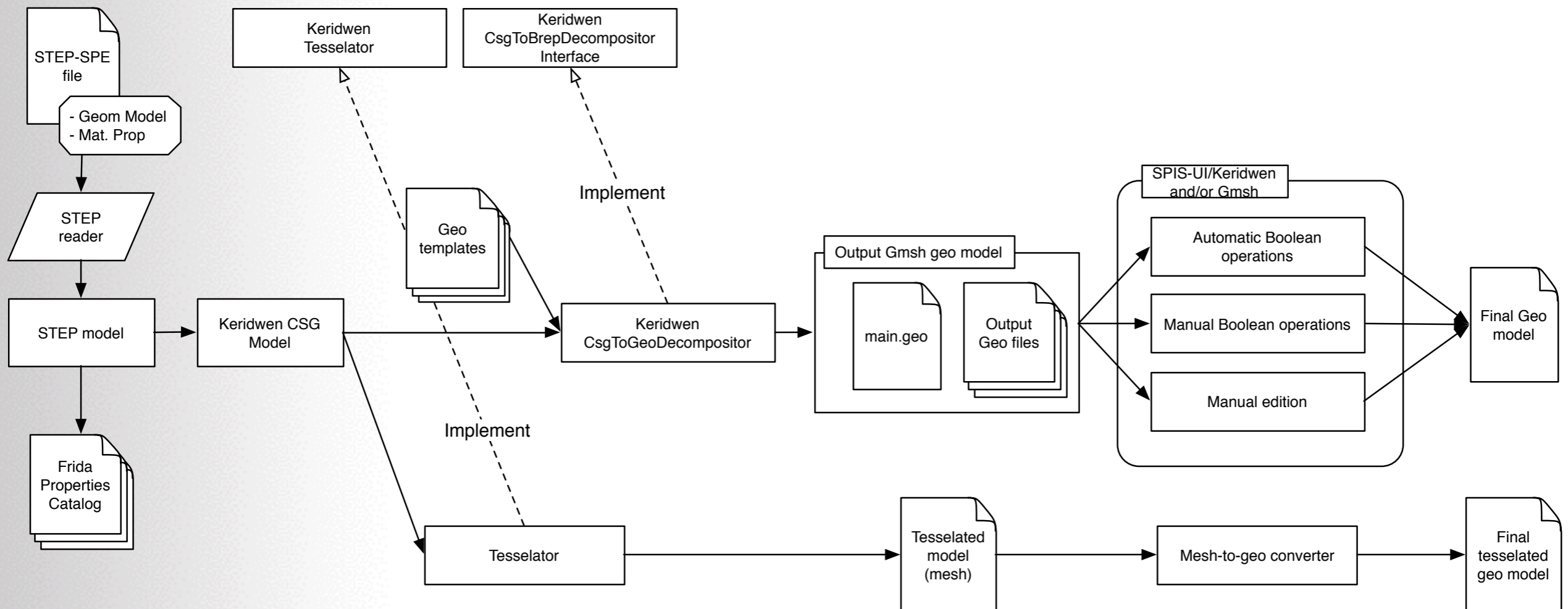
- **Command line application**
- **Two applications depending on the conversion direction**

```
SpisToStepApp.sh [FLAG] FileIn FileOut [opts.xml]  ← convert from SPIS to STEP
StepToSpisApp.sh [FLAG] FileIn FileOut [opts.xml]  ← convert from STEP to SPIS
```

- **Flags [FLAG] for fine settings and controls**

```
-c: To convert the input file into the targeted format and
    generates the default options file.
-g: To generate default option file only.
-o: Set the control option file and convert.
-h: Print the present help.

Example:
    StepToSpisApp.sh -o FileIn FileOut controlOptionsFile.xml
```

