



focussuite **WHITE PAPER**

Code: GMV-FOCUSSUITE-WP

Internal code: GMVAD 20463/12 V3/16

© GMV Aerospace and Defence, S.A.U., 2016 all rights reserved

GMV
www.gmv.com

 www.facebook.com/InfoGMV
 [@InfoGMV](https://twitter.com/InfoGMV)

gmv[®]
INNOVATING SOLUTIONS

THIS PAGE INTENTIONALLY LEFT BLANK

DOCUMENT STATUS SHEET

Version	Date	Pages	Changes
1.0	26/03/2012	56	First version
1.1	08/01/2014	56	Upgrade of focusleop computational layer
1.2	15/07/2016	75	Updates according to baseline product evolution

TABLE OF CONTENTS

1. INTRODUCTION TO FOCUSSUITE	8
2. REFERENCES	11
2.1. APPLICABLE DOCUMENTS	11
2.2. REFERENCE DOCUMENTS	11
3. FOCUSSUITE GENERAL DESCRIPTION	12
3.1. KEY REQUIREMENTS	12
3.2. FOCUSSUITE PRODUCTS ARCHITECTURE	12
4. MAIN FOCUSSUITE COMPONENTS	15
4.1. GRAPHICAL USER INTERFACE	15
4.1.1. GENERAL.....	15
4.1.2. USER LEVELS.....	16
4.1.3. UNDO/REDO.....	16
4.1.4. GUI OVERVIEW	16
4.2. GRAFOS : FOCUSSUITE PLOTTING TOOL	21
4.3. EBRO : FOCUSSUITE EVENTS BROWSER	23
4.4. FRAMES : REFERENCE FRAME CONVERSION	23
4.5. EVENTS LOGGER	24
4.6. FOCUSPROCESS MANAGER	26
4.7. FOCUSDATA MANAGER	26
4.8. COMMUNICATIONS LAYER	27
5. FOCUSASPECT	28
6. VISUALFOCUS	29
7. FOCUSUITESYNC	33
8. FOCUSCLOSEAP	35
8.1. FOCUSCLOSEAP OPERATION MODES	35
8.2. CONJUNCTION DETECTION	36
8.3. COLLISION RISK ASSESSMENT	36
8.4. RETRIEVAL OF INPUTS	37
8.5. GENERATION AND SUBMISSION OF OUTPUTS	37
9. AUTOMATION (AUTOFOCUS)	39
9.1. SOL: SPACECRAFT OPERATIONS LANGUAGE	39
9.2. AUTOFOCUS PROCEDURE DEFINITIONS	40
9.3. AUTOFOCUS PROCEDURE EXECUTIONS	40
10. COMPUTATION LAYER	42

10.1. FOCUSGEO COMPUTATIONAL LAYER	42
10.1.1. MISSION-INDEPENDENT MODULES.....	42
10.1.2. MISSION-DEPENDENT MODULES	53
10.2. FOCUSLEO, FOCUSLEOP AND FOCUSCN COMPUTATIONAL LAYER	56
10.2.1. METHODS AND MODELS.....	56
10.2.2. FOCUSLEO, FOCUSLEOP AND FOCUSCN COMMON FUNCTIONALITIES.....	58
10.2.3. FOCUSLEO SPECIFIC FUNCTIONALITIES	58
10.2.4. FOCUSLEOP SPECIFIC FUNCTIONALITIES.....	59
10.2.5. FOCUSCN SPECIFIC FUNCTIONALITIES.....	72
10.3. FOCUSSUITE SUPPORTED PLATFORMS	74

INDEX OF TABLES AND FIGURES

Table 2–1: Applicable Documents.....	11
Table 2–2: Reference Documents.....	11
Table 10–1: <i>focussuite</i> supported platforms (October 2016).....	74
Figure 1: <i>focussuite</i> products structure as a high level architecture	13
Figure 2: Main window showing the different working areas.....	17
Figure 3: <i>focussuite</i> main window working area: set of programs tab selector.....	18
Figure 4: <i>focussuite</i> main window working area: program selector.....	18
Figure 5: <i>focussuite</i> main window working area: edit window	19
Figure 6: Example of highlighted text.....	20
Figure 7: <i>ebro</i> application	20
Figure 8: <i>grafos</i> display.....	21
Figure 9: <i>grafos</i> Main Window	22
Figure 10: <i>frames</i> : Reference frame conversion tool user interface	24
Figure 11: evelog and its relationship with the rest of <i>focussuite</i>	25
Figure 12: <i>focusevelog</i> client main window.....	25
Figure 13: <i>focusdata</i> Manager Scenarios & Workspaces diagram.....	27
Figure 14: <i>focusaspect</i> Client main window	28
Figure 15: <i>visualfocus 3D Window</i>	30
Figure 16: <i>visualfocus 2D Window</i>	30
Figure 17: <i>visualfocus</i> covariance ellipsoid representation.....	31
Figure 18: <i>visualfocus</i> 3D representation in real time including mobile elements	31
Figure 19: <i>visualfocus</i> footprint for antenna radiation representation.....	32
Figure 20: <i>focussuitesync</i> Client Main Window	33
Figure 21: <i>focussuitesync</i> Client Main Window	34
Figure 22: <i>closeap</i> Conjunction detection report	38
Figure 23: <i>autofocus</i> SOL Editor	39
Figure 24: <i>autofocus</i> procedures editor window	40
Figure 25: <i>autofocus</i> procedures agenda window.....	41
Figure 26: Preread example of graphical output for condensed range measurements	43
Figure 27: GORDAM Example of Range residual plot	44
Figure 28: SORBAM Orbital evolution Plots (Osculating semi-major axis example).....	45
Figure 29: <i>focusgeo</i> : LEGO plots showing long term propagation	46
Figure 30: SOLONG Manoeuvre plots.....	47
Figure 31: INCLON Manoeuvre plots	47
Figure 32: PRODGEN Gantt diagram	48
Figure 33: Events XML format	49
Figure 34: <i>focusgeo</i> ISATPAR Interface File	49
Figure 35: COLLOC Orbit Plots: Eccentricity Vector.....	50

Figure 36: *focusgeo* COLLOC Inter-satellite Distance Plot 51
 Figure 37: *focusgeo* COLLOC Collocation Plots 51
 Figure 38: *focusgeo* INICOL Collocation Insertion Plot (Osculating longitude) 52
 Figure 39: *focusgeo*: HPLP plots example 54
 Figure 40: Strategy tree and strategy profile 59
 Figure 41: Example of INITGUESS constraints options panel - Transfer Constrains Panel 61
 Figure 42: Example of INITGUESS Constraints Options Panel - Sequencer Constrains 62
 Figure 43: INITGUESS output strategy file 62
 Figure 44: Example of part of the AEFOS output file 63
 Figure 45: Example of report generated by AEFOS 64
 Figure 46: EVENTS Orbital Events panel 65
 Figure 47: EVENTS Events Report File 66
 Figure 48: EVENTS EBRO Gantt chart 66
 Figure 49: Example of plot generated by VISIBILITY 67
 Figure 50: MONTECARLO injection dispersion errors 67
 Figure 51: *focusleop* for mission analysis 70
 Figure 52: *focusleop* for real operations 71
 Figure 53: FDF Scenario and workspace structure 73

1. INTRODUCTION TO **FOCUSSUITE**

This section provides a general introduction to **focusuite**, GMV's advanced off the shelf flight dynamics solution for satellite operations, and shows how our system provides the most complete solution to the needs of both end-user products and flight dynamics system developers. Our product addresses the specific requirements of satellite operators while at the same time providing a framework that allows further product development and evolution.

focusuite can therefore be regarded as a COTS product that can be customized to fit your needs, but also as a framework oriented to minimizing programme development schedule, costs and risks, and at the same time greatly improving the efficiency of operations (minimizing operations workload) and reducing the risk of human errors.

