

# SPIS 6

## User Manual

	Name	Date	Signature
<b>Prepared by</b>	Benoît Thiébault, Jean-Charles Mateo Velez, Julien Forest, Pierre Sarrailh	29/10/2013	
<b>Verified by</b>	Benjamin Jeanty-Ruard,	29/10/2013	
<b>Approved by</b>	Julien Forest, David Rodgers, Alain Hilgers, Fabrice Cipriani	29/10/2013	
<b>Updated by</b>	Benjamin Jeanty-Ruard	28/11/2019	

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

## Presentation

---

### Objectives

The objective of this document is to provide the user of SPIS the relevant information to understand how to install, launch and use the application.

This document also introduces to the use of additional tools and functions, like the post-analysis tools, material properties settings and logging system. The main data structures handled by the central Data Model and shared through the framework are also presented in a last part.

The import/export functions, especially with-respect-to the previous and other version of SPIS, is presented in details.

This document focuses on the main functionalities of SPIS. Please refer to the relevant annexes for the settings specific at each supported branch and especially for the settings of the numerical kernel.

This documentation has been initiated in the frame of the *Simplified Standard MEO/GEO Tools for Spacecraft Charging activity* (ITT ref. AO/1-6218/09/NL/AT, contract Nr 4000101174), here after called SPIS-GEO. It in this frame it constitutes the deliverable # 7.

It has been updated and extended according the software evolutions done in the frame of SPIS-Science project (contract Nr 4000102091/10/NL/AS). In this frame, it constitutes the deliverable # 5.

It has been updated and extended according the software evolutions done in the frame of SPIS-Propulsion project (contract Nr 4000116103/15/NL/LF).

This UM is compliant with the version 6 of SPIS, which incorporates models adapted to surface charging at GEO/MEO and advanced scientific tools. This includes the SPIS-GEO/MEO, SPIS-SCI and propulsion branches.

We remind that SPIS-6 is an expert application. For this reason, the present document will only present the step-by-step procedure to perform a whole simulation and extract the produced result. The relevance of these results remains of the responsibility of the user.

### Technical content summary

This document is divided in two main parts:

- The prerequisites details the hardware and software requirements to run the application, explains how to install and launch the application.
- The second part explains how to use the application and details its main functions.

Annexes detail specific uses:

- Annex 1: Surface charging in GEO/MEO
- Annex 2: Advanced use for scientific applications

### Diffusion

Name	Organisation
No restriction	No restriction

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

## Changes

Version	Revision	Date	Auteur / Observation
1	0	27/08/2012	Benoît Thiébault / Document creation
2	0	16/12/2012	Julien Forest / Update for SPIS 5.0.0
2	3	02/04/2013	Jean-Charles Matéo-Vélez / Update for numerical kernel
2	5	11/04/2013	Jean-Charles Matéo-Vélez / Update for numerical kernel
3	0	15/04/2013	Julien Forest, merge of all branches evolutions
3	1	16/04/2013	Jean-Charles Matéo-Vélez and Pierre Sarrailh / corrections
3	2	23/04/2013	Jean-Charles Matéo-Vélez / graphics
3	3	26/06/2013	Julien Forest / section 9.4 and gmsk versions
3	4	25/10/2013	Benoît Thiébault, Benjamin Jeanty-Ruard and Jean-Charles Matéo-Vélez / known bugs
3	5	28/11/2019	Benjamin Jeanty-Ruard / Update document with features developed for SPIS propulsions

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

This page is left intentionally blank.

## Table of contents

<b>1. INTRODUCTION .....</b>	<b>8</b>
1.1. SPIS, the Spacecraft Plasma Interaction Software .....	8
1.2. Applicable documents .....	9
1.3. Reference documents.....	9
1.4. Acronyms, abbreviations and definitions .....	9
<b>2. KEY CONCEPTS .....</b>	<b>10</b>
2.1. SPIS for Spacecraft Plasma Interaction Software .....	10
2.2. Caution and limit of warranty .....	10
2.3. Concept of Integrated Modelling Environment (IME).....	11
2.4. Modelling chain .....	11
2.5. Framework design and OSGi context.....	12
<b>3. PREREQUISITES.....</b>	<b>13</b>
3.1. Configuration .....	13
3.1.1. Hardware requirements.....	13
3.1.2. Software requirements .....	13
3.2. Get the software, the SPINE community platform.....	14
3.3. Installation.....	15
3.4. Running the application.....	16
<b>4. USING THE APPLICATION.....</b>	<b>17</b>
4.1. User interface .....	17
4.1.1. Overview.....	17
4.1.2. A wizard-based approach.....	17
4.1.3. Menu bar.....	18
4.1.4. Top tool bar .....	19
4.1.5. Left side tool bar.....	19
4.1.6. Right side tool bar.....	20
4.2. Exiting the application .....	20
4.3. Monitoring of the application with the Logs console.....	20
4.4. Controlling the application with Jython .....	21
<b>5. A PROJECT-BASED APPROACH .....</b>	<b>22</b>

Reference: ESA-SPIS-GEO-D7-SUM-2013-03-001	Version: 3	Revision: 5
Status: Final	Date: 28/11/2019	

5.1.	<b>Project structure</b> .....	<b>22</b>
5.2.	<b>Creation of a new project</b> .....	<b>23</b>
5.3.	<b>Open an existing project</b> .....	<b>25</b>
<b>6.</b>	<b>STEP-BY-STEP MODELLING PROCESS</b> .....	<b>26</b>
6.1.	<b>Pre-processing</b> .....	<b>26</b>
6.1.1.	<i>Geometrical modelling</i> .....	27
6.1.2.	<i>Meshing edition and control panel</i> .....	31
6.1.3.	<i>Groups edition and properties settings</i> .....	33
	i. <i>Properties editions</i> .....	39
	ii. <i>Materials definition</i> .....	39
	iii. <i>Properties deployment and DataFields generation</i> .....	41
6.1.4.	<i>Internal electrical circuit setting</i> .....	42
6.1.5.	<i>Global parameters settings</i> .....	43
6.1.1.	<i>Transitions modelling</i> .....	44
6.1.2.	<i>Model finalization and SPIS-NUM model building</i> .....	45
6.2.	<b>Quick start with the numerical kernel</b> .....	<b>46</b>
6.2.1.	<i>Overview and quick start settings</i> .....	46
6.2.2.	<i>Integration loops control - How to set time steps?</i> .....	51
6.2.3.	<i>Methodology for two typical situations</i> .....	54
6.2.4.	<i>Where should I put my input ASCII files?</i> .....	55
6.2.5.	<i>Where do I get extra ASCII files results?</i> .....	56
6.2.6.	<i>Where is the documentation of the full list of parameters?</i> .....	56
6.3.	<b>Simulation running and control</b> .....	<b>56</b>
6.4.	<b>Post-processing and data analysis</b> .....	<b>58</b>
6.4.1.	<i>Data miner</i> .....	58
6.4.2.	<i>Cassandra, 3D visualization tool</i> .....	60
6.4.3.	<i>Views and screenshot saving</i> .....	63
6.5.	<b>Reporting</b> .....	<b>63</b>
<b>7.</b>	<b>INSTRUMENTS (GENERAL CONFIGURATION AND USE)</b> .....	<b>65</b>
7.1.	<b>Principle of the SPIS instruments</b> .....	<b>65</b>
7.2.	<b>Requesting a new instrument</b> .....	<b>67</b>
7.3.	<b>General settings</b> .....	<b>68</b>
7.4.	<b>Live Monitoring specific settings</b> .....	<b>70</b>
<b>8.</b>	<b>ADVANCED USES</b> .....	<b>74</b>
8.1.	<b>Batch mode</b> .....	<b>74</b>
8.2.	<b>Thin elements modelling</b> .....	<b>74</b>
8.2.1.	<i>Concept of thin elements</i> .....	74
8.2.2.	<i>Thin wires (1D)</i> .....	74

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

8.2.3.	<i>Thin surfaces (2D)</i> .....	77
<b>8.3.</b>	<b>How-to-import of SPIS-Legacy projects</b> .....	<b>81</b>
<b>9.</b>	<b>OPTIMIZATION AND ADVANCED SETTINGS</b> .....	<b>82</b>
9.1.	3D visualization troubleshooting and degraded mode .....	82
9.2.	How to increase the memory stack? .....	82
9.3.	My computer is multi-core, does SPIS use all them? .....	83
9.4.	Property file and configuration of native components .....	83
<b>10.</b>	<b>KNOWN BUGS</b> .....	<b>85</b>
<b>ANNEX A</b>	<b>DATAFIELD/MESHMASK STRUCTURE AND FILE FORMATS</b> .....	<b>87</b>
<b>ANNEX B</b>	<b>EXAMPLE OF PROPERTIES CATALOGUE FILE</b> .....	<b>91</b>
<b>ANNEX C</b>	<b>LAUNCHING SCRIPT</b> .....	<b>95</b>
<b>ANNEX D</b>	<b>EXAMPLE OF GLOBAL PARAMETER FILE</b> .....	<b>96</b>
<b>ANNEX E</b>	<b>EXAMPLE OF CAD TOOL IN GEO GMSH FORMAT</b> .....	<b>98</b>
<b>ANNEX F</b>	<b>EXAMPLE OF CAD TEMPLATES</b> .....	<b>101</b>
<b>ANNEX G</b>	<b>LIST OF MAIN SPIS MATERIAL PROPERTIES</b> .....	<b>103</b>
<b>ANNEX H</b>	<b>LIST OF PROVIDED NASCAP BASED MATERIALS</b> .....	<b>105</b>

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

## 1. Introduction

---

### 1.1. SPIS, the Spacecraft Plasma Interaction Software

Initially released in March 2004, the SPIS software has become today the European standard for the modelling and the simulation of the spacecraft plasma interactions. Initially funded on an ESA effort and following an open-source approach in the frame of the SPINE community, SPIS knows today a real and dynamic community life. SPINE counts about 600 registered members today, inside and outside EU. The average number of posts on the forums overtakes 70 messages per month.

Originally designed to focus on scientific applications, the application scope of SPIS is largely wider now and is regularly extended to new engineering applications or domains of physics. This includes, for instance, the modelling of electrical propulsion systems, ESD prediction on solar arrays or link with radiation models through deep charging phenomena.

The prediction of the electrostatic charge (absolute and relative) of spacecraft for engineering purposes is also a key issue with modern platforms that are more and more complex, operating high-power and sensitive electronic devices or using modern materials. Differential charging can lead to arcing, dangerous for the electronic payload. This need is especially critical for GEO and MEO missions, where are located most of the commercial platforms. Moreover, the progressive generalisation of electrical propulsion systems on commercial platforms pushes integrators to perform much more detailed electrostatic analysis before the flight.

Legacy tools currently used in the industry, cannot address these modern constraints. Thanks to its modularity and the implemented models representing the present state-of-the-art in plasma-surface interactions, SPIS is currently probably the best basis to address these issues in a self-consistent manner. There is a real need of a specific version of SPIS, called SPIS-GEO/MEO, to model MEO and GEO missions that would simplify its usage in an engineering context.

SPIS-GEO/MEO has been developed in consortium by Artenum (SPIS-UI) and ONERA (SPIS-NUM), evaluated and validated by OHB-Sweden and ASTRIUM-France with the support of the European Space Agency (ESA) in the frame of the contract Nr 4000101174 - Order Nr AO/1-6218/09/NL/AT).

In addition, scientific tools were added by ONERA and ARTENUM to SPIS in order to answer SPINE community needs in term of accuracy, performance and advanced scientific capabilities, evaluated and validated by IRAP-France and IRFu-SWEDEN, in the frame of the SPIS-SCIENCE ESA contract Nr 4000102091/10/NL/AF - Order AO/1-6368/10/NL/AF).

New features have developed to be able to model the electric propulsion and its interaction with the surrounding plasma in the frame of the ESA contract 4000116103/15/NL/LF. A full refactoring of the SPIS Numerical kernel has been realized in this contract. Because of that, the new SPIS version is 6.X.X.

SPIS-GEO/MEO and SPIS-SCIENCE activities were based on the version 5.0 of the main branch of SPIS, which is maintained in the frame of the SPINE ([www.spis.org](http://www.spis.org)) open-source community.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



## 1.2. Applicable documents

[AD1] Statement Of Work “Simplified MEO/GEO Tools for Spacecraft Charging”, TEC-EES/2008.348/DR, issue 1.0, 30/09/2009.

[AD2] Technical and Administrative Proposal, ESA-SPISGEO-PTC-2009-12-001, 20/11/2009

[AD3] Software Requirements Document, ESA-SPISGEO-D3-SRD-2010-11-001.

[AD4] Guide to applying the ESA software engineering standards to small software projects, BSSC(96)2 Issue 1, 1996.

[AD5] Guide to the software architectural design, ESA PSS-05-04 Issue 1 Revision 1, March 1995

[AD6] ESA - Software engineering and standardisation, ECSS E-40, part 1 and 2, 31 March 2005

## 1.3. Reference documents

[RD1] SPINE community Web site, [www.spis.org](http://www.spis.org)

[RD2] Artenum’s Keridwen Web site, <http://www.artenum.com/EN/Products-Keridwen.html>

[RD3] SPIS services Web site, <http://www.spis-services.eu/>

[RD4] Apache Felix OSGI runtime Web page, <http://felix.apache.org/site/index.html>

[RD5] Gmsh User Manual, <http://geuz.org/gmsh/doc/texinfo/gmsh.html>

[RD6] Artenum, SPIS Project Documentation, Gmsh User Tutorial for CAD and Meshing, Issue: 1.0, October 2003.

[RD7] ONERA, SPIS Project Documentation, *How to control NUM from UI*, 2013.

[RD8] ONERA, SPIS Project Documentation, *How to define spacecraft equivalent circuit*, 2012.

[RD9] Cassandra Web site, <http://dev.artenum.com/projects/cassandra>

[RD10] Penelope Web page, <http://www.artenum.com/EN/Products-JFreeMesh.html>

[RD11] Frida Web page, <http://dev.artenum.com/projects/frida>

## 1.4. Acronyms, abbreviations and definitions

- **TBC**: to be confirmed
- **TBD**: to be determined
- **IME**: integrated modeling environment
- **IBC**: initial and boundary conditions

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

## 2. Key concepts

### 2.1. SPIS for Spacecraft Plasma Interaction Software

The Spacecraft Plasma Interaction Software, SPIS, is a simulation software based on a numerical kernel, called SPIS-NUM, an electrostatic 3D unstructured Particle-In-Cell plasma model consisting of a Java-based highly modular Object Oriented library. More accurate, adaptable and extensible than the existing simulation codes, SPIS is designed to be used for a broad range of industrial and scientific applications. The simulation kernel is integrated into a complete modular pre-processing/computation/post-processing framework, called SPIS-UI, allowing a high degree of integration of external tools, such as CAD modellers, meshers and 2D/3D visualization tools.

SPIS-GEO/MEO and SPIS-SCIENCE represent evolutions of the SPIS trunk especially adapted to the modelling of complex orbital structures in Geosynchronous Earth Orbit (GEO) and Medium Earth Orbit (MEO), and to advance use for scientific applications, respectively. In order to be used in an industrial context, the whole user interface (UI) has been deeply reviewed and simplified with respect to the previous versions of SPIS (hereafter called SPIS-Legacy) with the hope to simplify its use by new users. In order to be used by a scientific community, extended capabilities in term of accuracy and performance have been developed both at NUM and UI levels.

SPIS-GEO/MEO and SPIS-SCIENCE are based on the general refactoring of SPIS done in the frame of the version 5.x.x of the software. SPIS-UI is based on the general Integrated Modelling Environment (IME) Keridwen, developed by Artenum [RD2].

Developed in an Open Source approach and oriented towards a future community-based development, SPIS is already partially available for the whole community and is used by members of the European SPINE network [RD1]

SPIS should address a large majority of the new challenges in spacecraft plasma interactions, including the environment of electric thruster systems, solar arrays plasma interactions, commercial missions and accurate calibration of scientific plasma instruments.

### 2.2. Caution and limit of warranty

The present document is intended to be a simple User Manual (UM) to help the user to use SPIS in basic mode and follow a step-by-step simple simulation. We remind that SPIS is still an expert tool. This document does not attempt to provide to the future SPIS user the whole expertise and knowledge needed to perform the complete modelling of a spacecraft. The final confidence in the relevance and accuracy of simulation results depends on the user expertise and remains his responsibility.

FOR THIS REASON, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING, THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

In spite of the simplification of the user interface, it is still strongly recommended to have good knowledge in space environment, space engineering and mathematical modelling before any simulation of realistic and/or critical systems is performed.

For an improved handling of the key concept addressed by SPIS, we strongly recommend to contact ESA and ONERA space environment experts and/or follow one of the training programs proposed by ONERA and Artenum [RD3].

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

For further information, please contact:

- General expertise contact: [contact@spis.org](mailto:contact@spis.org)
- SPIS trainings offers [contact@artenum.com](mailto:contact@artenum.com) or see [www.artenum.com/services](http://www.artenum.com/services)

The present document will focus on the use of SPIS-UI, as a modelling framework.

### 2.3. Concept of Integrated Modelling Environment (IME)

SPIS-UI is the user interface of the SPIS system. The functions of SPIS-UI include the possibility of definition of all inputs parameters (CAD geometry, material properties, numerical settings...), during the pre-processing phase, the control of each step along the whole modelling chain (Figure 1) and the processing of outputs after the simulation. The SPIS-UI framework offers the possibility to access all shared data and dynamically edit and reload models and processing modules.

### 2.4. Modelling chain

One of the main objectives of the SPIS-UI framework is to help the user to follow the modelling chain and perform each step in a consistent approach in order to guarantee the results of the simulation as much as possible. The most common process starts with the definition of the geometry of the spacecraft with the Computer Assisted Design (CAD) module and the attribution of surfaces properties and boundary conditions. It is then necessary to define the spacecraft electrical circuit and the global parameters of the simulation. This phase corresponds to the pre-processing phase. At its end, the system to model should be fully defined and ready to be loaded into the simulation kernel, SPIS-NUM, for the simulation phase. The modelling process is closed by the post-processing phase with the access to several data analysis modules and 2D/3D visualization tools. Figure 1 illustrates this principle.

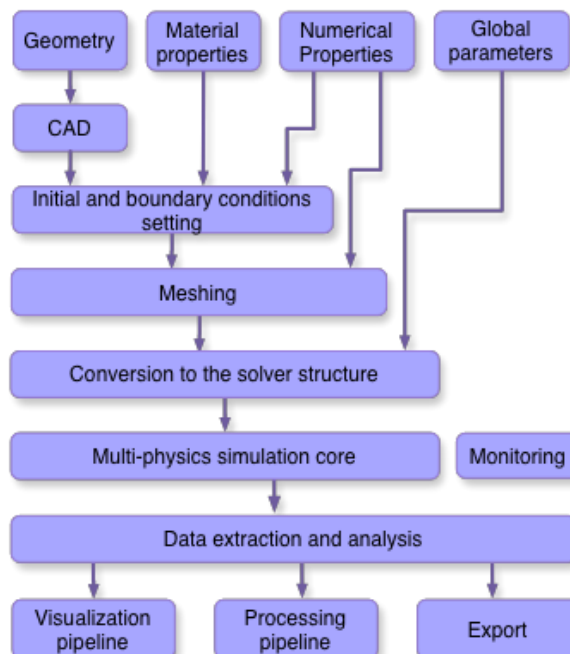


Figure 1: Schematic modelling chain

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

## 2.5. Framework design and OSGi context

As explained above, SPIS is divided into two main layers, SPIS-NUM, the numerical kernel and SPIS-UI the IME dedicated to the pre and post-processing phases.

SPIS-UI is built on top of Artenum Keridwen Integrated Modelling Environment (IME), a generic framework. Keridwen is both a generic IME, used in a large scope of scientific thematic, and a set of tools to help users and developers to build pre/post-processing chains of simulation software but also real data processing chains. To offer a modular design and a high degree of inter-operability with other existing processing tools, Keridwen is based on the OSGi standard. Keridwen can be run on any runtime respecting the OSGi standard. This includes the GUI, which is also embedded as an OSGi bundle and can be then easily replaced. In practice, the current version of the software is deployed on the Apache Felix runtime [RD4].

In such an approach, each tailored module is embedded in a standardized module, called bundle, which can be dynamically loaded into a central kernel. Tailored bundles are then available as services. This choice allows to easily adapt and extent the framework according the evolutions of the users' needs.

Behind Keridwen, several external libraries are used for specific functions. The main ones are:

- **Artenum Penelope library** to store and handle meshes and fields deployed on the meshes;
- **Artenum Frida library** to handle groups and properties deployed on groups;
- **Docking Frames** for the management of frames and views;
- **Artenum ArtTk** for scientific GUI and widgets.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

## 3. Prerequisites

---

### 3.1. Configuration

#### 3.1.1. Hardware requirements

The minimal hardware configuration to properly run SPIS is:

- Intel/AMD x86 32/64bit processor, with a frequency higher than 1.7GHz (multi-core CPU with frequency higher than 2.4 GHz recommended)
- 2Gb of RAM (more than 8Gb recommended, especially for large grids)
- Graphic card with more than 128Mb of video memory (NVIDIA Quadro FX 570M or equivalent recommended)
- Screen resolution of 1024x768 pixels (1280x1024 resolution recommended)
- 500Mb of free space on the hard drive

#### 3.1.2. Software requirements

SPIS 5 has been successfully tested on the following configurations. However, this list remains indicative, some native components, like the 3D rendering, may deeply depend on the local system configuration (e.g. version of Glibc or OpenGL libraries).

**Tableau 1: List of supported platform for SPIS**

Operating System	Version	Architecture
Linux Fedora	16 and higher	64 bits
Linux Ubuntu	10.10 and higher	64 bits
Linux Cent OS	5.4 and higher	64 bits
Linux Mint	13.0	64 bits
Apple OSX	10.6 (Snow Leopard)	64 bits
Microsoft Windows	XP SP3	32 bits
Microsoft Windows	7	64 bits

SPIS 6 has been successfully tested on the following configurations. However, this list remains indicative, some native components, like the 3D rendering, may deeply depend on the local system configuration (e.g. version of Glibc or OpenGL libraries). For further details about the supporting OS, please refer to the SPINE website.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

**Tableau 2: List of supported platform for SPIS**

Operating System	Version	Architecture
Linux Debian	8.7	64 bits
Linux ArchLinux	Kernel 5.3.13-arch1-1	64 bits
Apple OSX	10.10.5	64 bits
Microsoft Windows	10	32 bits
Microsoft Windows	10	64 bits

**Remark:** Some instability due to the 3D rendering has been observed on some old Linux systems, especially with ATI like video cards (including the video emulation of some virtual machines). This is due to an improper support of the OpenGL/JavaSwing interface and is deeply dependent on the used driver. More especially the open-source ATI and Vmware drivers may lead to instabilities. We recommend the official drivers or the basic VESA one in place. Please see the X11/Xorg documentation for further information.

**Remark:** On UNIX based systems; only 64 bits architectures will be supported.

On linux OS, a problem may occur when JBlas functions are called in SPIS if gfortran is not installed. An error should appear in the terminal explaining how to solve the problem. Usually, it is necessary to install the gfortran library.

Java jre 8 is embedded with SPIS and is not required to be installed before to run SPIS 6. Be careful, with old Linux version, the jre embedded with SPIS may not work. In this case the Java Jre1.8.0\_73 is recommended.

If you wish to compile the source codes or use you own JRE, please visit the Oracle Sun web site (cf. <http://www.java.com>) and download the latest relevant java 8 version. Please see the SPINE platform ([www.spis.org](http://www.spis.org)) for further information regarding the compilation of SPIS or contact Artenum's technical team (contact@artenum.com).

## 3.2. Get the software, the SPINE community platform

As all SPIS releases, SPIS-6 is hosted on the SPINE community platform at the following address:

<http://www.spis.org>

A registration as SPINE member may be needed before to download the software. Registration demands are available at the following address:

<http://dev.spis.org/projects/spine/home/captcha>

The core elements of SPIS are open-source software (see the Licenses directory or file). However some additional components might be diffused under close-source and/or proprietary license and/or have diffusion restriction, please check the licenses terms before any use and if necessary, please contact the ESA contact officer or the SPIS SDAB contact person at the following address:

**contact@spis.org**

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

### 3.3. Installation

The application should be provided in a compressed archive (.zip or .gz for instance). Simply uncompress the archive to the desired location of your hard drive. The folder obtained after decompression is illustrated on **Figure 1**.

```

Spis-5.1.0-beta1          -----> SPIS ROOT directory
├── COPYRIGHT.txt
├── GPL.txt                |
├── LICENSE.txt           |-----> Licenses files
├── QPL.txt               |
├── README.txt            -----> Readme file
├── Spis.sh               -----> Main launching script
├── SpisNo3D.sh           -----> Launching script without 3D rendering
├── THIRDPARTY.txt        -----> List of used external components
├── VERSION.txt           -----> Version of the current distribution
├── data                  -----> Example data directory
│   └── DefaultProject.spis5
│       └── (...)
├── dependencies          -----> Dependencies directories
│   ├── main              -----> Main dependencies (external libraries)
│   │   └── (...)
│   └── thirdparty        -----> External tools dependencies
│       ├── cassandra-2.5.0
│       │   └── (...)
│       ├── felix-4.0.2
│       │   └── (...)
│       ├── gmsH-2.4.2-osx64b
│       │   └── (...)
│       └── jython-2.5.1
│           └── cachedir
├── documentation        -----> Documentation directory
│   ├── SPIS_UM.pdf
│   └── SPIS_IPR_and_Licensing_Status.pdf
├── lib                  -----> SPIS libraries
│   └── (...)
├── resources            -----> Resources directory
│   ├── defaults
│   │   ├── ODFEmptyDocumentTemplate.odt
│   │   └── autoReportTemplateArtenumAsDocument.odt
│   │
│   │   ├── globalParameters          -----> SPIS-NUM preconfigured global parameters
│   │   │   ├── globalParam.xml
│   │   │   ├── globalParameters-SPISGEO-ECSSGeoWorstCase.xml
│   │   │   └── globalParameters-SPISGEO-NASAWorstCase.xml
│   │   └── properties                -----> Material properties catalogs
│   │       ├── defaultElecNodeList.xcat
│   │       │   └── (...)
│   │       ├── defaultMaterialList.xcat
│   │       │   └── (...)
│   │       └── defaultPlasmaList.xcat
│   │           └── (...)
│   └── externalTools.properties     ----> Settings files
├── felix
│   └── config.properties
├── images
│   └── splash_spis-geo-png24.png
└── jyconsole.properties

```

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

**Remark:** On some Microsoft Windows platforms, the default archiving tool (zip) may have difficulties to un-compress some files with a long path name. To avoid this problem, we recommend to use alternative archiving tools like 7Zip.

### 3.4. Running the application

In the created folder, there are two executable files.

On UNIX-based machines (e.g Linux, Apple OSX) the command-line executable files are:

- **Spis.sh** to execute SPIS.
- **SpisPropulsion.sh** to execute SPIS.
- **SpisGeo.sh** to execute SPIS in GEO mode, see Annex 1.

On Windows, these are:

- **Spis.bat** to execute SPIS.
- **SpisPropulsion.bat** to execute SPIS
- **SpisGeo.bat** to execute SPIS in GEO mode, see Annex 1.

For more details about the SPIS flavours you can visit the SPINE community web site.

If the application is properly launched, a GUI as illustrated in **Figure 2** should appear.

**NB:** **SpisGeoNo3D.sh/bat** permits to execute SPIS in degraded mode without 3D rendering. On some machines, this mode may present an improved software stability.

**NB:** SPIS can also be run in batch mode (also called headless mode) without any GUI. This mode can typically useful to run large simulations on remote computers. Its use is presented in detailed in a dedicated section 8.1 Batch mode.

**NB:** Several extra internal configuration and optimisation files are automatically generated at the first launch. For this reason the launching phase of the software can be longer than usual the first time.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



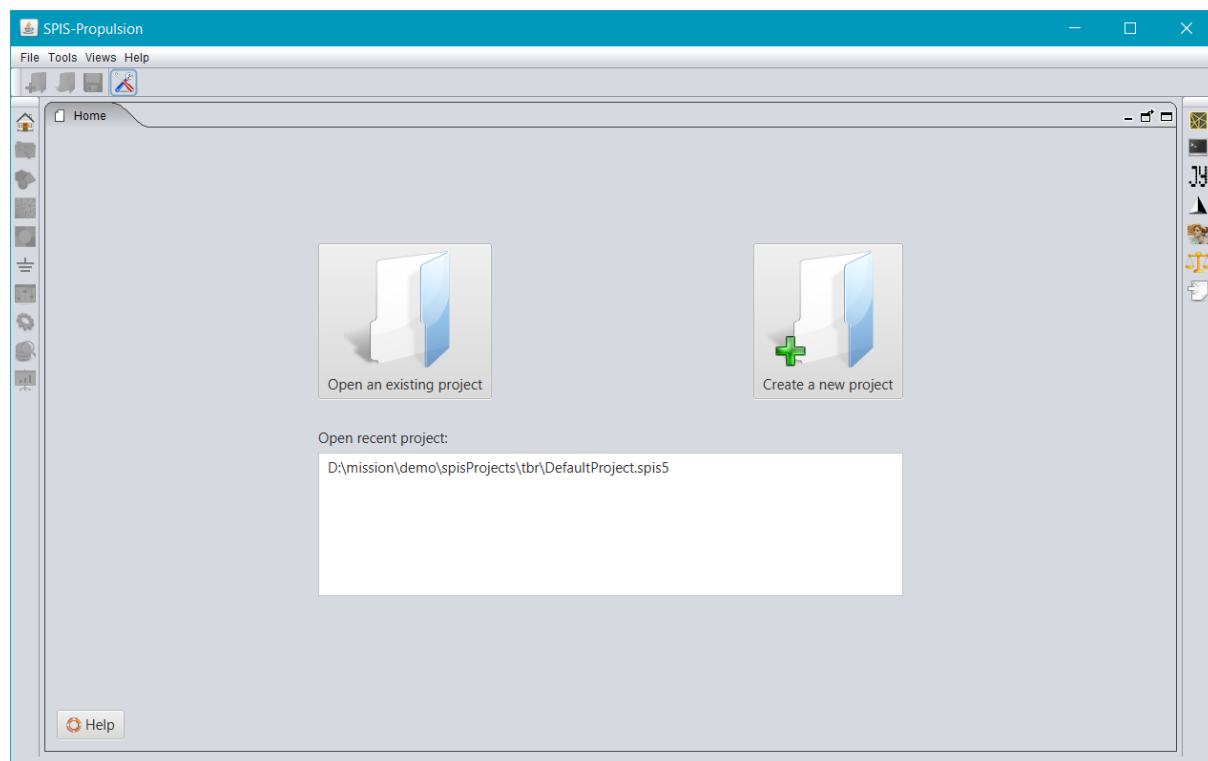
## 4. Using the application

### 4.1. User interface

#### 4.1.1. Overview

When the application is first opened, the window displayed in **Figure 2** appears on the screen. It is divided in five main parts:

- 1) The menu bar gives you access to some of the application main functions. It is composed of five menu items, **File**, **Edit**, **Views**, **Tools** and **Help**.
- 2) The top tool bar contains all generic functions of the framework;
- 3) The left side bar provides a direct access to each step of the modelling process.
- 4) The right side bar allows the access to the external tools of the application.
- 5) The central panel displays the step-by-step thematic panels of the modelling chain.



**Figure 2:** Interface overview

#### 4.1.2. A wizard-based approach

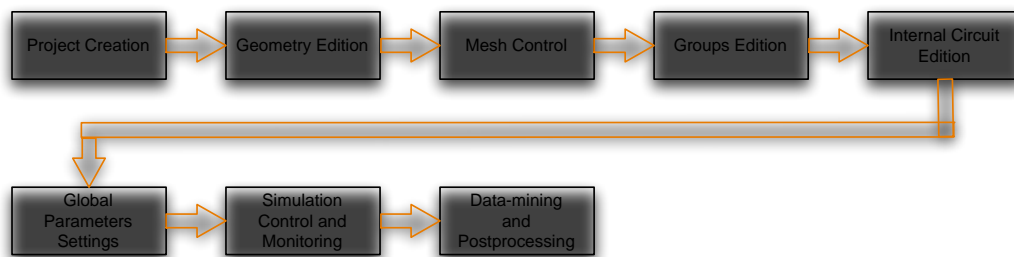
In order to simplify the modelling process for non-experts, the GUI of SPIS has been deeply refactored with respect to the previous versions (SPIS-Legacy) and a “wizard-based approach” has been selected.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

In such an approach, all key-steps of the modelling process have been. Each step corresponds to a specific action, like for instance, the geometry edition, gathering all needed tools for this task and only these ones.

The whole modelling chain is then processed step-by-step, the user passing from one panel to the following/previous one by simple *Next* or *Previous* actions.

The complete process is then, at least in first iteration, linear and direct for the user, minimizing the risk of errors and corruptions of the *Data Model*. The modelled system is then built-up progressively. Figure 3 shows the step-by-step modelling chain, with the thematic panels.



**Figure 3: The step-by-step modelling chain.**

In the same logic, tools are available only when all needed previous information has been set/computed. For instance, the Groups Editor, to apply boundary conditions, is available only if the CAD model and the corresponding mesh have been defined and loaded previously.

Reciprocally, the *Previous* action allows the possibility of “undo”, maintaining the self-consistency of the *Data Model*.

As soon as the model is fully defined, all tools are available and the user has the freedom to directly go to the panel of his choice to perform corrections or advanced settings, if he wants.

### 4.1.3. Menu bar

The menu bar contains five menus:





- The **File** menu contains the operations related to files:
  - o **Create a new project** to create a new project
  - o **Open an existing project** to open a .spis5 project
  - o **Save** to save the currently loaded project
  - o **Save as...** to save the currently loaded project with a new name
  - o **Exit** to quit the application
- The **Views** menu contains direct access to the modelling steps when available
- The **Tools** menu allows to execute external tools, to open the preferences panel, to apply patch and to add a new SPIS plugin like the mesh inspector for example
- The **Help** menu provides help and information about the software

Some of the menu items are deactivated by default. They activate automatically when necessary.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

#### 4.1.4. Top tool bar











The topside tool bar gathers all generic functions of the framework, like access to the project and/or the generic tools. The generic commands are:

-  - Creates a new project *ex nihilo*;
-  - Open a pre-existing project;
-  - Save the current state of the project (available if data have been generated or updated);
-  - Set the general preferences of the application (may depends on the version of SPIS);

This list is indicative and may depend on the current version of SPIS.