- **Running under Linux and Windows**

Example of control options file

```xml
<?xml version="1.0" encoding="UTF-8"?>
<options>
  <Converter inputConcept="95">
    <id>0</id>
    <typeName>TasCompoundMeshedGeometricItemStepToSpisConverter</typeName>
  </Converter>

(…)

  <Converter inputConcept="100">
    <id>4</id>
    <typeName>SolidBoxStepToSpisConverter</typeName>
    <option>
      <name>Mesh Resolution</name>
      <selectedOption value="0.1">automatic_adaptative_mesh_resolution</selectedOption>
      <availableOptions>automatic_adaptative_mesh_resolution manual_mesh_resolution </availableOptions>
      <descriptionOption>Triangle size of the geometry.
          If the value option selected is automatic_adaptative_mesh_resolutionthen the
          mesh refinement of the .geo file generated by this converter is equal to  "value"
          attribute multiplied by the bounding radius of the shape defined in the geo file.
          If the value option selected is manual_mesh_resolution then the "value" attribute
          is used to define the mesh refinement of the .geo file generated by this converter.
      </descriptionOption>
    </option>
  </Converter>
```
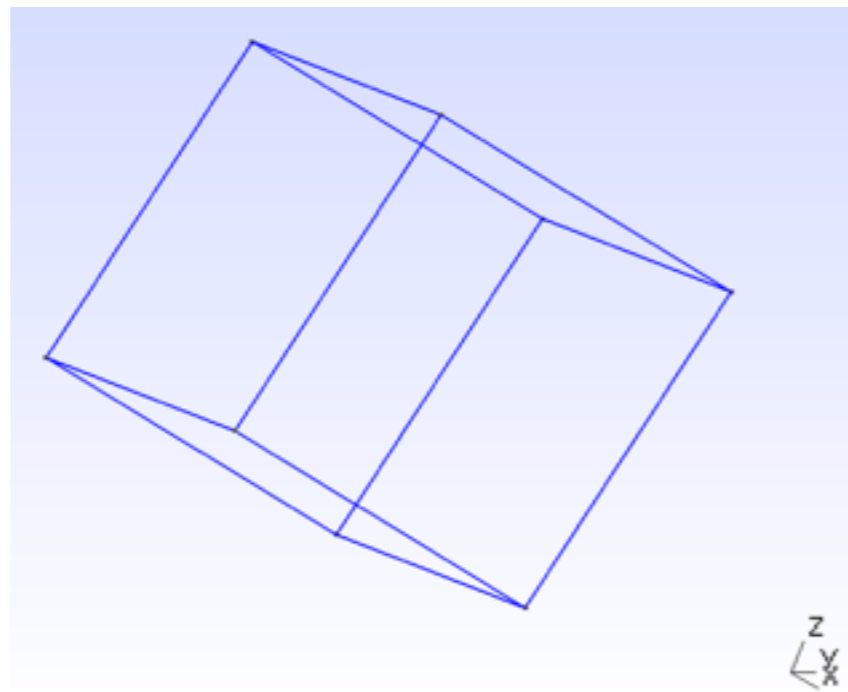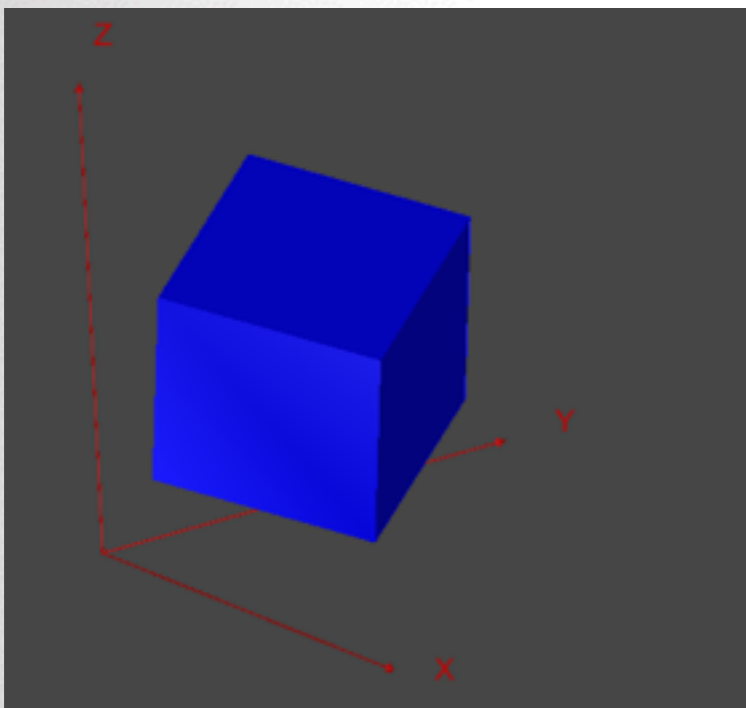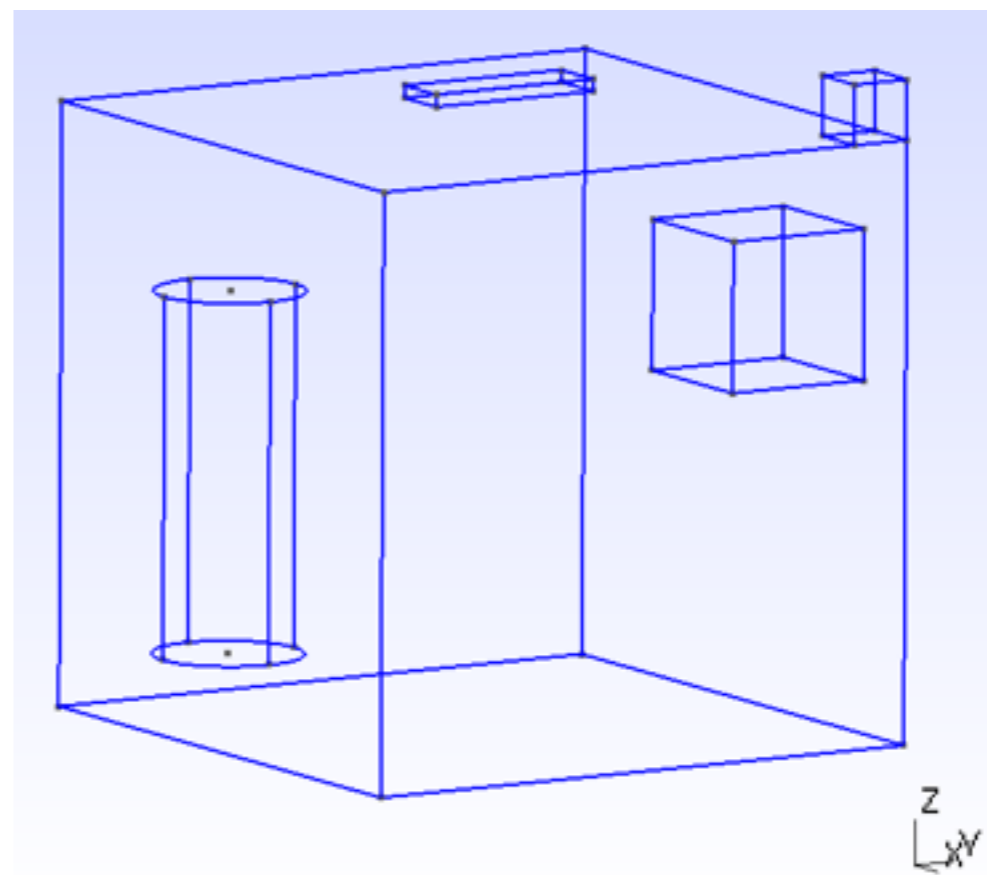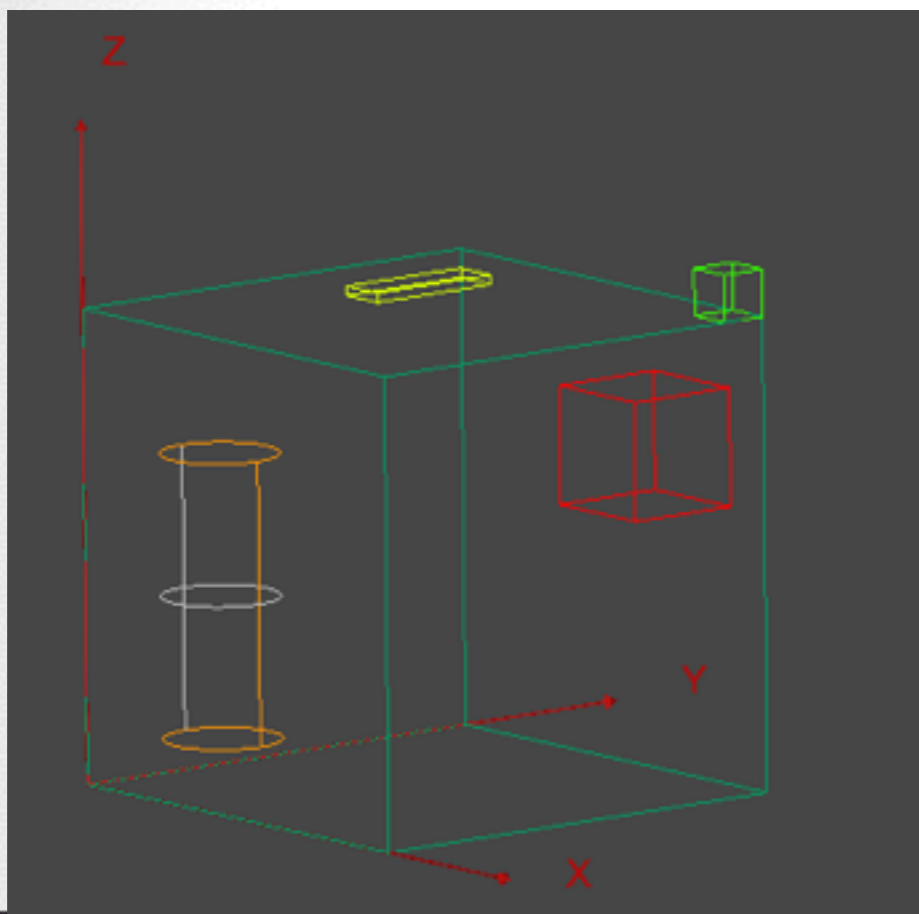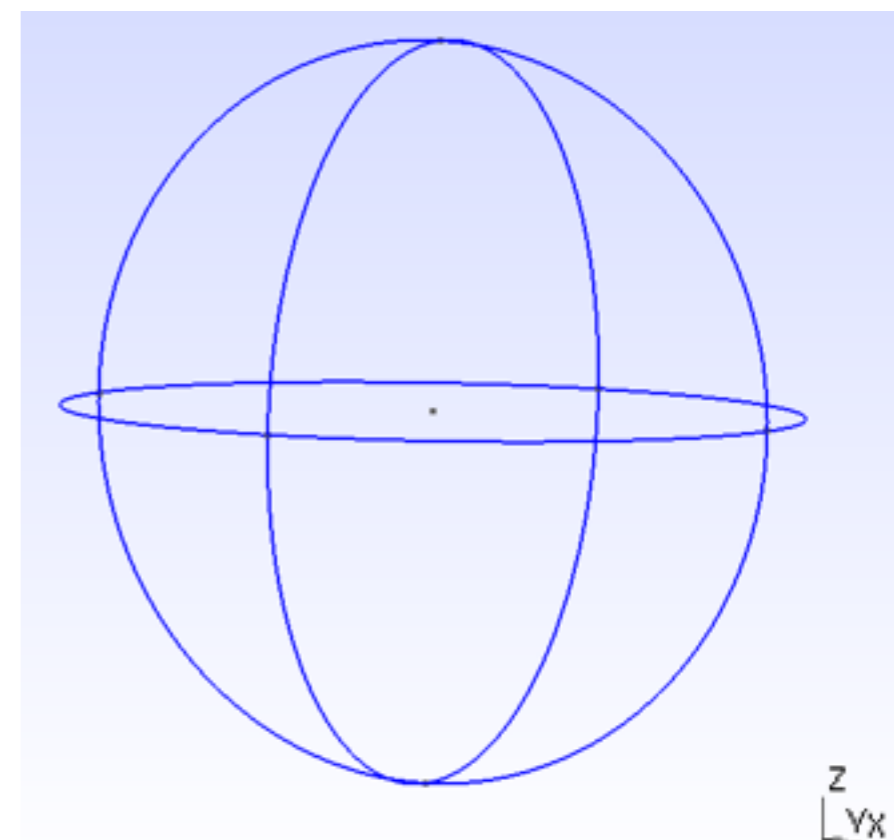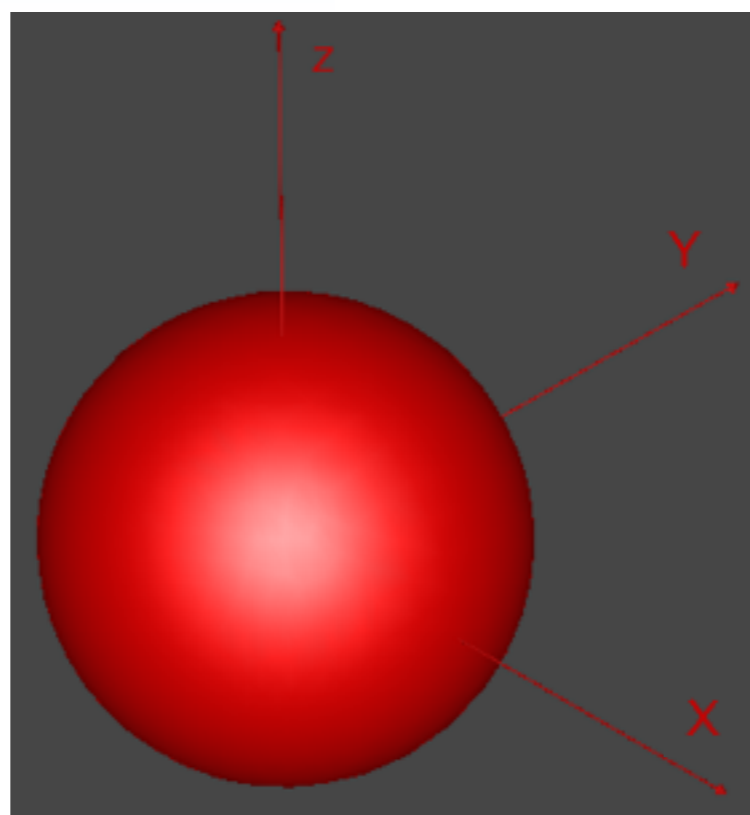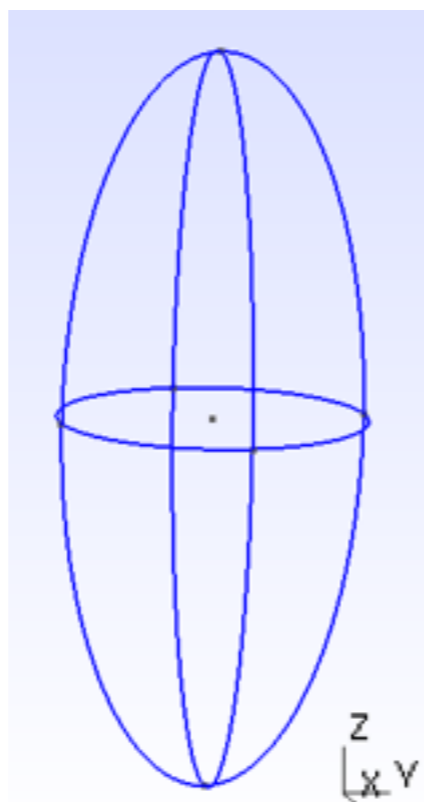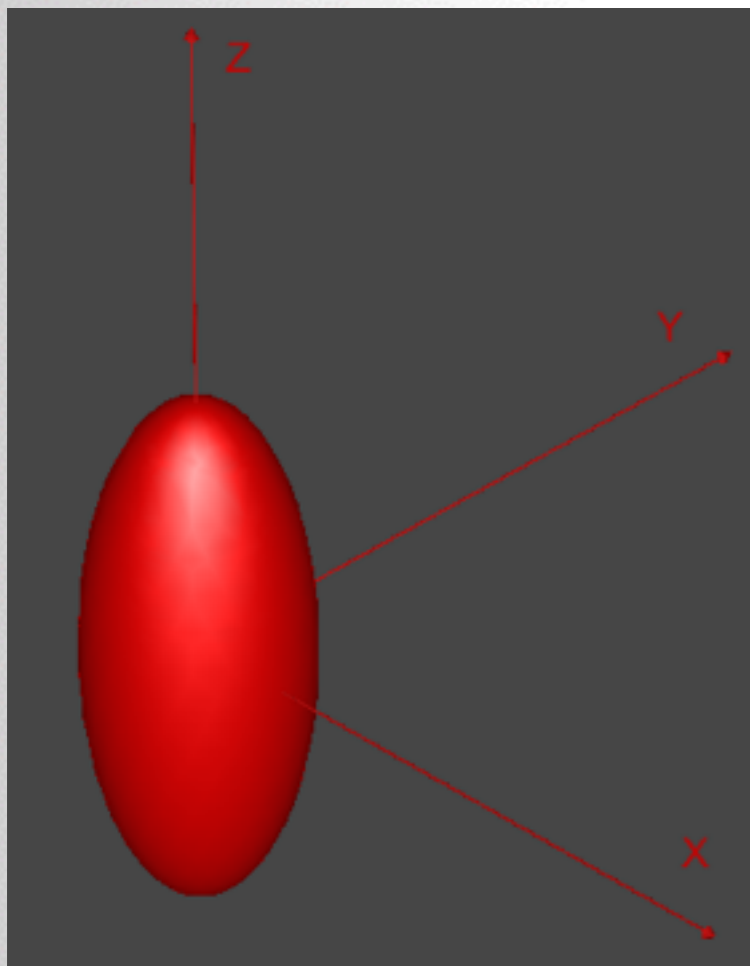
STEP_SPE-to-SPIS
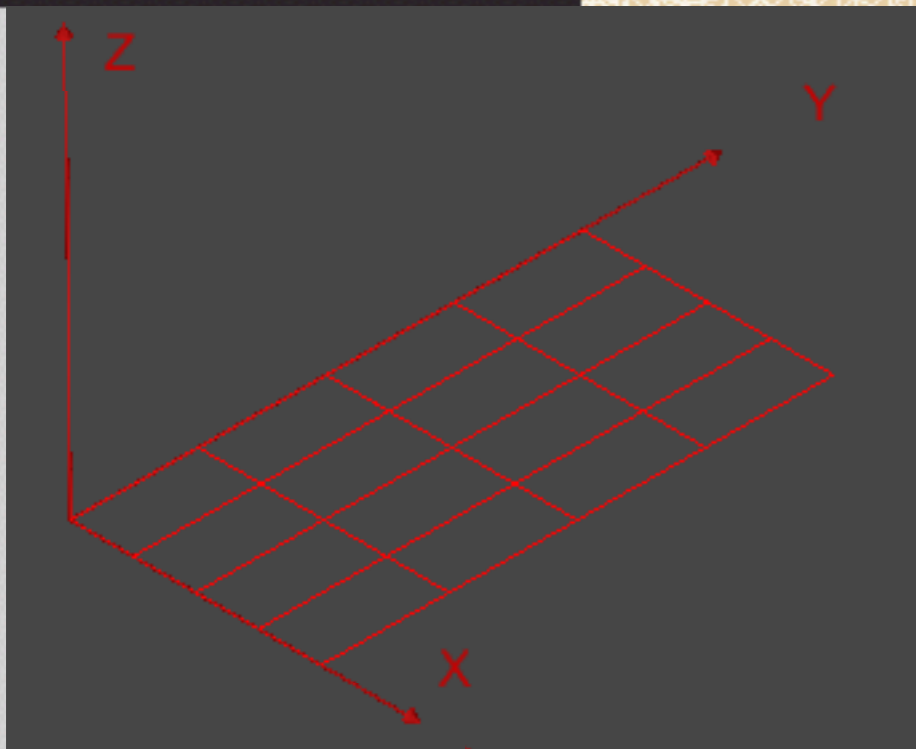
STEP_SPE-to-SPIS