GMV acknowledges the fact that every customer has different needs and requirements when it comes to covering their flight dynamics needs. This reflection comes from our privileged position as the main European supplier of flight dynamics systems and services after almost two decades of uninterrupted successful services to a large variety of clients. Consequently and consistently with this we are pleased to offer to the market **focusuite**, the ultimate flight dynamics solution for satellite operations. Our solution is characterized by the following unique high-level requirements:

- Advanced off-the-shelf, multi-mission, multi-satellite flight dynamics solution for flight dynamics satellite control that sets a new standard in functionality, reliability, flexibility and user friendliness.
- Capability to provide full lifecycle flight dynamics operations support through our unsurpassed collection of flight-proven mission independent and mission specific functionality. Flight-proven support of commercial satellite platforms is provided with accuracy fully consistent with native systems.
- Availability of a large collection of plug-and-play components providing unprecedented functionality and being able to customize the system to fit your specific needs. GMV provides custom solutions with strong customer focus and customer support.
- Provision of a generic framework that allows further product development and evolution, including the ability to integrate external applications with unprecedented ease. Our open framework dramatically boosts productivity, system usability, accessibility and stability.

focusuite has therefore been conceived as a real framework as it is demonstrated by its major functionalities: a **computation layer** based on the extensive reuse of existing and improved software, a **client/server architecture**, a **database driven** system, an **advanced GUI** (based on desktop applications philosophy: "everything-in-one-working-area" and "all-one-click-away" and using a GUI toolkit that allows a development of GUIs through configuration files rather than through code), procedures automation capability through the **autofocus** extension (based on a dedicated language **SOL** -Spacecraft Operations Language), advanced **graphical capabilities**, **portability** (e.g. Windows / UNIX / Linux), **extensibility** (any extra functionality following certain I/O rules can be easily integrated via configuration files) and finally, capability to perform **unlimited Undo/Redo** operations.

focusuite includes a suite of generic operational flight dynamics systems. They share the same architecture and many common components. Each of them has a different computational layer, all of them based originally on flight proven ESA's packages like PEPSOC and NAPEOS. The following stand-alone products are included within **focusuite**:

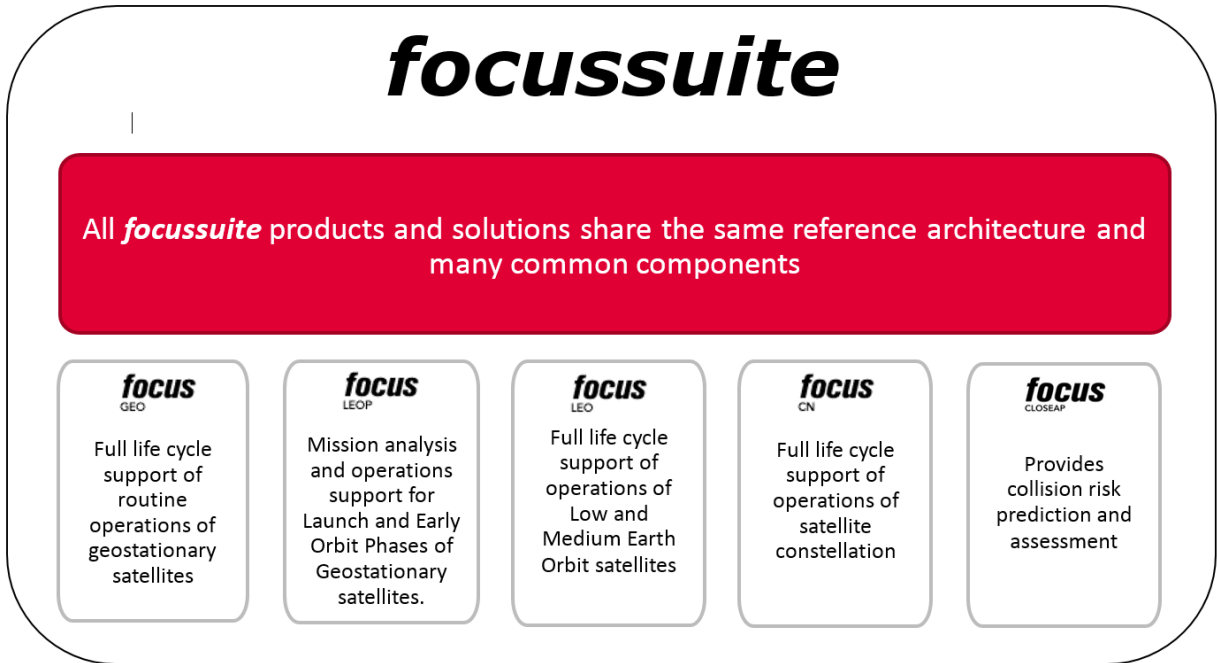


Table 1-1 –focussuite operational Flight Dynamics Systems

The following paragraphs describes the overall structure of **focussuite** products and a more detailed description of the common components used by all the products within **focussuite**, namely:

On the client side:

Component	Description
ebro	Generic Events Browser, shows a set of events in a Gantt chart. Self-contained tool, to be used in all focussuite products.
grafos	Generic plotting tool. Self-contained tool, to be used in all focussuite products.
focusclient	Client side of focussuite products, including the GUI and the communications layer
focusapi	API (Application Programming Interface) that can be used by external clients to interact with focusprocess manager . In particular, it shall be used by focusclient and autofocus
autofocus	focussuite Automation module.
focusframe	Stand-alone application used for conversion between different reference frames. This application has been provided by GMV in previous systems and shall be provided in focussuite products
focusaspect	Telemetry Monitoring tool
visualfocus	Visualisation tool providing 2D, 3D and Sensor's Views

On the server side:

Component	Description
<i>focusprocess manager</i>	Processing component in the Server Side of <i>focussuite</i> products.
<i>focusdata manager</i>	Manager of operational data files (including revision control), in the Server Side of <i>focussuite</i> products.
<i>focusart</i>	Optional component that provides Automatic Regression Testing functions, to let the developers or system administrators execute automatically a set of reference tests on the computation functions installed at the server side. It can alternatively be installed as a stand-alone tool that will use the same computation functions as the operational system.

On both sides:

Component	Description
<i>tkforms / tkflib</i>	Data access layer for computation programs and user interaction toolkit
<i>focusvelog (Events Logger)</i>	Generic logger of operational events. It is a generic client-server component, easily adaptable to <i>focussuite</i> products or any other system.
<i>focusComm</i>	Communications layer

At the same time, several **stand-alone products** based on ***focussuite*** infrastructure are available. A good example is ***focuscloseap***, a generic tool for automatic detection of close approaches between geostationary satellites and objects intersecting their orbit. An independent document providing complete information on ***focuscloseap*** is available at ***focussuite*** web page: <http://focussuite.gmv.com/>

Additionally, GMV has a long experience in developing customised ***focussuite*** -based solutions for specific needs. The existing framework and the multiple development tools associated to it make this process extremely efficient and reliable. GMV will be pleased to study with you ways to develop a ***focussuite*** -based solution for your own needs, both in operational and mission analysis contexts.

The next chapters provide a general description of ***focussuite*** and a high level overview of its main components.

2. REFERENCES

2.1. APPLICABLE DOCUMENTS

The following documents, of the exact issue shown, form part of this document to the extent specified herein. Applicable documents are those referenced in the Contract or approved by the Approval Authority. They are referenced in this document in the form [AD.X]:

Ref.	Title	Code	Ver.	Date
[AD. 1]				

Table 2-1: Applicable Documents

2.2. REFERENCE DOCUMENTS

The following documents, although not part of this document, amplify or clarify its contents. Reference documents are those not applicable and referenced within this document. They are referenced in this document in the form [RD.X]:

Ref.	Title	Code	Ver.	Date
[RD. 1]				
[RD. 2]				
[RD. 3]				
[RD. 4]				

Table 2-2: Reference Documents

3. FOCUSSUITE GENERAL DESCRIPTION

3.1. KEY REQUIREMENTS

Among others, the following key requirements have been taken into account in the design of the system:

- **Openness:** *focussuite* makes use of open standards such as XML for exchange of products and standard communication protocols with external systems. Moreover, it provides a well-documented API (Application Programming Interface) to enable execution of the system functions from external systems plus its unique capability of integrating external applications into the package without code modifications or software builds.
- **Portability (Windows / UNIX / Linux):** All the technical solutions chosen for each part of the system are portable to these platforms. Operating system dependent modules have been minimised and encapsulated.
- **Client / Server,** both available in Windows, UNIX and Linux. Heterogeneous combinations (e.g. client in Windows connecting to server in UNIX) are supported.
- **Flexible, expandable, scalable:** Ease of maintenance is a must; it is possible to perform quick updates of the system and inclusion of new modules without having to rebuild the system.
- **Ergonomic, Flexible and Visual GUI:** multiple datasets, highly configurable
- **Capability to perform “undo/redo” operations:** This lets the user go forward and backwards in the stack of actions performed in the system.
- **Full life-cycle support:** Mission independent + mission specific functions, GMV's support for customisation.
- Capability to support **multi-user, multi-satellite** and **multi-environment concepts**
- Highly **efficient:** Automation capabilities

3.2. FOCUSSUITE PRODUCTS ARCHITECTURE

focussuite is a generic framework allowing the generation of a family of products that share the same architecture and infrastructure. They are systems built following a three-tier architecture (User Tier / Process Management Tier / Data & Computation Tier). The following figure shows this architecture.

The three-tier architecture is consistent with state-of-the-art trends in system design. It is a system design pattern that has been used successfully since the early 1990s on thousands of systems of various types. The three tiers in *focussuite* work as follows:

- The top tier, User Tier, provides services to the end-user, in particular everything related to presentation (user interface). It can be seen as the client-side in client-server architectures. Multiple clients may be running concurrently at a given time.
- The middle-tier, Process Management Tier, handles the requests for data and process execution received from the clients. It is the heart of the system.
- The third tier (Data and Computation Tier) has two distinct layers or sub-tiers.
 - The **Data Management** Layer encapsulates all the data access (control of concurrent read / write requests coming from different users, data versions control to enable undo actions, etc). In the standard versions of *focussuite* the data are stored in a set of ad-hoc files. Potentially this layer could be upgraded to use a commercial database (a natural and feasible extension given the current architecture).

- The **Computation Layer** hosts all the temporary processes that can be running concurrently at a given time, as a result of the execution of computation programs per user request. These processes act on temporary data retrieved previously from the Data Management Layer by the **focusprocess manager**. This approach ensures that data will not be corrupted in case of process failure and enables concurrent processing.