#### 4.1.5. Left side tool bar

The left side bar provides a direct access to each tool panels of the modelling process, like the Geometrical Editor for instance. The buttons become active only if the corresponding tool-panel is available, according to the current progression in the modelling process. The potentially available tool-panels are:

-  - Allow the access to the Home View panel to create a new project or open one;
-  - Allow the access to the Project and Studies Setup panel;
-  - Allow the access to the Geometry Editor panel;
-  - Allow the access to the Mesh Control panel;
-  - Allow the access to the Groups Editor, to set properties and boundary conditions;
-  - Allow the access to the Internal Circuit Editor;
-  - Allow the access to the Global Parameters Editor;
-  - Allow to access to the Simulation Control and Monitoring Panel;
-  - Allow the access to the Data Miner;
-  - Call the auto-reporter for the simulation finalization.

This list is indicative and can be change according to the bundles loaded in the framework and/or the version of SPIS.

The menu **Views** of the top menu bar gathers the same items.







**IMPORTANT REMARK:** Please notice two important points. First, during the wizard progression, if the user clicks on the previous button, the button corresponding to the current step will be deactivated in the left bar (i.e. switched to grey). In most of the cases, the data corresponding to the present step and downstream steps are also removed in order to keep the consistency of the data model. However, if the user comes back to previous steps using the buttons of the left-side tool bar, the downstream data

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

are not removed automatically; they are cleaned if the user clicks on the next button of the step. The user should then manage himself the maintenance of the data model consistency.

#### 4.1.6. Right side tool bar

The tool bar on the right side allows the access to the permanent external tools, like the 3D viewer Cassandra. These tools are always available, what ever the current step in the modelling process.

-  - Call the legacy SPIS projects importer;
-  - Call the Log console;
-  - Call the Jython console (JyConsole);
-  - Call the Gmsh CAD and meshing tool.
-  - Call the 3D viewer (Cassandra)
-  - Call the data field comparator to compare data fields stored inside two SPIS projects


This list is indicative and can be change according to the bundles loaded in the framework and/or the version of SPIS. The menu **Tools** of the top menu bar gathers the same items.

## 4.2. Exiting the application

To close the application, you have to select the **File>Exit** menu. You will be proposed to confirm this before quitting.

## 4.3. Monitoring of the application with the Logs console

In order to simplify the general use of the application, most of the generated messages are hidden during the standard use. All messages are collected by a central logging system and forwarded to a common logging console, on one side, and a logging file (called debug.log and localised at the root directory of the application). Several levels of verbosity/criticality are defined for messages: trace, debug, info, warning and error. By default, the logging console is automatically opened only if warning or error messages are generated.

For a finer analysis of the control message, it is possible to open the logs console manually, by just clicking on the  icon. Figure 1 show an example of logging messages (indicative).

Important remark: All messages are gathered into the debug.log file and localised at the root directory of the application. In case of troubleshooting and errors or if you ask for support, please send this file to the SPIS team at the following address: [contact@spis.org](mailto:contact@spis.org).

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

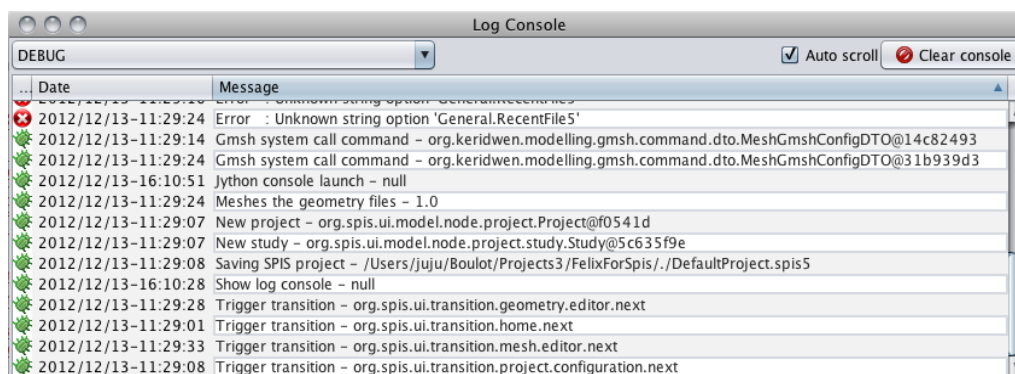



Figure 4: View of the log console.

## 4.4. Controlling the application with Jython

It is possible to activate a Jython console (cf. **Figure 5**) by clicking on the  icon. In the Jython console, you have access via introspection to classes and methods of the software. You can also access the model object stored in memory.

This way to control the application is thus reserved to advanced users who know what they are doing. Indeed, it is relatively easy via command line to corrupt a file or to make the data model inconsistent. As a consequence, it is highly recommended to backup your data before editing them via the command line.

The Jython console offers auto-completion by using the “control + space” key combination. It also permits to load existing scripts by right-clicking in the console and selecting **Load a script** in the contextual menu. To know what are the possibilities, please refer to the API documentation of the project.



Figure 5: Jython console

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

## 5. A project-based approach

The persistence model of the SPIS-UI modelling chain stands on a “project-based” approach, where all needed parameters and actions performed during each step are progressively saved. A SPIS project should be seen as the equivalent of the concept of workspace in some Integrated Development Environment (IDE) like Eclipse.

Before any new study, the user should define a new project first. A SPIS project links all needed data (CAD file, mesh, groups settings, material, global parameters...) needed to perform the simulation and recover computed data. Scalar and vector fields applied on the mesh (i.e. DataFields) are also saved and can be reloaded for visualization and post-processing.

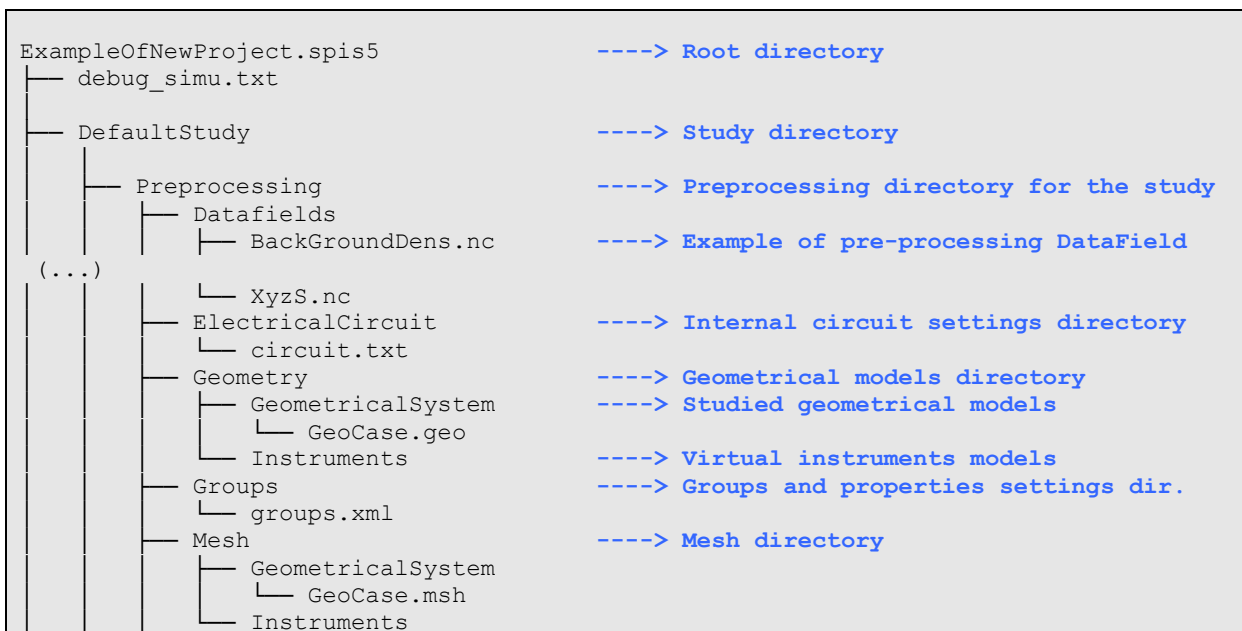
Practically, a SPIS project is a directory with a set of pre-built subdirectories and files. It can be packaged with classic archives tools, like tar or zip, to be moved to another place or computer. Most of the input data is saved under the form a XML files. These files can then edited manually with all current ASCII text editor. Output data, like fields mapped on the grid, are saved into a standardized NetCDF file format (binary) or in a dedicated XML format. The first one is used by default. The files formats are described in Annex A.

Please notice that, in order to optimize both memory footprint and I/O access, the project is loaded and saved in a “lazy way”. This means that only the project structure is initially loaded and nodes are themselves loaded individually only when they are requested. Reciprocally, only modified nodes are saved when the project is saved.

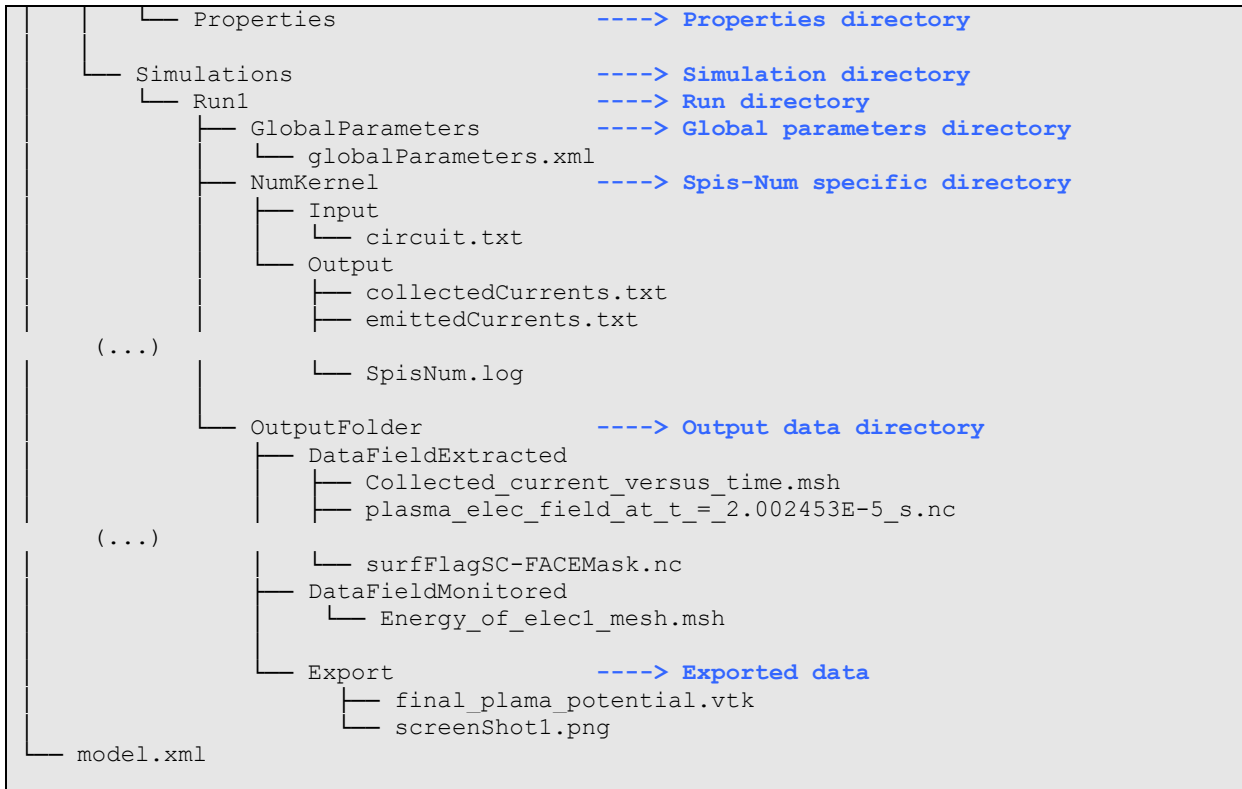
The project structure is still very modular and each component is loaded independently and new projects built by aggregation of existing components. The current project format is not stabilized and can be modified without any notification. For now, please avoid from using spaces in the project or study name as they are not correctly supported.

### 5.1. Project structure

The tree given in Figure 6 presents an example of a standard SPIS project at the end of the simulation. This structure is automatically generated by SPIS-UI and should not be modified manually. However, the presented structure is indicative and may differ depending on the data nodes generated during the modelling process. Files are distributed into dedicated sub-directories, as described here.




<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 6: Example of project structure**

A project can gather several studies. A study can gather several simulation runs. A study is defined by the needed preprocessing information and the corresponding runs. Main and large pre-processing data files, like the CAD or the mesh files, are shared by all runs of the parent study. Simulation runs store the global parameters and the corresponding outputs.

## 5.2. Creation of a new project

The creation of a new project (or the loading of an existing one) is the first operation to be performed. By default, the application starts on the Home View panel (see Figure 7), where the user can choose to create a new project or open a pre-existing one. If the Home View panel is not displayed, please click on the  icon to access to it.

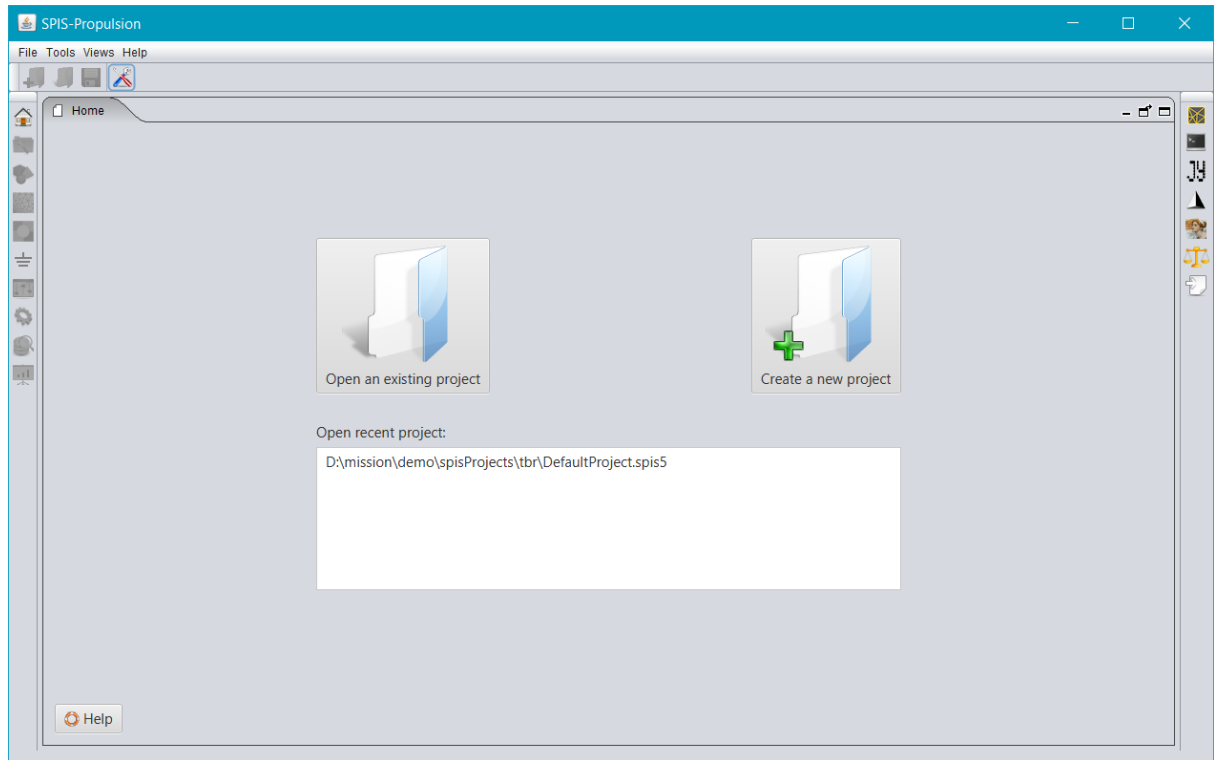
Please click on the “Create a new project” icon in order to access to the Project and Studies Setup panel, as illustrated in Figure 8. This panel allows setting the location of the parent directory, the name of the project itself and a free description. A new study is proposed by default. Its name can be changed.

The user can pass to the Geometry Editor panel just by clicking on the

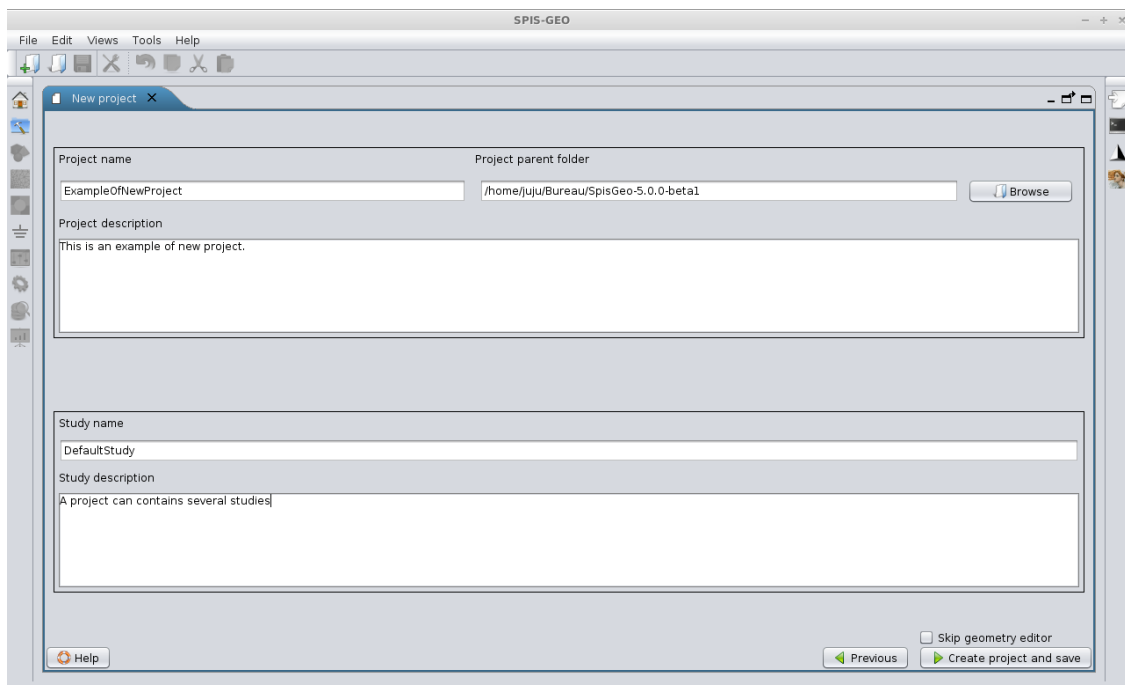


button. The project information is automatically saved.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 7: Project selection**



**Figure 8: Project edition panel.**

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



### 5.3. Open an existing project

To open an existing project, first go to the *Home View* panel and then click on the left icon “Open an existing project”. A file chooser will appear, please select the relevant project directory. The project should be with a .spis5.

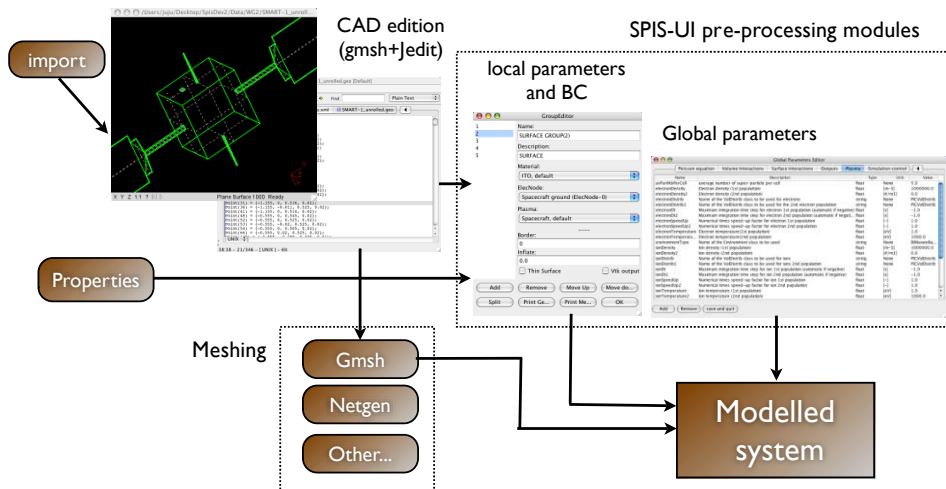
In the home panel, there is the list of the recent projects. To open one of them, you can click on one of them in this list. The number of recent projects can be set via the SPIS preferences panel.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

## 6. Step-by-step modelling process

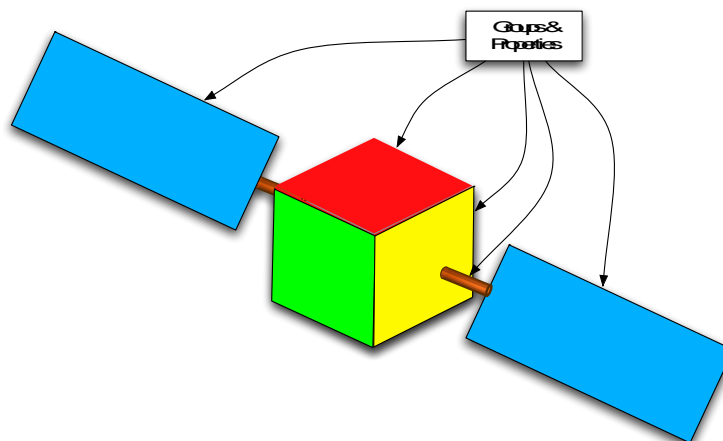
### 6.1. Pre-processing

The pre-processing phase corresponds to the construction phase of the modelled system. This phase should be performed prior to any simulation in order to define the studied system. During this phase, all parameters and settings needed by the simulation kernel should be set. This includes the geometry model, the local parameters, with different values depending on the localization on the mesh like initial and boundary condition, and global parameters. Figure 9 summarizes the pre-processing process.



**Figure 9: The pre-processing phase aims at the full definition of the modelled system.**

One of the key settings is the link between the geometry of the system (spacecraft and computing space) and the initial and boundary conditions (IBC). In the SPIS-UI framework, these parameters are generally called “local parameters” because their value may differ depending on their localization. These local parameters are Properties associated to the geometric model through a set of geometrical zones, called “physical” in the Gmsh format or groups in SPIS-UI. SPIS-UI provides a set of tools to attribute properties to groups and to deploy these properties on the mesh, as illustrated in Figure 10.



**Figure 10: Example of groups and properties setting. The spacecraft can be sub-divided into several “Gmsh Physicals” to which the user can associate properties, using the SPIS-UI Group Manager. These properties are converted into local fields (DataFields) deployed on the mesh and passed to the simulation kernel SPIS-NUN as Initial and Boundary Conditions (IBC).**

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

The building of the CAD model, the groups edition and the settings of local properties constitute a very delicate and complex phase. This is generally the main source of mistakes or failures in the modelling process. This process must be done step after step and taking into account constraints of the mathematical models of the solver.

### 6.1.1. Geometrical modelling

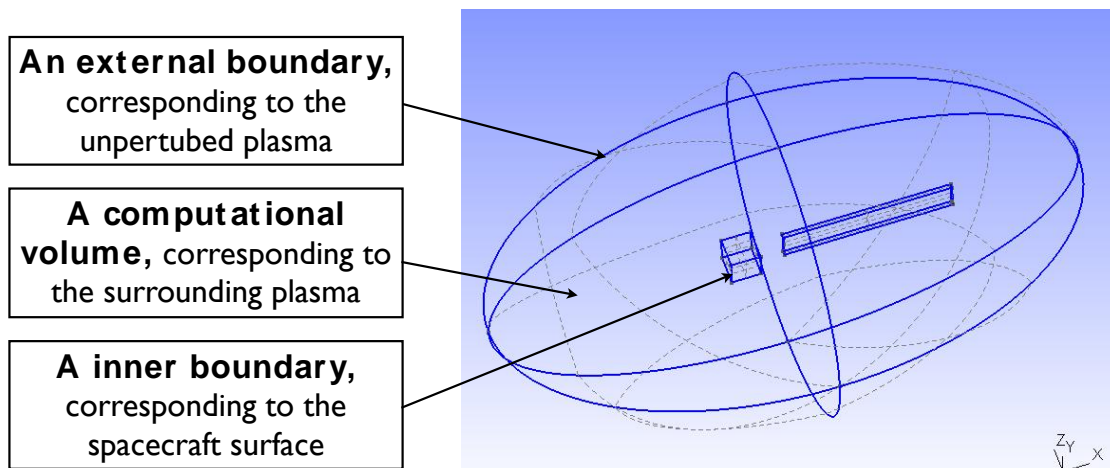
SPIS supports complex geometrical descriptions using a B-rep approach, as used in the Gmsh CAD/meshing tool and supports natively the geo Gmsh file format. Using the import capabilities of Gmsh, other files formats are also partially supported. An example of simple CAD model is given in Annex E. Please see the Gmsh documentation of the SPIS trainings for further information on this.



**Figure 11: Principle of the B-rep approach in the geometry definition.**

As illustrated in Figure 12, a complete geometrical system should be constituted, at least, of:

- The spacecraft structure, defined as one or several closed surfaces, that should correspond to the inner mathematical boundary;
- An external boundary, defined by one closed surface, that should correspond to the external mathematical boundary;
- The computational volume delimited by the above boundaries.



**Figure 12: Example of minimum geometrical system.**

Using the Include command of the geo file format, a complete geometrical system can be defined through several CAD files, as illustrated in the following example, where the spacecraft geometry, defining the inner boundary, and the external boundary are defined in two different files. Both are gathered in a common main file (e.g. main.geo) where the computational volume and needed physical are also defined.

```
// import of sub-systems
// import of the various sub-systems
Include "solarArray1.geo";
Include "solarArray2.geo";
```

**Reference:** ESA-SPIS-GEO-D7-SUM-2013-03-001

**Version:** 3

**Revision:** 5

**Status:** Final

**Date:** 28/11/2019

```

Include "hub.geo";
Include "extBound.geo";

// definition of the computational volume
Surface Loop(60062) = {60005}; // external surface loop
Surface Loop(60063) = {1014}; // inner surface loop
Volume(60064) = {60062,60063};

// definition of the physicals
//external boundary
Physical Surface(60065) = {60005};
//inner boundary
Physical Surface(60066) = {1014};
//computational volume
Physical Volume(60067) = {60064};




```

**Tableau 3: Example of CAD multi-files geometrical system.**

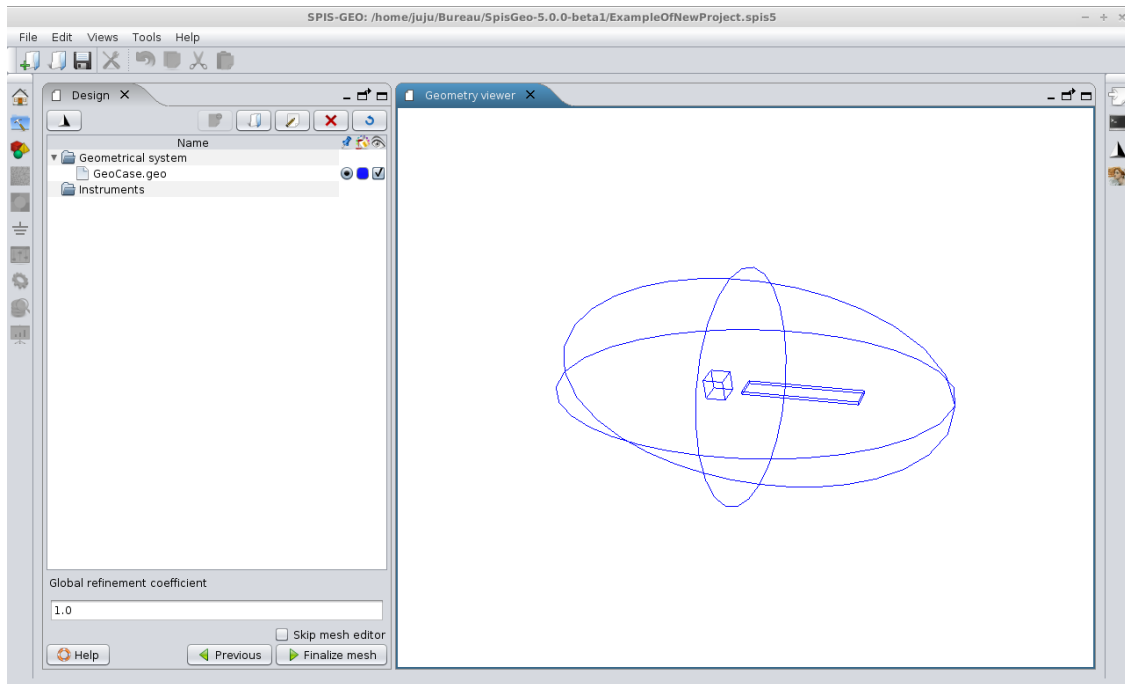
In addition, several additional objects, like virtual instruments, can be added to the system.

To help the CAD files edition and the management of geometry and CAD files, SPIS-UI provides a Geometry Editor, which gathers into a common workspace all relevant CAD files. Figure 13, here below, shows a view of the Geometry Editor.


The Geometry Editor is divided into two main panels, the Design panel, on the left, and the Geometry Viewer, on right side.

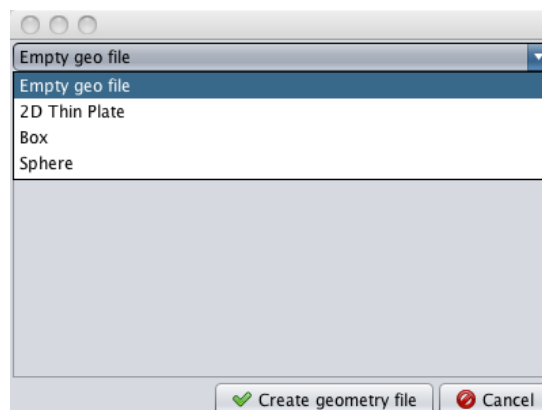
In the Design panel, the  will call the Gmsh tool to edit the selected CAD file with this 3D tool. The  icon will open and import a pre-existing CAD file in Gmsh .geo format. More formats are supported by the SPIS geometry editor. Step (.stp or .step), Stereolithography (.stl), Gmsh Mesh (.msh), Initial Graphics Exchange Specification (.iges) and UNV (.unv) files can be loaded in the geometry editor. A conversion from mesh to geometry is performed and a new Gmsh .geo file is created from these CAD files during the import. The loaded file should then appear in the tree list of the Design panel and in the 3D view on the right side panel. If needed, please type the R key when the mouse is over the 3D panel to perform a camera reset to see the whole geometry. The CAD file can also be directly edited as a simple ASCII text file by selecting it and clicking on the  icon.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



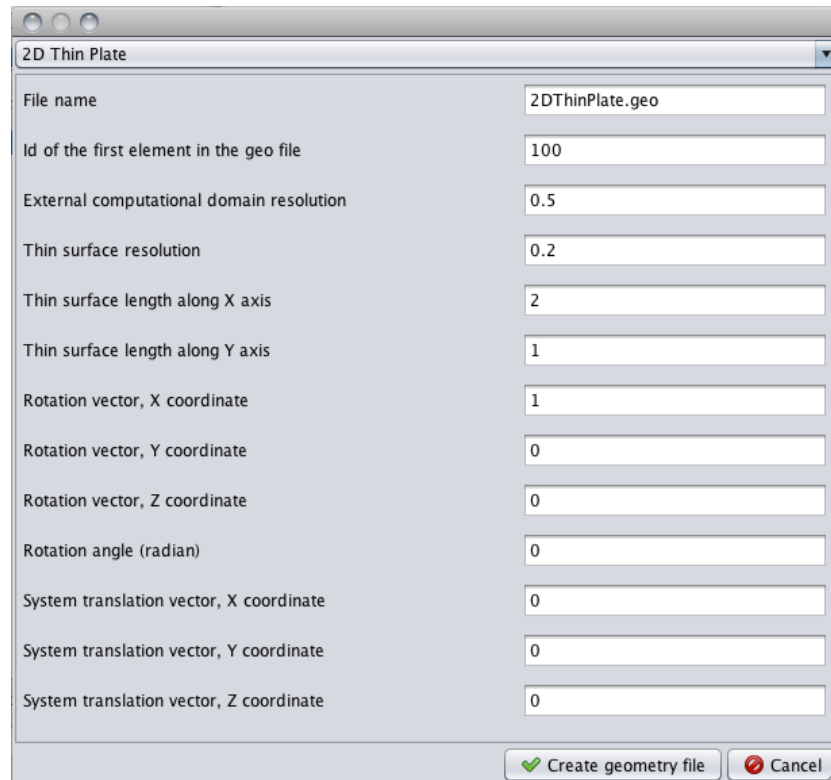
**Figure 13: Geometry Editor**

**Use of CAD templates:** In order to simplify the modelling of the geometrical system, template of CAD models are provided with SPIS. These shapes can be used for the definition of the external boundary, refinement boxes or specific objects like thin elements. To access to these template, you should click on the  button in the Design panel of the Geometry Manager. A selection panel as illustrated in Figure 14 should appear. Templates can largely parameterised to adjust their geometry according to the needs of the modelled system, as illustrated in Figure 15.



**Figure 14: CAD template selection panel, various pre-designed CAD models can be selected as well as a new and empty file.**

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



2D Thin Plate	
File name	2DThinPlate.geo
Id of the first element in the geo file	100
External computational domain resolution	0.5
Thin surface resolution	0.2
Thin surface length along X axis	2
Thin surface length along Y axis	1
Rotation vector, X coordinate	1
Rotation vector, Y coordinate	0
Rotation vector, Z coordinate	0
Rotation angle (radian)	0
System translation vector, X coordinate	0
System translation vector, Y coordinate	0
System translation vector, Z coordinate	0

**Figure 15: Example of configuration panel for a new geometry issued from a template. The listed fields may differ depending on the selected template.**

**WARNING:** SPIS uses Gmsh as CAD engine and mesher. Gmsh is a native component that depends on the targeted platform (OS and hardware architecture). On some OS, Gmsh may also depends on related external libraries (e.g. glibc, Fltk...). For these reasons, the CAD import/loading may fail if Gmsh does not run properly. In this case, please install manually a version of Gmsh appropriate to target platform (see <http://geuz.org/gmsh/>) and set the corresponding path as described in section 9.4-Property file and configuration of native components to fix such troubleshooting. As a rule of thumb, gmsh version 2.4.2 was adapted to most tested platforms.