STEP_SPE-to-SPIS
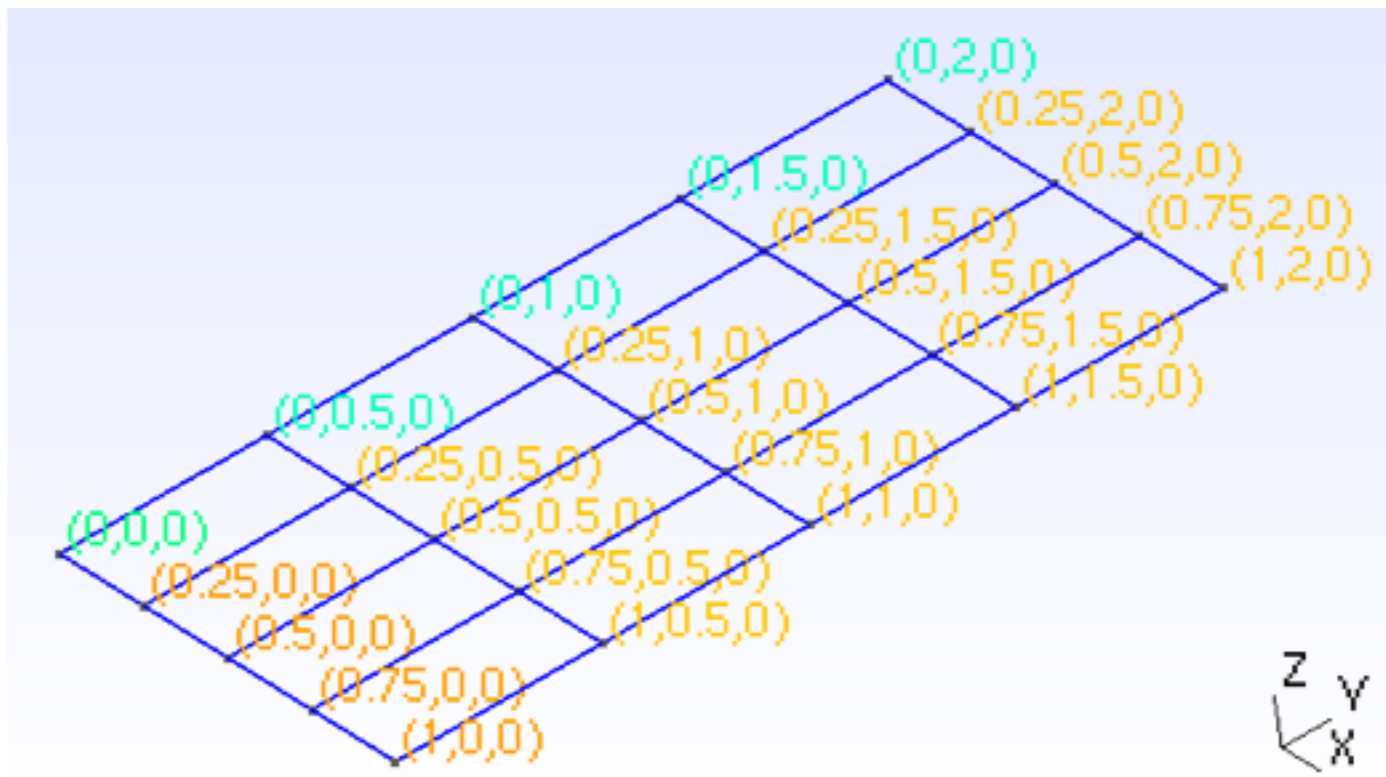
Recently added shapes
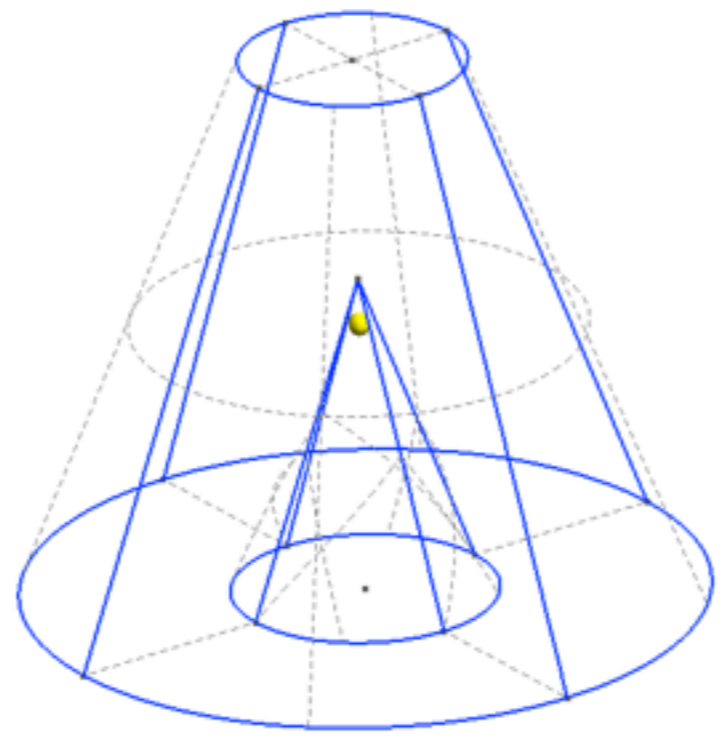
Truncated cone
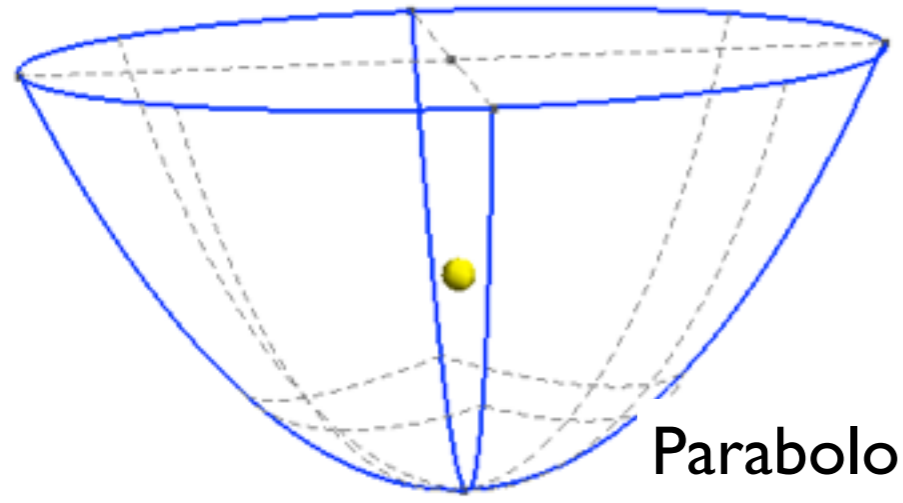
Paraboloid
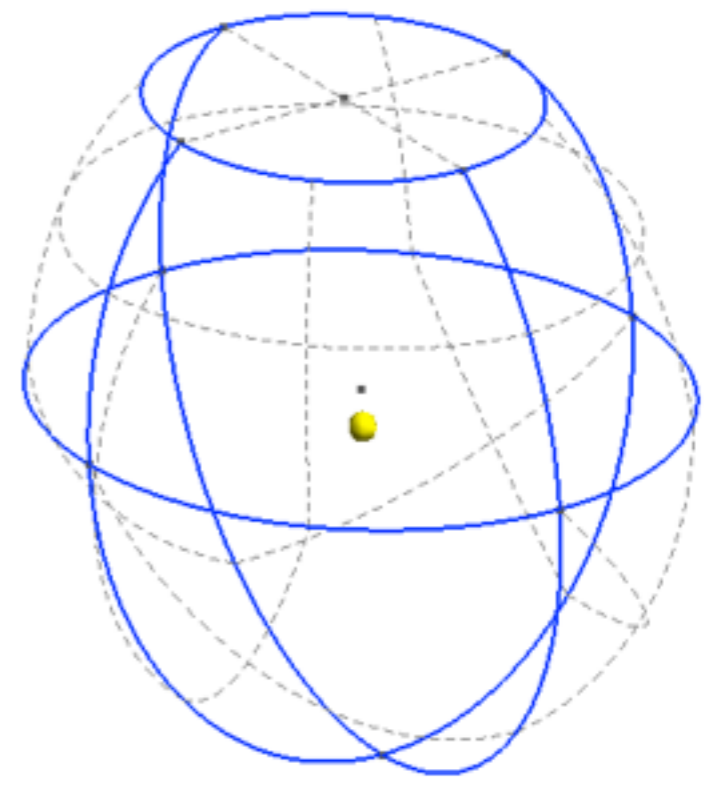
Ellipsoid
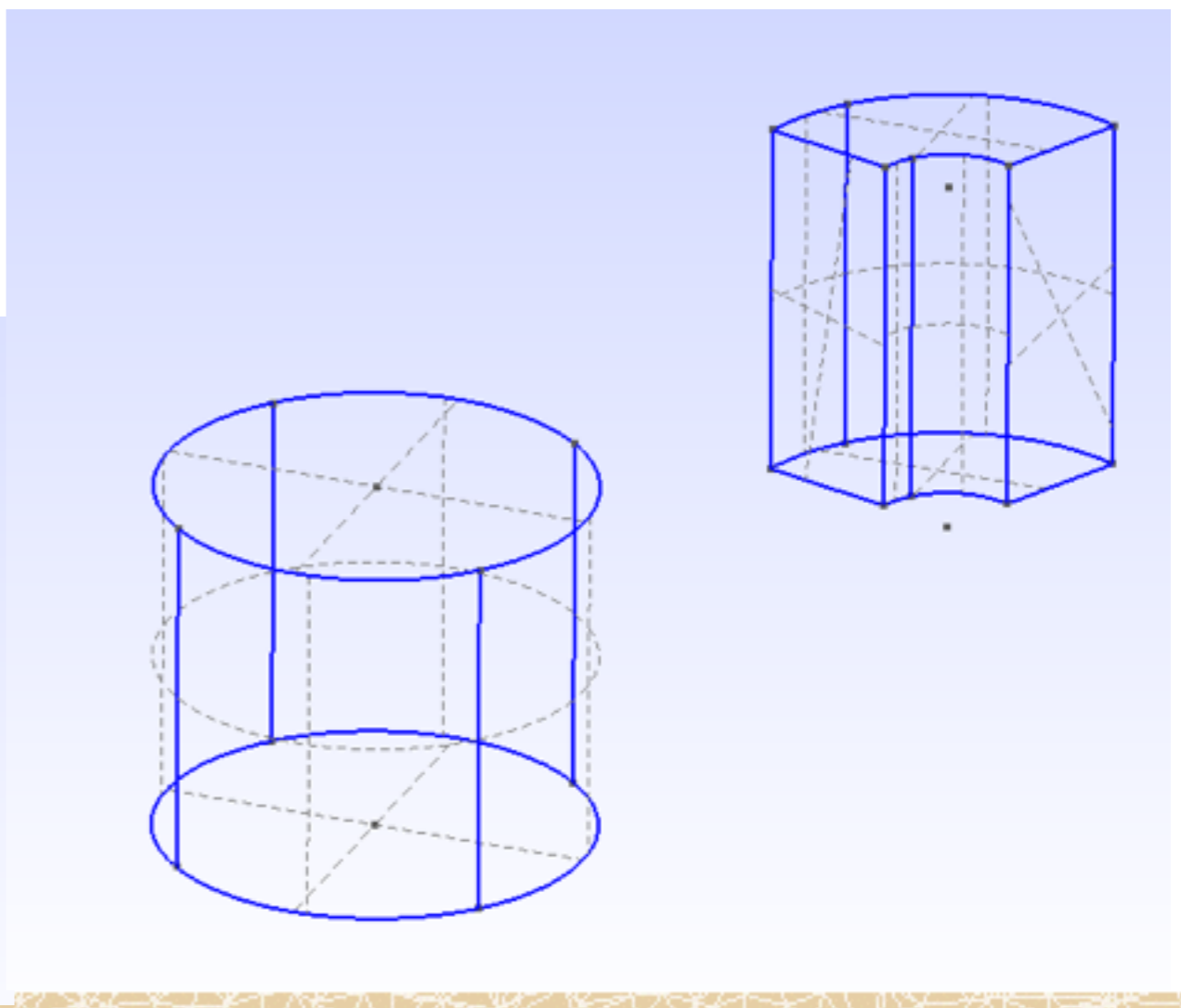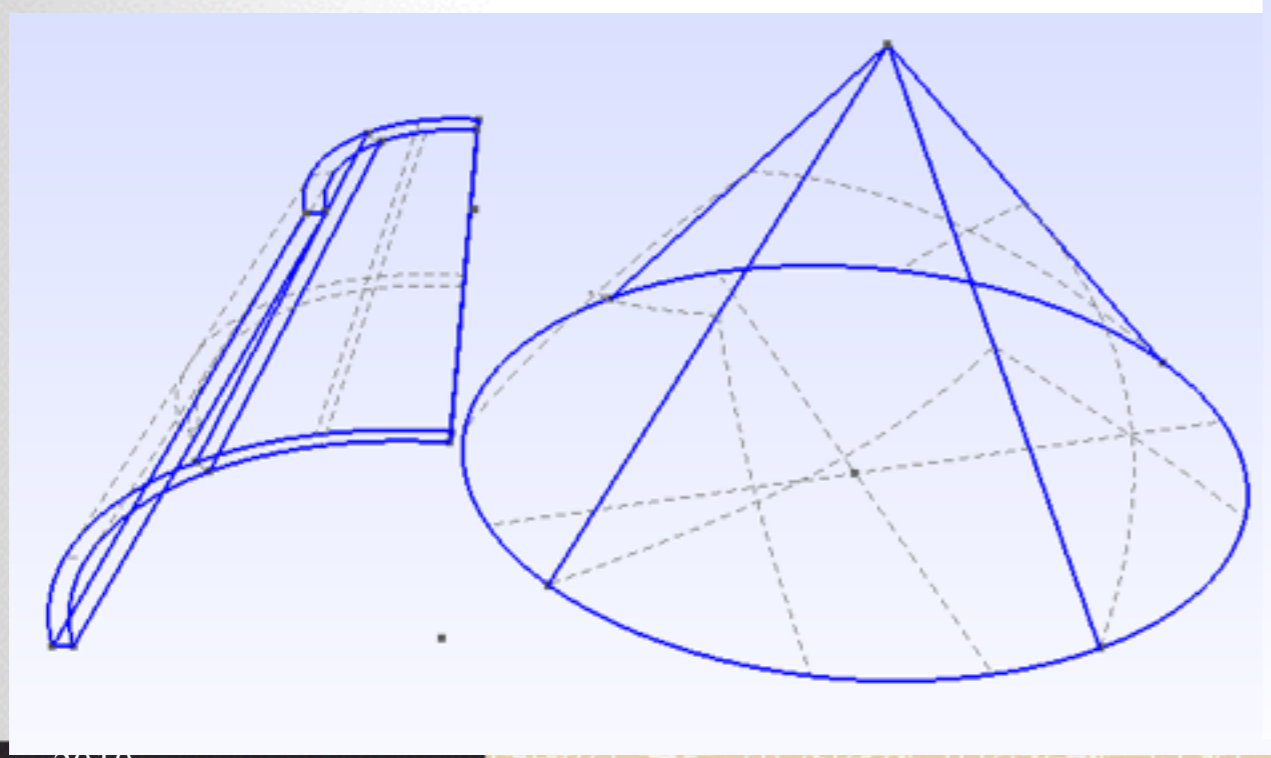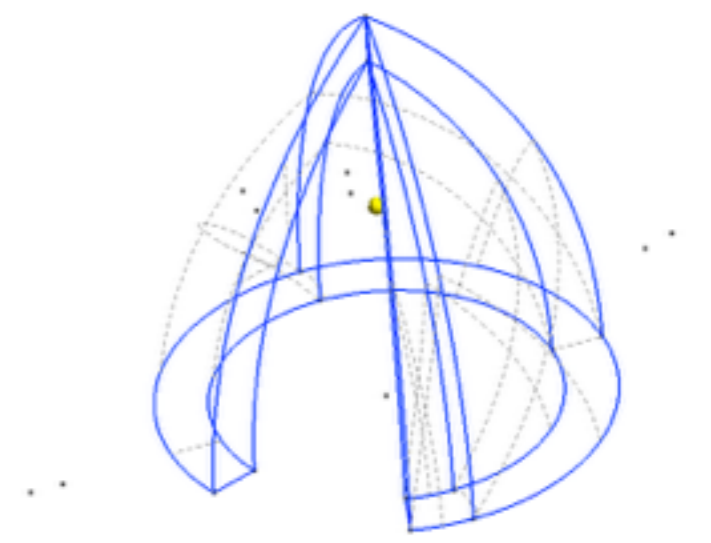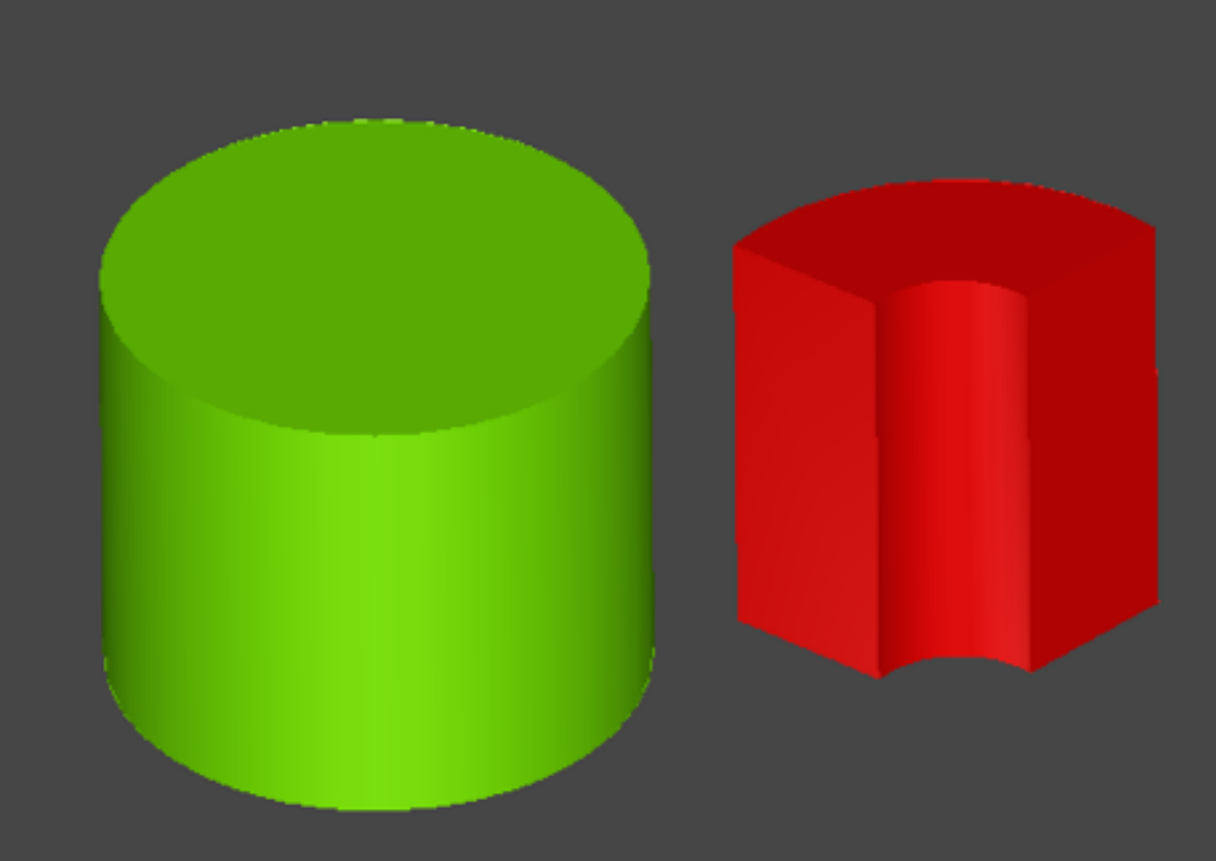
• **Impossibility to provide a clean, exact and robust B-Rep to CSG conversion in the general case**