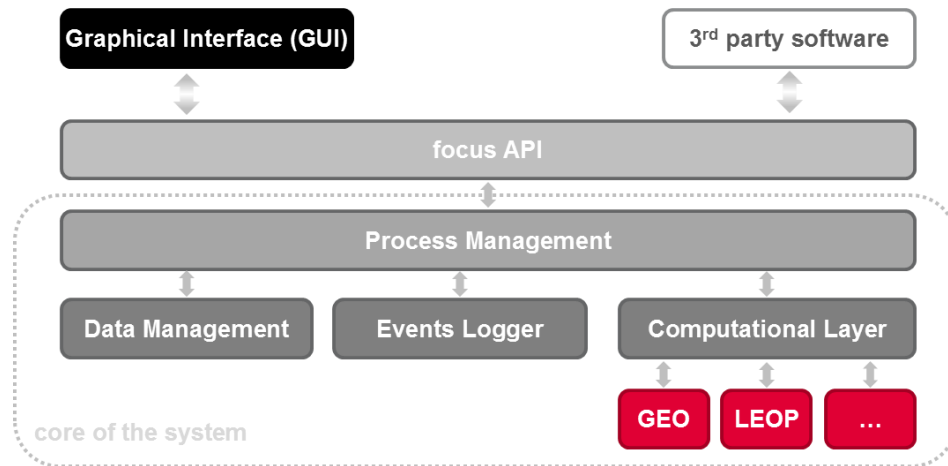


Figure 1: *focussuite* products structure as a high level architecture

Although the architecture has been defined as three-tier, this does not mean that three levels of hardware are needed. Nominally two levels (clients + servers) are foreseen, so that the second and third tiers work in the same server. However the architecture allows for an easy escalation towards a higher level of distribution.

In the frame of *focussuite* products, the nominal configuration in terms of system deployment is to install the User tier in a set of client workstations and the other tiers in a server. However, the architecture ensures the extensibility and scalability of the system. For example, the data management layer could be installed in a different machine for a large satellite constellation.

Many advantages derive from this architecture, in particular high performance (due to the higher distribution of the processing), flexibility, maintainability, reusability and scalability, by centralising process logic. Its management is a lot easier than in a general distributed system due to this centralisation.

In addition to the first decomposition of *focussuite* products, it is worth noting the following:

- **focusapi**: This API lets any component interact with the **focusprocess manager**. It is used by the GUI and by **autofocus** procedures, but it could also be used by external systems to interact with *focussuite* products. The API includes a front-end and a back-end (the one that talks to the **focusprocess manager**). This approach encapsulates the communications and allows easier integration. This way instead of an ICD, a documented API is provided.
- **tkforms/tkLib**: This toolkit or set of libraries provides all data access services including GUI panel edition and data access to/from computation programs. A common data file format (TKF) and the use of this toolkit improve commonality throughout the system and allows faster development of new components and an easier integration of new ones (such as new computation programs). The GUI, the **focusdata manager** and the Computation Programs use this toolkit. The format of this configuration files was specific for **tkforms**, but recently, the format of **tkforms** has been replaced by a more standard format as XML.

- **focuscomm:** *Communications Layer.* The purpose of this general use library is to encapsulate the communications between different components to make it possible to perform changes in the communications in future releases easily. Several modifications at this level could be explored, such as changing the communications to HTTP (to enable Internet connections between clients and server through a firewall) or converting them to CORBA services.
- **Revision controller:** This module supports the **focusdata manager** providing file revision control services. This function is needed to make it possible for the user to undo actions and retrieve old versions of data files.

4. MAIN **FOCUSSUITE** COMPONENTS

This section provides a high level overview of the main components of **focussuite**. Further sections provide more detailed descriptions on key components of the system such as the automation tools, plus an explanation on how to integrate external applications within **focussuite** framework.

4.1. GRAPHICAL USER INTERFACE

4.1.1. GENERAL

focussuite GUI is largely based on a proprietary toolkit for rapid GUI development called TkForms/TkFLib that has been developed by GMV. The TkForms/TkFLib toolkit is a piece of middleware that allows GUI development based on configuration files rather than on code (more details are provided in the following sections).

The most outstanding features of the **focussuite** GUI are:

- Efficiency and ergonomic: Current Flight Dynamics GUIs do not fully exploit capabilities for easing the access of the data and facilitating the use of the system. The figures below show a number screen shots of the **focussuite** GUI and highlight some of the advantages that the GUI brings.
- Additional functionality, reliability, efficiency: The new GUI not only brings substantial improvements in terms of efficiency and ergonomic but also provides additional functionality, including:
 - Detailed **Events Logging** mechanism including capabilities to visualise key results characterising the execution of the programs and therefore allowing for very efficient monitoring of station keeping planning procedure behaviour both in manual and automatic mode.
 - Capability to perform unlimited **Undo** and **Redo** operations, which represents a dramatic improvement of the system in terms of reliability. Every data modification and program execution operation can be undone and redone.
 - Advanced **data management** capabilities including the possibility to manage data sets (duplication, modification, deletion). The GUI includes the concept of private and public data sets thus reinforcing the reliability of the operations performed with the system and enabling far more efficient concurrent use of the system. Also capability for the user to perform on the fly duplication of operational data for performing trial executions before bringing data operational.
 - Improved mechanisms for work groups (**concurrent system access**), dramatically improved locking mechanisms.
 - Capability of **integrating new pieces of software** without the need of performing software rebuild, making maintenance a lot easier and dramatically less costly.
- Ease of use and learning curve: from the screen shots shown below it can be observed that the **focussuite** GUI represents a dramatic improvement in terms of ease of use and learning curve of the system. The system offers two levels of grouping of tasks that can be customised for each customer, therefore being capable of adapting to different operational approaches. Moreover, the system can be customised such that at any time the user is confronted with the normal sequence of operations making the system extremely easy to use.
- Additional **graphic capabilities** to the system will dramatically improve the productivity of the operations staff and provide a user-friendlier environment. A better output would also provide better view and better understanding of the results. In summary productivity and reliability will both be considerably improved. There several types of graphical information to be presented to the user, each of them visualised by a dedicated widget:
 - X-Y plots (for example orbital elements versus time or orbit determination measurements residuals)
 - Events (for example Orbital Events or a Timeline)

- 3D/2D displays

4.1.2. USER LEVELS

The system administrator assigns each user a user level. The user level is shown in the title bar in the main window of the GUI.

The number and names of the available user levels may vary depending on the customer. For example, the levels could be operator, expert and administrator. In the figure, a number is used as a user level.

The user level has an impact on the kind of actions that are authorised for each user (basically, process execution and dataset access). For example, the execution of a certain sensitive process may only be authorised to users at or above a given level. Another case is when the modification of a given dataset (either by edition or via the execution of a process) is restricted to a certain user level.

4.1.3. UNDO/REDO

The undo/redo functions are a very powerful feature that makes it possible for the user to correct the effects of a mistake when using the system. They also let the user go backwards and forward in the actions performed on the system.

Notice that our unique undo/redo feature works in a multi-user environment, which makes the process not trivial.

You can undo / redo a panel edition or a process execution, or even a complete automation procedure execution. In the second and third cases, all the datasets that were modified by the concerned process (es) will be consistently restored to the state they had before execution.

4.1.4. GUI OVERVIEW

This section provides a representative collection of screen shots from one *fofocussuite* family member, *focuscn* in order to provide an overview of the proposed GUI concept.

fofocussuite client includes standard menu bar, tool bar and status bar from which the user can get access to a large variety of options. Particularly noticeable are the **undo/redo**, **find** and **cut/paste** functions providing flight dynamics engineers with powerful editing means representing a significant improvement in reliability and productivity.

fofocussuite client includes a system log (called *fofocusevelog*) that is further described in a separate section.