**Setting of the main CAD file:** In Gmsh, a complex geometry can be defined by a set of subsystems defined in several CAD files that are imported into a central one, that can be considered as *Main CAD File (MCF)*.

SPIS-UI will read the whole geometry by first loading this *MCF*. For this, the user must absolutely indicate to the software which file should be considered as main, by selecting the relevant file in the tree with the radio-button column symbolized with a pin (📌). This operation must be absolutely done before the loading procedure. By default the first file loaded is automatically set as *MCF*.

Please see the Gmsh Technical Documentation [RD5] or SPIS training documents [RD3] for further information regarding the inclusion of sub-CAD models using the “Import” method.

**Definition of *Physicals* on the CAD model:** As introduced above, we outlined the fact that several *Physicals*, in the Gmsh sense, must be defined to set all Boundary Conditions (BC) needed by the

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

solver. This is to be done using Gmsh or by directly editing the corresponding CAD file. Please see the Gmsh Technical Documentation [RD5], the "Gmsh User Tutorial for CAD and Meshing" [RD6] or SPIS training documents [RD3] for further information regarding the *Physicals* definition and syntax.

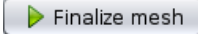
The practical repartition and setting of Physicals generally depends on the modelled system and also on the constraints of the numerical kernel. We strongly recommend here to see the SPIS-NUM documentation and more especially "How-to-control NUM from UI" document [RD7].

However, at least, the three following Physicals must be defined.

Object or zone to be marked	Type of physicals
Spacecraft surface	Surface
External boundary of the computational volume	Surface
Computational volume	Volume

**Figure 16: Minimum physical settings**

In some configurations of the SPIS-UI framework, extra CAD models can be added, like virtual instruments, for instance. These elements will not be considered as part of the modelled system and will be handled depending on the SPIS-NUM version. Please see the SPIS-NUM technical documentation for further information.

**Finalization and meshing of the whole geometric system:** The meshing of the whole geometrical system constitutes the last step of the 3D modelling. This is simply done by passing to the next wizard step by clicking on the  button. The project is automatically updated.

Before, it is possible to apply a global refinement factor through the field *Global refinement coefficient*. Please take care, that this value is a scaling factor applied to the local resolution characteristic lengths defined in the CAD file. This means that a *Global refinement coefficient* of 2.0 will multiply the local characteristic lengths by two and the final mesh will be coarser and not more refined. This value can be let as it is by default.

By selecting the *Skip mesh editor* check box, the framework will skip the next step and directly go to the Group Editor Panel. This value can be let as-it by default.

**Instruments and extra-mesh supports:** For measurements and monitoring extra instruments can be defined to measure and extract various physical and numerical quantities. Additional meshes of support for these instruments can be defined and added into the Instrument section. There is no limitation in meshes defined here.

Take care that the types of implemented instruments may depend on the version/branch of SPIS and the fact that the introduction of additional instruments may break the inter-compatibility of stored projects and SPIS versions.


### 6.1.2. Meshing edition and control panel

The mesh edition and control panel has as final purpose to centralize all tools related to the mesh edition, visualization, metrics and quality control.

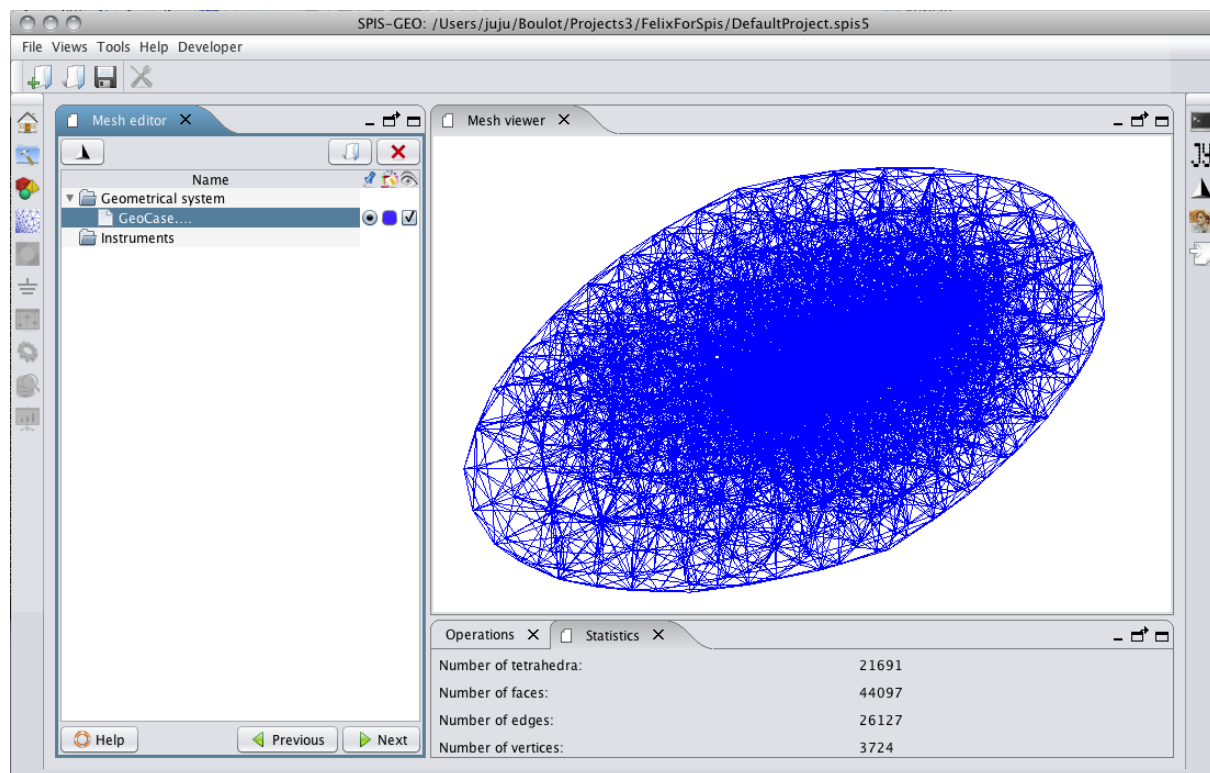
When a mesh is selected, the corresponding statistical information, like the numbers of cells is displayed.

At this level, it is still possible to define several meshes for the Geometrical System as well as for the instruments. However, only one mesh will be taken into account for the Geometrical System. If several

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

meshes are defined here, the selected one can be chosen as in the Geometry Editor with the pin icon ()

If *Instruments* are defined, several meshes can be present in the Instruments subsection without limitation.



**Figure 17: Mesh control by 3D view of the mesh.**

Several meshes can be visualised and controlled at the same time, for instance to control virtual instruments immersed into the computational domain.

When the user selects a mesh in the selection tree of the Mesh statistics panel, the main mesh information is displayed, like the total number of nodes or cells.

The  icon will open and import a pre-existing mesh file.

Extra-tools for operation on the mesh are available on the Operation index. The types of implemented operators may depend on the version/branch of SPIS.

The access to the next panel is done by clicking on the Next button.

**WARNING:** SPIS uses Gmsh as CAD engine and mesher. Gmsh is a native component that depends on the targeted platform (OS and hardware architecture). On some OS, Gmsh may also depend on related external libraries (e.g. glibc, Fltk...). For these reasons, the meshing may fail if Gmsh does not run properly. In this case, please install manually a version of Gmsh appropriate to the target platform (see <http://geuz.org/gmsh/>) and set the corresponding path as described in section 9.4- *Property file and configuration of native components* to fix such troubleshooting. As a rule of thumb, gmsh version 2.4.2 was adapted to most tested platforms.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



### 6.1.3. Groups edition and properties settings

In SPIS, the notion of properties corresponds to the definition of local parameters, like boundary conditions, used in the definition of the model. This includes material, physical and numerical properties. As introduced above, the purpose of the pre-processing part is to attribute these properties to each specific parts of the spacecraft or the computational domain, convert them into fields (i.e. DataFields) mapped on the mesh that the numerical model can understand.

In order to define properly the needed IBC, the various properties should be attributed to their respective Physicals in order to build properties groups. This should be done using the Group Editor, as shown in Figure 18, hereafter.

Based on the Artenum's Frida library and its generic Properties Groups Editor, the SPIS Groups Editor allows to:

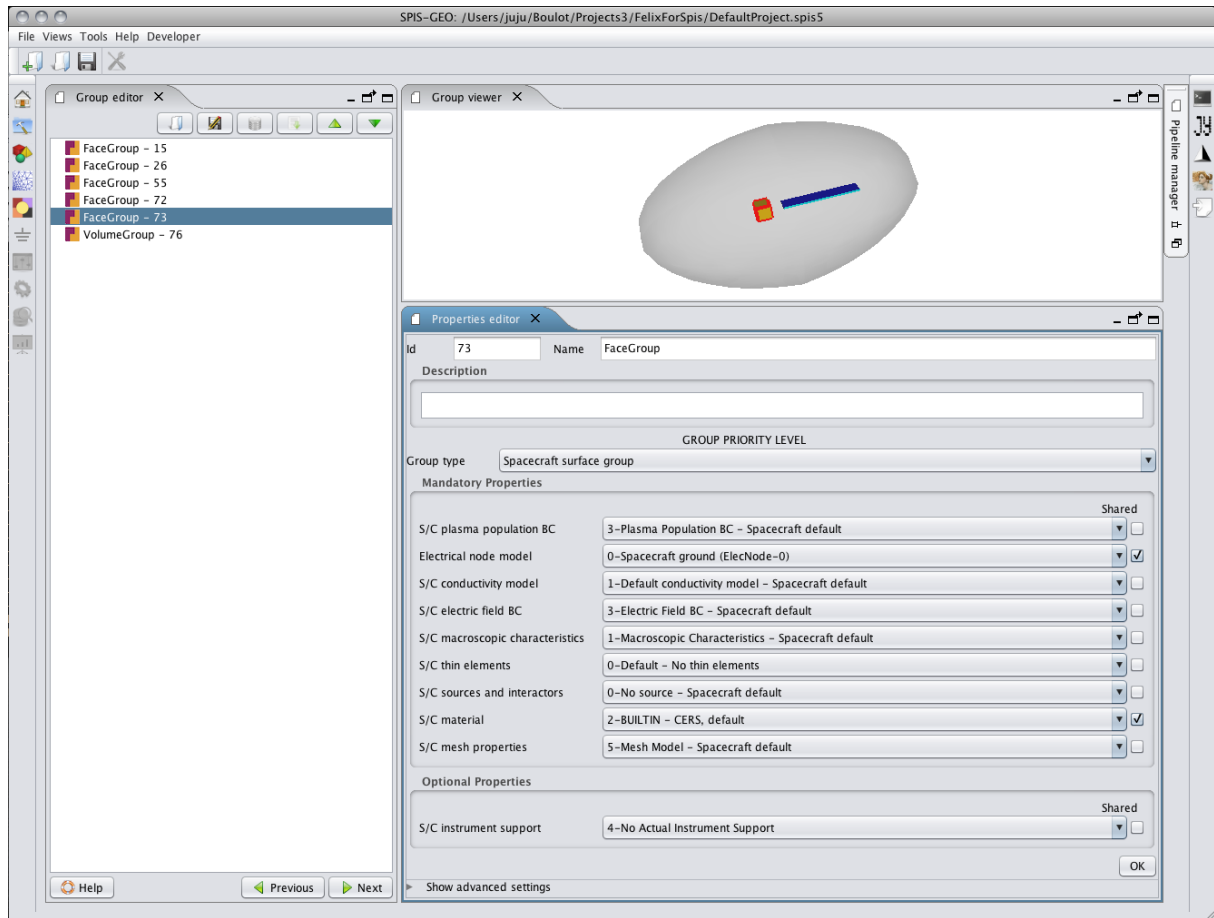
- Load generic properties (e.g. material, numerical flags, meta-data, etc...) for external catalogues;
- Create, edit and modify properties;
- Build, edit properties groups by making the link between Physical on a CAD models and properties.
- Deploy the Properties characteristics on the mesh and build-up the corresponding DataFields.

The Groups Editor GUI is divided into three main panels. On the left side, a tree lists all empty groups automatically generated from the *Physicals* of the loaded CAD file. On the top right, a panel gives a 3D view of the whole system with its various groups. The user may have to type the R-key to resize the view and see the whole system. The groups are shown with different colours. The most external group, generally corresponding to the external boundary group, is displayed in transparent colour to facilitate the view of the inner system.

The group selected in the left side tree will be outlined in red in the 3D view.



The down right side corner corresponds to the edition panel of the selected object in the tree. This edition panel may change depending on the type of the selected object (group, property, characteristics).

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

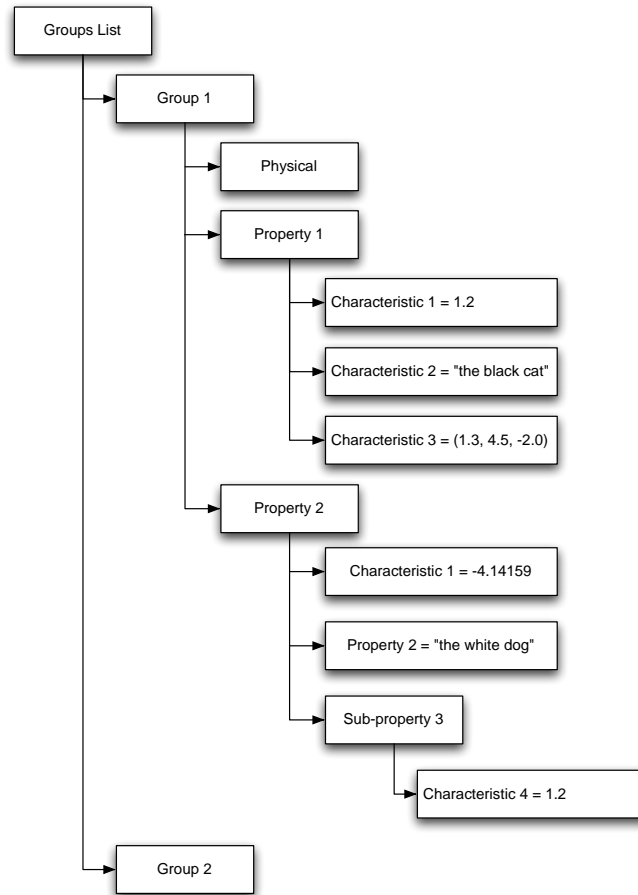


**Figure 18: Groups Editor and properties settings**

The Frida library offers a very versatile and adaptable model to attribute all types of data to properties to groups [RD11]. Groups are gathered into a central groups list. A *Group* can contain several *Properties*. Each *Property* can contain several *Characteristics*. A *Characteristic* can be a scalar (integer, float, double), a vector or all other types of data (object). A *Property* can have a sub-property, that itself contains its own characteristics. Such an approach allows to build-up or enrich properties by composition of existing ones. The same *Characteristic* can have different values when it is linked to different *Properties*, in order to set different values for the corresponding field. Figure 19 illustrates this structure.

The group manager allows also the possibility to re-define the priority in the properties mapping procedure. This is useful for the specific setting of elements owned by two groups. Value finally mapped will correspond to the group of higher priority. In the group manager, the priority is decreasing from the top to the bottom. The user can change the priority of the selected group with the  and  buttons. At the loading phase, groups are loaded according to the Ids of their respective Gmsh physicals defined through the loaded mesh.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 19: Properties and groups edition with Artemum Frida.**

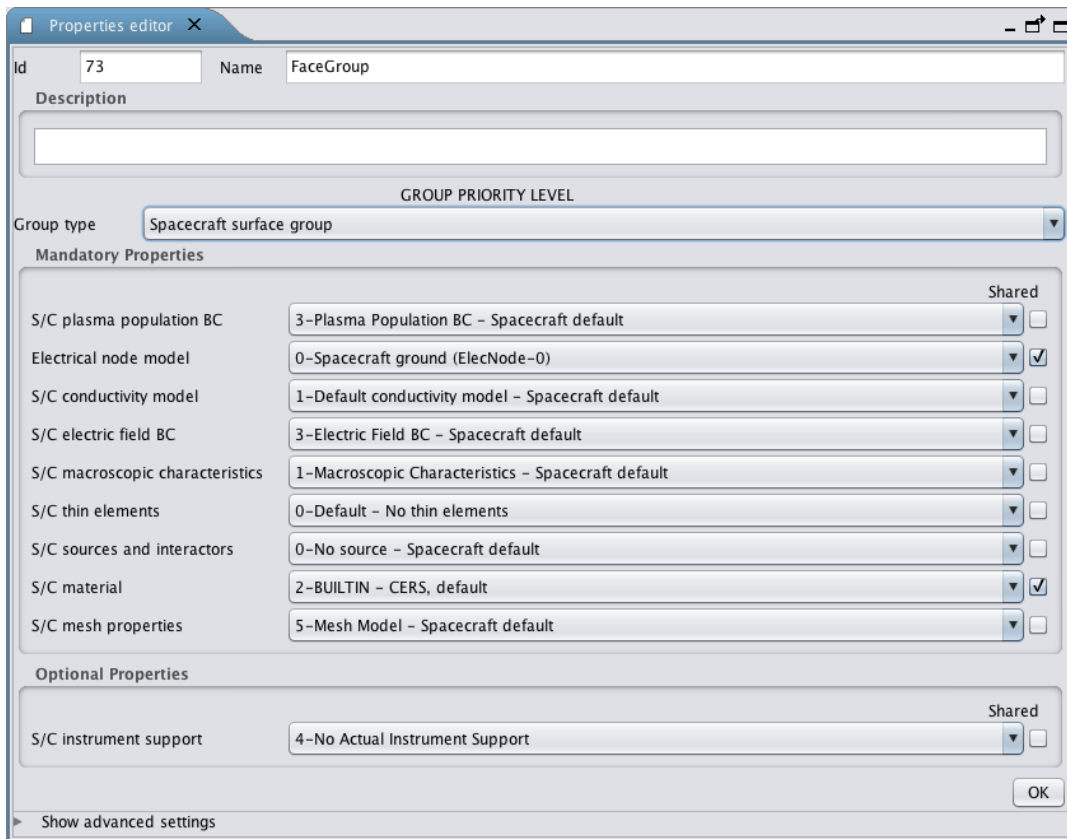
Properties can be shared by several groups or not. In the first case, the instance of the property (and its linked characteristics) are shared by all groups to which ones this property is attributed. If the value of a characteristic is changed, it is changed for all them. In the second case, a different instance is attributed group by groups. Characteristic may have different values for each group.

**Caution:** An inconsistent attribution of groups will automatically lead to errors or a crash during the mapping phase or the simulation. The relevance of the simulation is strongly dependent on a correct definition of groups.

The Groups Editor provides a facility to help to select goods property types for each given type of group. First, select the group on the left side panel. The group edition panel should appear in the right down corner.

Allowed properties types are pre-set according the different group types. In the first combo box, select the relevant group type, for instance Spacecraft surface group, if the selected group is on the surface of the spacecraft. Only corresponding mandatory and optional property types supported for the present group type will appear in the panel through several combo boxes, as illustrated in Figure 20. Property instances can be shared on several groups by clicking on the *Shared* selection boxes.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 20: Property types pre-selection in the Groups Editor.**

Select the properties of your choice and click on the OK button to validate your choice. The left side tree should appear as illustrated in Figure 21.

In some versions of SPIS, extra options and additional tools may be added in the lower sub-pane, called Show advanced settings. Please see the modelling thin element section for an example of such tools.

In the current version of SPIS (5.1), the group types versus property types attributions are defined in the following table. Default values are pre-set and for most of the properties do not need any modifications. Please see the SPIS-NUM documentation for further information regarding the models selections.

As a matter of fact, the user usually needs only to define the property type in *italic* in Tableau 4. For other property types, the default settings are generally sufficient, see [RD7].

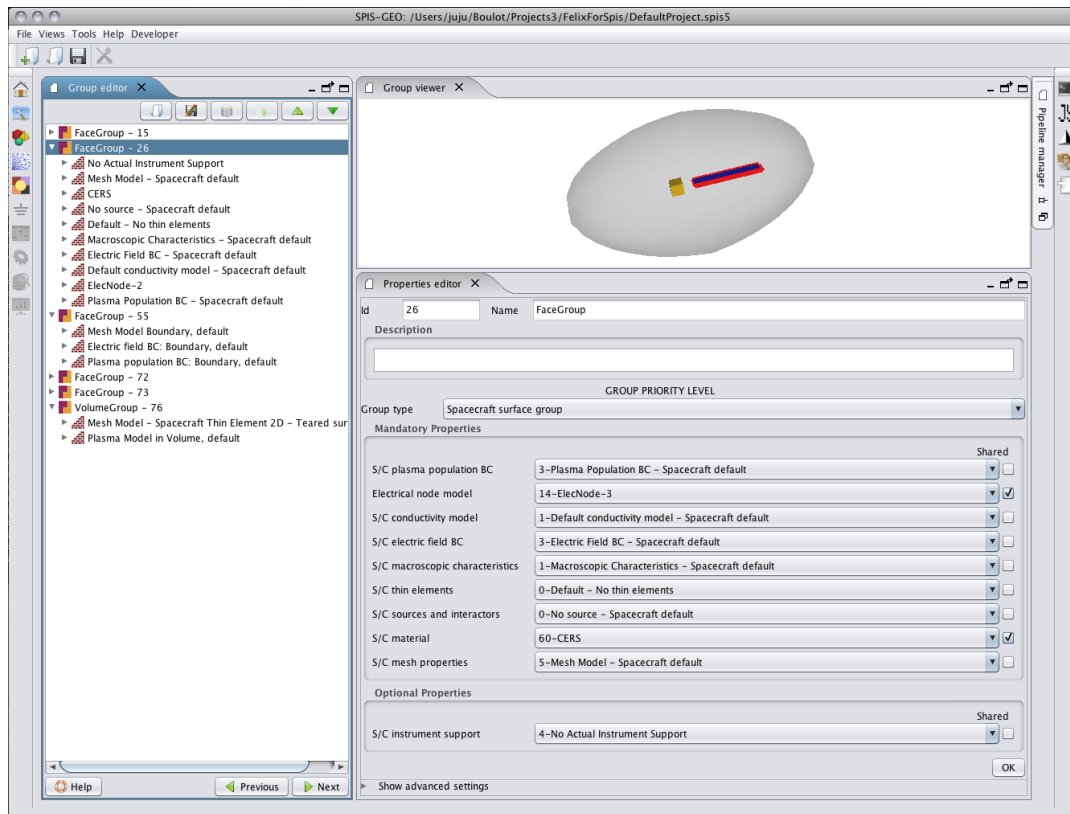
<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

**Tableau 4: Groups and properties types. Description of properties characteristics is given in the annex Controlling NUM from UI [RD7].**

Group type		Property type	Description
<b>Spacecraft surface group</b>	<b>Mandatory Properties</b>	S/C Plasma population BC	Define all characteristics related to the plasma model on the boundary defined by the S/C surface.
		<i>Electrical node model</i>	Select the electrical node for the present group. Please notice, that this property care shared with other groups.
		S/C conductivity model	Define all characteristics related to the conductivity model.
		S/C electric field	Define all characteristics related to the electric field model on the present S/C surface boundary.
		<i>S/C macroscopic characteristics</i>	Define all macroscopic characteristics, including thickness and temperature .
		S/C thin elements	Set if the present group corresponds to a thin element or not. See the corresponding section.
		<i>S/C sources and interactors</i>	Defined all characteristic related to the interactions models and sources.
		<i>S/C material</i>	Select the material for the present group.
	<b>Optional Properties</b>	S/C instrument support	Set if the present group correspond to an instrument on the S/C surface or not. This option is not supported on all SPIS versions.
<b>External boundary group</b>	<b>Mandatory Properties</b>	<i>Ext. Bound. Plasma Population BC</i>	Define all characteristics of the plasma model on the external boundary defined by far plasma.
		<i>Ext. Bound. Electric field BC</i>	Define all characteristics of the electric field model on the external boundary defined by far plasma.
		Ext. Bound. mesh properties	Define all characteristics related to the mesh model the external boundary defined by far plasma.
	<b>Optional Properties</b>	N/A	N/A
<b>Computational volume group</b>	<b>Mandatory Properties</b>	Volume plasma model	Define all characteristics related to the plasma model on the external boundary defined by far plasma.
		Volume mesh properties	Define all characteristics related to the mesh model the external boundary defined by far plasma.
	<b>Optional Properties</b>	N/A	N/A

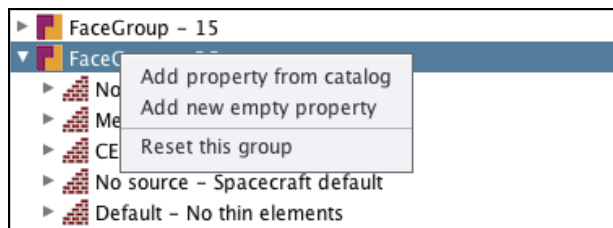
Properties that can be shared are pre-set as shared.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 21: Example of properties attributions.**

In case of error and if the group should be corrected, a right-button click on the selected group in the tree opens a contextual menu and allows to access to more individual actions like, remove, add or copy properties.



**Figure 22: Example of advanced edition through the contextual menu, here for the selected group.**

The  icon will open and import a pre-existing XML group file.

To create new Properties, several approaches are possible. New properties files can be edited manually as all XML file. However, the simplest one is to use the advanced editor in the selection tree, as follow:

1. Select a group,
2. On the selected group, access to the contextual menu with a right-button click, add a new Property.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

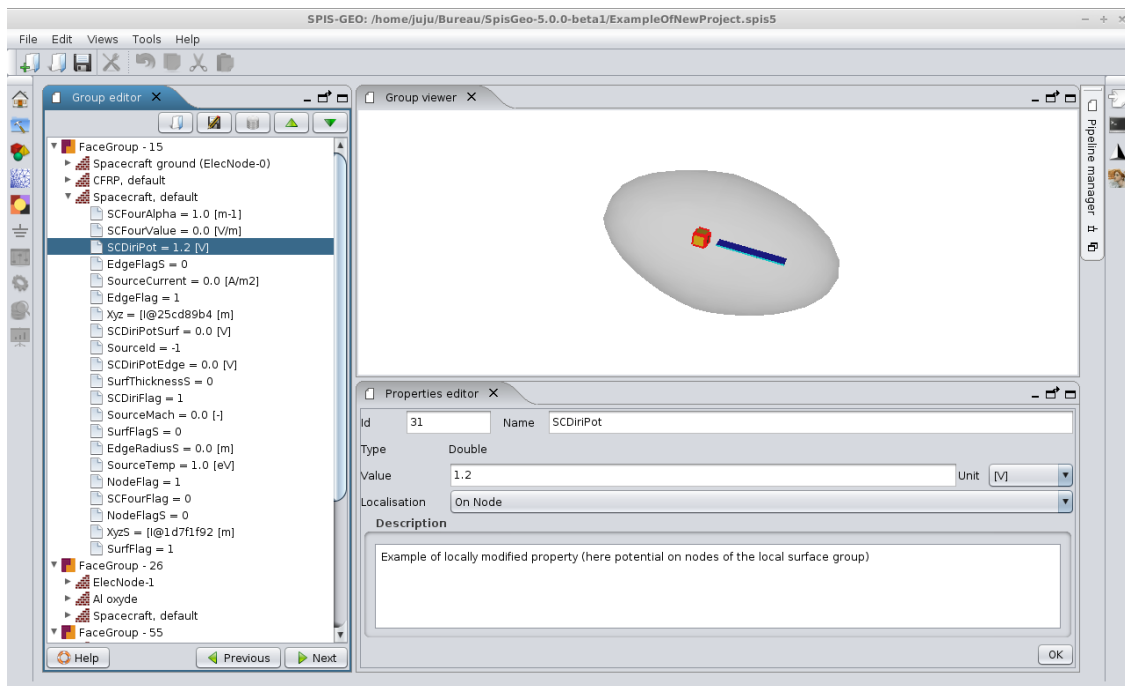
3. Select the newly created property and add, in the same way, all needed Characteristics.
4. After a complete setting of the Property and its related Characteristics, select the built Properties, with the right-click button, access to the contextual module and select the “Save this property” item.

We recommend to save all selected Property into a directory with an “.xcat” extension (should be created manually before) to constitute a new catalogue. Such directory can then loaded as usual.

### *i. Properties editions*


Default values are normally defined for most of the applications cases and may have not to be changed, but Properties can be edited and linked values (i.e. Characteristics) can be changed. In the selection tree, expand the selected property by double clicking on it. The stored characteristics should appear. Please select one to be modified. The lower right panel should be updated as illustrated in Figure 23.

Change the value as you want and validate by clicking on the OK button. If a Characteristic is defined on several properties, it may have different values but should have the same type (e.g. float).



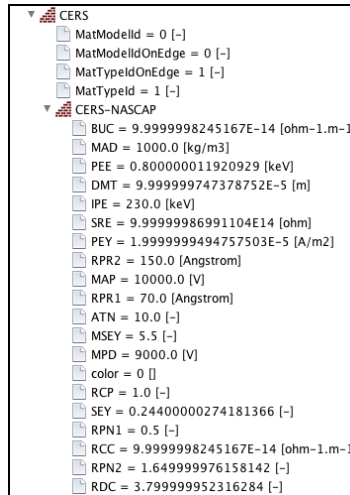
**Figure 23: Properties values change.**

### *ii. Materials definition*

Material can be attributed to spacecraft surface groups. In the SPIS context materials gathers numerical attribution flags and physical characteristics. Catalogues of material are automatically pre-loaded with a large set of predefined material or can be loaded by clicking on the  icon. Each characteristic can be selected and edited. It is possible to import NASCAP-2K based materials, with the 19 canonical parameters. Figure 24 show an example of SPIS material. Material properties significations are given in Annex G. A detailed list of NASCAP based material provided with SPIS is given in Annex H.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

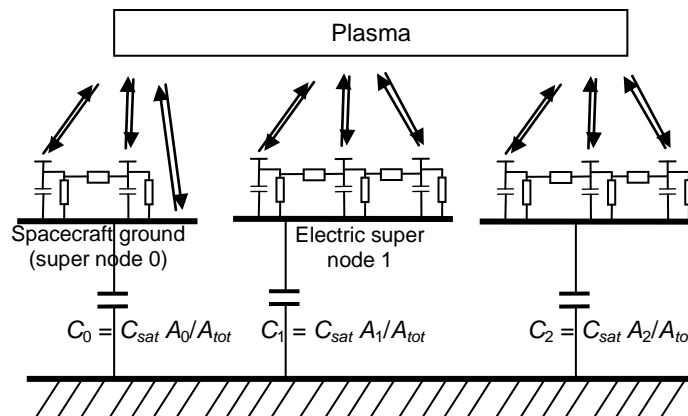
Materials are stored in XML based formats as given in Annexe B. Default material catalogues are provided in the \$SPIS\_ROOT/resources/defaults/properties/NascapMaterials sub-directory.



**Figure 24: Example of SPIS material, here from an import NASCAP based material.**

**More specifically concerning resistivity**, dielectrics are modelled as a capacitor in parallel with a resistor between the dielectric top surface facing the plasma and the back conducting surface, in order to model the bulk conductivity. Surface resistivity is modelled by a resistor between adjacent dielectric top surfaces.

Plasma spacecraft current exchanges (collection/emission) are achieved on all spacecraft surface groups, so-called electrical super nodes (ESN), which can be dielectrics or conductors. Absolute potential is calculating using the absolute spacecraft capacitance  $C_{sat}$  wrt to undisturbed plasma. Differential charging is calculated using the dielectric capacitance and resistivity.

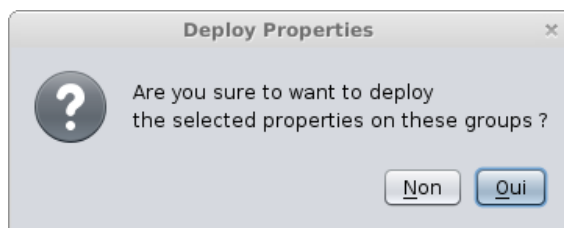


**Figure 25 - Spacecraft potential calculation modelling. In the present example, all separate ESN are floating with each other. They are linked to the undisturbed plasma potential through the local absolute capacitance (ratio of ESN surface area wrt to total spacecraft area)**

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



iii. *Properties deployment and DataFields generation*



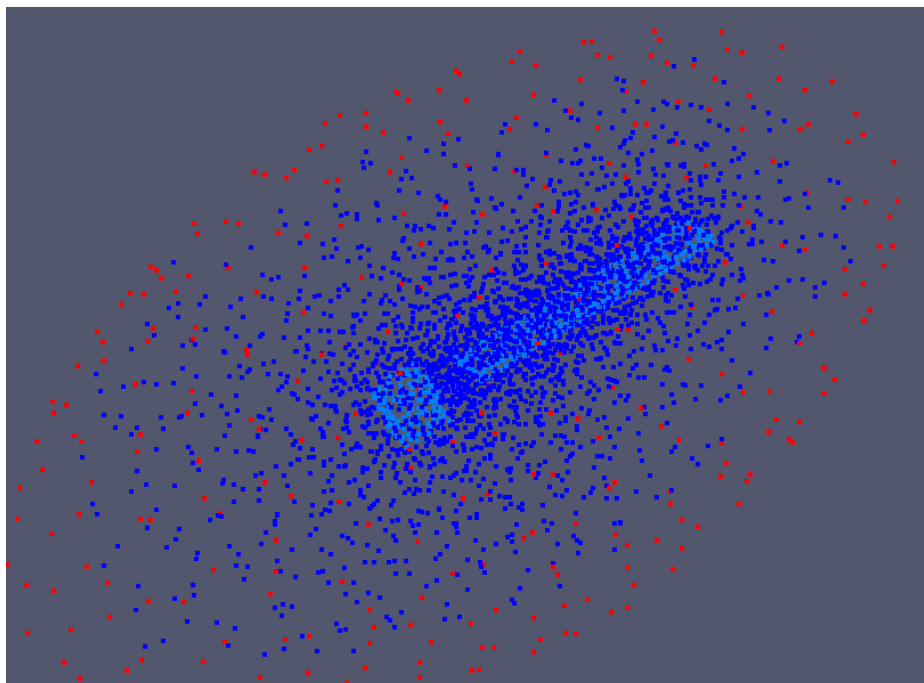
**Figure 26 Confirmation dialog box panel for the properties deployment**

The deployment of the properties information will lead to the construction of DataFields deployed on the mesh with the corresponding values. As illustration, Figure 27 shows the NodeFlag value DataField mapped on the whole mesh. Here NodeFlag equals 1 on the spacecraft (as defined in *SpacecraftDefault*), 0 inside the computational domain (as defined in *Plasma Model in Volume, default*) and 8 for the external boundary (as defined in *Boundary, default*). In most of the cases, this is such DataFields that are passed to the numerical kernels to set the Initial and Boundary Condition (IBC).