• **Conversion from SPIS to STEP-SPE done by tessellation only with a simple « soup of triangles » (no inner connectivity due to STEP-SPE limitations)**

- Meshing directly handled by the converter using Gmsh (as in SPIS).

- Support of Gmsh's Physical

- Support of length units
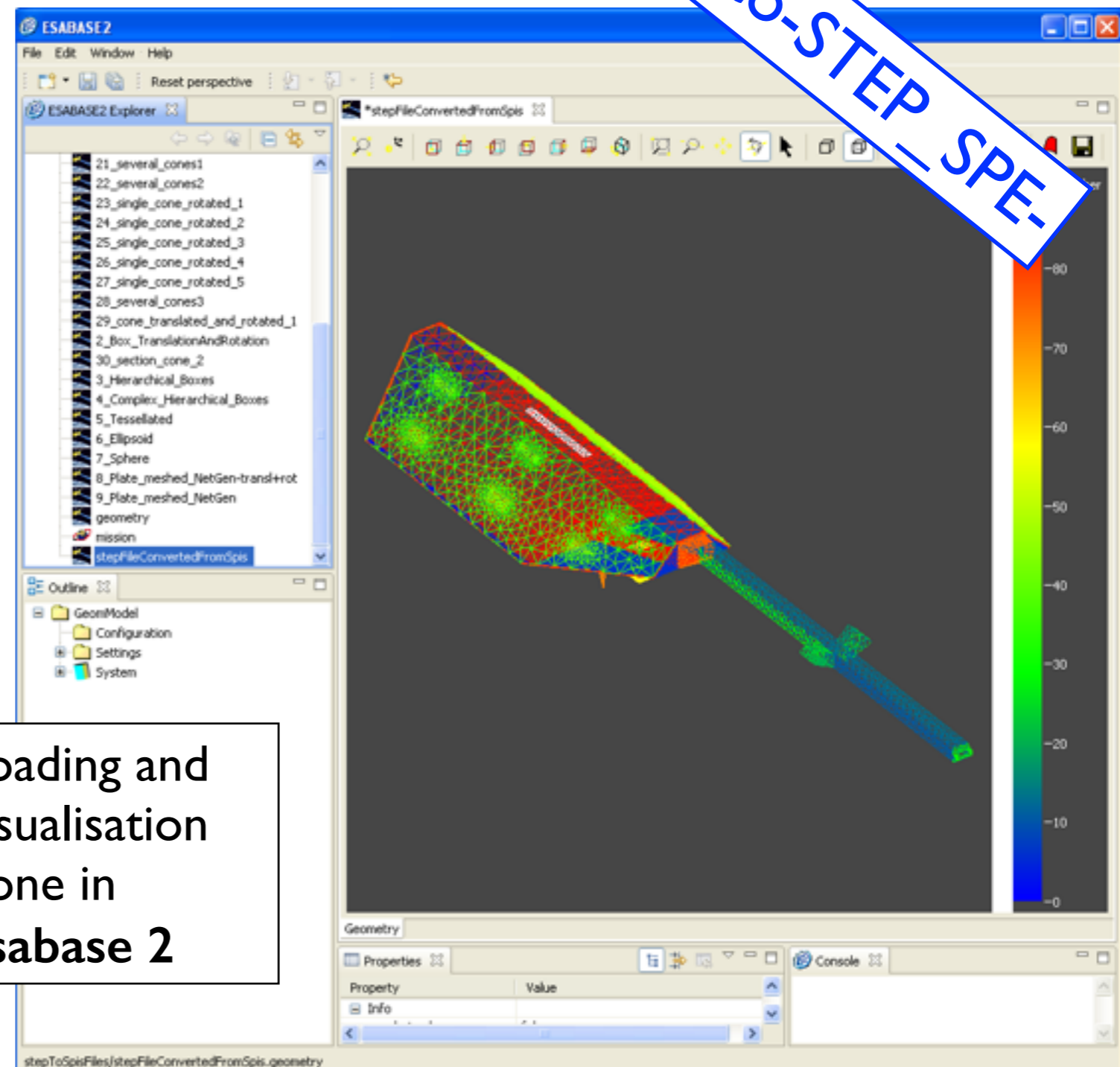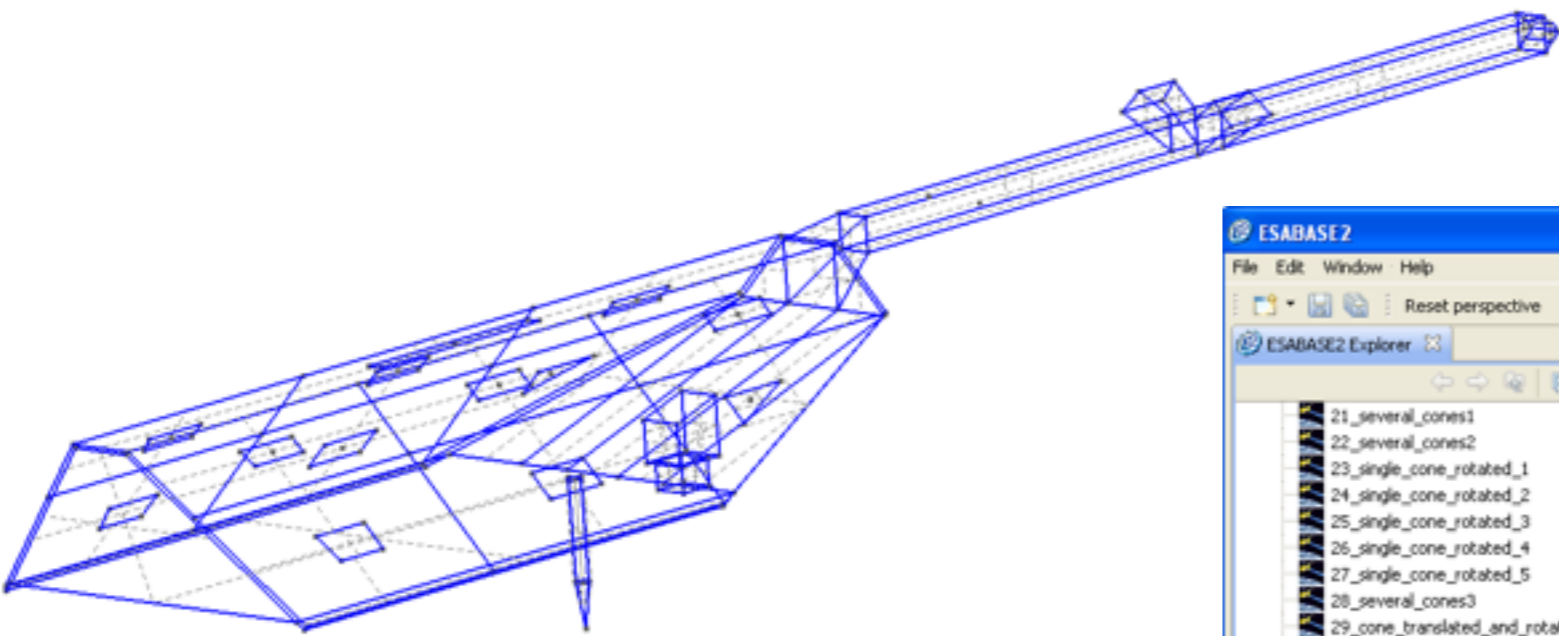
SPIS-to-STEP_SPE
 Conversion time = 10s
 Output STEP-file = 3Mo