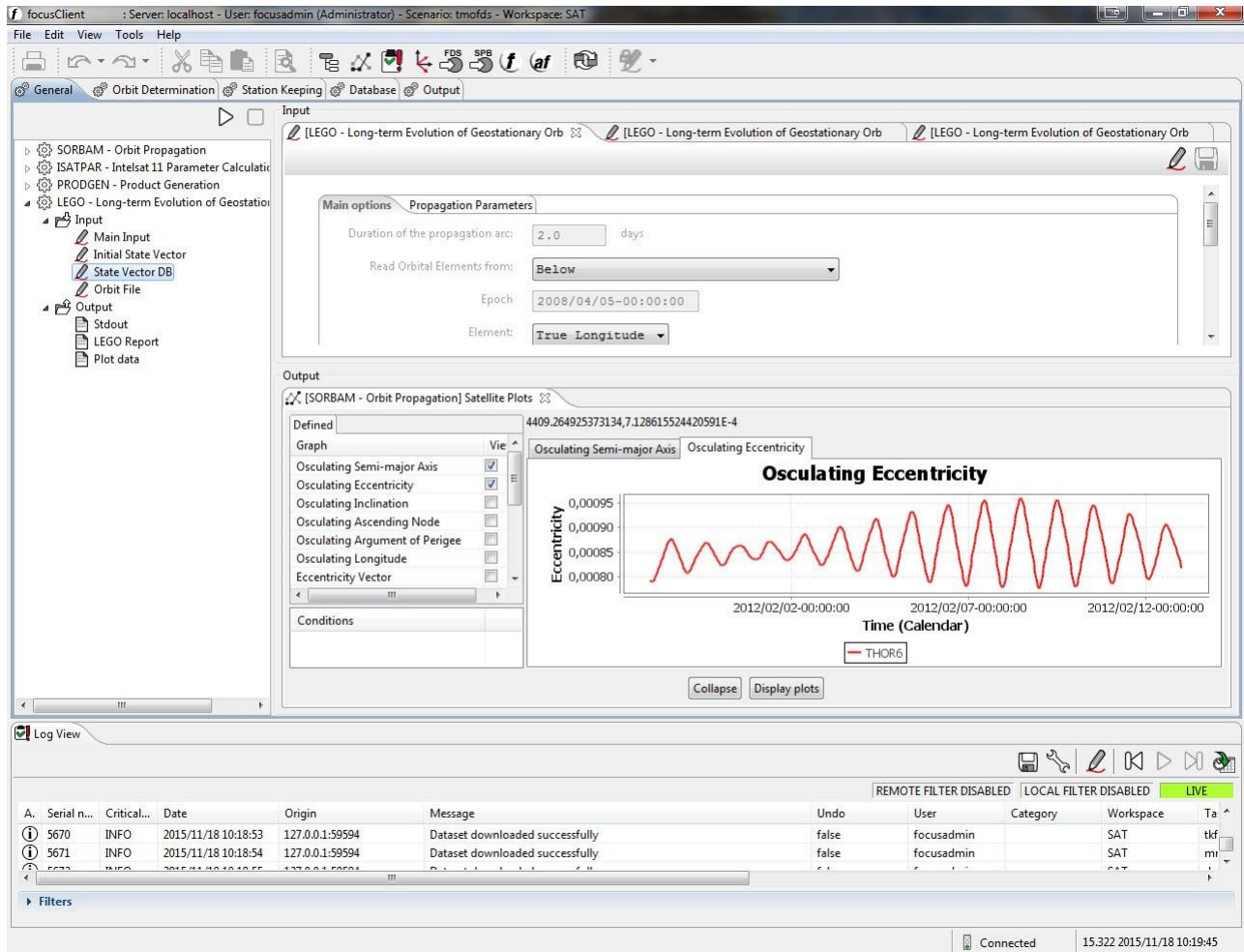


Figure 2: Main window showing the different working areas

The main GUI also provides direct access to other components embedded in the package such as **ebro** (events browsing application), **grafos** (XY plots), **frames** (conversion between reference frames), **focuselog** (standalone events logging client), new instance of the same client, **focusauto** (automation), **focuspect** (telemetry monitoring and display), backup/restore facilities, etc.

The main GUIs are described further in the following subsections.

4.1.4.1.1. **focussuite** Working Area

focussuite working area (see previous figure) includes four major elements (in addition to the satellite name) as follows:

- Tab selection (set of programs/functions selection)
- Program function selection
- Input area

- Output area

4.1.4.1.2. Tab selection (set of program functions)

The complete **focusuite** set of functions are logically grouped at the time of setup (the product allows full degree of flexibility on the number of integrated functions and how they are grouped together) according to a number of tabs that the user shall select in order to switch from one set of functions to another.

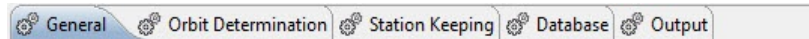


Figure 3: focusuite main window working area: set of programs tab selector

4.1.4.2. Program/function selection

The previously described programs tab selector allows getting access to the given set of **focusuite** programs or functions. To select from one to another the user shall simply 'click' on the corresponding program tree element. This deploys the main options associated to the corresponding function, namely:

- Input data setting
- Execution of the corresponding program
- Results browser

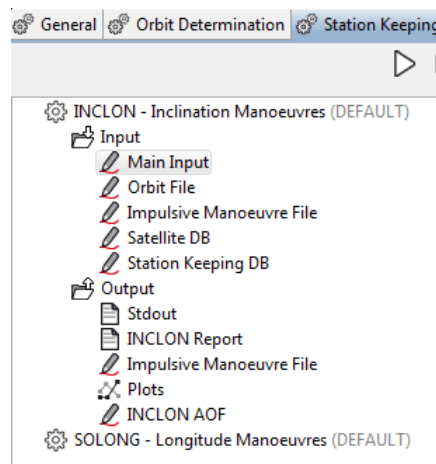


Figure 4: focusuite main window working area: program selector

4.1.4.3. Edit (Input) Area

As indicated earlier, all the edition process within **focusuite** is supported by **tkforms/tkflib**. **tkforms/tkflib** is a component of **focusuite**, which allows a common access layer for the edition of data that ensures the consistency between the different data across programs in the same **focusuite** implementation.

tkforms/tkflib allows the user to edit input data prior to program execution in a comfortable, secure and efficient manner. By default the corresponding data is made available to the user in 'view' mode. Note that **focusuite** is a multi-user system that requires reserving the data set for the user before modifying it. This is done through the 'Edit' button. When the Edit button is pressed then the data set is reserved to the user. In 'edit' mode the user can modify any data and either 'Save' or 'Cancel'. In 'view' mode the user can 'Quit' from the Edit area which means that the area is released (and taken over by the Output area).

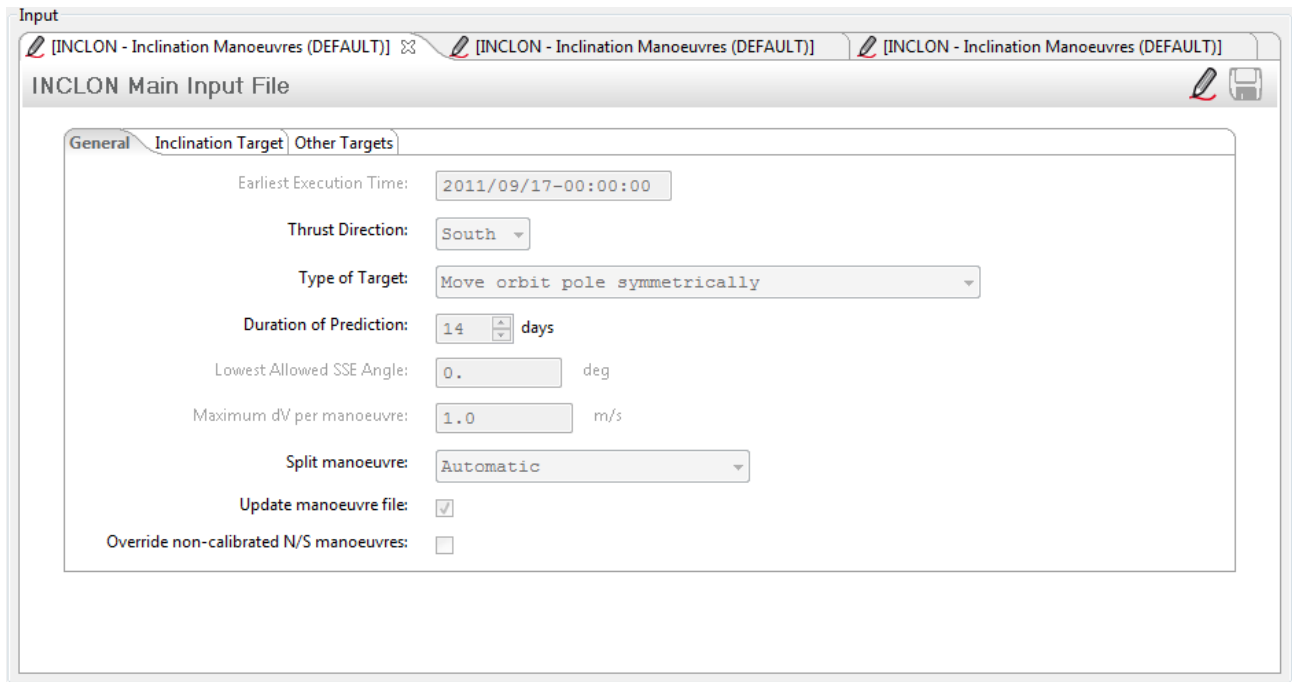


Figure 5: **focusuite** main window working area: edit window

4.1.4.4. Output Area

4.1.4.4.1. Text (ASCII) Output

When the user selects the Stdout option in the Output from a Program command (programs tree) then the Output Area is filled with the corresponding text as shown in the figure below. When double clicking the 'Stdout' option from the Program Selector a predefined (usually WordPad) application will be started and load the corresponding text file in a separate window.

In addition it has to be noted that should an error occur during execution of any flight dynamics function, then the output area is split into two sections, one of them showing the standard output of the program and a second one showing the standard error.