Please take care that a different group order may impact on the field mapping. For instance, in the present case, if the group of volume has the highest priority the final DataField will be uniformly set to 0. Such wrong settings may lead to errors at the model finalization (i.e. UltoNum phase).

The user can pass to the next panel, just by clicking on the *Next* button. The project is than automatically updated.

**Remark:** This method is costly in CPU resources and memory. Its complete processing might be long.

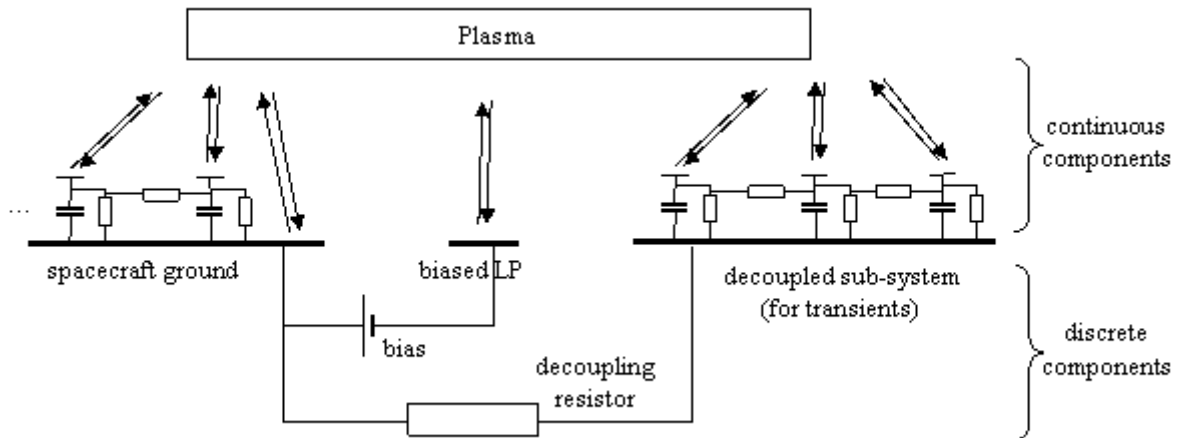


**Figure 27 Example of pre-processing DataField after deployment (i.e. SPIS-NUM NodeFlag). The value is set according the geometry groups and the properties initially defined. Here NodeFlag equals 1 on the spacecraft, 0 inside the computational domain and 8 for the external boundary. The mapping is done according to the priorities of groups.**

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

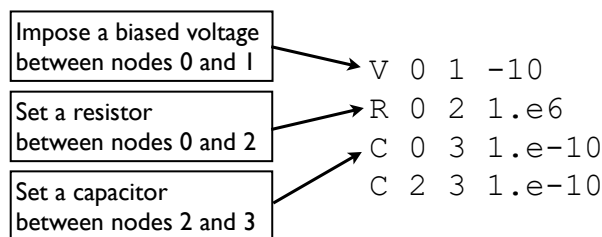
### 6.1.4. Internal electrical circuit setting

SPIS supports to describe the spacecraft structure with several macro electrical-nodes, in order to model differential charging and real time dynamics transient evolutions. By default, macro electrical nodes are floating wrt to undisturbed plasma with no connection with each other, as in Figure 25. The use of an internal circuit permits a better control of internally connected elements via an internal RCV, such as in .Figure 28. Electrical Super Nodes can be connected through resistors (R), capacitance (C) or voltage generator (V).



**Figure 28 - Example of internal circuit connections. Here, the ESN are not in floating wrt to the plasma. A potential bias is imposed between spacecraft ground and the “biased LP”. A resistor is imposed between the ground and the group on the right. The absolute capacitance is applied to the full spacecraft (not separated between ESNs).**

Figure 29 gives an example on internal circuit definition syntax.

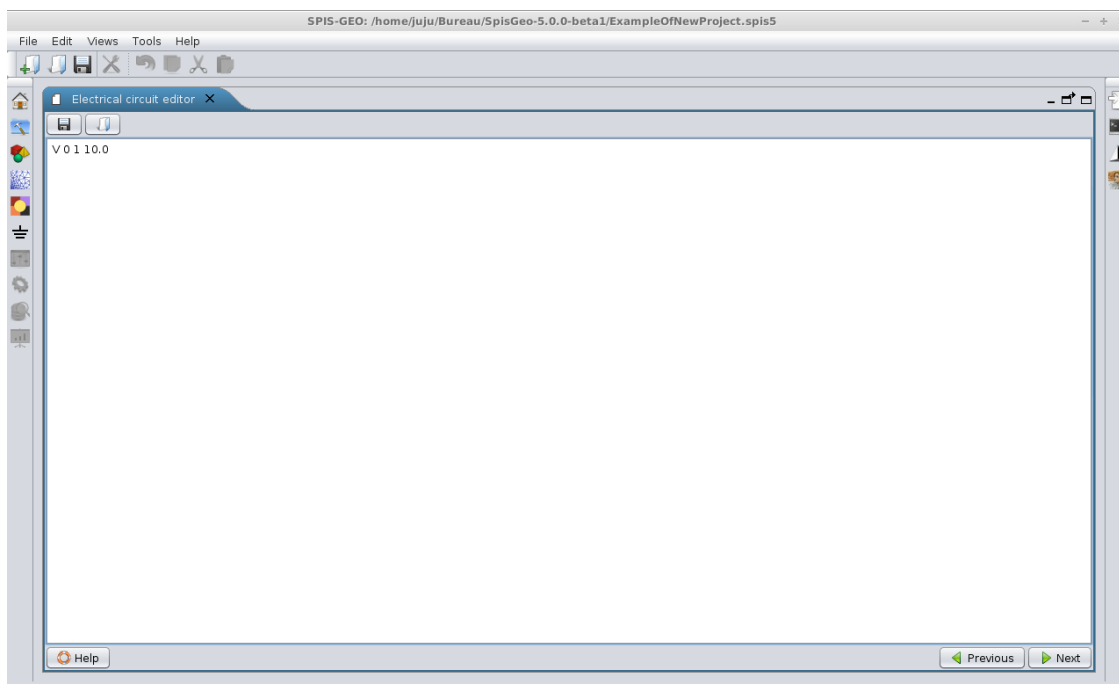


**Figure 29: Example of basic internal circuit definition.**

The internal circuit can be simply edited in the provided text editor as illustrated in Figure 30. The corresponding data are saved in a simple ASCII circuit.txt file in the project repository /NumKernel/Inputs.

The user can pass to the next panel, just by clicking on the *Next* button. The project is then automatically updated and internal circuit data saved.


<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 30: Edition of the internal electrical circuit**

### 6.1.5. Global parameters settings

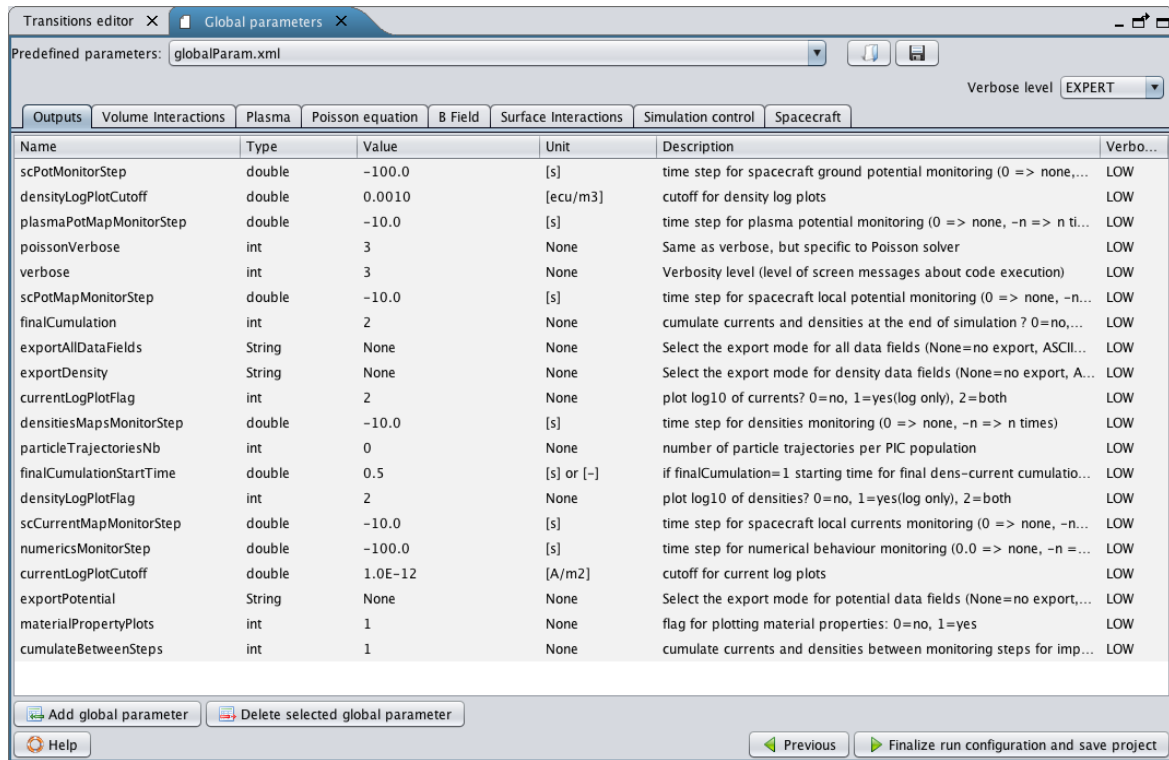
Global Parameters corresponds to parameters like global physical values of the space environment (e.g. Temperature) or numerical values (e.g. simulation duration) that are general to the simulation and have not to be deployed locally on the mesh. They can be set using the Global Parameters Editor. Global parameters are sub-divided into many categories and are available through several indexes and tables.

Global parameters are saved under the form of an XML file, as described in Annex D. Several preset templates of Global Parameters are available via the  icon in the editor panel. Most of these templates correspond to predefined flight conditions, like GEO worst cases or eclipse exit, "Annex 1 - Spacecraft surface charging in GEO/MEO".

**Remark:** The global parameter file format changed from SPIS 5 and SPIS 6. There is an extra field in the SPIS 6 global parameter file format "<valueAsLong>". If you want to open a global parameter file generated from SPIS 6 in SPIS5 or if you want to open a SPIS project with SPIS 5 and generated by SPIS 6, you have to remove in the global parameter xml file all the "<valueAsLong>0</valueAsLong>" fields. You can realize that with a research/relace feature available in any text editor for example.


**For a quick start,** we recommend to charge the globalParam-5.1.0.xml file. However, the user must keep in mind that these values are indicative and should be checked and adjusted to the studied case. Please see section 6.2 and the SPIS-NUM Documentation "How to control Num from UI" [RD7] for further information.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 31: Setting of the global parameters of the simulation.**

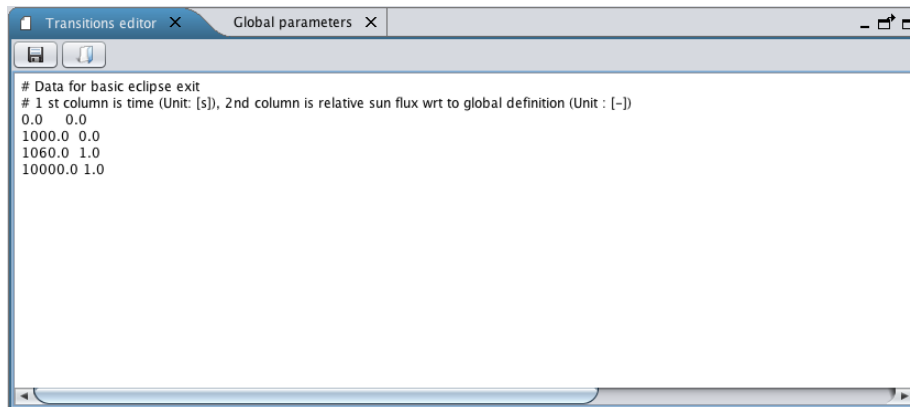
The *Global Parameters Editor* provides a field validation mechanism, checking if the input value corresponds to the set type (type column). Several levels (i.e. by increasing value low, medium, advanced, expert) of verbosity can be attributed parameter by parameter (Verbosity column). This feature allows to hide/show parameters by their level of relevance and/or user expertise. The global visualization verbosity level is set by the combo-box *Verbosity level* at the top right corner. Parameters with a verbosity level higher than the selected one will be hidden, as illustrated in Figure 33.

Modified parameters can manually saved by clicking on the  icon. This can be used to produce new templates also.

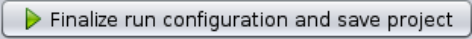
### 6.1.1. Transitions modelling

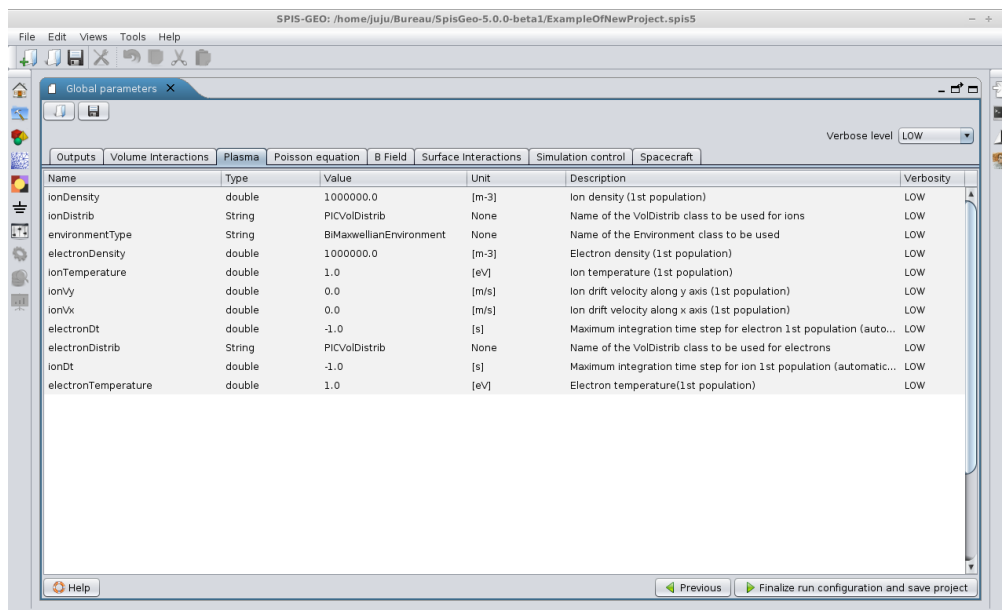
With SPIS 6 the possibility of parameters changing during the simulation integration, to model a transition for instance, has been introduced. These transitions can be edited and set by the Transition Editor index in the space step than the global parameters. Figure 32 shows an example of transition corresponding to an eclipse exit, with a modification of the sun flux versus the time.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 32: Edition of the transition models.**

The user can pass to the next panel just by clicking on the  button of the *Global Parameters Editor* index. The project is automatically updated and global parameters saved.



**Figure 33: Example of use of the verbosity level to focus on the most relevant parameters. The display parameters set can change according to the pre-settings of the project or the template.**

### 6.1.2. Model finalization and SPIS-NUM model building

The last step of the pre-processing procedure is the conversion of fields from the SPIS-UI data structure to the SPIS-NUM one. This action will also build the corresponding numerical model. This phase corresponds to UI2NUM action in *SPIS-Legacy*.

Error messages at this stage mean that the system definition is incorrect and must be corrected.

This phase includes several complex data conversion processes and can be long on large simulations.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

## 6.2. Quick start with the numerical kernel

The main parameters that must be set by users concern global parameters. SPIS uses more than 200 global parameters and local parameters to control physical models but also numerical algorithms. This is not the goal of the present document to list and describe them all.

The first objective of this section is rather to give an overview of the main physical and numerical models used in SPIS.

The second objective is to provide basic users with a quick start methodology.

### 6.2.1. Overview and quick start settings

**Preliminary remark:** The pre-defined settings developed in the frame of SPIS-GEO/MEO activity are adapted to a very quick start with SPIS (quasi in a black box mode) in geosynchronous and middle Earth orbits charging environments, see corresponding "Annex 1 - Spacecraft surface charging in GEO/MEO". Here, we present an intermediate level of details in order to fit with more applications. Users can easily modify the pre-defined XML file provided with the release (settings for ECSS and NASA worst case environment).

SPIS can be divided in 4 domains:

- **Matter** : electrons, ions, artificial sources, etc
- **Fields** : electric and magnetic field
- **Spacecraft surface interactions** : photoemission, secondary emission, etc
- **Spacecraft surface potential** : conductivities, bias potentials, etc

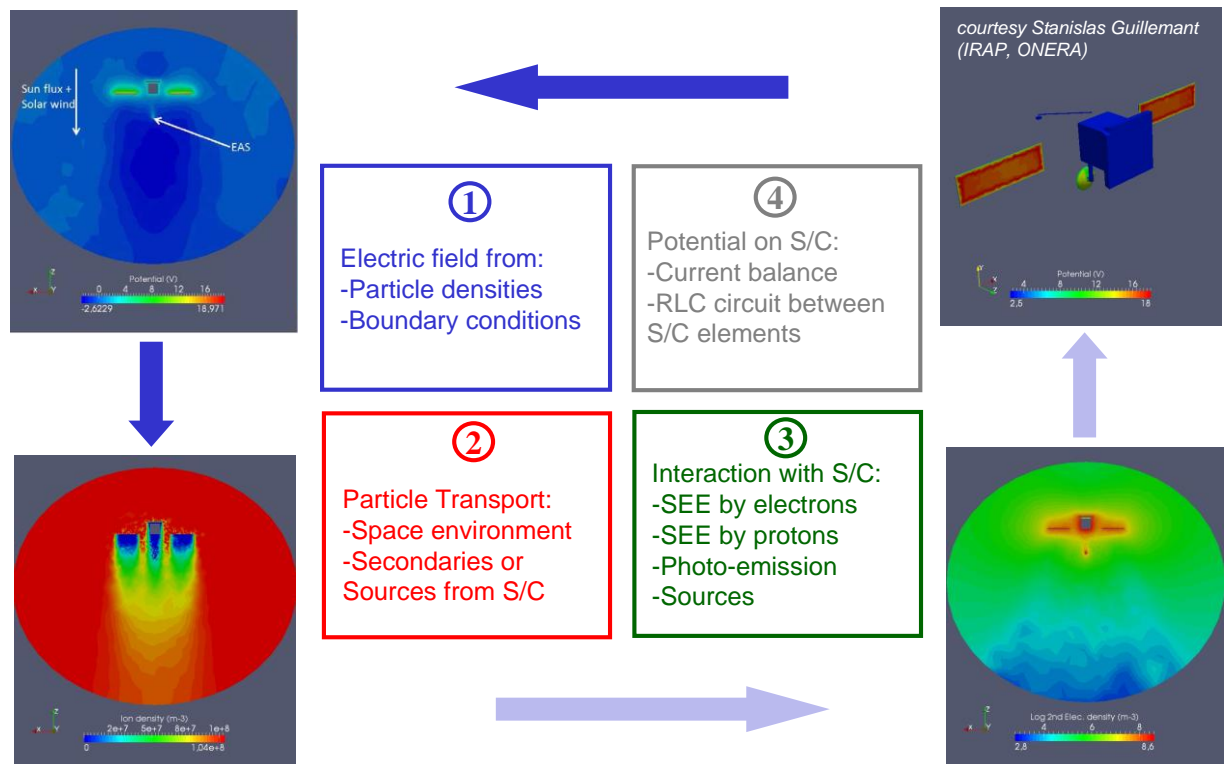
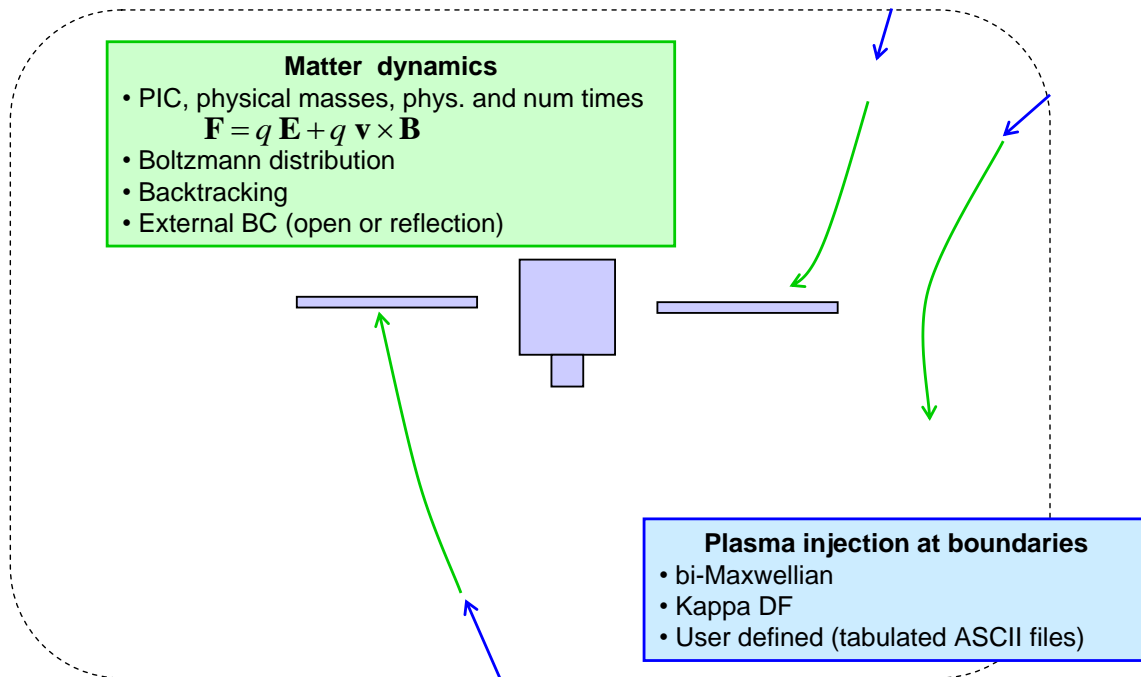


Figure 34 - SPIS numerical simulation principle

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



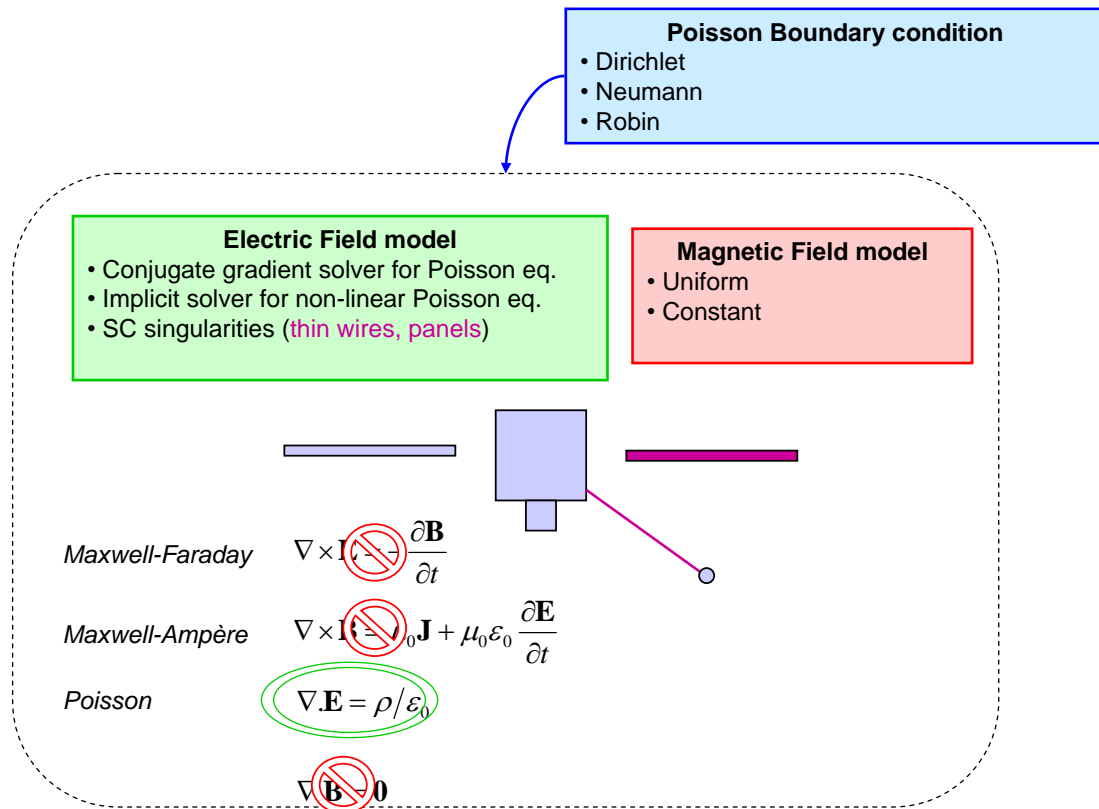
**Figure 35 - Matter model**

Concerning matter, the main models are a Particle-In-Cell (PIC) solver for a Monte Carlo solving of Vlasov equation, and a Boltzmann distribution to describe the thermal equilibrium distribution of electrons. Particle source distributions include a whole library. Gas phase collisions can be modelled through a MCC method, with only charge exchange reactions implemented as of today. Available surface interaction models cover most interactions relevant to space environment: secondary electron emissions (under electron or proton), photo emission, radiation-induced-conductivity, erosion. A complex 3D multizone model was defined to handle simultaneously in a single simulation a dense quasi neutral region and a space charge region, typically influenced by an unscreened positive potential nearby.

**For a quick start**, users may select the following global parameters of the "plasma" box

- Bi maxwellian ambient environment is selected by default: environmentType → BiMaxwellianEnvironment
- Ions may be treated as PIC (ionDistrib → PICVolDistrib) and electrons as a Maxwell Boltzmann equilibrium distribution (electronDistrib → GlobalMaxwellBoltzmannVolDistrib) assuming only negative potentials and no barriers of potential, and a "transparent" spacecraft
- Simply set the populations density and temperature
- Time steps for PIC populations may be calculated using time integration notice of paragraph 6.2.2.
- On a multi-core Operating System, select a multi-thread method for particle pushing (as e.g. pusherThreadNb → 4)
- Control the statistics by using more or less super particles (avPartNbPerCell)

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 36 - Field model**

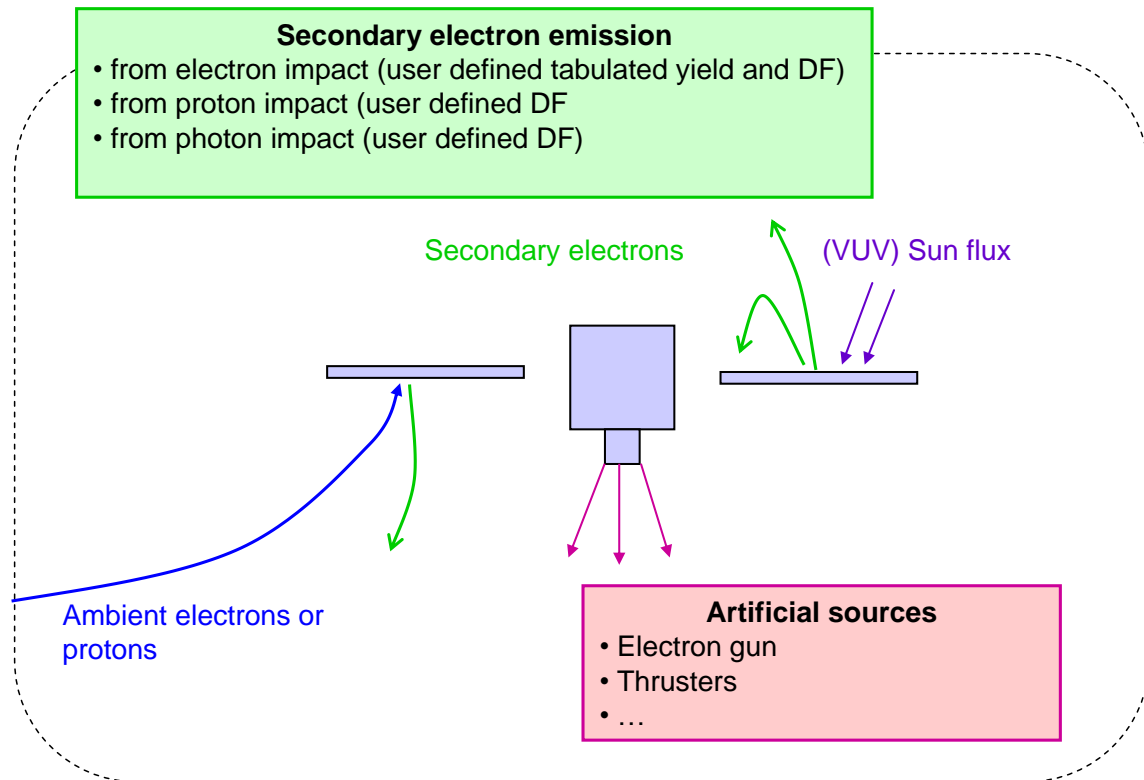
Concerning fields, the Poisson equation finite element solver follows a pre-conditioned conjugate gradient method. The boundary conditions can either be Dirichlet, Neumann or a mix of them (known as Robin or Fourier), which allows a better modelling of pre-sheath conditions ( $1/r^n$  behaviour). Non linear Poisson equation (i.e. Poisson including Boltzmann distributions for the electrons) can also be solved with an implicit method, offering stability even for a Debye length smaller than cells. Specific features were added to handle singular geometries, either very thin wires (of which radius cannot be meshed without generating degenerated elements) or very thin plates (of which edges cannot be meshed for the same reason). The method consists in analytically subtracting the singular part of the potential field resulting from the geometrical singularity, and solving the regular part (this splitting being of course transparent for the user).

**For a quick start**, users may select the following global parameters of the "Poisson equation" box:

- In case of ambient electrons modelled with Maxwell-Boltzmann equilibrium distribution, use the non-linear Poisson solver: linearPoisson  $\rightarrow$  0
- Electric field boundary conditions: vacuum-like or pre-sheath condition, respectively poissonBCTYPE  $\rightarrow$  1 and 2
- Time steps for Poisson equation solving may be calculated using time integration notice of paragraph 6.2.2.
- For some demanding cases, as e.g. small Debye length, the user is invited to refine the Poisson iterative solver convergence criteria (decrease "tol" and increase "iter" parameters)
- MagnetizedPlasmaFlag = 1 will calculate particle dynamics including the BX-Y-Z component of the electric field

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	





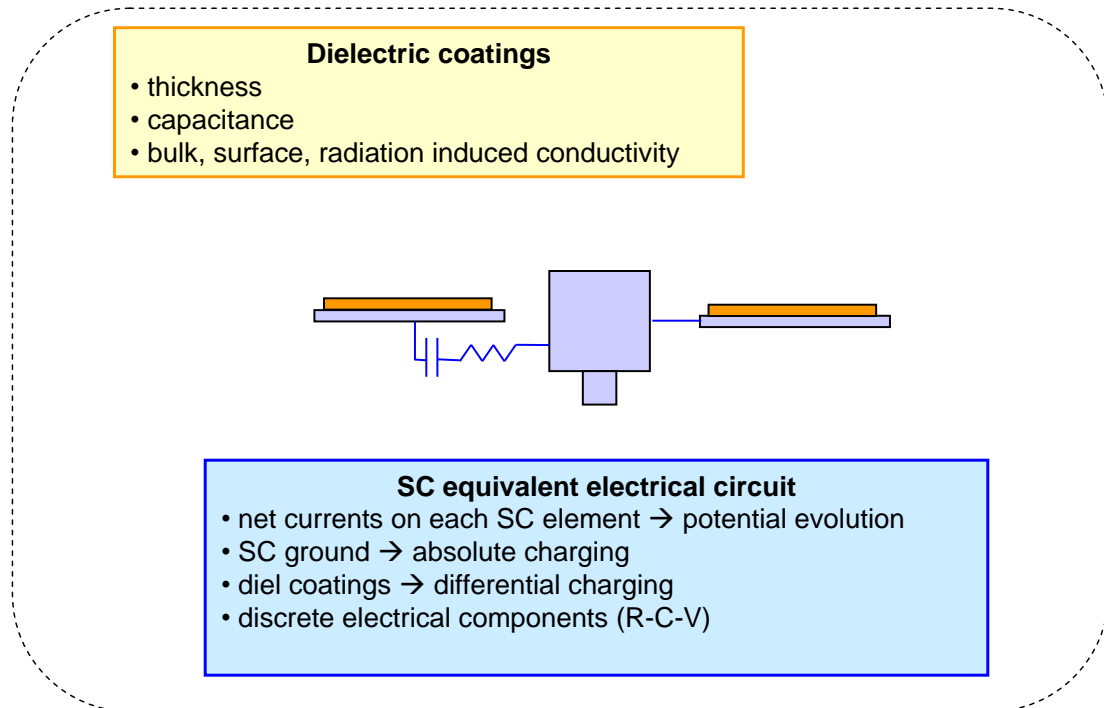
**Figure 37 - Spacecraft surface interaction with environment**

Surface interactions with plasma considered in SPIS are the secondary electron emission under electron, proton and photon impact. Yields are calculated automatically within SPIS pending on macroscopic material properties defined at user level. For instance the yield of true electron emission under electron impact is computed using the maximal yield and the energy of the maximum. Distribution functions are Maxwellian in volume (or Lambertian in surface). Secondaries from secondaries is activated for electrons from environment and artificial sources on spacecraft. Artificial sources models are maxwellian (possibly with a drift), axisymmetric and two axis symmetry (plus sources like field effect emission). Erosion model is applicable to Xenon ions.