Loading and visualisation done in **esabase 2**
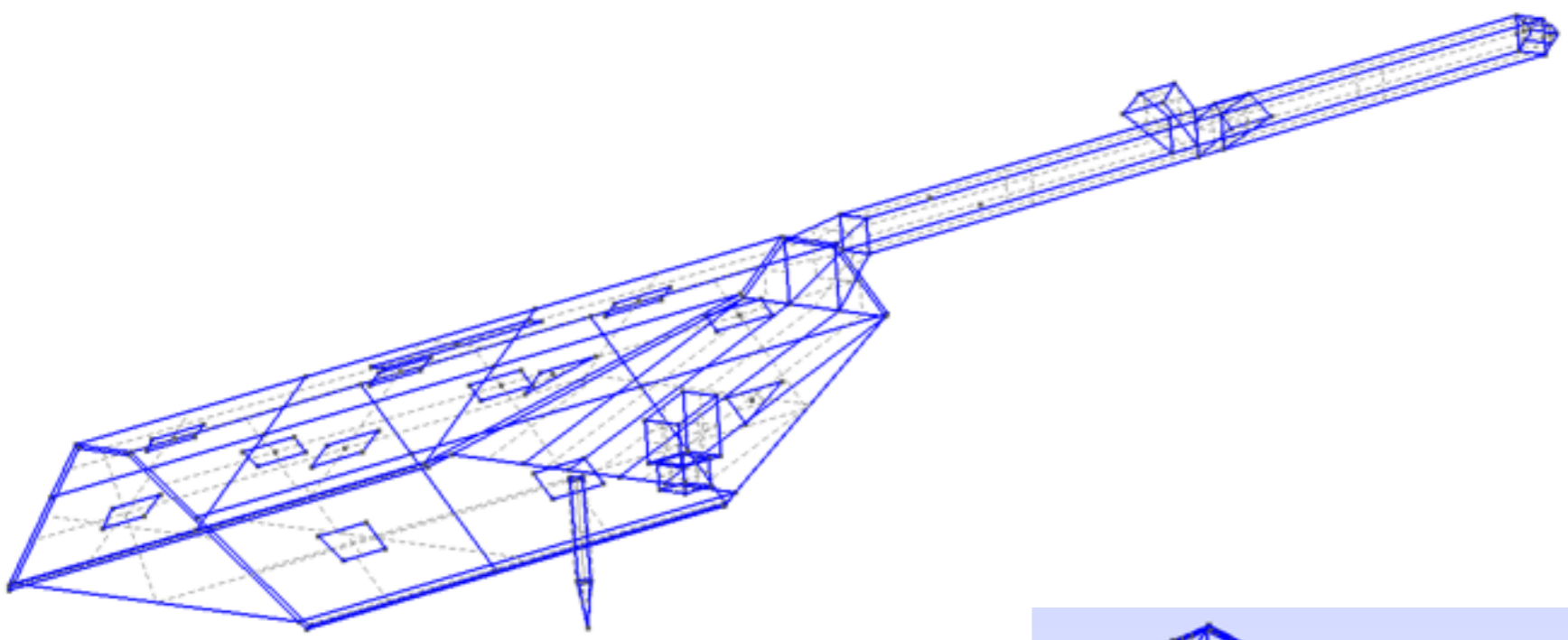
Conversion loop