The capability of the widget has been extended, supporting the highlighting of a program output, edition functions (copy, cut and paste), printing in colour and find

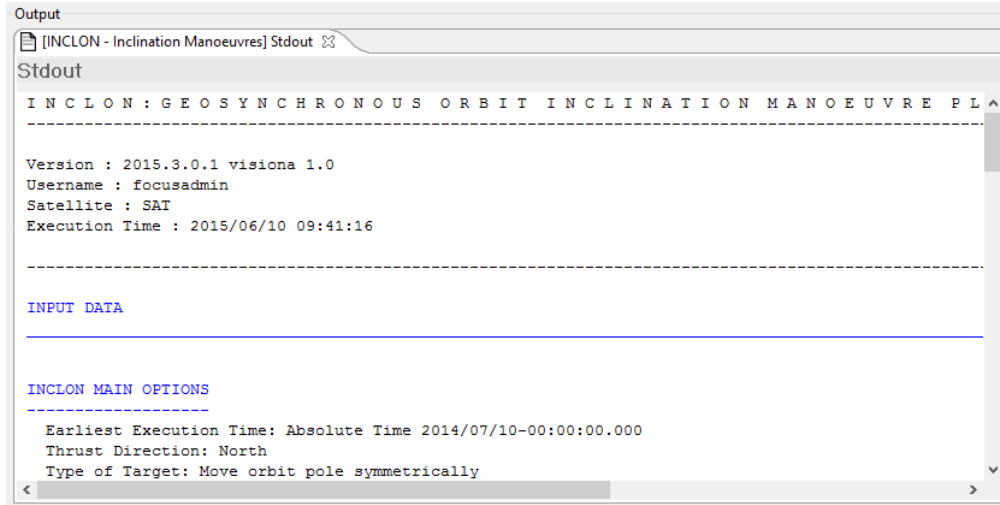


Figure 6: Example of highlighted text

4.1.4.4.2. Events Browser – Gantt Chart Output

Certain **focusuite** programs have the capability of generating an events file that can be visualized by the user in the form of a Gantt chart. As in the previous cases the Gantt chart is shown within the corresponding Output Area of **focusuite** GUI by means of a single click.

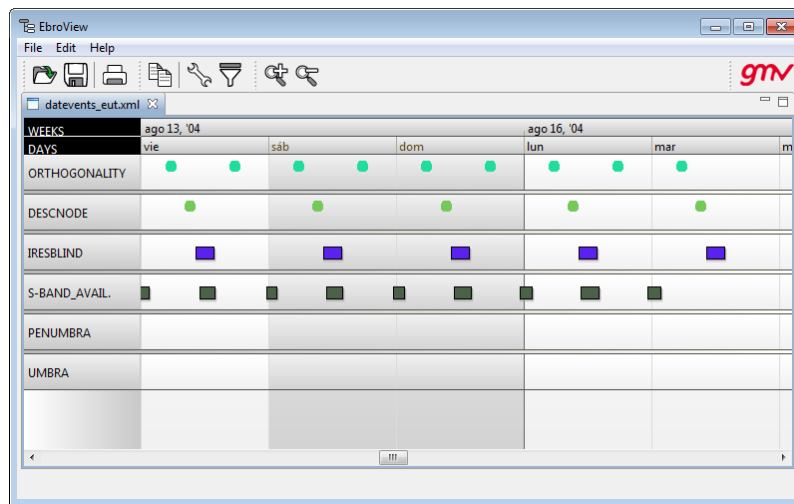


Figure 7: *ebro* application

When double clicking the 'Ebro' option (if applicable) from the Program Selector the complete **ebro** application will be started and load the corresponding Events browser file.

4.1.4.4.3. Graphics (X-Y plots)

Certain **focussuite** programs have the capability of generating X-Y plots that can be visualized by the user through a dedicated plot facility. As in the previous cases the plots are shown within the corresponding Output Area of **focussuite** GUI by means of a single click.

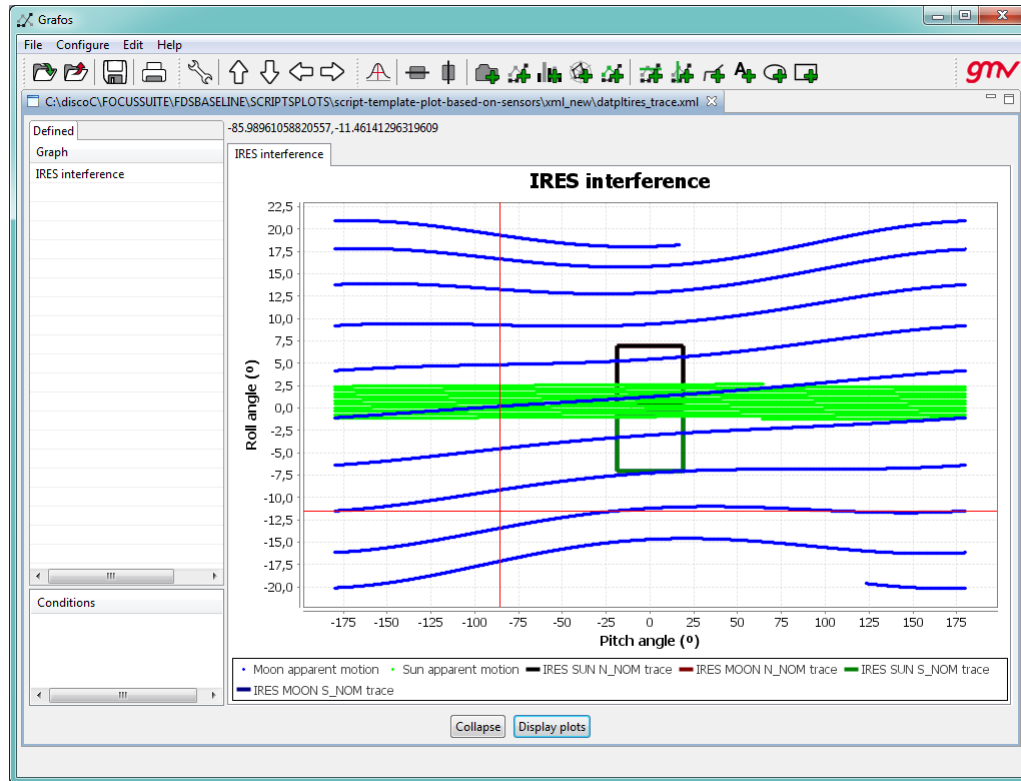


Figure 8: *grafos* display

When double clicking the 'Plot file' option (if applicable) from the Program Selector **grafos** will be started as an independent application.

Refer to next section where the specific **grafos** application is described in more detail.

4.2. **GRAFOS**: FOCUSSUITE PLOTTING TOOL

grafos is a generic, simple, robust X-Y plotting facility and fully integrated module available for all **focussuite** products. **grafos** allows to generate a wide variety of X-Y plots like residuals, orbit comparisons, station-keeping analyses, etc., and basic image manipulation functionality like zooming, labelling, scaling and printing. **grafos** covers these requirements as an external application as well as an integrated widget in flight dynamic systems.

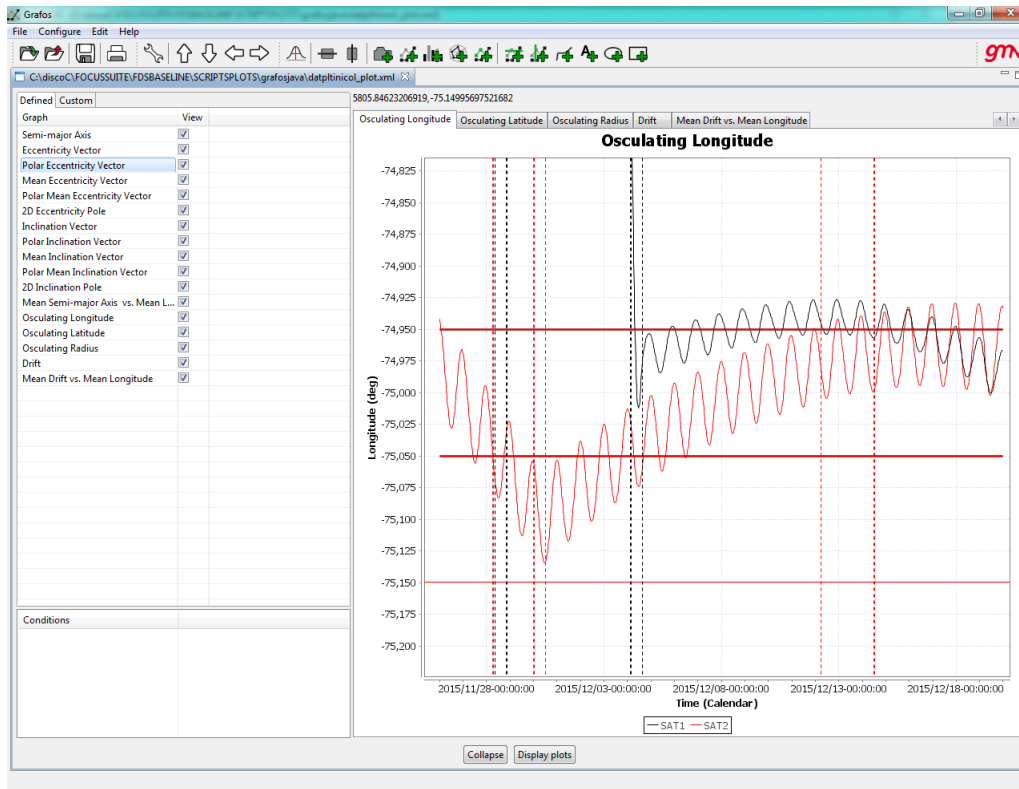


Figure 9: grafos Main Window

grafos provides the user with many plot manipulation capabilities such as:

- Creation of new plots through a series of dialogue windows and plot manipulation that allow the user to freely generate a new plot file.
- Adding or deleting sheets (tabs) in a plot
- Adding or deleting graphs (curves) in a plot
- Adding or deleting objects in a plot. The objects currently supported by **grafos** are text, lines, polygons and ovals.
- Clearing a Plot
- Setting options and properties: Setting options for each sheet (tab) through a configuration panel that allows setting the following options (fonts, colors, line widths and styles, etc.): general options (title, legend), axes options, attributes of the selected graph and attributes of the selected object.
- Changing the visualization window
 - Zoom In/Out: Allows you widen or narrow the x/y axis range
 - Shift left/right/up/down: Allows you move the x/y axis to subsequent or preceding values without changing the range.
- Deleting points from a graph
- Statistical analysis of the points in a graph, **grafos** shows the following information:
- Moving objects

- Working with multiple plots at once
- Deleting plots
- Saving plots
- Printing a plot File.