**For a quick start**, users may select the following global parameters of the "Surface interactions" box:

- photoemission : 1 → activates photoemission, 3 → also model photoelectrons with PIC. |Sun|=1 will assume a spacecraft at 1 AU from the Sun. The Sun Flux can be multiplied by using SunX-Y-Z.
- electronSecondaryEmission: 1 → activates SEE from electron impact for all ambient electron populations; 3 → also models SEE electrons as PIC; 7 → also perform secondaries from secondaries
- protonSecondaryEmission: 1 → activates SEE from proton impact for all ambient protons; 3 → model SEE electrons as PIC; 7 → also perform secondaries from secondaries
- "densification" parameters permits to control the number of super particles emitted
- Time steps for PIC populations may be calculated using time integration notice of paragraph 6.2.2.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 38 - Surface potential model in SPIS 4.3 version**

The evolution of potentials on the spacecraft involves an equivalent circuit taking into account the coating capacitances and conductances (surface and volume conductivities), and also user-defined "discrete" components (extra resistors, biases or capacitors added between subsystems). A spacecraft circuit implicit solver was designed to handle problems with very different time scales thanks to physics predictors of the plasma current variations in reaction to potential changes.

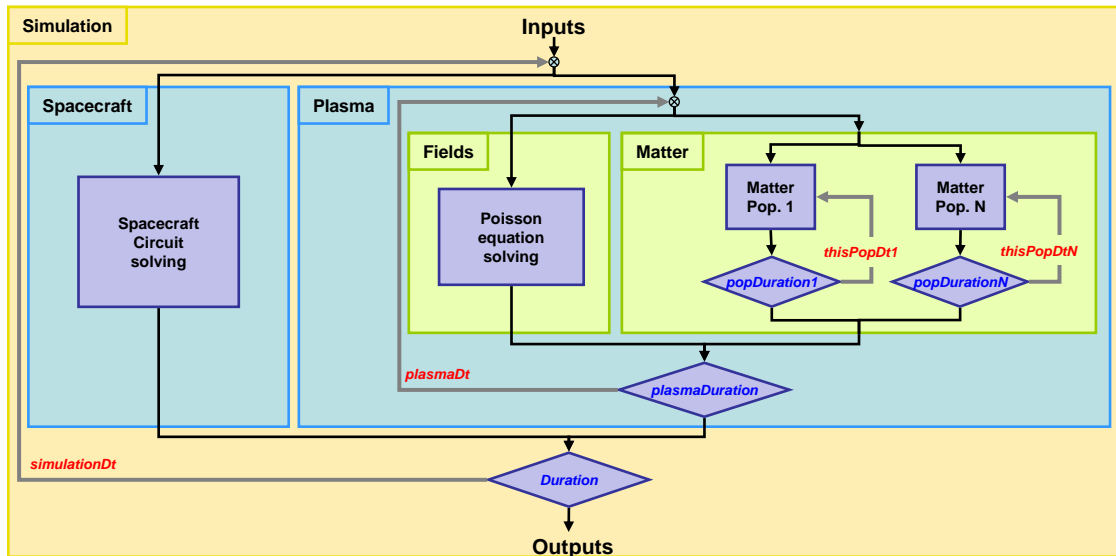
**For a quick start**, users may select the following global parameters of the "Spacecraft" box:

- electricCircuitIntegrate: 1 → yes (automatic adaptation of time steps see section 6.2.2); 0 → no (fixed potentials)
- In case of circuit integration, absolute spacecraft capacitance CSat of 10 nF or larger is generally acceptable. As the steady-state regime does not depend on CSat, it may be good to increase the value (to 1 μF) in order to reduce numerical constraints on circuit solver.
- initPotFlag = 2 is combined with initial voltage conditions defined in the group editor, see group editor section 6.1.3. Most common use is however: initPotFlag = 1 and initPot = 0 for a spacecraft in floating mode.
- Conductivity activation is controlled by 3 parameters: volumeConductivity, surfaceConductivity and (radiation) inducedConductivity. (1 → activate)
- validityRenormalisation larger than 1 permits to speed the simulation by increasing time steps. but possibly with a decrease in the accuracy. Special care should be taken.
- In addition it is recommended not to let electrical super node (ESN) grounds float with respect to each other. In the internal circuit description described in paragraph 6.1.4, prefer using **V 0 X 0** for any ESN of index  $X \in [1; \text{max ESN index}]$ .
- Time steps for surface potential updates may be calculated using time integration notice of paragraph 6.2.2.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

## 6.2.2. Integration loops control - How to set time steps?

The hierarchical structure of a simulation is outlined in the figure below. The nested boxes reflect the object structure of the code (what the basic user may not care about) while the arrows represent the time evolution of a simulation (what he needs to be aware of).



**Figure 39: Simulation cycles.**

The SPIS simulations can be divided in three levels of control:

- At simulation level: interaction between the satellite and the plasma (calculation of spacecraft potentials as a function of collected and emitted currents, emission from spacecraft)
- At plasma level: resolution of the particle transport coupled with Poisson equation
- At particles – populations level: particle trajectories in the Electric Field and the Magnetic Field, weighting algorithm to calculate densities in volume and currents on surfaces

For each level of control, a duration of integration (*xxxDuration*) and a time step (*xxxDt*) can be defined to control the integration process during the simulation. The ratio of the duration on the time step is the number of sub-cycling at this level (*xxxLoopNb*). In the figure below, these different levels are represented. In the present version of SPIS UI, only the three nested levels can be controlled: Simulation, Plasma and Matter.

In the most general case, all the components of the simulation are resolved in real time, the duration of integration of one level is equal to the time step of the upper level (for example  $popDuration\# = plasmaDt$ , where # is the index of the population). Nevertheless, in certain cases, the characteristic time of two processes can be very different. In such a case, the simulation may be sped up by considering that the fast process dynamics is quasi-static as compared to the slow one. Thus the stationary state of the fast process can be attained without integrating over the complete duration of the upper level time step. Consequently, it could be useful to define an integration duration different from the upper level time step (for example  $popDuration\# < plasmaDt$ ). Two examples can be cited:

- Example 1: the time evolution of the plasma compared to the spacecraft differential charging. In certain case, the potential evolution can be very slow (in order of seconds) compared to the

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

plasma dynamic (in order of micro-seconds to nanoseconds). Plasma can thus often be considered as stationary at the large time scale of surface potential dynamics (to be checked by the user in each case). In such a case, it is not useful to integrate the plasma evolution over the complete duration of the simulation time step. The *plasmaDuration* may be set by the user to a value several order of magnitude lower compared to the *simulationDt* (choice of this value is the responsibility of the users as a function of possible physical assumptions in each cases).

- Example 2: two populations can have very different velocity. It is quasi-systematically the case between ions and electrons in plasma due to the mass ratio, but it can also be the case between two populations which have very different energies. In such a case, the efficiency in term of calculation velocity can be substantially increased by selecting appropriate integration duration for both populations. For the slow population, the integration duration should be  $popDuration1 = plasmaDt$  (i.e. real time calculation) and for the fast population  $popDuration2 \ll plasmaDt$  (i.e. quasi-stationary state) taking into account the velocity ratio.

In SPIS v4.2 and earlier versions, two mechanisms permitted to define an integration duration different from the upper level time step: the speed up ratio and the *fixedDt = 1* mode.

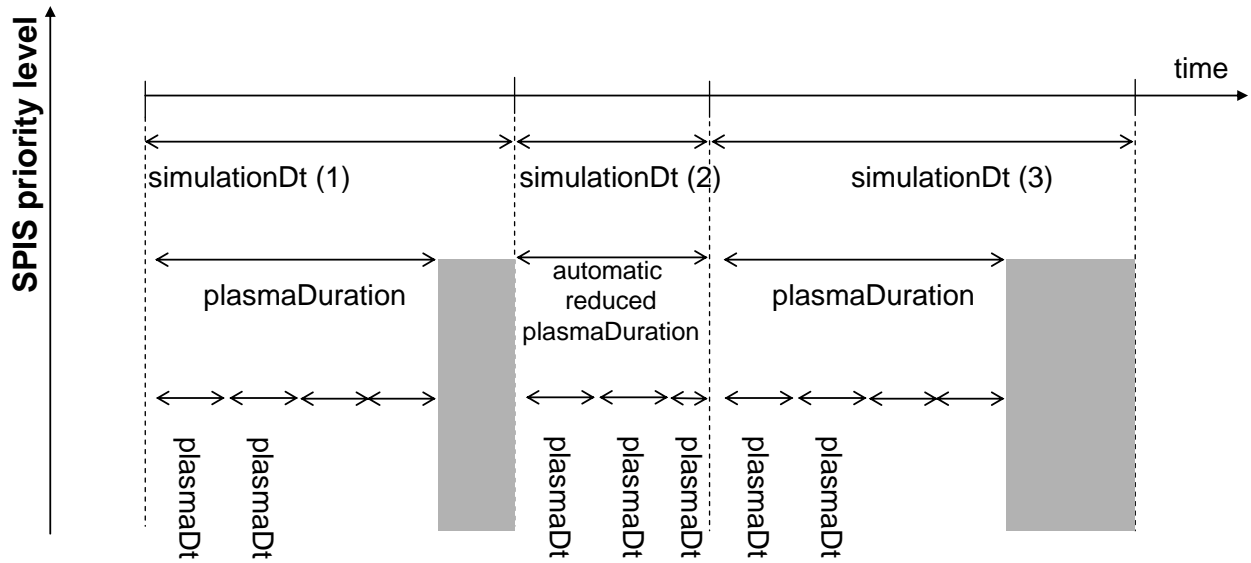
In the present version of SPIS, a new possibility is offered to users to determine the time steps and durations. **This recommended mode** is described in the following table that describes how to use global parameters (please refer to Time\_Step\_SPIS\_RC\_4.3.html page for details on other modes).

**Tableau 5: Main parameters of settings of simulation times steps and duration.**

Integration loop Parameter	Global Parameter	How to control
Simulation duration	<i>duration</i>	fixed by user
Simulation time step	<i>simulationDtInit</i>	Initial simulation time step
Simulation time step	<i>simulationDt</i>	If positive, automatic: only considered as a maximum and <i>simulationDt</i> is increased or decreased within validity limits (validity of linear extrapolation of collected/emitted currents as a function of local potentials) starting from $simulationDt = simulationDtInit$  If negative, fixed by user to its absolute value (an infinite validity is set to the current scalers to be consistent – flag to enable or disable the current scaler $dI/dV - dIdVFlag = 0 / 1$ )
Plasma loop duration	<i>plasmaDuration</i>	If positive, duration fixed by user with automatic control ensuring that duration is smaller than <i>simulationDt</i> .
Plasma loop time step	<i>plasmaDt</i>	If positive, time step fixed by user.
Pop# loop duration	<i>pop#Duration</i>	If positive, the duration is fixed by user with automatic control ensuring that duration is smaller than <i>plasmaDt</i>
Pop# loop time step	<i>pop#Dt</i>	If positive, time step fixed by user with automatic control ensuring that duration is smaller than to <i>pop#Duration</i>

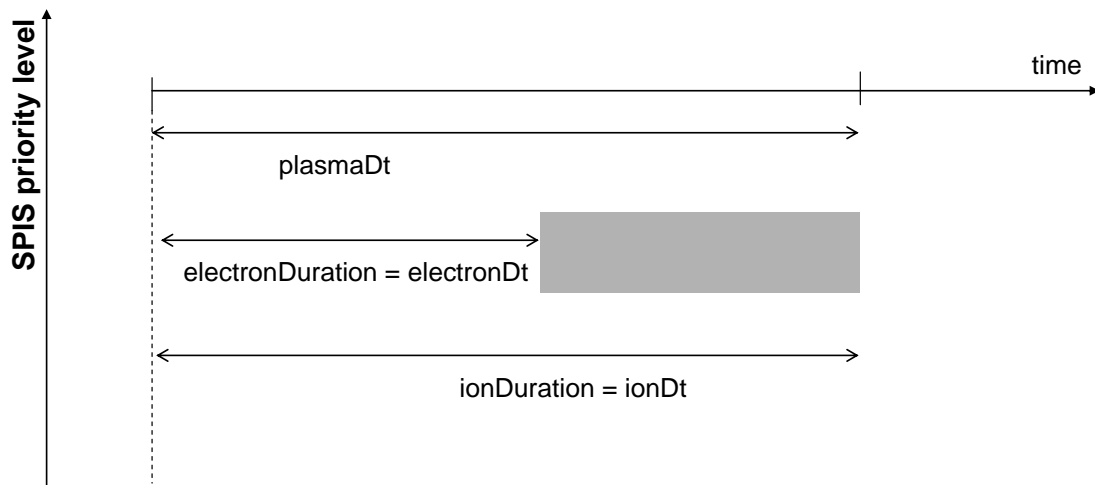
An illustrative diagram of time steps and integration duration is given in Figure 40 for the Example 1 cited above. The simulation time step being positive, it is automatically adapted by the numerical kernel during the simulation, in order to prevent large potential variations between two time steps. The plasma duration is kept constant, but still lower than the current simulation time step. The same paradigm prevails for the plasma/particle loop

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 40 - Time step series illustrating the Example 1 of a simulation with numerically accelerated plasma integration wrt to simulation loop**

A second illustrative diagram is given in Figure 41 for the Example 2 cited above. The electron population need not being computed in the same footing as ions since they can reach local steady-state much faster.



**Figure 41 - Time step series illustrating the Example 2 of a simulation with numerically accelerated electron population integration wrt to plasma loop**

**Recommendations for a quick start:**

We recommend to set time steps and durations in the following order

- Plasma - spacecraft loop
  - duration = the estimated time to reach stable steady-state equilibrium
  - simulationDt <= duration/10

Reference: ESA-SPIS-GEO-D7-SUM-2013-03-001	Version: 3	Revision: 5
Status: Final	Date: 28/11/2019	

- simulationDtInit <= duration/1000
- Population loop (only for PICVoIDistrib or CompositePICBacktrackingVoIDistrib)
  - popDt = time to cross the box = size / mean velocity
  - popDuration = time to cross the box = size / mean velocity
- Plasma loop
  - plasmaDt <= 0.2 \* plasma period for PIC plasma modelling stability
  - plasmaDuration >= max( 2\*plasmaDt; max(popDuration))

NB: in some modelling cases, these parameters are irrelevant (Boltzmann electrons, SC at fixed potential...).

**Remark for an even-quicker start:** if equal to 1, the spisGEO global parameter flag permits to calculate automatically all time steps and durations, except *duration* and *simulationDtInit* and *simulationDt*. However this mode is only compatible to a **simplified use in GEO/MEO surface charging simulation**, see "Annex 1 - Spacecraft surface charging in GEO/MEO".

### 6.2.3. Methodology for two typical situations

In this paragraph, we provide basic users with a general methodology to quickly set ambient populations in two configurations:

- Negative spacecraft (often in GEO and sometimes in LEO)
- Positive spacecraft (often in LEO and in solar wind)

The methodology for secondary electrons is also described

#### Negative spacecraft

- Use of the Boltzmann electron model + Non-linear Poisson solver
- Main advantage: you can mesh with grid much larger than Debye length !
- In global parameters
  - select GlobalMaxwellBoltzmannVoIDistrib instead of PICVoIDistrib for ambient elec1 and elec2
    - no time step and duration: not necessary to set electronDt and electronDuration
  - select linearPoisson = 0
  - for ions distribution
    - use PICVoIDistrib or CompositePICBacktrackingVoIDistrib if the Debye length is small <= spacecraft size

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

- select the time step and duration for ions: ionDt and ionDuration populations
  - use BacktrackingVolDistrib for ions if Debye length is large >> spacecraft size
    - no time step and duration for ions: not necessary
- Plasma loop
  - select the time step and duration for plasma : plasmaDt and plasmaDuration

### Positive spacecraft

- Use PIC modelling for electrons and ions
- In global parameters
  - For electrons and ions distributions
    - use PICVolDistrib or CompositePICBacktrackingVolDistrib if the Debye length is small
      - select the time step and duration for populations
    - use BacktrackingVolDistrib for ions if the Debye length is large
  - select the time step and duration for plasma
  - select linear Poisson = 1
- Mesh the plasma volume with grids smaller than  $0.5 * \text{Debye length}$

### Secondary electrons

- set photoemission and secondary electron emission flags > 1 as explained earlier
- set the temperature of secondary electrons and photoelectrons
- if PIC modelling, set secondaryDt and secondaryDuration

**Remark:** there is no warrantee that this simple configuration models apply to any situation listed above. **It is the responsibility of the user to verify the relevance with specific cases studied.**

### 6.2.4. Where should I put my input ASCII files?

The numerical kernel of SPIS uses the directory located at mySPISprojectRoot/DefaultStudy/Simulations/RunId/NumKernel/Input.

All inputs files shall be put there. It includes the circuit file, artificial sources definitions, and files for GEO/MEO applications and advanced use for scientific applications.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

### 6.2.5. Where do I get extra ASCII files results?

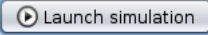
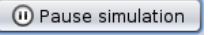
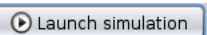
In complement with results provided in the Data Miner, the numerical kernel offers full list of ASCII tables. They are located at SPISprojectRoot/ DefaultStudy/Simulations/RunId/NumKernel/Output.

### 6.2.6. Where is the documentation of the full list of parameters?

For a complete view of SPIS capabilities, we invite the reader to read the reference html document in complement to this [RD7].

## 6.3. Simulation running and control

The Simulation Running and Control panel, as illustrated in Figure 42, allows to follow in real time the evolution of the simulation and control this one. The left panel, Instrument List Viewer, allows to select the instruments used to monitor the simulation, see section 7. Outputs are displayed on the right side panel. Several indexes can be displayed. Monitors are automatically updated during the simulation evolution.

The simulation run is started simply just by clicking on the  button. It can reciprocally be paused by clicking on the  button. This will also extract the computed data at the current simulation time. The simulation may be resumed with a new duration by clicking on  after changing the final duration. The progress bar in the bottom panel is updated according the simulation progression.

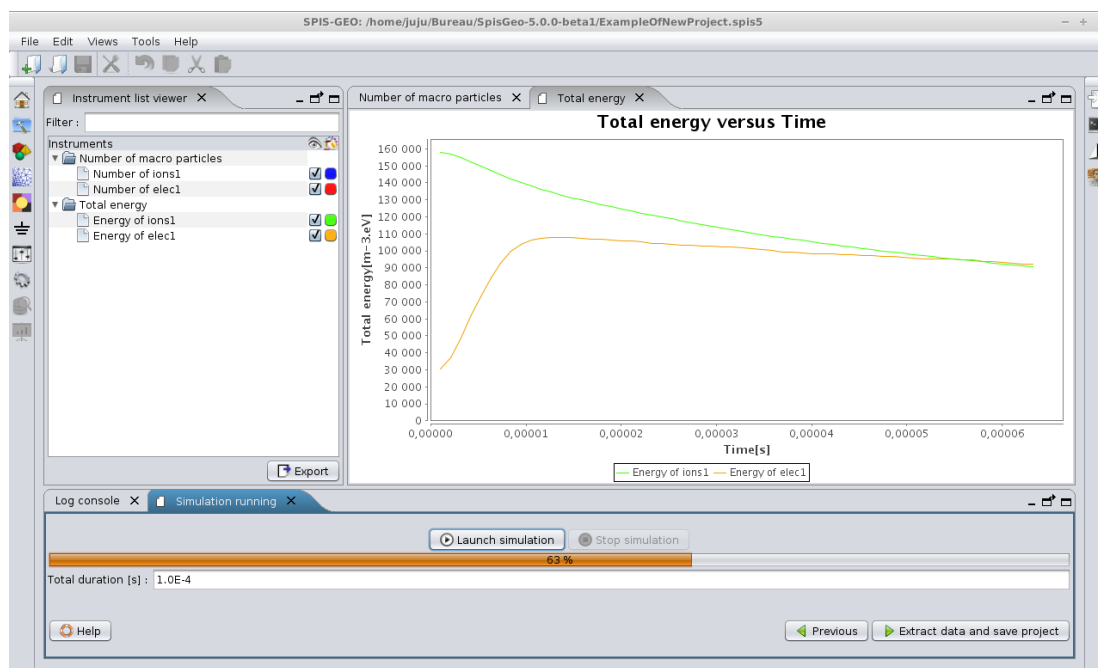
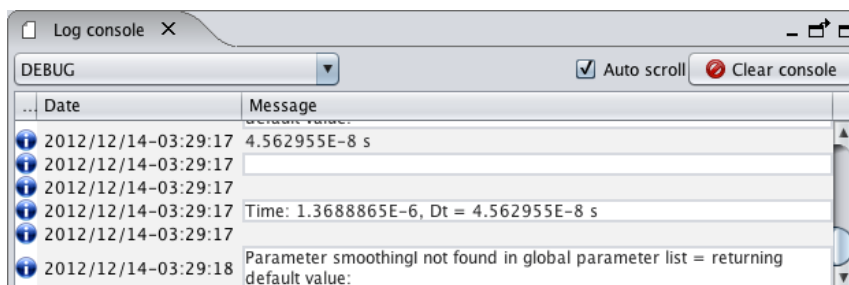


Figure 42: Simulation control panel.

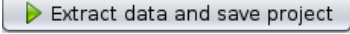
<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



A detailed log window of the simulation kernel (SPIS-NUM) is available through the *Log console* index of the bottom panel. This console gathers all outputs and information messages generated by SPIS-NUM. We strongly recommend to look carefully at the generated message to check the consistency of the simulation. Please see the SPIS-NUM documentation [RD7] or the SPIS training support [RD3] for further information regarding the meaning of the generated messages.



**Figure 43: Log console of the SPIS-NUM simulation kernel.**

The user can go the post-processing panel and the results analysis just by clicking on the  button. The data generated by SPIS-NUM will then extracted, converted as DataFields and saved into the project. This phase can be long on large simulations.

When the simulation reaches the final duration time, the simulation is automatically paused. The user can choose to

- Finish the simulation
- Or set a new final time and continue the simulation

Each time the final time is reached, detailed information on the computational time is monitored in the log console (global simulation duration, plasma loop, circuit solver, etc.), as shown in Figure 44.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

```

|-- SPIS numerical simulation --|
|-- Task durations
|  |-- Task Simulation integration      | Cumulative duration :      24 MINUTES
|  |-- Task Plasma                    | Cumulative duration :      19 MINUTES
|  |-- Task Plasma/SC Interactions    | Cumulative duration :      17 SECONDS
|  |-- Task SC Circuit                | Cumulative duration :     177 SECONDS
|  |-- Task Results storing           | Cumulative duration :       81 SECONDS
|  |-- Task Transitions                | Cumulative duration :        7 MILLISECONDS
|  |-- Task Instruments               | Cumulative duration :        5 MILLISECONDS
|--
|----- Plasma subtasks
|  |-- Task Poisson Solver            | Cumulative duration :        3 SECONDS
|  |-- Task Move all populations      | Cumulative duration :      19 MINUTES
|  | At population level-----
|  |-- Task Injection of ions1 density component | Cumulative duration :      15 SECONDS
|  |-- Task Push of ions1 density component | Cumulative duration :     100 SECONDS|
|  |-- Task Move of ions1 density component | Cumulative duration :     130 SECONDS
|  |-- Task Injection of ions1 current component | Cumulative duration :      47 SECONDS
|  |-- Task Push of ions1 current component | Cumulative duration :     129 SECONDS
|  |-- Task Move of ions1 current component | Cumulative duration :     177 SECONDS
|  |-- Task Injection of ions2 density component | Cumulative duration :        7 SECONDS
|  |-- Task Push of ions2 density component | Cumulative duration :     36 SECONDS
|  |-- Task Move of ions2 density component | Cumulative duration :     57 SECONDS
|  |-- Task Injection of ions2 current component | Cumulative duration :     43 SECONDS
|  |-- Task Push of ions2 current component | Cumulative duration :     112 SECONDS
|  |-- Task Move of ions2 current component | Cumulative duration :     156 SECONDS
|  |-- Task Injection of photoElec     | Cumulative duration :     120 SECONDS
|  |-- Task Push of photoElec         | Cumulative duration :     170 SECONDS
|  |-- Task Move of photoElec         | Cumulative duration :        5 MINUTES
|  |-- Task Injection of secondElec   | Cumulative duration :     108 SECONDS
|  |-- Task Push of secondElec        | Cumulative duration :     198 SECONDS
|  |-- Task Move of secondElec        | Cumulative duration :        5 MINUTES
|-----

```

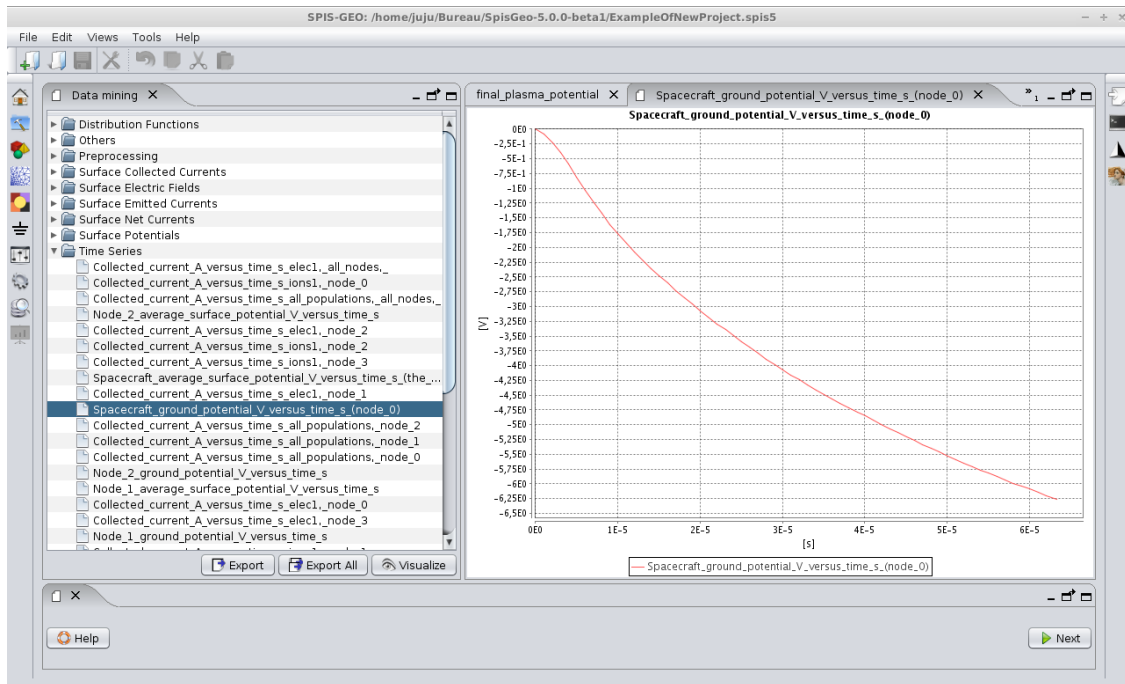
**Figure 44 - Example of detailed information provided at the end of the simulation. CPU cost of each integration loops or process is monitored.**

## 6.4. Post-processing and data analysis

### 6.4.1. Data miner

With respect to the previous versions of SPIS (i.e. SPIS-Legacy) and improved Data Miner has been introduced in order to both simplify the data analysis/visualization and allows a finer access to each data. Figure 45 shows the *Post-processing* panel, with the Data Miner, on the left, and the rendering panel, on the right side.

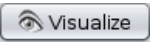
<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



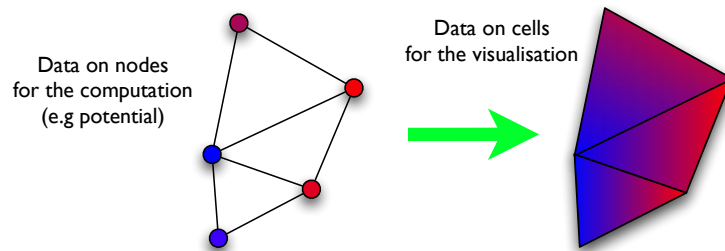
**Figure 45: Data mining and post-processing, example of 2D  $y = f(x)$  plot type rendering.**

Depending on the data type, several rendering are possible and/or are pre-selected as follows:


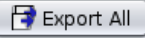
- **Time series and  $y=f(x)$  type data:** 2D plots ( $y=f(x)$ , histogram...). Figure 45 shows an example of such plot.
- **Surface data:** 3D surface view, with the following rules regarding the cell trans-typing (i.e. conversion of the localization on cells):
  - Computed on nodes, viewed as surfaces (linear interpolation between values at the cells nodes);
  - Computed on edges, viewed as edges;
  - Computed on faces, viewed as surfaces;
- **Volume data:** 3D volume view, with the following rules for the cell trans-typing:
  - Computed on nodes, viewed as cells (linear interpolation between values at the cells nodes);
  - Computed on edges, viewed as edges;
  - Computed on faces, viewed as surfaces;
  - Computed on cells, viewed as cell.

To visualize a given data, the user should first select it in the tree and then click on the  button. The corresponding viewer should appear on the right side panel. For Surface and Volume data, it is also possible to visualize the position of the sun computed from the global parameters. In the 3D view, open the control panel and perform a right click on the sun actor to change its visibility.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

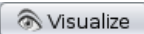


**Figure 46: Concept of cell trans-typing: Data computed on nodes are displayed on cells with an interpolation to visualize a continuous field.**

The user can export the selected data just by clicking on the  button. A file chooser should appear and offer a combo box with the supported export formats. The  button will export all data present in the Data Miner selection tree when the export is possible with the selected format.

#### 6.4.2. Cassandra, 3D visualization tool

To visualize 2D and 3D fields of data (DataFields) deployed on the mesh, SPIS-UI integrates Cassandra, a generic 3D scientific data viewer based on VTK. Cassandra provides a dynamic interaction with the VTK pipeline and enables to load plugins dynamically in order to perform specific tasks in data manipulation and visualization. More information about Cassandra is available on the Artemum Cassandra Web page [RD9].

When the user clicks on the  button, the selected DataField is converted into an equivalent vtkDataSet and loaded automatically in Cassandra. Figure 47 shows an example of visualized data set, with a view of the final plasma potential on the whole computational domain.

Once exported with the export button, it is possible to reload data later and outside the SPIS context, with any software able to load vtk files for further analysis.

A large list of filters is already available in the **Filters** menu of Cassandra and can be extended by additional plug-ins:

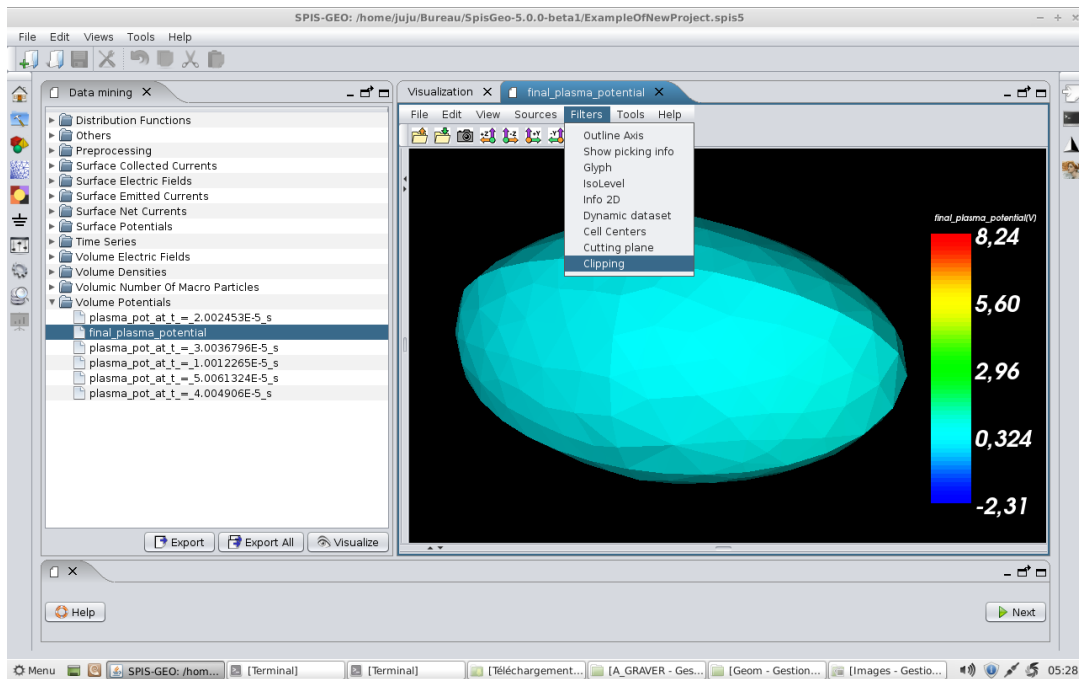
- **Cell Centers:** Compute an averaged (linear interpolation) value at the centre of the cells.
- **Clipping:** Perform a plan clipping (i.e. half-space extraction) of the grid.
- **Cutting plane:** Perform a cut of the grid from a given plan.
- **Outline Axis:** Compute and display the bounding box and the axis of the grid.
- **Show picking info:** Given the local information (value, Id, coordinates) of the cell or node pointed with the mouse.
- **Dynamic dataset:** Allow to load several data set of the same type in order to process them automatically with the same visualization pipeline.
- **Glyph:** Display a glyph (e.g. small arrow) on each cell according the local vector value of the data.
- **Info 2D:** Display the Ids of the nodes or cells of the grid.
- **Isolevel:** Compute the surfaces or levels of iso-values in the grid.

Some filters are depending on the type of processed data, please check before use.

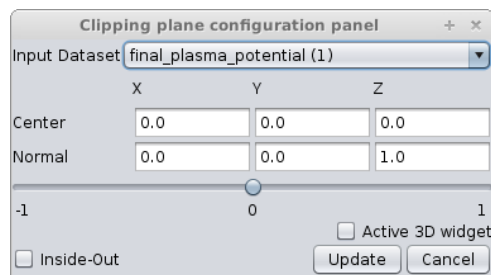
<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

Figure 47 shows how to perform a clipping on a volume data (here final plasma potential). The user should go to the Cassandra filters menu and select the *Clipping* filter. A control panel, as illustrated in Figure 48 should appear.