SPIS-to-STEP_SPE-to-SPIS loop

SPIS-to-STEP_SPE
    Conversion time = 10s
    Output STEP-file = 3Mo

STEP_SPE-to-SPIS
    Conversion time about 15s

Visualisation done in Gmsh

STEP-SPE

SPIS

STEP-to-SPIS

SPIS-to-STEP
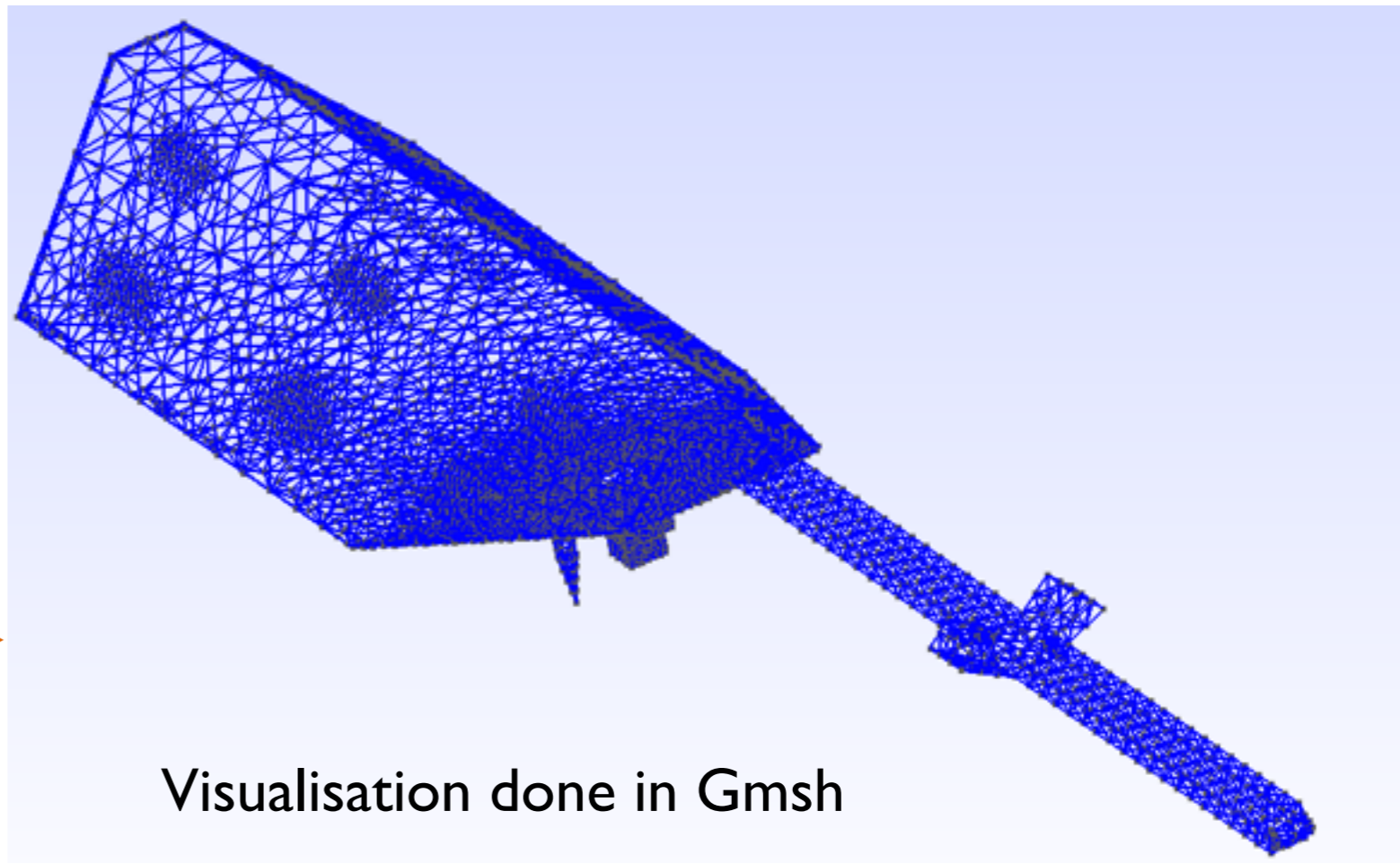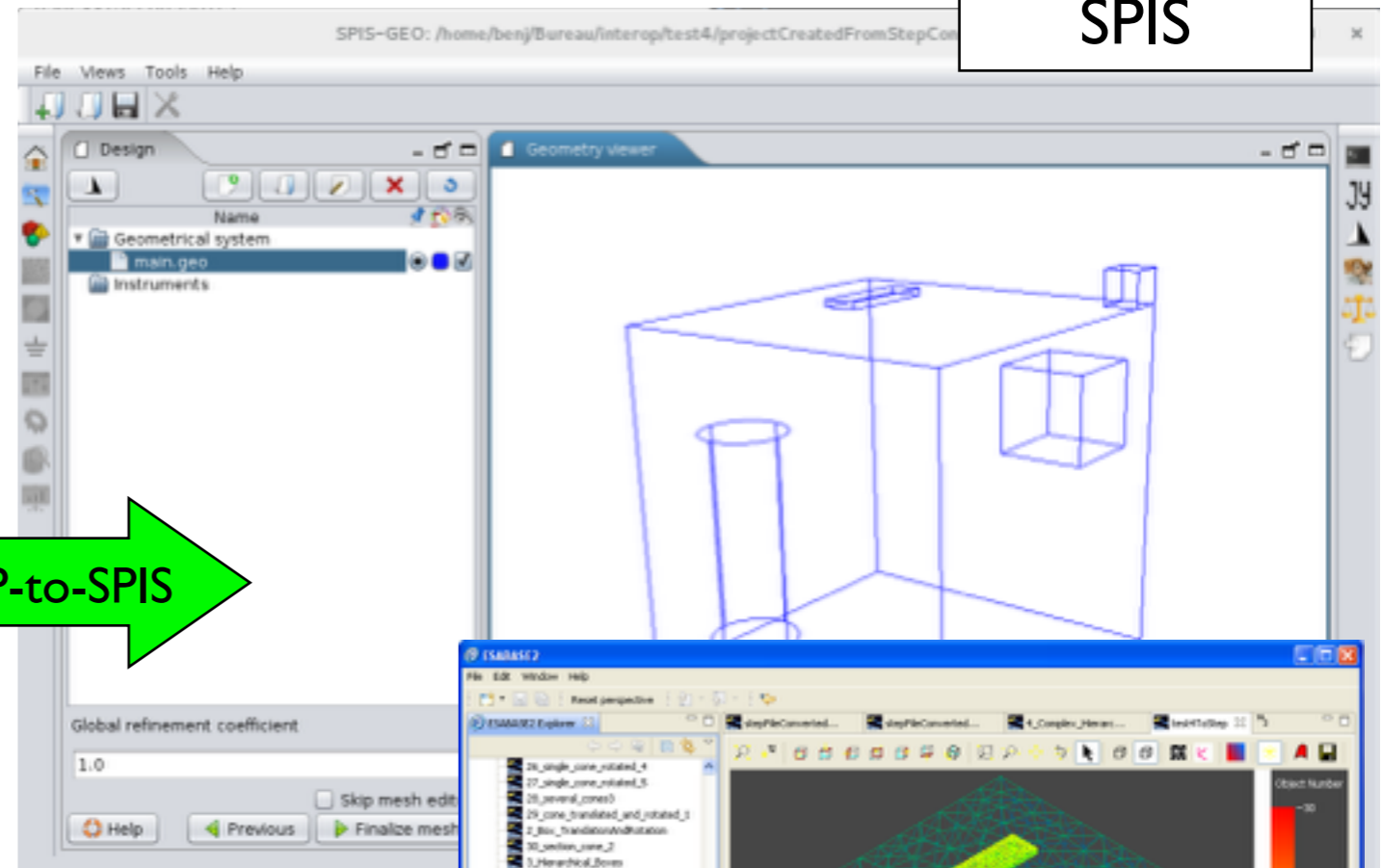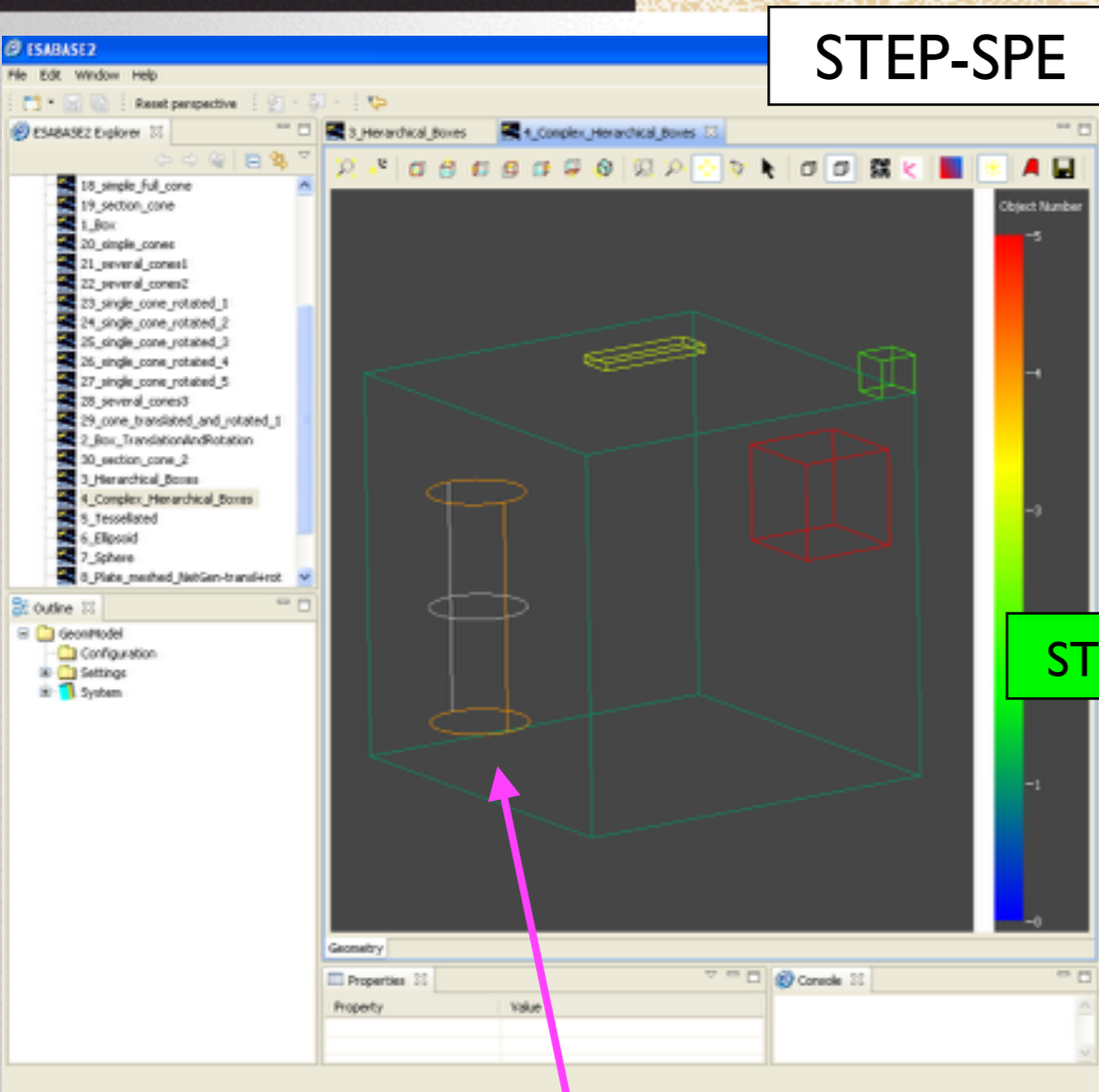
STEP-SPE

On the way back (STEP-to-SPIS), one surface Gmsh physical per STEP-SPE concept.
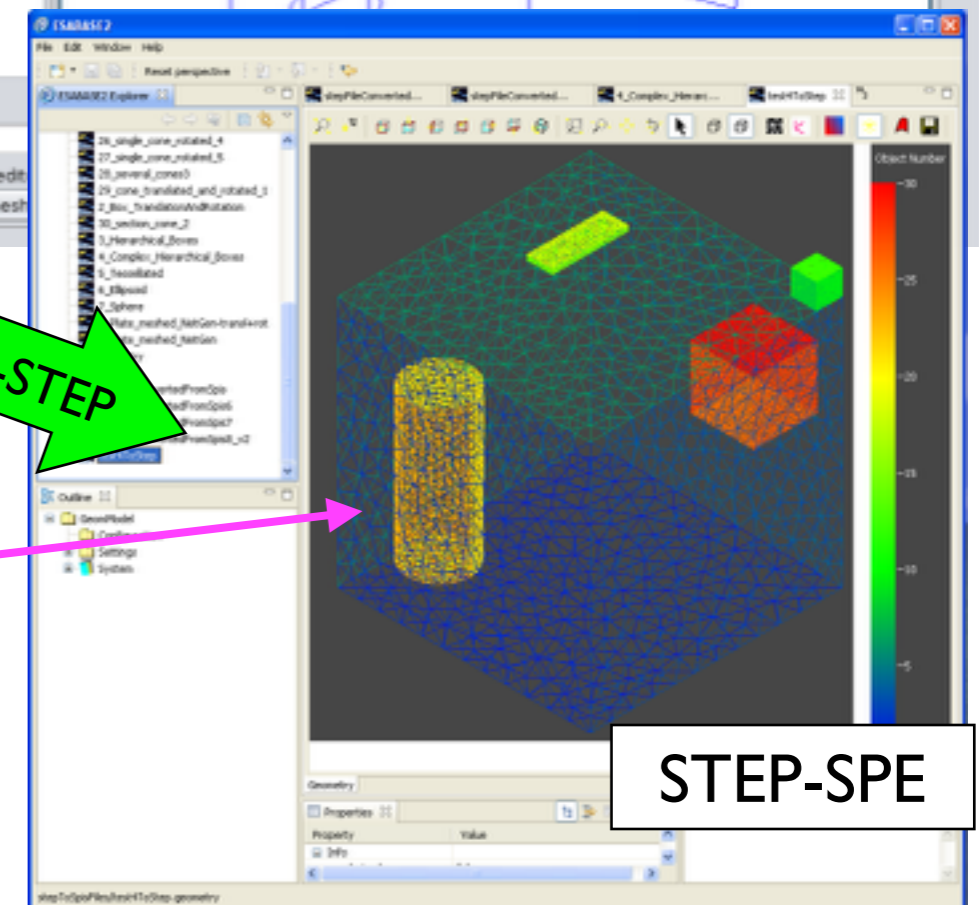
STEP-to-SPIS options
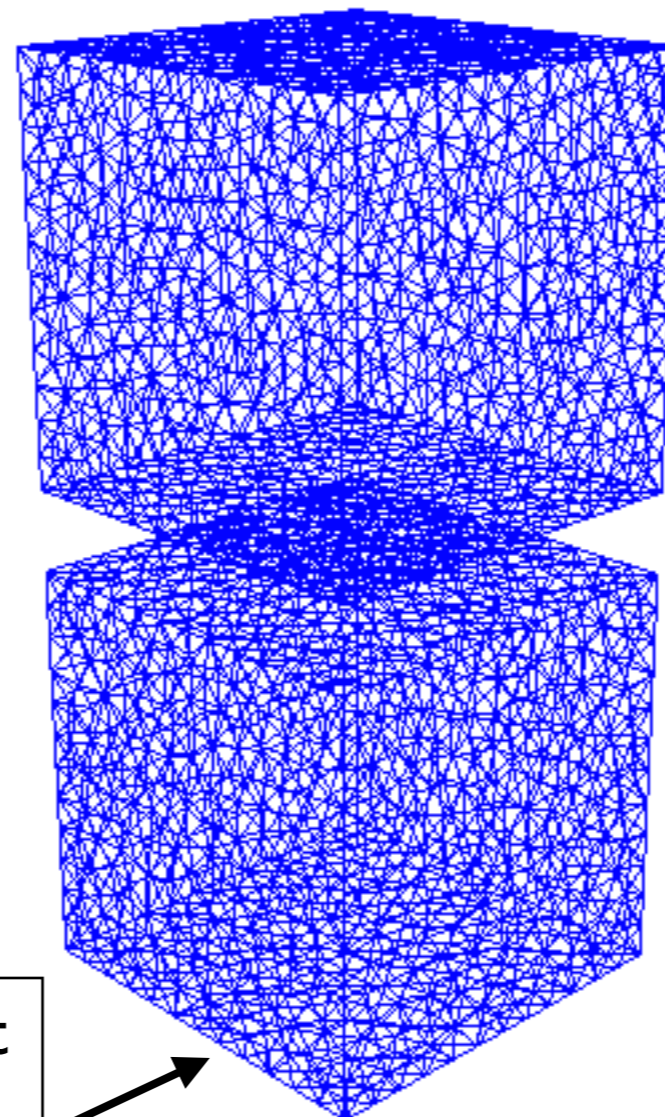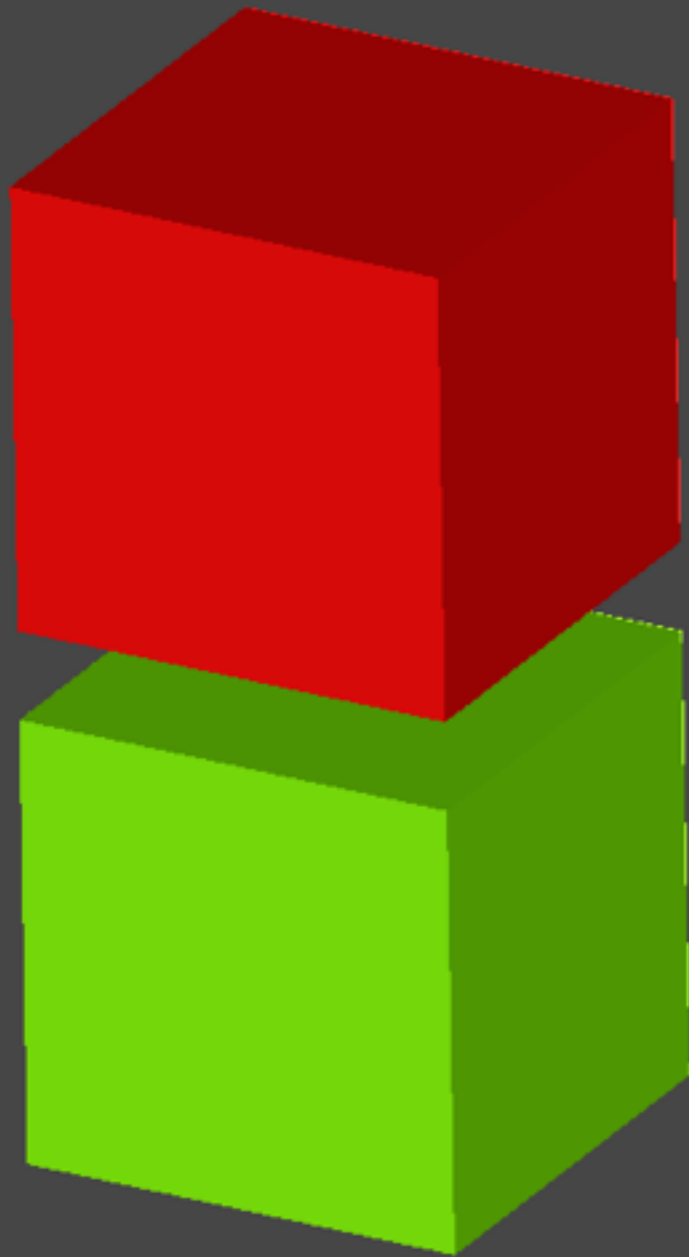
SPIS's Mesh Editor view
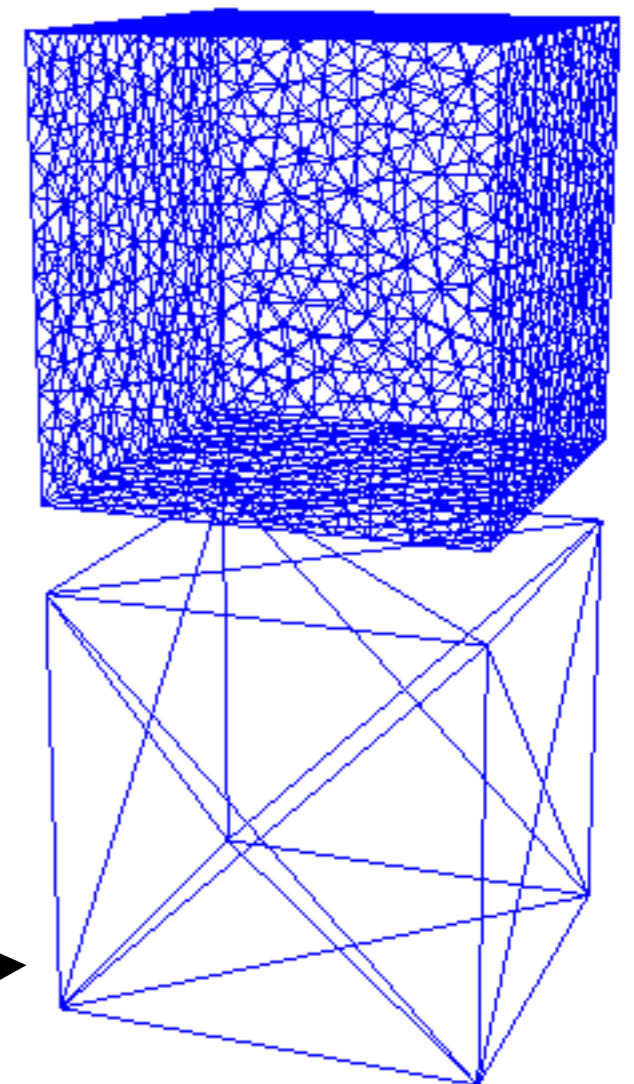
Various mesh resolution per input concept on the generated B-Rep CAD model