4.3. **EBRO**: FOCUSSUITE EVENTS BROWSER

ebro is a generic, simple event browsing application available for all **focussuite** products. **ebro** is available both as a stand-alone application and as fully integrated within **focussuite** products.

ebro supports changing the time scales, configuring events and managing event files.

When the cursor stays over an event, an event information window pops-up to display all the information about the event: Event type, event start and end times, event duration and comments section.

ebro provides the user with Events Files management capabilities such as:

- Opening and Closing existing Events file
- Browsing and navigating through the Gantt Chart Diagram
- Setting options for the Gantt Chart diagram
 - Event type configuration, to change the appearance of each type of event.
 - Adding and deleting event types, new types of events can be added and configured according to user needs.
 - Filtering events, allowing choosing the event types you want to display and their order along the vertical axis.
- Zooming: there are four scales for the time axis: year, month, day and hour
- Printing an event File.

4.4. **FRAMES**: REFERENCE FRAME CONVERSION

frames is the reference frames and time formats conversion tool of **focusgeo**.

frames converts orbital elements between different formats and reference frames. It also supports the conversion between osculating and mean elements, conversion between different time formats, and conversion to and from TLE (two line elements).

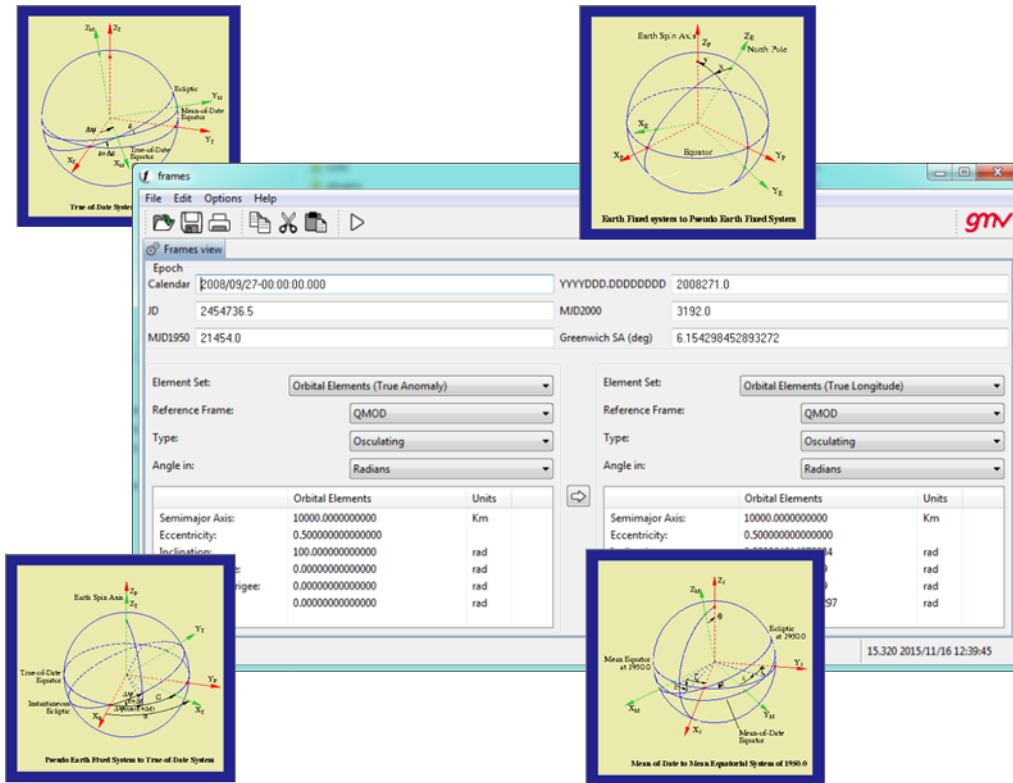


Figure 10: frames: Reference frame conversion tool user interface

4.5. EVENTS LOGGER

focusvelog is a generic, flexible and extensible events logging facility available as either stand-alone or fully integrated module for all **focussuite** products. **focusvelog** is in charge of receiving, archiving, managing and distributing all information, warning or error messages to operators.

focusvelog implements a robust client-server solution composed of several modules:

- The server module is in charge of receiving the events from the emitters, archive them and broadcast to the clients, on both real-time or on demand. The server is also in charge of handling the alarm system.
- The emitting module is provided as a library that can be invoked by other processes (other **focussuite** components or third party application). It generates events to the Events logger.
- A variety of client modules, one of the most important being the **focusvelog** client.

The following figure shows the relationship between **focusvelog** and **focussuite**.

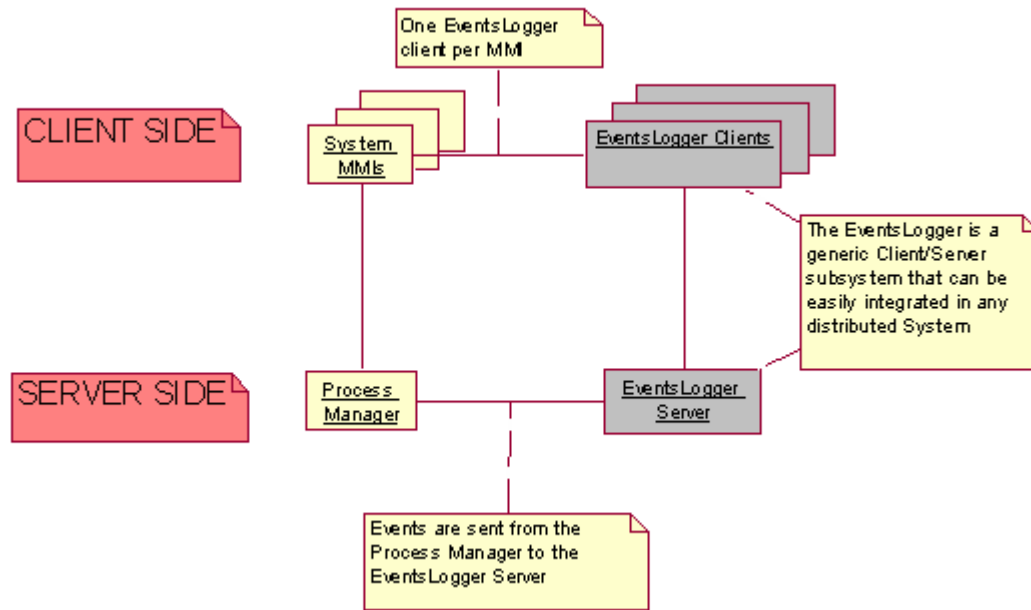


Figure 11: evelog and its relationship with the rest of focussuite

focusevelog provides a user-friendly interface with the ability for handling real-time and archived messages, alarm notifications and acknowledgements.