In the control panel, select the input data set, set the cutting plan position and orientation and click on the **Update** button. The data should be processed and the view updated. As illustration, Figure 49 shows the corresponding result.

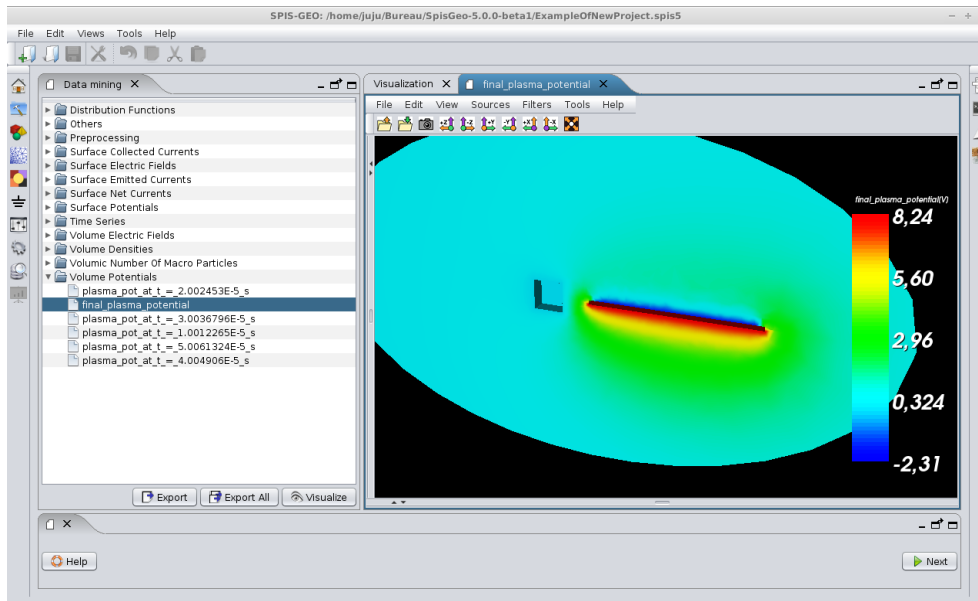


**Figure 47: Example of 3D post-processing**



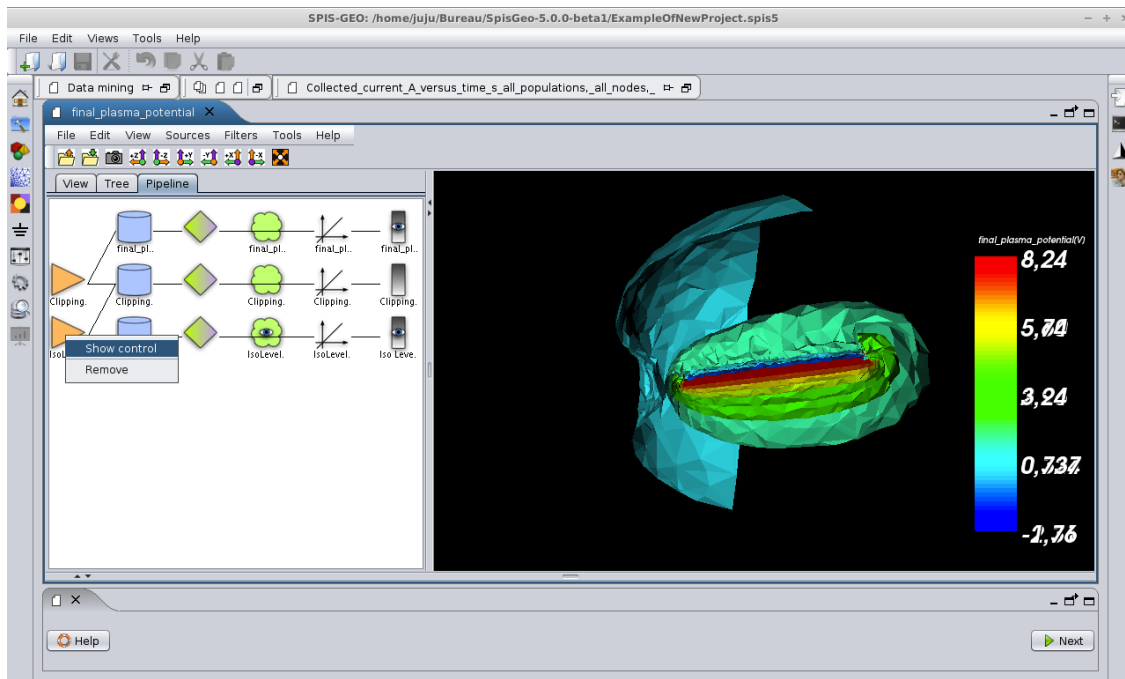
**Figure 48: Example of control panel of a Cassandra filter.**

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 49: Example of clipping of a 3D map around the spacecraft (here final plasma potential) performed using Cassandra.**

To access again to the control panel the user can expand the split pane on the left side of the 3D view. The Cassandra visualization pipeline editor will appear. For all nodes of the visualization pipeline, i.e. icon, the settings and control panels are accessible through the contextual menu, itself accessible by a simple right-button click on the selected icon. Figure 50 illustrates this procedure.




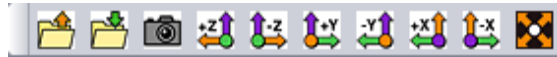
**Figure 50: Example of a complex processing**

Thanks to the Cassandra graphical pipeline editor, it is particularly easy to combine filters to build-up complex processing and refined data extraction. As illustrated in Figure 50, a second filter can be applied on the outputs of a first one, just by selection of the relevant input data set in the control panel

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

of the second filter. In the example shown here, an iso-level filter is applied on the output of the previous clipping operation.


The tool bar of Cassandra offers a set of classic actions on the 3D view, like the load/saving of selected data set, the possibility of screenshot or the reorientation of the view with respect to the main axis. The  icon resets the camera position to have a complete view of the loaded object.



**Figure 51: Tool bar of Cassandra, offer classic actions on the 3D view.**

For further information regarding the use of Cassandra please see the Cassandra's user manual and tutorial screencast on the Cassandra Web site [RD9].

### 6.4.3. Views and screenshot saving

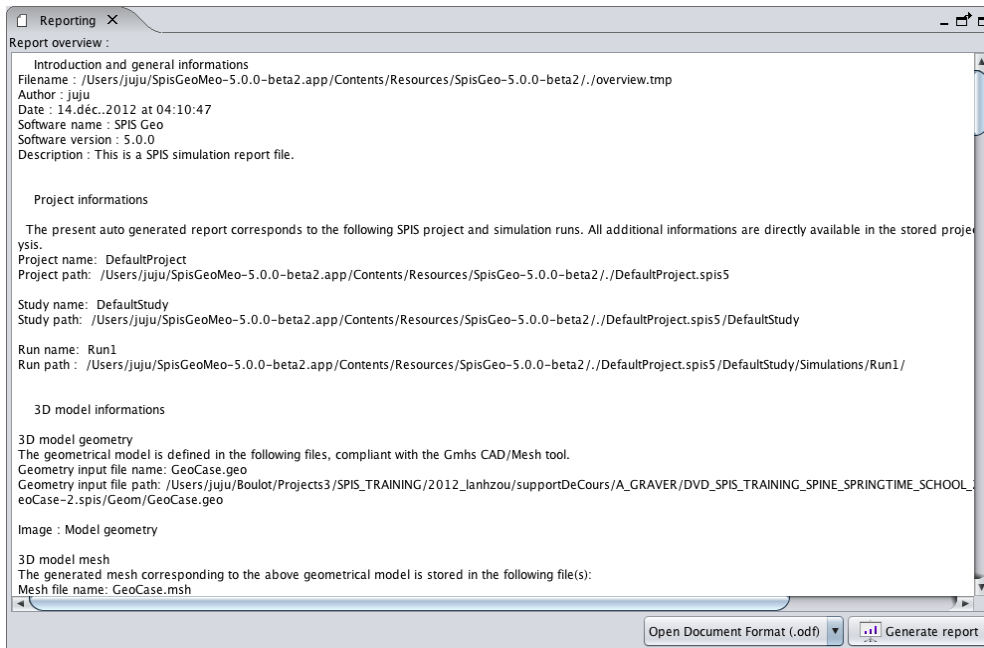
By clicking on the  button, all displayed views will automatically be exported as images (PNG) and saved into the *OutputFolder->Export* directory of the current simulation run.

## 6.5. Reporting

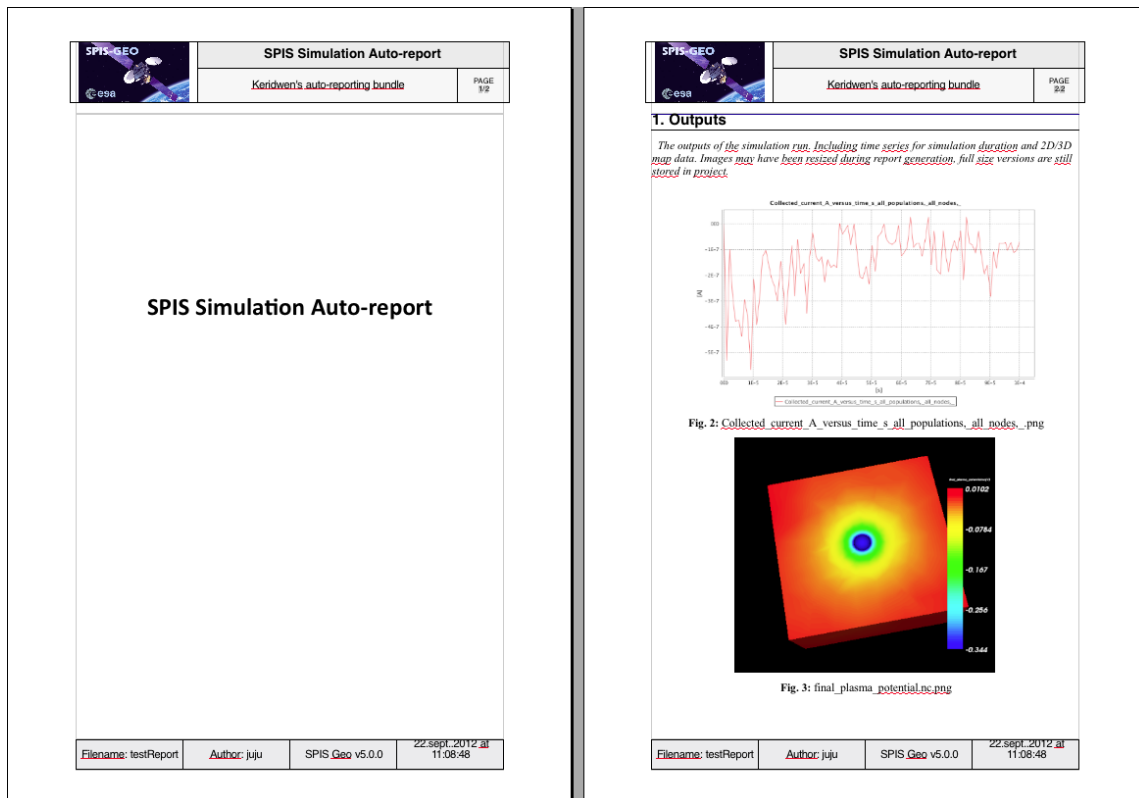
In order to help the user to present his key results a module of auto-reporting has been introduced. The auto-reporting module gathers into an Open Office document all inputs parameters and information needed to set and defined the whole simulation. The same is done for the outputs studied by the user in the post-processing phase. Screenshots, as viewed by the user, are also collected into the document. Figure 52 shows the GUI of the auto-reporting module.

The output document can be in plain text or Open Office (.odt) format. In the second case, the report is generated from an Open Office template. A default SPIS template is provided in the  $\$SPIS\_ROOT/resources/defaults$  directory, but it can be changed to be adapted to the user need and own graphical rules. Figure 53 shows an example of generated report with of the default template.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



**Figure 52: GUI of the auto-reporting module**



**Figure 53: Example of auto-generated report on the basis of the default template.**

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

This document is the property of Artemum and ONERA. It cannot be used, copied or distributed without prior written approval.



## 7. Instruments (General configuration and use)

### 7.1. Principle of the SPIS instruments

In order to improve the data extraction and probing from SPIS-NUM simulation, a generic instrument interface, called *SPIS Instruments*, has been introduced at the interface between SPIS-UI and SPIS-NUM. SPIS Instruments allow performing advanced measurements inside the whole computational domain and/or on the spacecraft surface.

The objective of the instruments is to provide the user with specific information on the simulation outputs. The results provided depend on the sub-type of instruments. The current implementation includes three main categories of instruments:

- **Particle detectors** (see details in the annex “Advance use for scientific applications”), in the large sense of the term, including:
  - **Particles Detector** itself provides mainly the particle distribution functions on dedicated spacecraft surfaces. They basically rely on a Test Particle (TP) method which consists in calculating the particle trajectories in a frozen electromagnetic field, by a series of forward and backward tracking.
  - **Langmuir probe** instrument extends the Particle Detector class by introducing IV sweep coupled with TP.
- **Virtual particle detectors** (see details in the annex “Advance use for scientific applications”) provide the same information as real particle detectors (defined on spacecraft surfaces) but on a surface independent from the spacecraft. They do not interact with the plasma and spacecraft dynamics. They are used to estimate currents flowing through a virtual surface immersed in the computational volume.
- **Plasma sensor** performs simple and quick diagnostics in the plasma volume. They do not interact with the plasma and spacecraft dynamics. Within the course of the SPIS-SCI activity, diagnostics of values living on spacecraft surfaces were also developed and reversed in this category. So “plasma sensors” should be understood by users as a **Live Monitoring of the simulation**. This category includes the following concrete implementations (in italic, sensors which are set automatically at simulation start), see 6.3:
  - *Average differential spacecraft potential monitor (dV)*
  - Density sensor
  - *Electrical super node current monitor*
  - *Electrical super node individual current monitor*
  - *Energy distribution function sensor*
  - Line plasma potential sensor
  - Mean kinetic energy sensor
  - Plasma potential sensor
  - *Surface potential of an electrical super node: average (V)*
  - *Surface potential of an electrical super node: difference w.r.t local ground (dV)*
  - *Surface potential of an electrical super node: time variation (V/s)*
  - Surfacic flux distribution function sensor
  - Total Energy distribution function sensor

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

- Total energy sensor
- Total super particle number sensor
- Velocity distribution function sensor

In order to simplify and normalize the access to scientific data, observations (i.e. data extraction) are done through a set of classes of instruments, each of them implementing a normalized interface called Instrument. All of classes listed above implement this interface. The Instrument interface offers a normalized API to extract scientific data and information from the simulation kernel and return them to the modelling framework under a standardized form. Each concrete implementation of the Instrument interface should implement it.

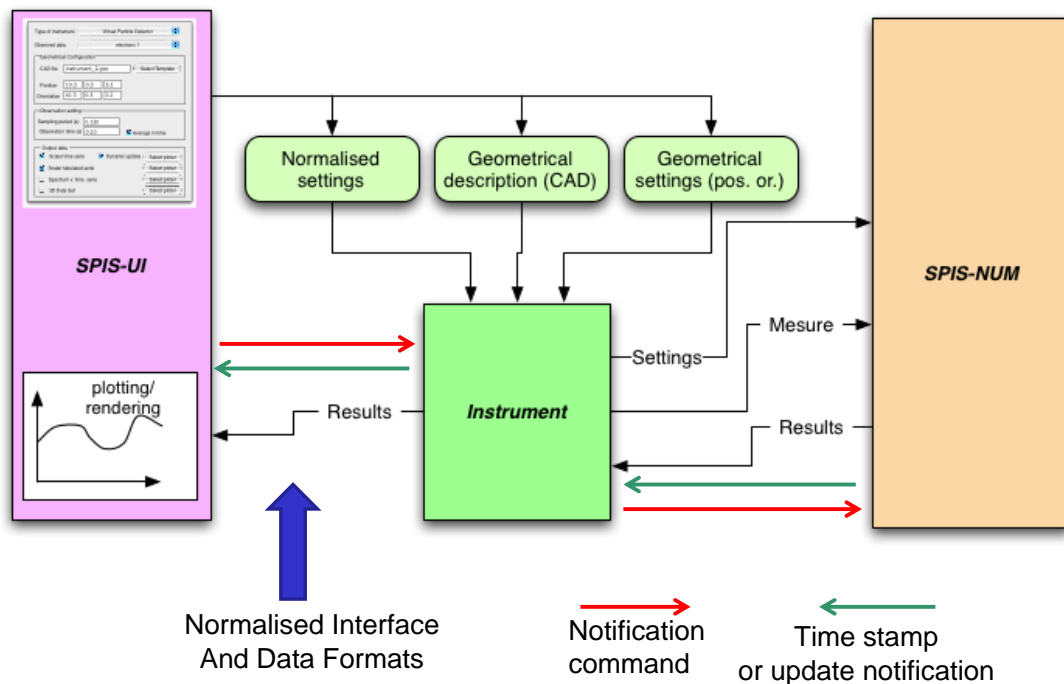
As real scientific instruments and/or detectors, Instruments can be executed in two modes:

- On user demand, from the modelling framework, for a punctual and specific observation;
- On the basis of regular observations using a sample frequency defined at the framework level;

Outputs from Instruments can be:

- DataFields of local data deployed on the grid (e.g. maps of potential);
- Times series, including spectrograms;
- Tabulated data and histograms.

Settings are defined through the pre-processing information like, global parameters, geometrical description and settings. A notification/listener system between the Instrument and the simulation kernel has been implemented. The Instrument access directly to the SPIS–NUM objects and is itself accessible from SPIS–UI. Figure 54 summarizes this approach and provides a simplified collaboration diagram.



**Figure 54: The general SPIS Instrument principle.**





<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

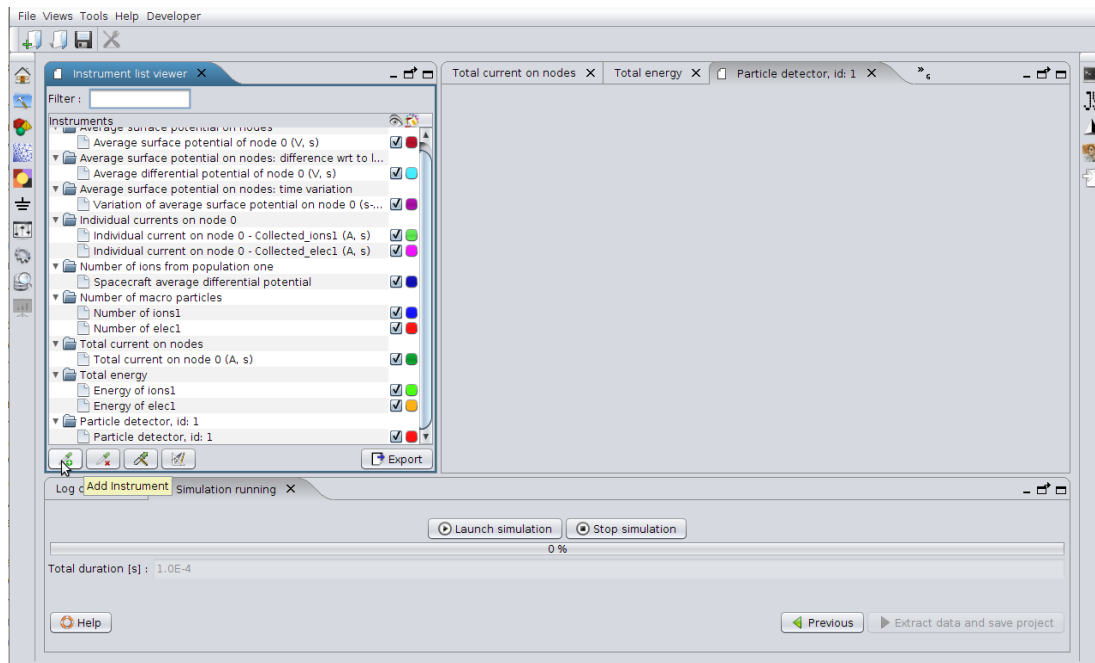
**NB:** *Monitoring Tools* used to control the simulation evolution in interactive mode are also based on the *Instrument* interface. Most of them are automatically instantiated by the SPIS-UI framework and listed as *Instruments* in the *Instrument List Viewer* of the *Simulation Panel*.

## 7.2. Requesting a new instrument


In order to follow the evolutions of the capabilities of the simulation kernel, this is directly SPIS-NUM that provides the available types of instruments and generates, through a central factory, requested new instances.

To request a new instance of instruments, you must be at the *Simulation Panel* step, as illustrated in Figure 55. All instantiated instruments are listed in the *Instruments List Viewer*, on the left side panel. This includes all monitoring tools, automatically instantiated by the SPIS-UI framework to control the simulation. The access to instruments is done with the following buttons at the bottom on this panel:

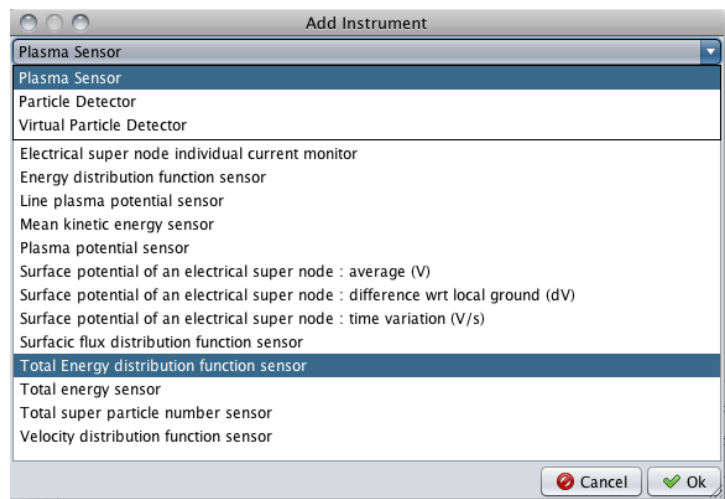
-  - Allow adding a new instrument **(including within the course of the simulation)**;
-  - Remove the selected instrument;
-  - Edit the selected instrument;
-  - Perform a measure on user's demand (for Virtual Particle Detectors only).



**Figure 55: Request and edition of an instrument at the Simulation Panel level.**

By clicking on the  button, a dialog frame will appear displaying the whole set of available types of instruments, as illustrated in Figure 56. For each type of instrument, the physical quantity to be measured and displayed can be selected.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	



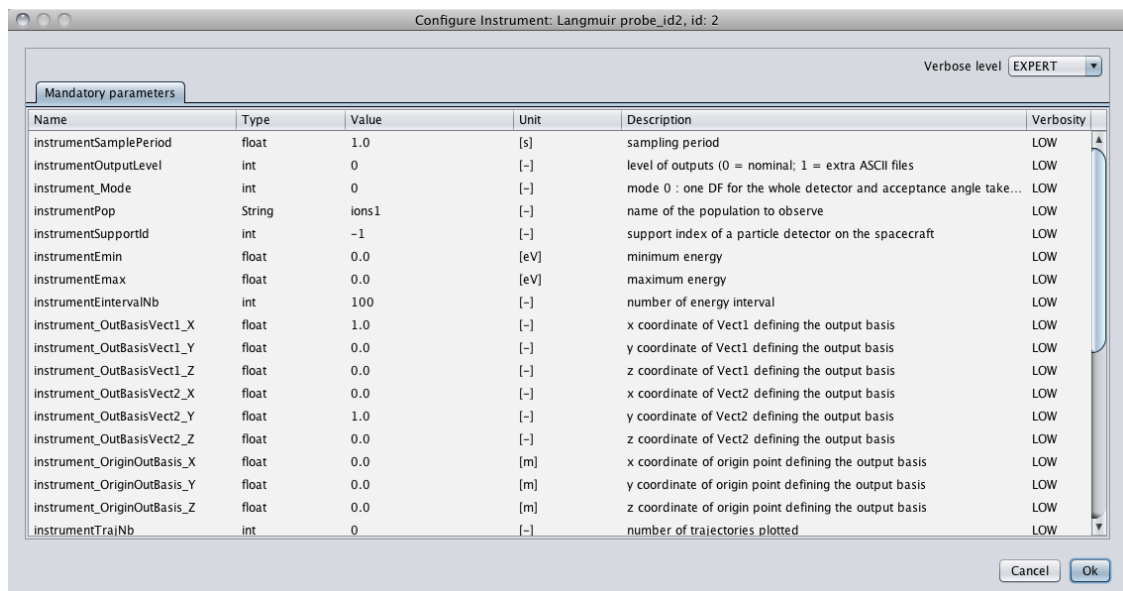
**Figure 56: Selection of an instrument type for a new instantiation..**

Action on the OK button will lead to the Setting Panel of the instrument.

Some instruments, like Virtual Particles Detector, may require, specific setting as virtual mesh of support, previously defined in the Geometry or Mesh editors. Please see Annex on “Advanced uses for scientific applications” for further information.

### 7.3. General settings

Instruments are set through instrument parameters, as illustrated in Figure 57. Most of instruments present two types of parameters: Mandatory Parameters, that must imperatively set to run he instrument and Optional Parameters, that may used for advanced settings. In SPIS 6, all parameters are mandatory however.



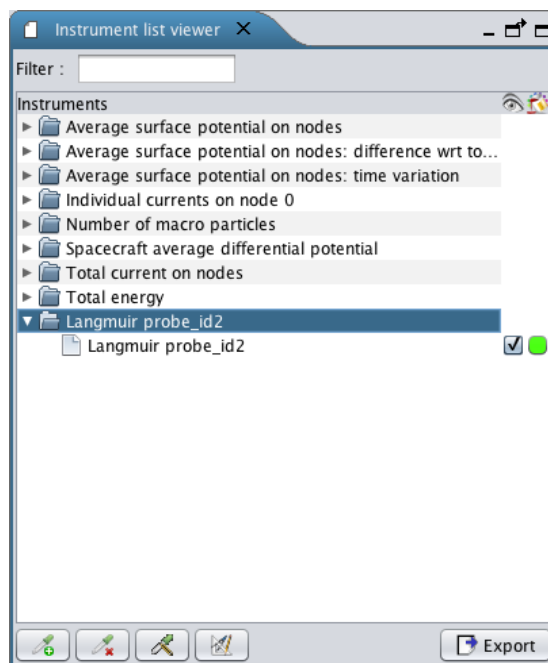
**Figure 57: Setting panel of Instruments. Some instruments present a second index, for optional settings.**

**WARNING:** The relevance of the parameters values is checked by the numerical kernel and a pop-up window appears whenever a value is not acceptable. The associated message indicates the reason



<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

and may propose solutions. For instance, a population name given by the user may not correspond to an existing plasma populations. In this case, the popup window exhibits which populations may be monitored. The user is invited to correct the setting before running or continuing the simulation. In the opposite case, the simulation may crash at next measurement step.


After validation, the newly instantiated instrument should appear in the Instrument List Viewer, as illustrated in Figure 58, with the DataField corresponding to the resulting measurements. Please notice, that some instruments may produce several outputs. The generated DataFields are updated at each measurement.



**Figure 58: Instantiated instrument in the Instrument List Viewer.**

It is possible to edit the settings of an existing instrument just by selecting it on the Instrument List Viewer and clicking on the  button. In the same logic, the selected instrument can be removed by clicking on the  button.

**WARNING:** adding a new instrument or modifying an existing one within the course of the simulation should be done after pausing the simulation using the “pause” button in the simulation wizard. Once the instruments are created/updated, the simulation can be resumed by clicking on “launch simulation”. It may avoid unexpected side-effects such as modifying an instrument while it performs a measurement.


All instruments have, at least, a sampling period to define the period between two measures. In addition, for all selected instrument one can perform an “on-user-demand” measure just by clicking on the  button. In this case, the measurement is performed as soon as the numerical kernel has finished the current time step loop.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

## 7.4. Live Monitoring specific settings

Plasma Sensors used as Live Monitors do not require any specific settings regarding their geometrical descriptions. If relevant of the instantiated live sensors, their position should be set through the InstrumentParam Editor of the instrument. Some of them need a radius (to compute averages of particles inside a sphere for instance). Some others are defined on the spacecraft surface and rather refer to the electrical node of the electrical circuit.

The type of results and their exploitation by the framework differ from one instrument to the other. Generally, all monitors are automatically plotted by UI. They also generates ASCII files that may be post-treated by users.

Next table is a summary of the role and of the inputs parameters of available plasma sensors. It is reminded that plasma sensors should be more generally understood as live monitors of the simulation, including data living in the plasma volume but also on spacecraft surface. Crosses indicate which parameters will have to be defined by users. Some live monitors are automatically built at simulation start. It remains however possible to update their parameters (as e.g. sampling) with the  button.

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

Instrument Type	Quantity	Lives in	Automatically built at start	Position	Sampling Period	Sampling Duration	Pop.	Radius	Emin	Emax	Nb of Eintervals	OutBasis	ESN Id
PotentialPS	Potential	plasma		X	X	X							
DensityPS	population density	plasma		X	X	X	X						
EnergyDistFuncPS	Kinetic energy histogram	plasma		X	X	X	X	X	X	X	X		
TotalEnergyDistFuncPS	Total energy histogram	plasma		X	X	X	X	X	X	X	X		
KineticEnergyPS	Mean Kinetic energy	plasma		X	X	X	X	X					
VelocityDistFunctionPS	3D velocity DF	plasma		X	X	X	X	X	X	X	X	X	
SurfaceFluxDistFunctionPS		plasma	X	X	X	X	X	X	X	X	X	X	
TotalEnergyPS	Total pop energy in domain	plasma	X		X	X	X						
TotalSuperParticlePS	Total nb pop super particles	plasma	X		X	X	X						
AvSCPotentialMonitor	Mean SC potential	S/C	X		X	X							

<b>Reference:</b> ESA-SPIS-GEO-D7-SUM-2013-03-001	<b>Version:</b> 3	<b>Revision:</b> 5
<b>Status:</b> Final	<b>Date:</b> 28/11/2019	

ESNCurrentMonitor	Total current on ESN	S/C	X		X	X							X
ESNIndividualCurrentMonitor	Detailed currents	S/C	X		X	X							X
ESNPotentialMonitor	Potential of ESN	S/C	X		X	X							X
ESNPotentialDifferenceMonitor	Diff pot vs local ground	S/C	X		X	X							X
ESNPotentialVariationMonitor	dV/dt of ESN	S/C	X		X	X							X



The parameters referring to data in previous tables are:

- Position: instrumentCenterX-Y-Z = the coordinates in the Gmsh basis
- Sampling Period: instrumentSamplePeriod which = the time step between two measurements
- SamplingDuration: instrumentObservationDuration = the duration used to average the data between two sampling period
- Pop: instrumentPop = name of the population to observe
- Radius: instrumentRadius = radius of a spherical sensor (it is the responsibility of the user to be sure the sphere is inside the computational volume)
- Emin: instrumentEmin = minimum energy used to plot particle characteristics
- Emax: instrumentEmax = maximum energy used to plot particle characteristics
- Nb of Eintervals: instrumentEintervalNb = number of energy intervals
- OutBasis: instrument\_OutBasisVect1\_X-Y-Z and instrument\_OutBasisVect2\_X-Y-Z = coordinates of first two vectors defining the basis in which velocity distribution function sare plotted.
- ESNId: instrumentESN = electrical node index of the spacecraft circuit

**NB:** other parameters may be used. They are all concisely described in the comment box of the InstrumentParam wizard.

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

## 8. Advanced uses

### 8.1. Batch mode

In order to run large simulation on remote computer, SPIS can be launched in batch or head-less mode. The batch mode can be used in two different ways:

- **Python based scripts driven:** The whole SPIS-UI framework can be run through Python (Jython) based scripts, also called SPIS-tracks. To do it, just execute the following command:

```
./Spis.sh -b mySpisTrask.py
```

Such approach can typically used for parametric studies.

- **Project execution:** For a fully set project, i.e. where the whole pre-processing is done, the simulation phase can be directly be performed without GUI with the following command:

```
./Spis.sh -p $PATH_TO_MY_PROJECT/myProject.spis5
```

Such approach is useful to run a long or large simulation on a remote computer.

Currently, the batch mode can be used under UNIX type systems only.

### 8.2. Thin elements modelling

#### 8.2.1. Concept of thin elements

Thin elements are special geometries and numerical models introduced to simulated very thin or sharp geometrical objects, likes wires, antennas, solar arrays, without the necessity of local prohibitive mesh refinement. The specific models in SPIS-NUM also address the presences of singularities like at the edges of a thin surface with different potentials on both faces. Two types of thin elements are defined:

- **1D thin elements or thin wires:** to model mono-dimensional objects, like wire or antennas, characterised by a length and a radius, this last one being typically small with respect to the local mesh refinement;
- **2D thin elements or thin surfaces:** to model bi-dimensional objects, like thin solar arrays, characterised by a thickness typically small with respect to the local mesh refinement.

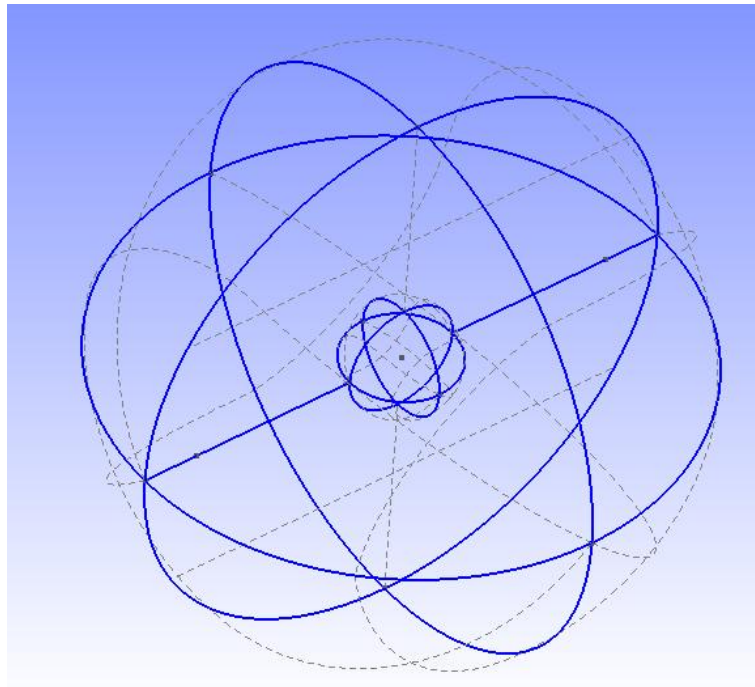
The use of thin elements remains a more complex operation than the modelling of a standard geometry. It may require a direct modification of the mesh structure (i.e. mesh splitting) and attributions of specific properties on the related groups. For these reasons, they should be considered as an advanced use and the modelling process should be considered carefully in a try-error approach.