- **In case of conversion through tessellated surface, the continuity and the self-consistency of the mesh is a critical point**
  - Explicitly demanded by the community (Airbus, ONERA…)
  - Needed by most of simulation tools (like SPIS) to rebuild the mathematical boundaries of the computation domain
- **This means:**
  - Avoid duplicate elements
  - Provide or rebuilt the connectivity between neighboring elements
- **But not supported by the STEP-SPE protocol**
  - Only set of independent triangles
  - Duplication of nodes (vertices of triangles)
- **To solutions are possible:**
  - Use the duplicated node detection / fusion capability of recent Gmsh versions (experimental)
  - Use Artenum's GEFICOT tool (recently released)

See, http://dev.artenum.com/projects/GEFICOT

Re-built .geo file with continuous surface and no duplication

Spacecraft
(inner boundary)

External boundary
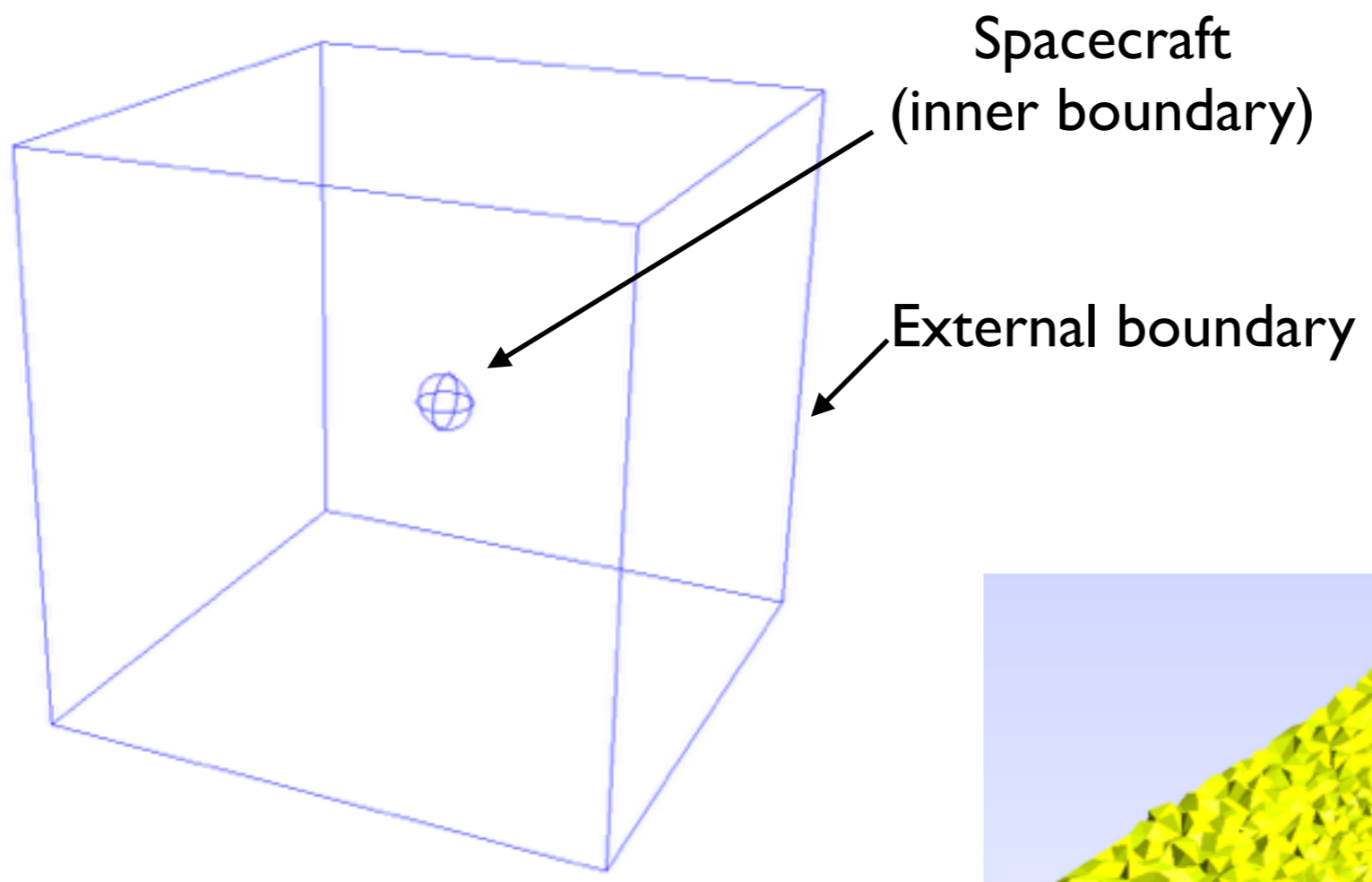
SPIS-to-STEP_SPE-to-SPIS loop