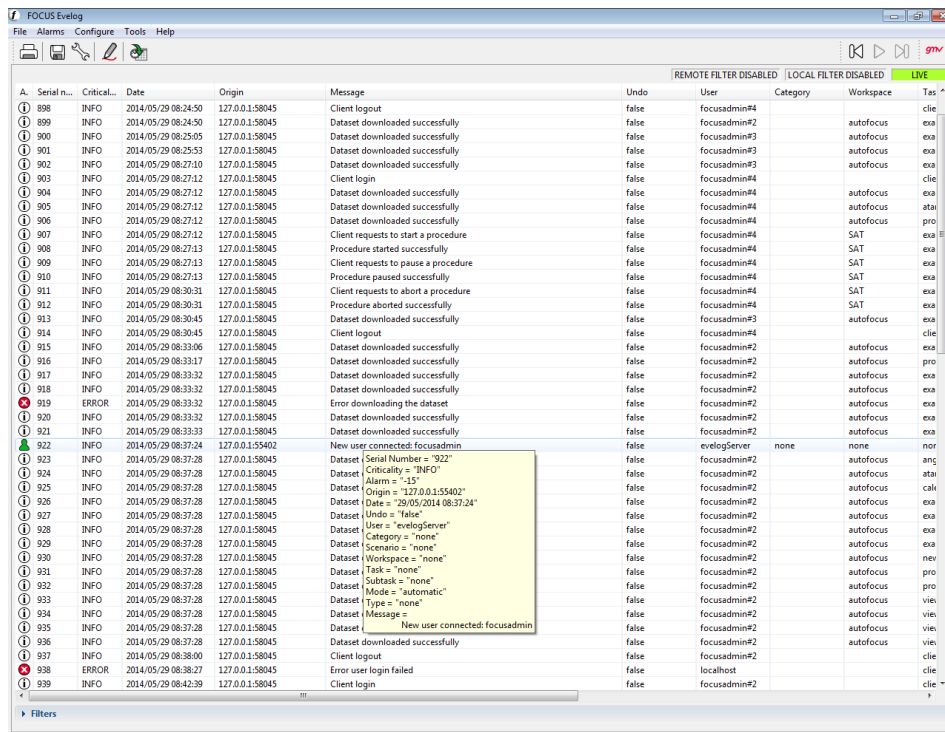


Figure 12: focusevelog client main window

4.6. FOCUSPROCESS MANAGER

The **focusprocess manager** is the main “brain” of the system, centralising all the critical coordination tasks among the different components. Among others, it is responsible for the following tasks:

- **Handle requests for connection** from new clients (which shall include the userid, password, satellite family, satellite, environment and project corresponding to the connection), check the password and establish the connection or reject the request depending on the case.
- **Manage requests from clients for execution of computation layer processes.** Ensure that the client requesting the process is authorised (according to an external table). Assign a process ID. Check conflicting locks on data files. Send corresponding event to all clients acting on the same data area (with a process ID). Prepare the temporary environment, checking out the operational files from the configuration management repository, establishing the needed locks on them and creating the temporary copies on which the process shall act. Launch the process and monitor its execution. Enable the transmission of the output generated by the process to the corresponding client in real time. Check in the updated files in the configuration management repository and update the log of changes entering the ones associated to the process ID (to support the undo operation). Send event with exit status of process to clients acting on the same data area.
- **Receive requests from clients to stop an active process** (identified by its process ID). Check that the process is running and the client is authorised to stop it. Kill process and re-establish operational files in their original state, guaranteeing the integrity of the operational data. Send event to all clients acting on the same data area.
- **Send events to all the clients** corresponding to the relevant data area (combination of satellite family, satellite, environment and project). For this, the **focusprocess manager** needs to maintain a list of active clients and their corresponding data area. For example, if a client runs a process in a given data area, an event such as “Process X run by client Y” needs to be sent to all the clients using the same data area. This shall provide a common events log for all clients in each data area, a feature needed for the purpose of the “undo” function.
- **Send any type of data to a given client.** For example, send files for editing via the user’s interface, whenever it is requested.
- **Update centralised data with data provided by a client:** For example, after the user has edited it and has decided to upload it so that it is stored in the central database.
- **Undo an action:** Check that the process ID corresponds to the last process run in the data area. Disable the possibility of any process execution until the undo is completed. Retrieve the corresponding data changes from the local log of changes. Undo the changes on the data files using the configuration management files, re-establishing the original state before the process was run. Send an event to all clients acting on the relevant data area.
- **Redo an action:** “Undo the undo”

4.7. FOCUSDATA MANAGER

The **focusdata manager** is the component that handles the data in **focussuite**. This component receives the requests for data access from the **focusprocess manager**, as needed depending on the needs of active processes and user requests (e.g., data edition).

The data is organised in a tree tier structure as follows:

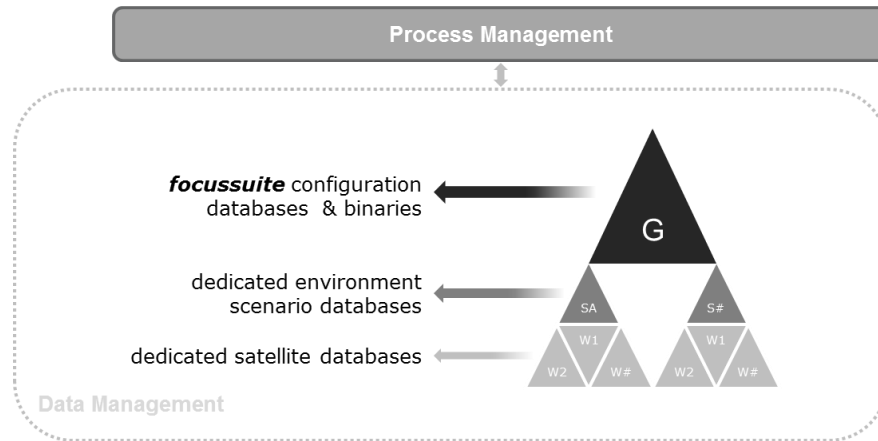


Figure 13: focussuite Manager Scenarios & Workspaces diagram

- A set of scenarios exist for a given **focussuite** server. A typical use of the scenario concept is to define a different one per environment (operational, simulation,). Another option is to define one per satellite platform, typical for operators that own big fleets.
- Each scenario includes a set of workspaces. A typical use of the workspace concept is to define a different one per satellite.
- Each workspace includes a set of datasets. A dataset is a group of data that are related somehow. In the standard products in **focussuite**, a dataset maps to a file of any type (e.g. a plain ASCII file, a postscript file, an XML file, etc.) or a structured file in a standard format. An optional extension of the system, enabled by its overall architecture, is to plug a commercial relational database manager into the **focussuite manager**.

When the user logs into the system, this can access only data from a single workspace. However you can have several clients open at the same time on a given machine, each working on a different workspace. The active scenario and workspace are shown in the title bar on the client GUI.

The **tree tier structure** makes it possible to define functions or data at a level higher than the workspace (e.g. scenario, or even global). They will be shared by all the workspaces under that branch. Therefore a certain function may be defined at the highest level (global), which means that all the workspaces in the system will share it.

Given that the system allows multiple users working at the same time, a system of locks is needed to avoid conflict due to concurrent access. This means that the system may tell you at some point that you cannot edit a certain dataset or you cannot run a certain process because of conflict with on-going tasks. For example, whenever there is another user editing that dataset or running a process that will modify it and therefore has set a lock on it.

4.8. COMMUNICATIONS LAYER

The purpose of this general-purpose library **focuscomm** is to encapsulate the communications between different components.

The functions available at **focuscomm** are grouped in two interfaces:

- **focuscomm-Client:** This interface includes all the services needed by any client component to start and stop a connection, send messages, upload and download files, ping the server, etc.
- **focuscomm-Server:** This interface includes all the services needed by any server component to be started or stopped, process incoming messages, etc.

5. FOCUSASPECT

focuspect client is a completely integrated application that provides a real time system for the on-line acquisition, processing and display of satellite telemetry. **focuspect** client provides the monitoring capabilities through graphical and alphanumeric displays arranged in multiples tabs.

After successfully login for a valid session the main window appears. It is observed that **focuspect** Client has been designed for having 'all in one' working area and the overall HMI has been carefully designed to allow a great degree of operations efficiency and reliability.