#### 8.2.2. Thin wires (1D)

Thin wires are 1D objects like antennas and wire. In this model, nodes of the volume mesh are aligned on the wire (i.e. the 1D object) and special models are applied to the equivalent spacecraft "surface".

A specific CAD approach should be followed to force the mesher to align mesh nodes on the wire. The wire should constitute a border (1D) between two surfaces. Themselves should constitute a border (2D) between two sub-volumes. As usual, physical of surfaces should be on inner (i.e. spacecraft) and external boundaries. Additional physicals of line should be set on the thin wires. In order to have a unique computational volume for SPIS-NUM, the sub-volumes should be gathered in a unique physical of volume. Figure 1 here after show an example of such system, with a spacecraft (inner sphere) and two thin wires. This file is given in example in the \$SPIS\_ROOT\_DIR/Data directory.

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	



**Figure 59: Example of sytem with *Thin Wires* (1D).**

The corresponding CAD file is given in the table here below.

```

// physical groups

// groups of thin wires
Physical Line(1052) = {1040};
Physical Line(1053) = {1039};

// external boundary
Physical Surface(1054) = {1034,1028,1026,1032,1036,1022,1030,1024};

// S/C hub
Physical Surface(1055) = {128,134,126,132,124,136,122,130};

// computationnal volume
Physical Volume(1060) = {1057, 1059};a
  
```

**Tableau 6: Example of Gmsh's physicals definitions for thin wires**

Mesh model of type « Spacecraft Thin Element 1D » should be applied on groups corresponding to the thin wires, as illustrated in Figure 60. Figure 61 shows the corresponding final plasma potential computed at the issue of the simulation.

**Remark:** Please notice that the relevance of the present approach and the selected numerical model may depend on the local mesh refinement and the selected meshing algorithm. Please see the Gmsh documentation for further information.

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

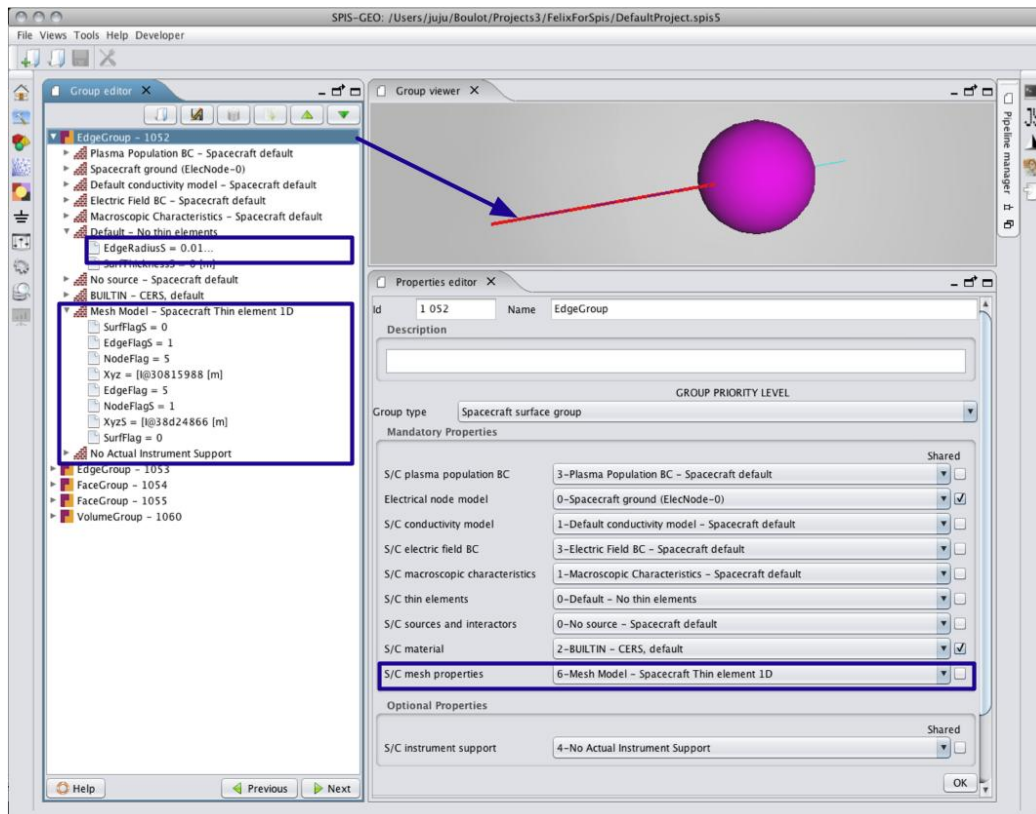


Figure 60: Example of properties attribution for groups corresponding to thin wires.

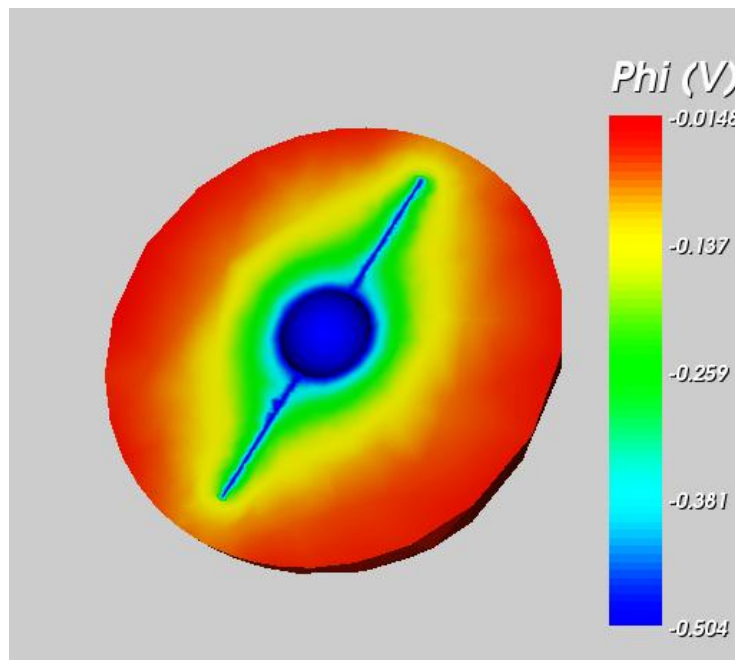


Figure 61: Result of the computation with a thin wire, here the final plasma potential.

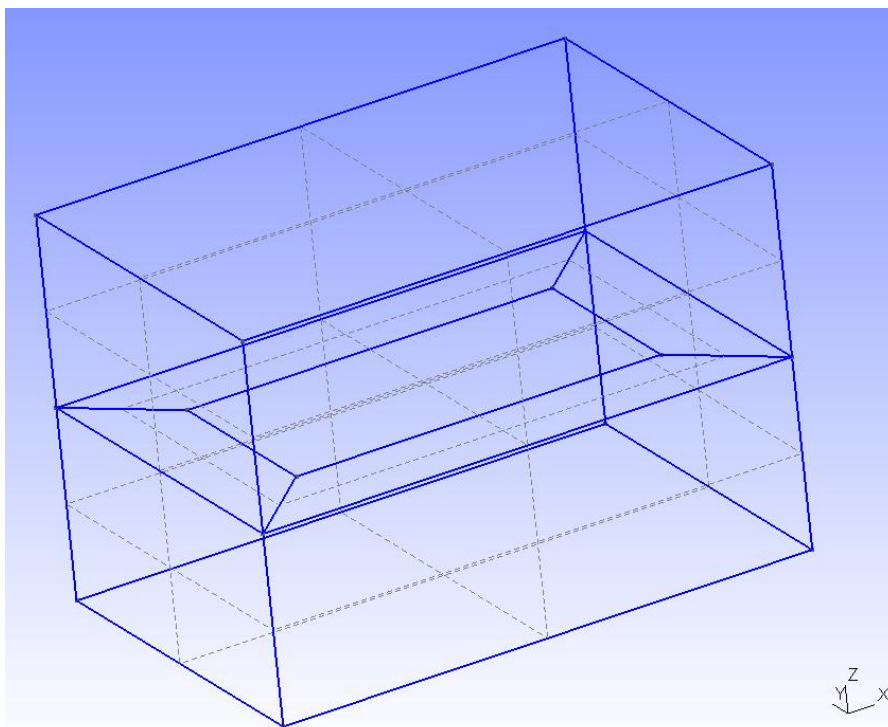
Reference: ESA-SPISGEO-D7-SUM-2012-08-002	Version: 2	Revision: 2
Status: Validated	Date: 28/03/2013	

### 8.2.3. Thin surfaces (2D)

2D thin elements or thin surfaces are bi-dimensional objects where the thickness is typically small with the respect to the local mesh refinement. Such model can typically used to model thin solar arrays, parabolic antenna, etc...

**CAD modelling:** As for thin wires, the nodes of the computational volume should be aligned on the surface. CAD modelling is the same than in the case of the thin wires. Thin surfaces should constitute a common border between two sub-volumes of the computational volume. The CAD template shown in Figure 62 illustrates this configuration.

Physicals of surface should be defined on the thin surfaces as for standard spacecraft surfaces. In a same manner than for thin wires, only one computational domain should be defined for SPIS-NUM by the definition of a unique physical of volume gathering all sub-volumes.



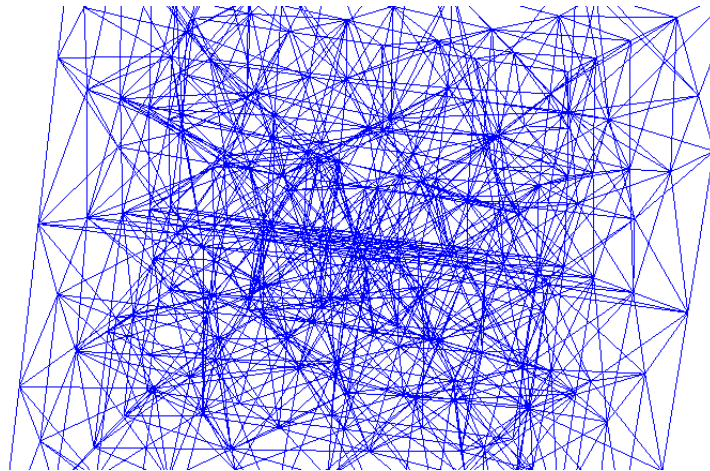
**Figure 62: Example of thin plate (2D) model**

**Mesh splitting and edition:** To be correctly processed by the Poisson's solver of SPIS-NUM and have the possibility to set different characteristics (e.g. electrical potential) on both, the mesh should be "split". In this specific process, mesh elements on the thin surface (i.e. nodes, edges, faces) are duplicated and the volume mesh "cracked". A hole of null thickness is then created.

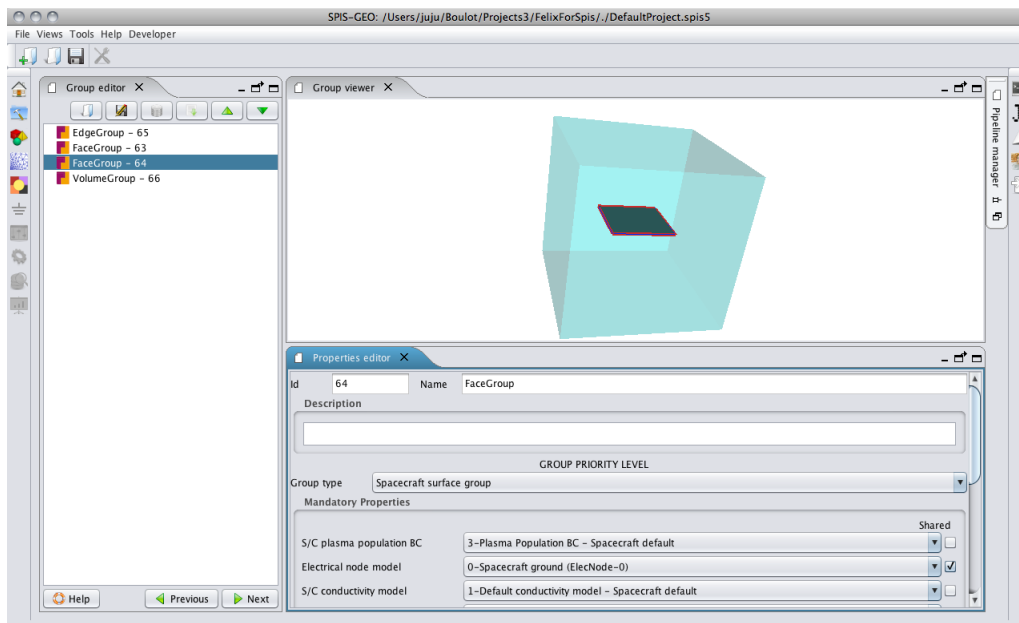
This operation is done in the Groups Editor.

First, load and mesh the CAD model as usual. Figure 63 shows an example of volume mesh with a thin surface in it. In the Groups Editor select the group corresponding to the thin surface as illustrated in Figure 64. Set the group type to "Spacecraft surface group", open the "Show advanced settings" sub-panel at the bottom of the Group Editor panel and perform the splitting operation, as illustrated on Figure 65.

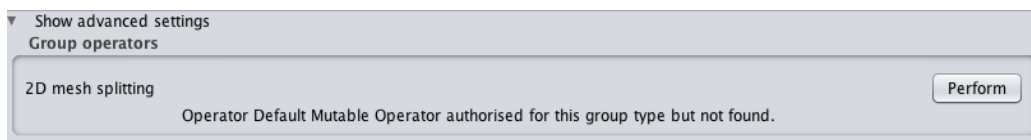
<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	



**Figure 63: Example of meshed thin surface. One can see that nodes are aligned on the surface.**



**Figure 64: Thin surface in the Groups Editor before the mesh splitting.**

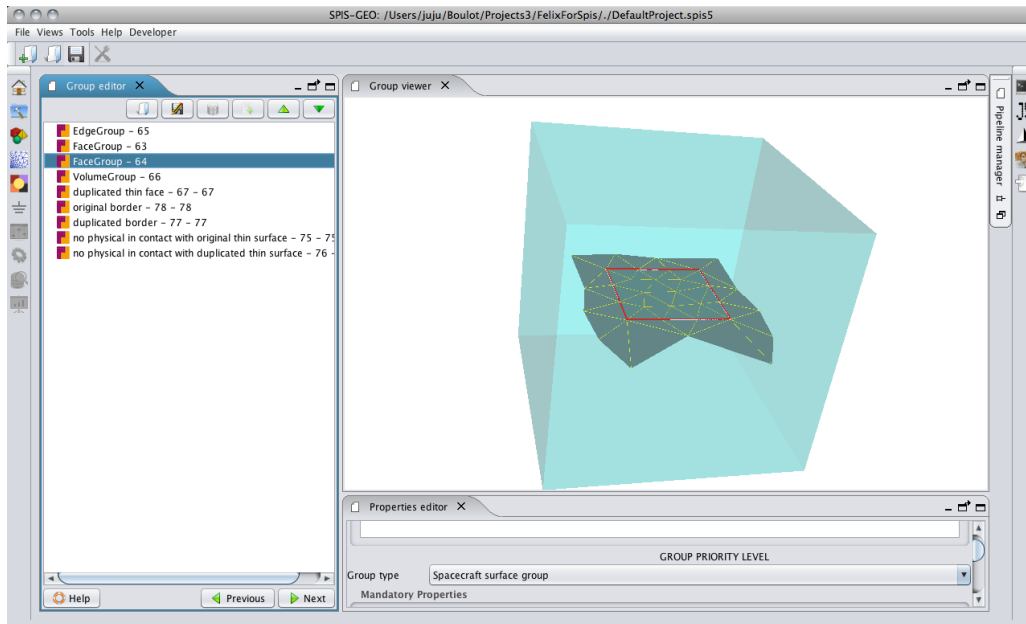


**Figure 65: Mesh splitting panel.**

After splitting, the mesh should present an aspect similar to the view in Figure 66, where the thin surface is duplicated and elements connected to the edge of the surface are cracked. Additional groups are created for the duplicated surface, the edge of the original surface, the edge of the duplicated surface and the crack or teared surfaces.

All of these newly generated groups should be considered as “Spacraft surface group” and set as given Tableau 7.

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	



**Figure 66: Surface in the Groups Editor after mesh splitting.**

Priority	Group	Mesh model
1	Original border	Spacecraft Thin Element 2D – Original border
2	Duplicated border	Spacecraft Thin Element 2D – Duplicated border
3	Original thin surface	Spacecraft Thin Element 2D – Original surface
4	Duplicated thin surface	Spacecraft Thin Element 2D – Duplicated surface
5	No physical in contact with original surface	Spacecraft Thin Element 2D - Teared surface
6	No physical in contact with duplicated surface	Spacecraft Thin Element 2D - Teared surface

**Tableau 7: Mesh model properties attribution on 2D thin surfaces after mesh splitting.**

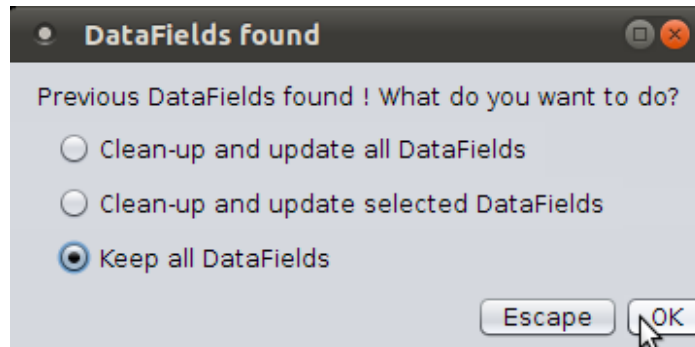
The surface to be split must imperatively be monolithic (i.e. in one unique piece) and convex (i.e. without hole). If several solar arrays should be modelled, please consider each part as an independent thin surface and the splitting operating should be repeated for each individual part.

**Important remark:** Groups of border should have the highest priority. Surface groups should have a higher priority than the volume group.

To finalise the process, apply the other needed properties and apply them as usual.

When the “next button” is clicked, and after the pop up asking for field deployment confirmation is passed, a new pop up appears. It indicates that there are some data fields that already exist in the SPIS model. These data fields define the parity between the original thin surface and the duplicated thin surface. The user has to keep them in the model. The radio button “Keep all DataFields” must be selected. This is illustrated on the Figure 67.

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

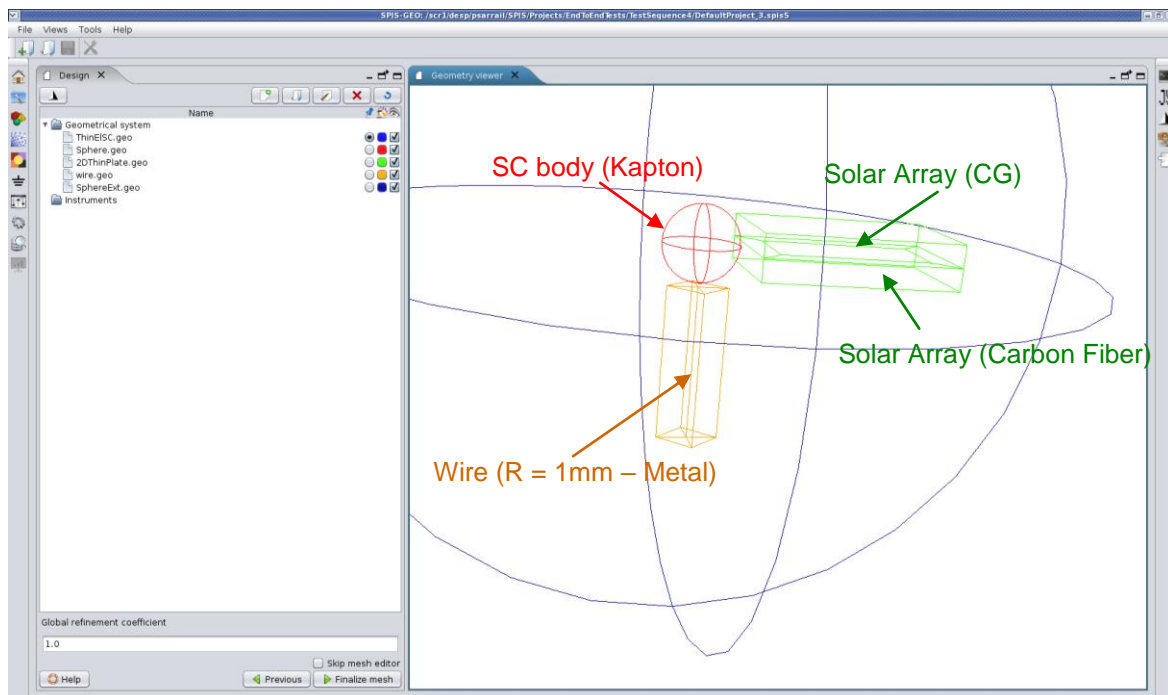


**Figure 67: Keep parity DataFields.**

**Important remark:** As for thin wire, the local mesh quality should impact the final result and the relevance of the final computation.

**Recommendations on when to use thin panels (as solar panels) :**

- if the mesh cells on top of panels are large (two or three times) as compared to the solar thickness: use thin 2D panel
- otherwise use the legacy parallelepiped box method.



**Figure 68 -Illustration of Cad geometry using thin solar panels and thin wires**

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	




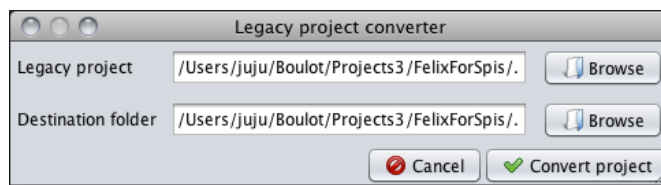
### 8.3. How-to-import of SPIS-Legacy projects

It is possible to convert SPIS-Legacy (i.e. version lower than 5.0.) projects into .spis5, however some conditions and a specific protocol should be respected:

- Currently, only systems where the mesh, the global parameters and the whole preprocessing DataFields are fully defined and deployed. This corresponds to the state at the “UI-to-Num” phase in SPIS-Legacy. Currently, CAD groups/properties settings cannot be imported.
- The global parameters should be converted from their historical format (Python serialisation) into the XML format of .spis5 by using the following command:

```
globalParametersConverter.sh pathToLegacySpisProject.spis \\
                             pathToLegacySpisProject.spis/outputGlobalParam.xml
```

- The output global parameters file should be saved at the root directory of the legacy project.
- To convert the whole project, you should click on the  icon on the right-side tool bar. This should open the conversion panel displayed in Figure 69 and convert the project.
- Load the converted project into the framework.



**Figure 69: SPIS-Legacy project conversion panel.**

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

## 9. Optimization and advanced settings

### 9.1. 3D visualization troubleshooting and degraded mode

The core modules of SPIS are written in Java and are operational without any modification and/or re-compilation on most of the current platforms (i.e. OS/hardware architecture). However some components, like the meshing (based on Gmsh) and 3D visualization tools (based on VTK) are in native languages and depend on the targeted platform. Most of the potential difficulties of deployment and/or troubleshooting are due to these native components and their relationship with installed native systems libraries.

More especially, 3D rendering modules use low level 3D accelerated functions and their behaviour and stability may depends on several factors:

- Version of the operating system;
- Type of architecture (32/64bits);
- Types of video card, knowing that a 3D acceleration is required to run these modules;
- Version of some low level systems libraries, like the glibc library under Linux based systems;
- Version of the 3D rendering system libraries and more especially OpenGL or MesaGL.

In case of crash, lost of control of the application, impossibility to visualize 3D views, we strongly recommend to check these elements with respect to the compliance tables available on the SPINE Web site [RD1].

If it is still impossible to run SPIS with 3D rendering modules, you can run it in degraded mode, without 3D views by using the following launching command:

**SpisGeoNo3D.sh**

### 9.2. How to increase the memory stack?

3D PIC models and mesh analysis may be very intensive in memory and CPU resources. A proper setting of your JVM might be critical to run your realistic models.

The recent JVMs (1.6 and higher) have removed memory limitation on most of the platforms. Several runs have been successfully performed with more than 20Gb of RAM (i.e. about  $10^6$  tetrahedrons mesh in full PIC mode with about 15 particles by cells) on UNIX/Linux 64 bits systems (Java 1.6 Sun). The maximum addressable memory size can be set by editing the launching script in the SPIS\_ROOT directory.

Just adjust the value of the memory Java stack size, i.e. the variable -Xmx, as follow in the section:

```
java -Xmx1024M -splash:resources/images/splash_spis-geo-png24.png \\  
-jar -Dfelix.config.properties=file:resources/felix/config.properties \\  
-Dactivate3DViews=true dependencies/thirdparty/felix-4.0.2/felix.jar \\  
> debug.txt
```

By default, it is set to 1Gb in order that most of users can at least start SPIS on small computers. Set it to 4Gb (i.e. -Xmx=4000M) or more if you computer has more RAM. This should still be lower than the real total RAM. On a standard laptop (Mac OSX with 4Gb), it is normally possible to run easily systems with up to  $9.0E5$  tetrahedrons.

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

**IMPORTANT REMARK:** Above comments are valuable only for UNIX based system. Under most of standard desktop Windows system (Windows XP 32-bits), the total amount of memory dedicated to one job is limited to less than 2Gb. In this case, it is impossible to overtake this limitation due to the OS capabilities.

This is why, we strongly recommend for large modelled systems to perform your simulations on UNIX-based systems or Windows server OS.

Remark: The total memory cost may also depend on the used JVM and its settings. Please see the JVM documentation for further information.

### 9.3. My computer is multi-core, does SPIS use all them?

Yes! Most of the bundles of the SPIS-UI framework are multi-threaded for the critical procedures and make benefits of moderns multi-cores / multi-CPU architectures. Some critical methods of the SPIS-UM simulation kernel, like the particle pusher, are also multi-threaded.

However, take care that on some operating systems, like Microsoft Windows, too many threads running in parallel can lead to a global performances decrease.

Regarding the SPIS-UI framework, the default thread-pool size is four.

### 9.4. Property file and configuration of native components

Most of the platform and packaging configuration settings are defined in the relevant property file located at \$SPIS\_ROOT/resources, like for example \$SPIS\_ROOT/resources/org-spis-osx64b.properties. The relevant file may depend on the targeted platform and the packaged flavour of SPIS. The simplest way to identify the relevant file, please edit the launch script at the \$SPIS\_ROOT and look at the value set for the -Dorg.keridwen.config JVM flag.

Please notice that on some specific platform (OS/hardware couple), some native components, like Gmsh, may have to be adjusted in this file.

Here below an example of commented property file.

```
# Define the title name (i.e at the top of the main window) of the application
application.name=SPIS-GEO

# Define the default size of the main window of the application
application.height=700
application.width=1100

# Set the project validation file (Do not edit).
project.format=org-spis-geo-5.1.0.xsd

# Set the path to the relevant Gmsh runnable application. Please,
# notice that this setting depends on the targeted platform
org.geuz.gmsh.launch.path=./dependencies/thirdparty/gmsh-2.7.0-osx64b/\
    Gmsh.app/Contents/MacOS/gmsh

# set the documentation entry point for the called browser (see help menu)
org.spis.ui.geo.wizard.help.file=./documentation/index.html

# set various properties related to the external tools
```

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

```
org.spis.ui.tools.properties=./resources/externalTools-osx64b.properties

# set the default spis-track to run SPIS in batch mode.
org.spis.ui.tools.track=./resources/defaults/tracks/spisTrack.py

# Set the directory of default CAD templates
org.spis.ui.geometry.editor.templates=./resources/defaults/geometryTemplates

# Set the directory of the default global parameters files
org.spis.ui.global.parameters.directory=./resources/defaults/globalParameters

# Set the directory of the defaults SPIS-NUM transition scenarios
org.spis.ui.global.parameters.transitions=./resources/defaults/transitions

# Set the level of verbosity of the main logger
org.spis.ui.global.parameters.verbose=LOW

# Set the default SPIS-UI DataFields to SPIS-NUM LocalParam and ParamSet mapping
# table. Please notice that this table is overridden by the table stored in saved
# spis-projects. This table may also differs in function of the SPIS-NUM kernel
# evolutions.
org.spis.ui.local.parameters.default.mapping.table=./resources/defaults/\
    defaultDataFieldMappingTable.xml

# Set the directory of the default properties catalogues
org.spis.ui.group.editor.properties.folder=./resources/defaults/properties

# Set the default state machine configuration file. This typically defines the
# wizard order.
org.spis.ui.model.state.machine.properties=./resources/state/\
    org-spis-ui-state-machine-config-geo.xml

# Set the path to the Jyconsole properties file
com.artenum.jyconsole.properties=./resources/jyconsole.properties
```

## 10. Known bugs

---

### Known bug #1:

On some Windows systems, the generation of the unique name for the cache directory, based on a unique time stamp, may fail. In that case, the name of the cache directory must be defined manually by editing the launching script as follows

- Open the Spis.bat file with the text editor of your choice
- The first line of the script file corresponds to the definition of the unique time stamp:

```
set NOW=%date:~-4,4%%date:~-7,2%%date:~-10,2%%time:~-11,2%%time:~-8,2%%time:~-5,2%%time:~-2,2%
```

On some Windows systems, this line generates spaces in the variable name that prevent the execution script from working properly. In this case, replace this line by the following:

```
set NOW=1
```

!!! Beware that if you run several SPIS instances simultaneously, the value of the NOW variable should differ for each execution

### Known bug #2:

Some OS do not support files whose name contains a long chain of characters. To avoid that, it is recommended to create the SPIS projects in a short arborescence tree, as for instance "C:/SpisProjects".

### Known bug #3:

SPIS uses Gmsh as CAD engine and mesher. Gmsh is a native component that depends on the targeted platform (OS and hardware architecture). On some OS, Gmsh may also depend on related external libraries (e.g. glibc, Fltk...). For these reasons, the CAD import/loading may fail if Gmsh does not run properly. In this case, please install manually a version of Gmsh appropriate to target platform (see <http://geuz.org/gmsh/>) and set the corresponding path as described in Software User Manual (section 9.4-Property file and configuration of native components) to fix such troubleshooting. As a rule of thumb, gmsh version 2.4.2 was adapted to most tested platforms.

### Known bug #4:

There have been some problems launching SPIS 6 on some Windows: the launch time of SPIS can take up to 10 minutes. This was due to McAfee antivirus that prevents SPIS from starting properly. It can be verified by launching Windows "Task Manager" and looking in the "Processes" tab while starting SPIS. Upon SPIS startup, the task associated with McAfee software (called "McShield") takes 100% of processor usage. Stopping SPIS stops this "McShield" task.

Disabling McAfee solves the problem and SPIS startup is then instantaneous. To disable McAfee, right-click on McAfee logo at the bottom right of the task bar and select "disable scan". This is however a temporary fix and this is not recommended on the long term as the computer will be vulnerable to viruses. A more sustainable solution will have to be fixed in the future to allow SPIS running properly with any antivirus.

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

**Known bug #5:**

Some OS do not support file or directory name that include spaces. It is recommended to remove any of them from your project arborescence tree, project name, mesh file name, geometrical file name, etc...

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

## Annex A DataField/MeshMask structure and file formats

SPIS-UI, and more generally the Keridwen IME, handles, proceeds and converts a large set of data (called here fields) deployed on one or several unstructured meshes. These fields are, of course, the outputs recovered from the simulation, but also most of the fields given in inputs to the simulation kernel to set the initial and boundary conditions. These operations are fully transparent for the user. The purpose of the present section is to explain the key concepts handled by SPIS-UI to perform all data conversion and processing needed along the modelling chain.

This work is delegated to the Artenum Penelope library. The objective of Penelope is to provide a highly portable and optimized Java based library to store and handle 2D/3D unstructured meshes and data deployed on these meshes. Penelope is an independent Artenum technology available on demand on the Penelope Web site [RD10].

The effort has been focused on the optimization of the memory footprint keeping the maximum versatility of the mesh structure. With respect to existing mesh libraries, Penelope provides an access to all intermediate ascending and descending connectivities. In order to reduce the memory cost, a part of the connectivity is not stored but either implicitly defined for each type of element or dynamically re-built by the mesh.

A very large set of data types can be deployed on the meshes, like scalars (integer, float, double), vectors or more generally objects. Data can be freely localized on cells nodes, edges, faces or centre.

Penelope provides a large set of mesh operations, like sub-meshes extraction, and data and file formats conversion (Gmsh, Vtk, NetCDF, XML...). See the Penelope documentation for further information.

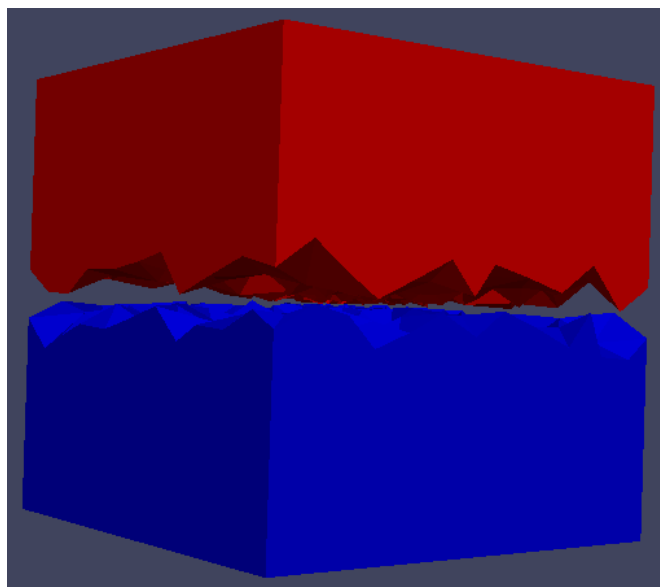


Figure 70: Example of sub-domain extraction done Penelope.