Reconstruction of the
computational domain
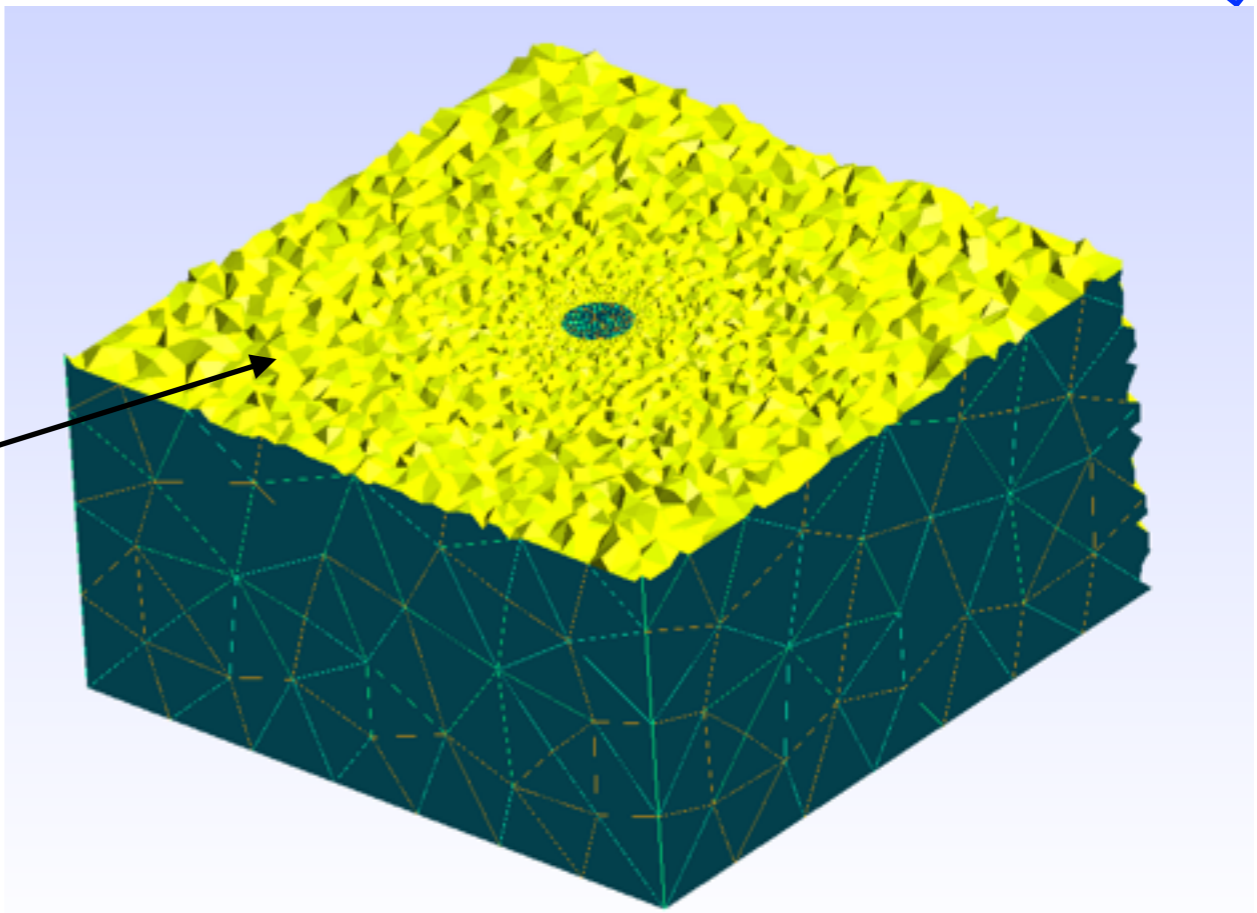
Visualisation done in Gmsh
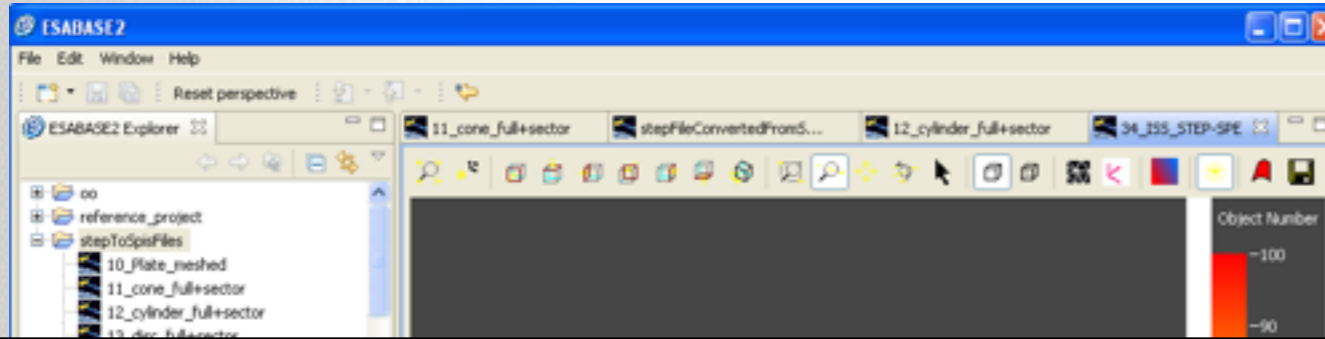
ISS model initially done in esabase 2
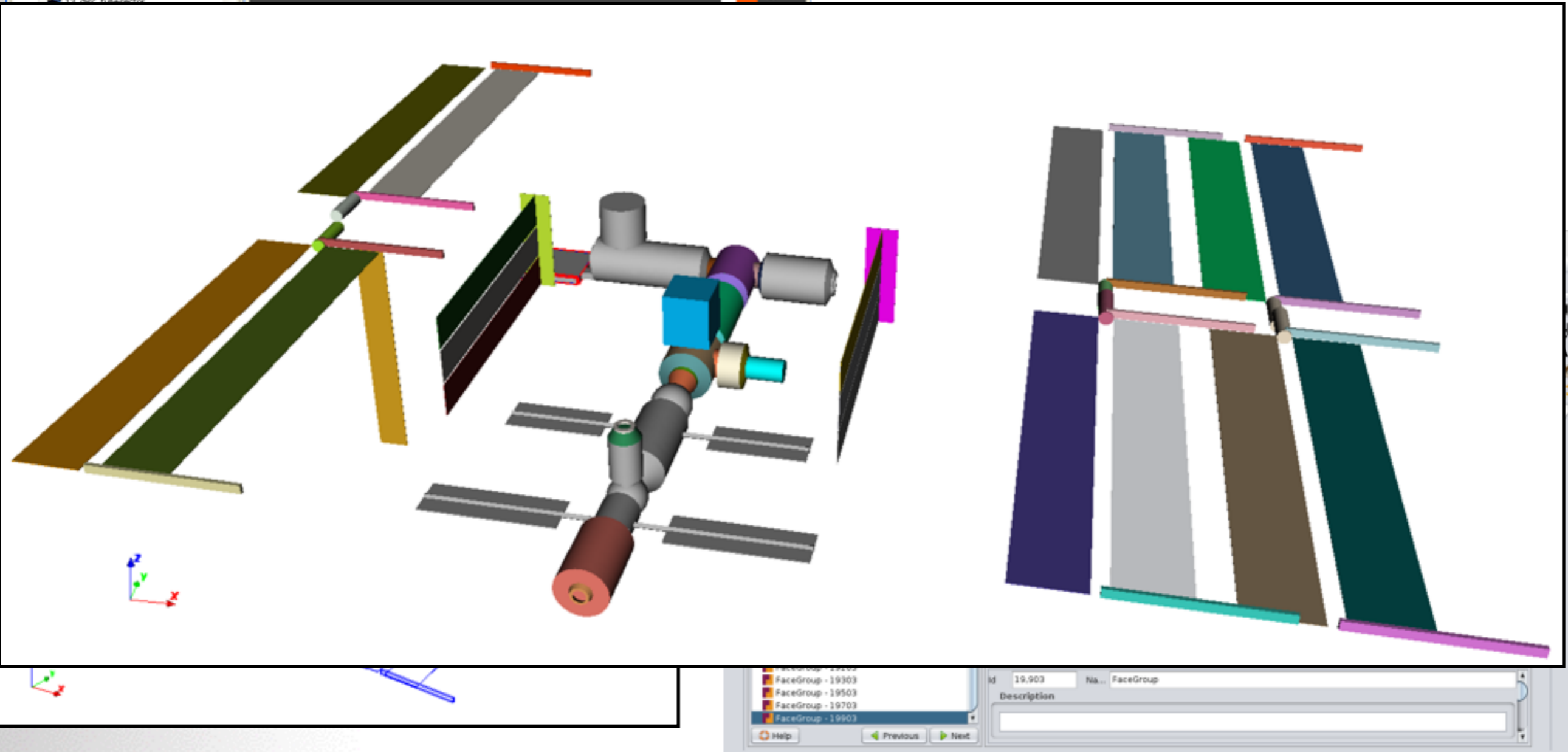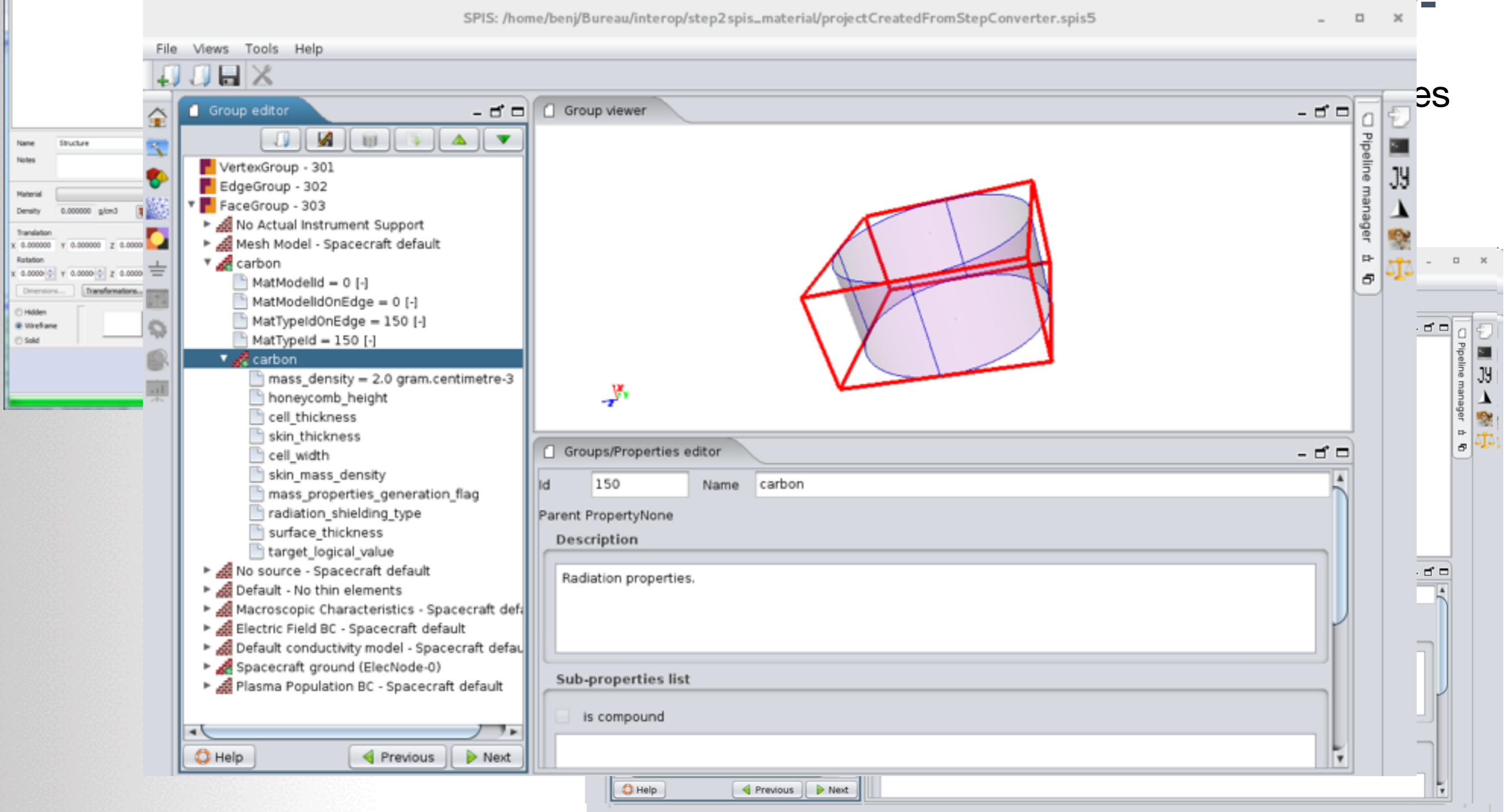
- **Tested from FasTrad (not supported by esabase 2)**
- **Conversion supported from STEP-**

Material properties conversion

- **Confirm the cross compliance between STEP-SPE and Frida for the material properties, at least on:**
  - The supported data structures
  - The functional richness
- **The import of material properties from STEP-SPE to SPIS is done « as-it » and without interpretation**
  - The loaded material, typically coming from another domain of expertise, does not fulfill the required of a « spis-material » description and characteristics are not rename properly
  - As direct geometrical attributions via the groups list, characteristics are imported directly without interpretations nor analysis.
- **Additional filtering/analysis/conversion modules, or a good expert user, are still needed for a proper materials exchanges in multi-SEAT context.**

- **WP dedicated to the validation**
  - Intense effort
  - Mainly done by etamax in double-blind approach
  - Tested against several reference tools:
    - esabase 2
    - FastRad
  - Necessity to define an objective metric
    - On the accuracy of the converted shape
    - Identification of lost data
  - Focused on the conversion of geometrical concepts
  - Software validation done on various platforms (Linux, Windows…)

XXXXXX
- XXXXx

Conclusion (I)

- **A first version of a STEP-SPE to/from SPIS converter has been developed and is operational!**

- A normalised packaging of developed codes done in the perspective of a future wide distribution into the community

- First tests are successfully passed for key geometrical concepts and including realistic hierarchised geometries and large size model

- The STEP-SPE to SPIS conversion of CAD model exact (or almost) using the Keridwen's CSG-to-BRep decomposition approach.

- Most of concepts listed in UR-DC-003 supported

- SPIS-to-STEP-SPE conversion of CAD model through tessellated surfaces

- SPIS-to-STEP_SPE-to-SPIS loop validated modulo the tessellation

- STEP-SPE to SPIS conversion of material properties

- The functional scope of the SPIS model surpasses the current capabilities of STEP-SPE, but this first implementation still confirm the approach and the global cross-compliance of STEP-SPE and SPIS.

- Bridges toward other simulation codes in space environment have been initiated.

- If the whole spis data model is not supported, all information needed to rebuild/exchange the geometrical description of the modeled system are supported by the converters.

- Possibility to rebuild the computational volume in SPIS

- This first version aims to be the support for:

  - Iterate on this basis with the community to identify and analyze the real usages.

  - See the need in case of real multi-physics / multi-models uses (e.g. CIRSOS).

- Results of this application case and your feedbacks will be usefull to improve/extend the STEP-SPE protocol.

Please contact ESA's technical officer or etamax prime manager for further informations:
- Fabrice Cipriani (Fabrice.Cipriani _at_ esa.int)
- Karl Dietrich Bunte (k.bunte _at_ etamax.de)