### A.1. Data structure and logic

In order to reduce the memory footprint, several techniques are used. First, the nodes-edges-faces-cells connectivity is computed dynamically in most of the cases. Second, the structure is divided in three main layers:

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

- **The Mesh object**, that stores the geometry (nodes coordinates) and the topology (cells types and connectivities);
- **The MeshMask objects**, which defines a “mask on the mesh” by selecting a sub-set of the mesh elements. Several MeshMasks can be defined on a given unique Mesh. MeshMasks can be defined on nodes, edges, face, cells type elements.
- **The DataFields objects**, which store the data to be deployed on the Mesh. Each DataField refers to a MeshMask (and a MeshMask can refer to several DataField). There is a bijective relationship between values stored in the DataFields and the mesh elements list stored in the corresponding MeshMask.

Such an approach, illustrated in Figure 71, allows a high degree of memory and access times optimization.

Additional classes, like the DataFieldsManager, helps developers and users to manipulates a large set of DataFields, as it is done in the SPIS-UI central data model.

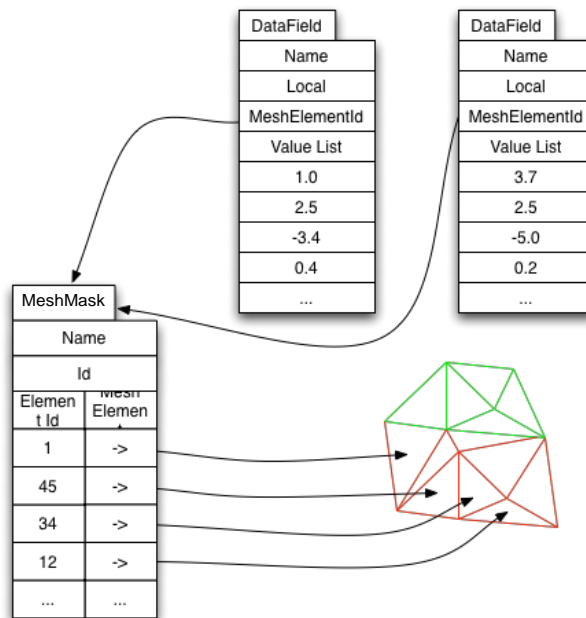


Figure 71: Logical principle of the Penelope mesh library.

## A.2. Penelope NetCDF-based file format

Penelope can save data on the disk into a NetCDF based format. By default, this format is used for the persistence scheme of SPIS.

The description header and some example values are given here, if the user wishes to develop his own importer using the CDF implementation of his choice. The file can be saved in ASCII or binary format. Two files will be produced, one for the DataField, one for the MeshMask.

**Remark:** Please do not forget that the mesh is saved separately under the Gmsh 2.xx format.

Readers and writers implemented in Java are provided with Penelope.

```
netcdf BackgroundDens {
    dimensions:
```

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	



```

nbData = 6532 ;
dataDimension = 1 ;
variables:
  double dataArray(nbData, dataDimension) ;
  dataArray:dataType = "DOUBLE" ;
  dataArray:nbData = 6532 ;
  dataArray:dataDimension = 0 ;
// global attributes:
:id = 47 ;
:isRelative = "true" ;
:name = "BackGroundDens" ;
:description = "" ;
:nbMeshElements = 6532 ;
:meshType = "POLYHEDRON" ;
:unit = "[m-3]" ;
:meshMaskId = 47 ;
:meshMaskNormalizedURI = "cad_sav.msh-PolyhedronMask-47.nc" ;
:meshMaskURI = "cad_sav.msh-PolyhedronMask-47.nc" ;
:vtkVisualizationPreference = "POLYHEDRON" ;
}

```

### A.3. Penelope XML-based file format

Penelope can save data on the disk into an XML based file format. Two files will be produced, one for the DataField, one for the MeshMask. The meta-information header can be read/write independently on the data themselves, in order to offer an overview capacity.

**Remark:** Please do not forget that the mesh is saved separately under the Gmsh 2.xx format.

Readers and writers implemented in Java are provided with Penelope.

**Example of DataField saved with the XML Penelope format.**

```

<?xml version="1.0" encoding="UTF-8"?>
<penelope.datafield version="3.0">
  <metadata>
    <id>24</id>
    <name>NodeFlag</name>
    <description></description>
    <localisation>VERTEX</localisation>
    <type>Int</type>
    <nb.data>3611</nb.data>
    <dimension>0</dimension>
    <unit></unit>
    <vtkFormatPreferred>VERTEX</vtkFormatPreferred>
  </metadata>
  <meshmask>
    <meshmask.type Value="VERTEX"/>
    <meshmask.normalised.uri>

```

```

    /home/user1/NodeFlag/meshMask.xml
  </meshmask.normalised.uri>
  <meshmask.uri>
    /home/user1/NodeFlag/meshMask.xml
  </meshmask.uri>
</meshmask>
<data>
  1,
  8,
  0,
  (...)
  8, "
</data>
</penelope.datafield>

```

### Example of corresponding MeshMask.

```

<?xml version="1.0" encoding="UTF-8"?>
<penelope.meshmask version="3.0">
  <metadata>
    <id>19</id>
    <name></name>
    <description></description>
    <localisation>VERTEX</localisation>
    <type>VERTEX</type>
    <nb.mesh.element>3611</nb.mesh.element>
  </metadata>
  <mesh>
    <mesh.format Value="gmsH 2.2" />
    <normalised.uri>
      /home/user1/Undefined
    </normalised.uri>
    <uri>file: /home/user1/Undefined</uri>
  </mesh>
  <mesh.elm.id>      55,
    286,
    1838,
    2282,
  (...)
    172, "
</mesh.elm.id>
</penelope.meshmask>

```

## Annex B Example of properties catalogue file

The following file gives an example of material numerical property catalogue (.xcat). Please notice, that the list of fields listed here may depends on the version of SPIS.

```

<Property>
  <id>0</id>
  <name>ITO, default</name>
  <description>Material coated with ITO, other parameters set to default: sunlit, interactions on, 125 microns thick, 300K, basic material model (based on NASCAP properties)</description>
  <characteristicList>
    <entry>
      <string>MatModelId</string>
      <IntegerCharacteristic>
        <id>1</id>
        <name>MatModelId</name>
        <parentProperty class="Property" reference="../../../../"/>
        <unit class="com.artenum.penelope.mesh.unit.SimpleStringUnit">
          <unit></unit>
          <standardUnit></standardUnit>
        </unit>
        <localisation>2</localisation>
        <informationLink></informationLink>
        <value>0</value>
      </IntegerCharacteristic>
    </entry>
    <entry>
      <string>ProtSecEmis</string>
      <IntegerCharacteristic>
        <id>6</id>
        <name>ProtSecEmis</name>
        <parentProperty class="Property" reference="../../../../"/>
        <unit class="com.artenum.penelope.mesh.unit.SimpleStringUnit">
          <unit></unit>
          <standardUnit></standardUnit>
        </unit>
        <localisation>2</localisation>
        <informationLink></informationLink>
        <value>1</value>
      </IntegerCharacteristic>
    </entry>
    <entry>
      <string>VolConduct</string>
      <IntegerCharacteristic>
        <id>7</id>
        <name>VolConduct</name>
        <parentProperty class="Property" reference="../../../../"/>
  
```

**Reference:** ESA-SPISGEO-D7-SUM-2012-08-002

**Version:** 2

**Revision:** 2

**Status:** Validated

**Date:** 28/03/2013

```

<unit class="com.artenum.penelope.mesh.unit.SimpleStringUnit">
  <unit></unit>
  <standardUnit></standardUnit>
</unit>
<localisation>2</localisation>
<informationLink></informationLink>
<value>1</value>
</IntegerCharacteristic>
</entry>
<entry>
  <string>ElecSecEmis</string>
  <IntegerCharacteristic>
    <id>5</id>
    <name>ElecSecEmis</name>
    <parentProperty class="Property" reference="../../../../.."/>
    <unit class="com.artenum.penelope.mesh.unit.SimpleStringUnit">
      <unit></unit>
      <standardUnit></standardUnit>
    </unit>
    <localisation>2</localisation>
    <informationLink></informationLink>
    <value>1</value>
  </IntegerCharacteristic>
</entry>
<entry>
  <string>Temperature</string>
  <DoubleCharacteristic>
    <id>10</id>
    <name>Temperature</name>
    <parentProperty class="Property" reference="../../../../.."/>
    <unit class="com.artenum.penelope.mesh.unit.SimpleStringUnit">
      <unit>[K]</unit>
      <standardUnit>K</standardUnit>
    </unit>
    <localisation>2</localisation>
    <informationLink></informationLink>
    <value>300.0</value>
  </DoubleCharacteristic>
</entry>
<entry>
  <string>IndConduct</string>
  <IntegerCharacteristic>
    <id>8</id>
    <name>IndConduct</name>
    <parentProperty class="Property" reference="../../../../.."/>
    <unit class="com.artenum.penelope.mesh.unit.SimpleStringUnit">

```

**Reference:** ESA-SPISGEO-D7-SUM-2012-08-002

**Version:** 2

**Revision:** 2

**Status:** Validated

**Date:** 28/03/2013

```

    <unit></unit>
  <standardUnit></standardUnit>
</unit>
<localisation>2</localisation>
<informationLink></informationLink>
<value>1</value>
</IntegerCharacteristic>
</entry>
<entry>
  <string>MatThickness</string>
  <DoubleCharacteristic>
    <id>3</id>
    <name>MatThickness</name>
    <parentProperty class="Property" reference="../../../../"/>
    <unit class="com.artenum.penelope.mesh.unit.SimpleStringUnit">
      <unit>[m]</unit>
      <standardUnit>m</standardUnit>
    </unit>
    <localisation>2</localisation>
    <informationLink></informationLink>
    <value>1.25E-4</value>
  </DoubleCharacteristic>
</entry>
<entry>
  <string>PhotoEmis</string>
  <IntegerCharacteristic>
    <id>4</id>
    <name>PhotoEmis</name>
    <parentProperty class="Property" reference="../../../../"/>
    <unit class="com.artenum.penelope.mesh.unit.SimpleStringUnit">
      <unit></unit>
      <standardUnit></standardUnit>
    </unit>
    <localisation>2</localisation>
    <informationLink></informationLink>
    <value>1</value>
  </IntegerCharacteristic>
</entry>
<entry>
  <string>MatTypeId</string>
  <IntegerCharacteristic>
    <id>2</id>
    <name>MatTypeId</name>
    <parentProperty class="Property" reference="../../../../"/>
    <unit class="com.artenum.penelope.mesh.unit.SimpleStringUnit">
      <unit></unit>

```

**Reference:** ESA-SPISGEO-D7-SUM-2012-08-002

**Version:** 2

**Revision:** 2

**Status:** Validated

**Date:** 28/03/2013

```

    <standardUnit></standardUnit>
  </unit>
  <localisation>2</localisation>
  <informationLink></informationLink>
  <value>0</value>
</IntegerCharacteristic>
</entry>
<entry>
  <string>SunFlux</string>
  <DoubleCharacteristic>
    <id>11</id>
    <name>SunFlux</name>
    <parentProperty class="Property" reference="../../../../"/>
    <unit class="com.artenum.penelope.mesh.unit.SimpleStringUnit">
      <unit>[-]</unit>
      <standardUnit>-</standardUnit>
    </unit>
    <localisation>2</localisation>
    <informationLink></informationLink>
    <value>300.0</value>
  </DoubleCharacteristic>
</entry>
<entry>
  <string>SurfConduct</string>
  <IntegerCharacteristic>
    <id>9</id>
    <name>SurfConduct</name>
    <parentProperty class="Property" reference="../../../../"/>
    <unit class="com.artenum.penelope.mesh.unit.SimpleStringUnit">
      <unit></unit>
      <standardUnit></standardUnit>
    </unit>
    <localisation>2</localisation>
    <informationLink></informationLink>
    <value>1</value>
  </IntegerCharacteristic>
</entry>
</characteristicList>
<isCompound>false</isCompound>
<type>Surface material model</type>
</Property>

```

## Annex C Launching script

The script hereafter corresponds to the launching script under UNIX based systems. It shows the equivalent launching command line and illustrates the key system variable being adjusted. Equivalent scripts are available for Windows-based systems, however on Windows based systems the batch (-b) and project (-p) modes are not available.

```
#!/bin/sh
while getopts p:b:h:m option
do
  echo ""
  do
    case $option in
      p)
        java -Xmx1024M -jar -
Dlogback.configurationFile=resources/logging/logback-headless.xml -
Dfelix.config.properties=file:resources/felix/config.properties -
Dactivate3DViews=true -Dorg.keridwen.config=./resources/org-spis-geo-
osx64b.properties -Dorg.keridwen.headless=true -
Dorg.spis.batch.project.path=$OPTARG dependencies/thirdparty/felix-4.0.2/felix.jar
        exit 0
        ;;
      b)
        java -Xmx1024M -jar -
Dlogback.configurationFile=resources/logging/logback-headless.xml -
Dfelix.config.properties=file:resources/felix/config.properties -
Dactivate3DViews=true -Dorg.keridwen.config=./resources/org-spis-geo-
osx64b.properties -Dorg.keridwen.headless=true -Dorg.spis.batch.script.path=$OPTARG
dependencies/thirdparty/felix-4.0.2/felix.jar
        exit 0
        ;;
      h)
        printf "Usage:\n"
        printf "* To run SPIS with its graphical user interface:\n"
        printf "$0 \n"
        printf "* To run SPIS in batch mode and execute a
simulation from a given SPIS project:\n"
        printf "$0 -p path/to/project.spis5\n"
        printf "* To run SPIS in batch mode and execute a given
Jython script:\n"
        printf "$0 -b path/to/jython/script\n"
        exit 2
        ;;
      m)
        java -Xmx1024M -splash:resources/images/splash_spis-geo-png24.png -
jar -Dlogback.configurationFile=resources/logging/logback.xml -
Dfelix.config.properties=file:resources/felix/config.properties -
Dactivate3DViews=true -Dorg.keridwen.config=./resources/org-spis-geo-
osx64b.properties -DforceLAF=Metal dependencies/thirdparty/felix-4.0.2/felix.jar
        exit 0
        ;;
      ?)
        java -Xmx1024M -splash:resources/images/splash_spis-geo-png24.png -jar
-Dlogback.configurationFile=resources/logging/logback.xml -
Dfelix.config.properties=file:resources/felix/config.properties -
Dactivate3DViews=true -Dorg.keridwen.config=./resources/org-spis-geo-
osx64b.properties dependencies/thirdparty/felix-4.0.2/felix.jar
        exit 0
        ;;
    esac
  done
done
```

**Reference:** ESA-SPISGEO-D7-SUM-2012-08-002

**Version:** 2

**Revision:** 2

**Status:** Validated

**Date:** 28/03/2013

## Annex D Example of global parameter file.

Example of XML-based global parameter file (partial).

```

<list>
  <GlobalParameter>
    <keyName>inPop2VollInteract</keyName>
    <category>Volume Interactions</category>
    <typeAString>String</typeAString>
    <valueAsFloat>0.0</valueAsFloat>
    <valueAsDouble>0.0</valueAsDouble>
    <valueAsInt>0</valueAsInt>
    <valueAsString>fractionOfFirstPopSource</valueAsString>
    <unit>None</unit>
    <description>Defines second interacting population (e.g. neutrals for CEX), must be one of ions1, ions2, elec1, elec2,
source1... source4, photoElec, secondElec</description>
  </GlobalParameter>
  <GlobalParameter>
    <keyName>scPotMonitorStep</keyName>
    <category>Outputs</category>
    <typeAString>double</typeAString>
    <valueAsFloat>0.0</valueAsFloat>
    <valueAsDouble>-100.0</valueAsDouble>
    <valueAsInt>0</valueAsInt>
    <unit>[s]</unit>
    <description>time step for spacecraft ground potential monitoring (0 => none, -n => n times)</description>
  </GlobalParameter>
  <GlobalParameter>
    <keyName>electronDistrib2</keyName>
    <category>Plasma</category>
    <typeAString>String</typeAString>
    <valueAsFloat>0.0</valueAsFloat>
    <valueAsDouble>0.0</valueAsDouble>
    <valueAsInt>0</valueAsInt>
    <valueAsString>PICVolDistrib</valueAsString>
    <unit>None</unit>
    <description>Name of the VolDistrib class to be used for the 2nd electron population</description>
  </GlobalParameter>
  (...)
  <GlobalParameter>
    <keyName>electronSpeedUp2</keyName>
    <category>Plasma</category>
    <typeAString>double</typeAString>
    <valueAsFloat>0.0</valueAsFloat>
    <valueAsDouble>1.0</valueAsDouble>
  </GlobalParameter>

```

**Reference:** ESA-SPISGEO-D7-SUM-2012-08-002

**Version:** 2

**Revision:** 2

**Status:** Validated

**Date:** 28/03/2013



```

<valueAsInt>0</valueAsInt>
<unit>[-]</unit>
<description>Numerical times speed-up factor for electron 2nd population</description>
</GlobalParameter>
</list>

```

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

## Annex E Example of CAD tool in geo Gmsh format.

Example of CAD model in Gmsh geo format. This example corresponds to a simple sphere inside a cubic computational volume.

```
// model nodes definition

// inner sphere (i.e. the spacecraft)
Point(1) = {1, 0, 0, 0.2};
Point(2) = {-1, 0, 0, 0.2};
Point(3) = {0, 1, 0, 0.2};
Point(4) = {0, -1, 0, 0.2};
Point(5) = {0, 0, 1, 0.2};
Point(6) = {0, 0, -1, 0.2};
Point(7) = {0, 0, 0, 0.2};

// external boundary (i.e. the far plasma)
Point(8) = {-10, 10, 10, 3};
Point(9) = {10, 10, 10, 3};
Point(10) = {10, -10, 10, 3};
Point(11) = {10, -10, -10, 3};
Point(12) = {10, 10, -10, 3};
Point(13) = {-10, 10, -10, 3};
Point(14) = {-10, -10, -10, 3};
Point(15) = {-10, -10, 10, 3};

// model edges definition
Circle (1) = {4, 7, 1};
Circle (2) = {4, 7, 5};
Circle (3) = {1, 7, 5};
Circle (4) = {4, 7, 2};
Circle (5) = {2, 7, 5};
Circle (6) = {2, 7, 3};
Circle (7) = {3, 7, 1};
Circle (8) = {6, 7, 3};
Circle (9) = {6, 7, 2};
Circle (10) = {6, 7, 4};
Circle (11) = {6, 7, 1};
Circle (12) = {3, 7, 5};
Line (13) = {8, 9};
Line (14) = {9, 12};
Line (15) = {12, 13};
Line (16) = {13, 8};
Line (17) = {8, 15};
Line (18) = {15, 10};
Line (19) = {10, 9};
```

**Reference:** ESA-SPISGEO-D7-SUM-2012-08-002

**Version:** 2

**Revision:** 2

**Status:** Validated

**Date:** 28/03/2013

```

Line (20) = {15, 14};
Line (21) = {14, 11};
Line (22) = {11, 10};
Line (23) = {11, 12};
Line (24) = {14, 13};

// model surfaces definition

// the inner sphere (i.e. the spacecraft)
Line Loop (1) = {4, -9, 10};
Ruled Surface (1) = {1};
Line Loop (2) = {2, -5, -4};
Ruled Surface (2) = {2};
Line Loop (3) = {5, -12, -6};
Ruled Surface (3) = {3};
Line Loop (4) = {6, -8, 9};
Ruled Surface (4) = {4};
Line Loop (5) = {3, -2, 1};
Ruled Surface (5) = {5};
Line Loop (6) = {3, -12, 7};
Ruled Surface (6) = {6};
Line Loop (7) = {7, -11, 8};
Ruled Surface (7) = {7};
Line Loop (8) = {1, -11, 10};
Ruled Surface (8) = {8};
Line Loop (9) = {14, -23, 22, 19};

// the external boundary (i.e. the far plasma)
Plane Surface (9) = {9};
Line Loop (10) = {21, 22, -18, 20};
Plane Surface (10) = {10};
Line Loop (11) = {15, -24, 21, 23};
Plane Surface (11) = {11};
Line Loop (12) = {20, 24, 16, 17};
Plane Surface (12) = {12};
Line Loop (13) = {13, -19, -18, -17};
Plane Surface (13) = {13};
Line Loop (14) = {13, 14, 15, 16};
Plane Surface (14) = {14};

// computational volume definition
Surface Loop (1) = {10, 11, 14, 13, 12, 9, 2, 5, 6, 7, 4, 3, 1, 8};
Volume (1) = {1};

// Physical marker definition.
Physical Surface (56) = {7, 4, 6, 3, 5, 2, 1, 8};

```

**Reference:** ESA-SPISGEO-D7-SUM-2012-08-002

**Version:** 2

**Revision:** 2

**Status:** Validated

**Date:** 28/03/2013

Physical Surface (57) = {11, 9, 14, 12, 10, 13};

Physical Volume (58) = {1};

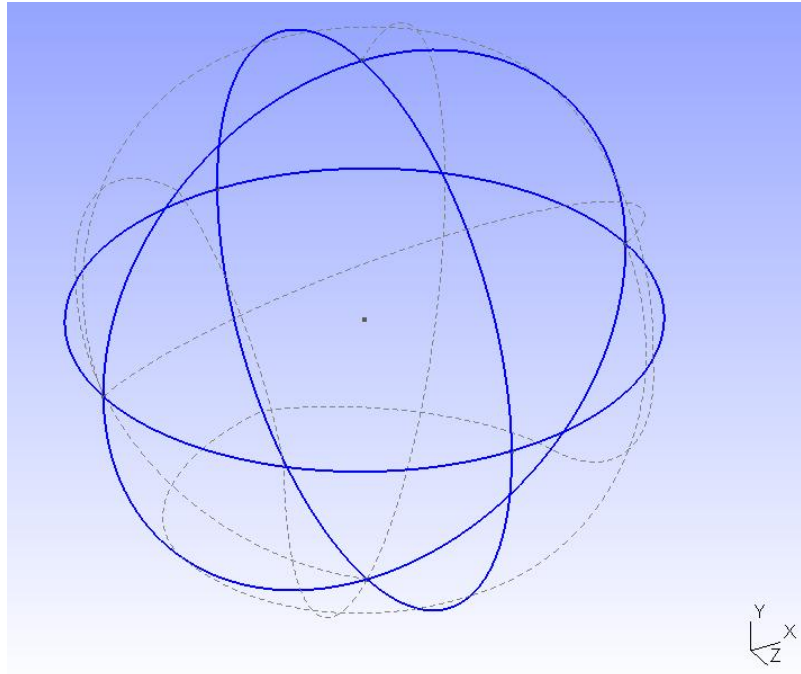
Physical Point (59) = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15};

Physical Line (60) = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24};

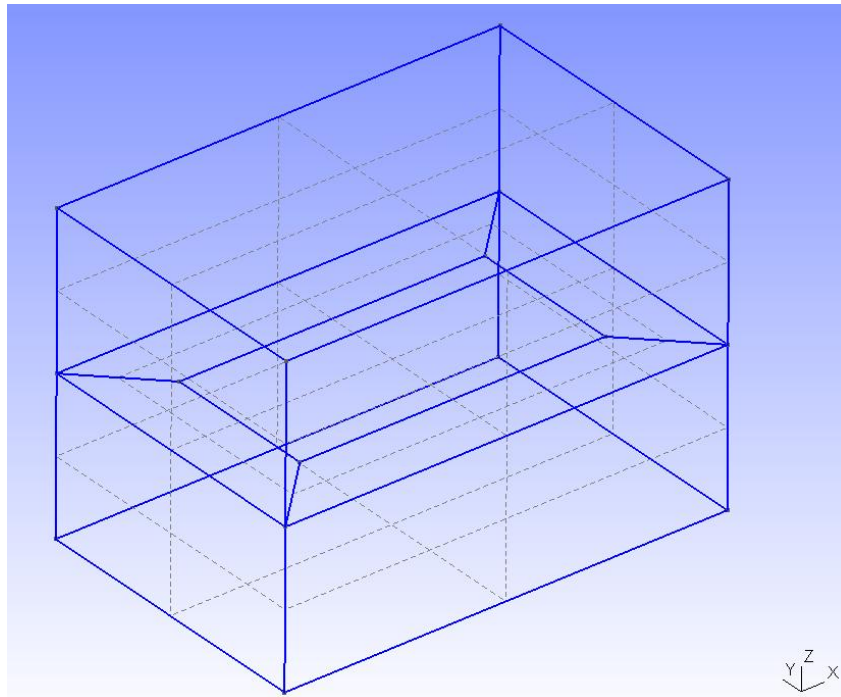
<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

## Annex F Example of CAD templates

Example of CAD templates in Gmsh geo format.

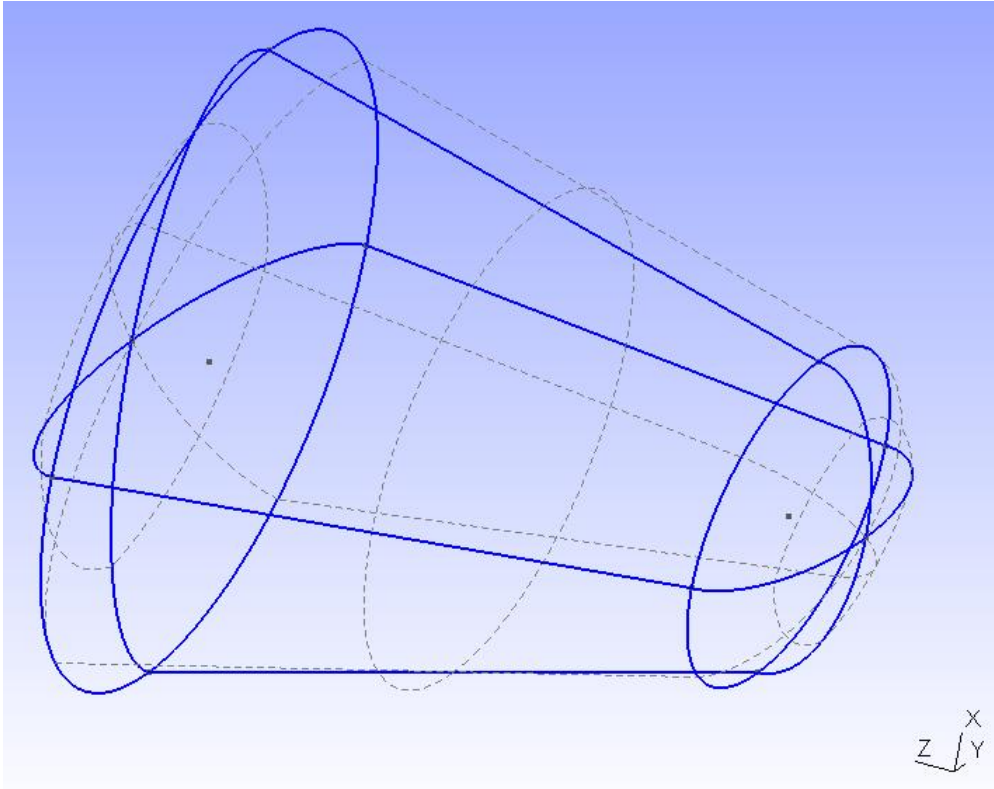


**Tableau 8 : Simple sphere, typically used to define the external boundary of a system immersed into a Maxwellian plasma at rest.**



**Tableau 9 : Example of thin surface (2D), can be included into more complex system and used as local refinement box.**

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	



**Tableau 10: Example of ellipsoidal-truncated external boundary. Such shape, that can be finely parameterised, can be used as external boundary for the system immersed into drifting plasma, like in LEO.**

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

## Annex G List of main SPIS material properties

MaterialProperty list				
Name	Type	UnitName	Comment	Used In
RDC	float	[-]	Relative dielectric constant	Used in SPIS 4 - Nascap parameter 0
DTM	float	[m]	Dielectric material thickness ( <b>NOT USED</b> , please see MatThickness in the Macroscopic Characteristic property).	<b>Not Used in SPIS 4</b> - Nascap parameter 1
BUC	float	[ohm-1.m-1]	Bulk conductivity, must be set negative if a conductor (in place of infinite for a perfect conductor)	Used in SPIS 4 - Nascap parameter 2
ATN	float	[-]	Atomic number	Used in SPIS 4 - Nascap parameter 3
MSEY	float	[-]	Maximum secondary electron emission (SEE) yield for electron impact	Used in SPIS 4 - Nascap parameter 4
PEE	float	[keV]	Primary electron energy that produces maximum SEE yield	Used in SPIS 4 - Nascap parameter 5
RPN1	float	[Angstroms]	Range parameter r1 in the range expression $r1 (E/E0)^{n1} + r2 (E/E0)^{n2}$ , with $E0 = 1$ keV (or equivalently with no E0 coefficient and E expressed in keV)	Used in SPIS 4 - Nascap parameter 6
RPR1	float	[-]	Range parameter n1	Used in SPIS 4 - Nascap parameter 7
RPN2	float	[Angstroms]	Range parameter r2	Used in SPIS 4 - Nascap parameter 8
RPR2	float	[-]	Range parameter n2	Used in SPIS 4 - Nascap parameter 9
SEY	float	[-]	Secondary electron yield due to impact of 1 keV protons	Used in SPIS 4 - Nascap parameter 10
IPE	float	[keV]	Incident proton energy that produces maximum secondary electron yield	Used in SPIS 4 - Nascap parameter 11
PEY	float	[A/m2 at 1 AU]	Photoelectron current for normally incident sunlight	Used in SPIS 4 - Nascap parameter 12
SRE	float	[ohms]	Surface resistivity	Used in SPIS 4 - Nascap parameter 13
MAP	float	[V]	Maximum (absolute) potential attainable before a discharge occurs	Used in SPIS 4 - Nascap parameter 14
MPD	float	[V]	Maximum potential difference between surface and underlying conductor before a discharge occurs	Used in SPIS 4 - Nascap parameter 15
RCC	float	[ohm-1.m-1]	Radiation induced conductivity coefficient K in the law $K*(rate/rate0)^D$ , with $rate0 = 1$ Rad/s (TBC), (or equivalently with no rate0 coefficient and rate expressed in	Used in SPIS 4 - Nascap parameter 16

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	

			Rad/s)	
RCP	float	[-]	Radiation induced conductivity power (D)	Used in SPIS 4 - Nascap parameter 17
MAD	float	[kg/m3]	Material density	Used in SPIS 4 - Nascap parameter 18
WORK	float	[eV]	work function	1) Used in the v4.3 for FowlerNordheim source 2) SPIS-ESD
ACEN	float	[J]	Activation Energy	Parameter dedicated to the DCSolver in SPIS-DC
THCO	float	[W/K/m]	Thermal conductivity	Used in SPIS-ESD
HEAT	float	[J/K/kg]	Specific heat	Used in SPIS-ESD
ATOM	float	[kg]	Atomic mass	Used in SPIS-ESD
ELRE	float	[ohm.m]	Electric resistivity of metal	Used in SPIS-ESD
PREF	float	[Pa]	Reference pressure for evaporation	Used in SPIS-ESD
TREF	float	[K]	Reference temperature for evaporation	Used in SPIS-ESD
VAPE	float	[J]	Vaporization energy per mole	Used in SPIS-ESD
FUSE	float	[J]	Fusion energy per mole	Used in SPIS-ESD
VMAS	float	[kg/m3]	Volumic mass	Used in SPIS-ESD
MELT	float	[K]	Melting temperature	Used in SPIS-ESD
CION	float	[m2]	Ionization cross section	Used in SPIS-ESD
BUCT	XY series	X unit [s] - Y Unit [ohm-1.m-1]	Bulk conductivity versus time	Used in SPIS-5
SEEEYFlag	Int	[-]	SEE by electron yield dependence in energy flag (0-analytical model / 1-tabulated mode)	Used in SPIS-5
SEEEYETab	Array	[eV,-]	SEE by electron dependence in energy tab	Used in SPIS-5
SEEEYTTab	Array	[rad,-]	SEE by electron dependence in angle tab	Used in SPIS-5
BCKEAFlag	Int	[-]	Backscattering of electron albedo dependence in energy flag (0-analytical model / 1-tabulated mode)	Used in SPIS-5
BCKEAETab	Array	[eV,-]	Backscattering of electron albedo dependence in energy tab	Used in SPIS-5
BCKEATTab	Array	[rad,-]	Backscattering of electron albedo dependence in angle tab	Used in SPIS-5

**\*WARNING:** The DTM property is not used. To define the dielectric material thickness, use the local parameter *MatThickness* defined in [RD7].

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	



## Annex H List of provided NASCAP based materials

- 0: ITOC (Material coated with ITO)
- 1: CERS (Solar cell material. Cerium doped silicon with MgF2 coating)
- 2: CFRP (Carbon fibre, conducting, no resin layer)
- 3: KAPT (Kapton, average values for SEE...)
- 4: COSR (Optical solar reflector without MgF2 coating. Cerium doped glass type)
- 5: EPOX (Epoxy. Thin layer of Epoxy resin on (conducting) Carbon fibre)
- 6: BLKP (Non conductive black paint. SEE yields are as measured for Electrodag 501)
- 7: BLKH (Non conductive black paint HERBERTS 1002-E. Values updated 3.10.88.)
- 8: BLKC (Conductive black paint Electrodag 501)
- 9: PCBZ (White paint PCB-Z assumed to be conductive in space)
- 10: PSG1 (White paint PSG 120 FD assumed to be conductive in space.)
- 11: TEFL (Teflon, DERTS measurements of SEE)
- 12: CONT (Generic Dielectric after 5 years in GEO environment.)
- 13: GOLD
- 14: SILV (Silver as from NASCAP library)
- 15: ALOX (Oxydized Aluminium. SEE yields from DERTS for Aluminium/Kapton)
- 16: STEE (Steel, SEE sigma +Emax from DERTS, curve shape from CONT material)
- 17: AL2K (Aluminium according to NASCAP-2k)
- 18: AU2K (Gold according to NASACP-2k)
- 19: KA2K (Kapton according to NASACP-2k)
- 20: TE2K (Teflon according to NASACP-2k)
- 21: OSR2K (OSR according to NASACP-2k)
- 22: BK2K (Black Kapton according to NASACP-2k)
- 23: SC2K (Solar Cells according to NASACP-2k)
- 24: NP2K (Non-conductive paint according to NASACP-2k)
- 25: GP2K (Graphite according to NASACP-2k)

<b>Reference:</b> ESA-SPISGEO-D7-SUM-2012-08-002	<b>Version:</b> 2	<b>Revision:</b> 2
<b>Status:</b> Validated	<b>Date:</b> 28/03/2013	