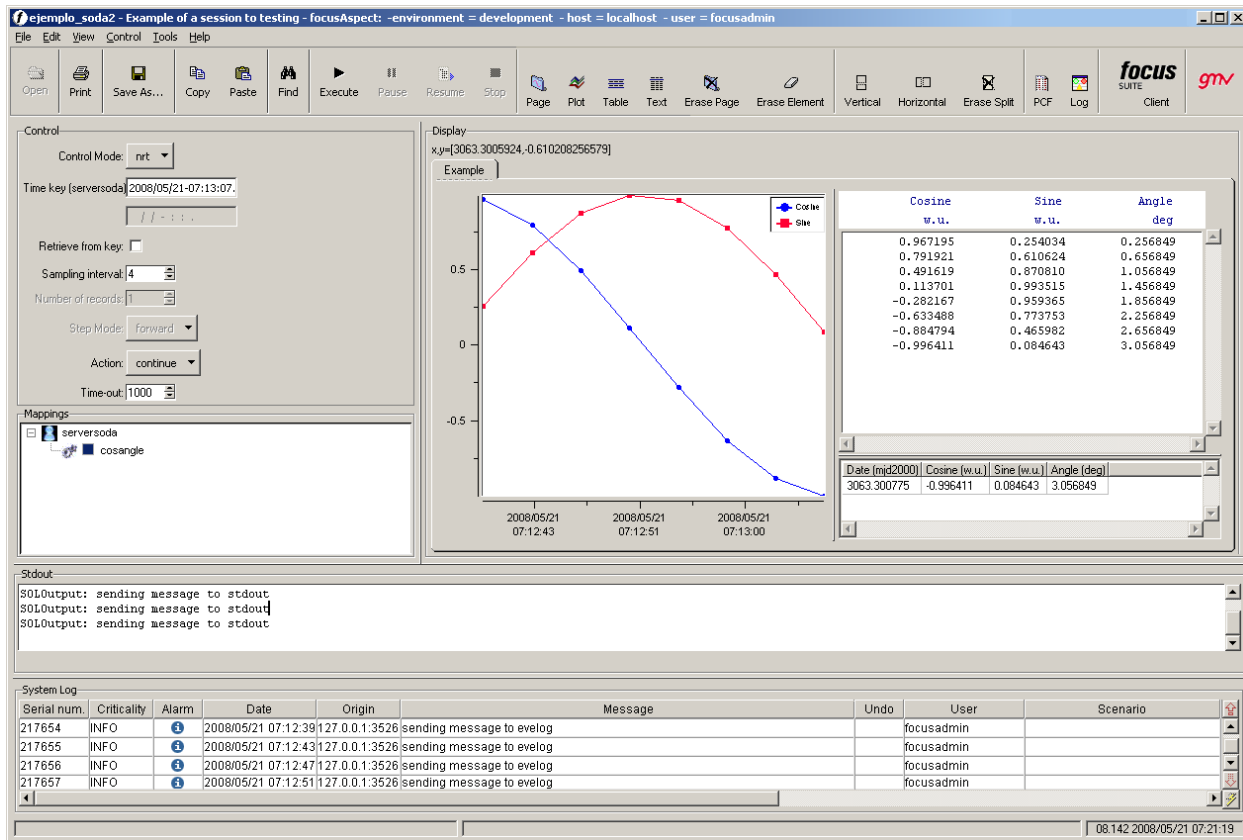


Figure 14: **focuspect** Client main window

6. VISUALFOCUS

visualfocus is a member of GMV's **fofocussuite** set of products and it is aimed to display the evolution of the satellite mission in realistic 3D environment and over a 2D earth projection.

visualfocus displays the flight of one or more satellites in real or simulated time. The configuration of the different displayed elements, input orbits and some more information is obtained from a configuration file stored in the **visualfocus** server.

Configurations are organised in scenarios. Each scenario defines the available elements; common to all the configurations within it that can be combined to define various configurations. Common elements comprise the definition of the available orbit files, satellites, sensors, maps, etc.

The **visualfocus** server stores the scenario and configuration files that are served to the remote clients upon request.

The **visualfocus** clients should first login to one of the available **visualfocus** servers providing the appropriate user and password values.

When the client is logged, the user can build or modify any configuration in any scenario using the available elements in that scenario database (satellites, sensors, orbits, maps, etc), this configuration are automatically saved in the server so other users may re-use them.

When the configuration is ready, the user launches the graphical tool, and all the files required are downloaded from the server side to the client (only missing or changed files are downloaded) and the graphical displays are launched.

The time can be scaled on the client side to make the simulation run faster or slower than real time (To run at real time speed, use time factor scale 1)

visualfocus is composed of the following tools:

- **2D visualization tool:** It shows the ground-track of all active satellites with respect to a certain projection of an Earth map. Ground stations and their visibility areas can be shown. Many features are provided to let the user control the display.
- **3D visualization tool:** This tool shows a three-dimensional view of the Earth and all active satellites, their orbits and ground-tracks, their sensor's footprint, ground stations and celestial objects. The user can control the position and configuration of the camera and activate the display of certain vectors, lines of sight, penumbra/umbra cones in eclipses, day/night zones, etc.

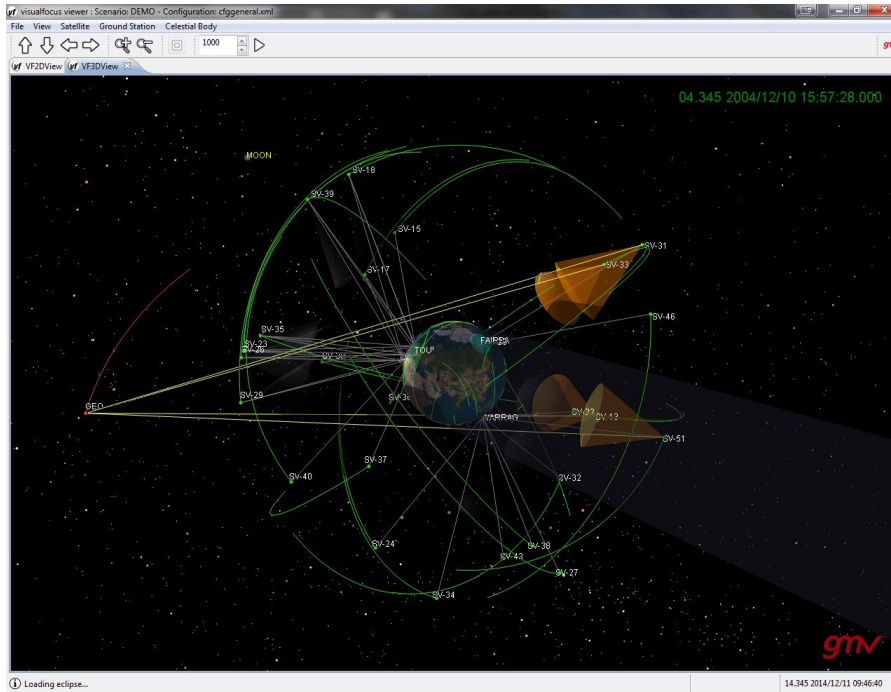


Figure 15: visualfocus 3D Window

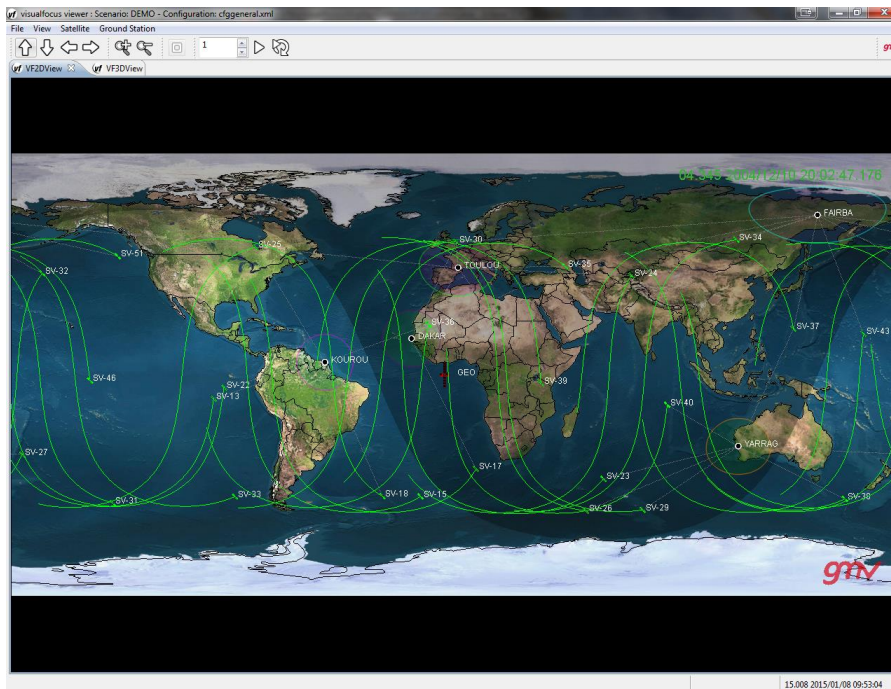


Figure 16: visualfocus 2D Window

All of them are synchronized to show the evolution of the mission from different points of view.

visualfocus has two data input modes:

- **File mode:** input data must all be stored in a file prior to presentation. Situation and name of the file will be provided to **visualfocus** and the tool will gather all the data to monitor the satellite mission.
- **NRT mode:** input data will be continuously provided via a socket or pipe with the option to enter data through a command interface, to modify dynamically scenario settings.

Among other features **visualfocus** supports:

- CDM and OEM formats that allows covariance ellipsoid representation (**focuscloseap**)

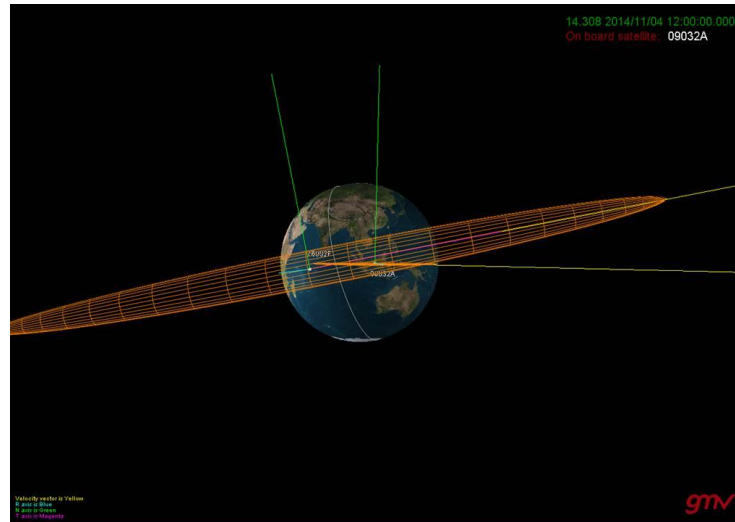


Figure 17: **visualfocus** covariance ellipsoid representation

- Connection with a TM server, to visualize TM data in Real Time
- Mobile elements: solar array rotation and thruster animation.

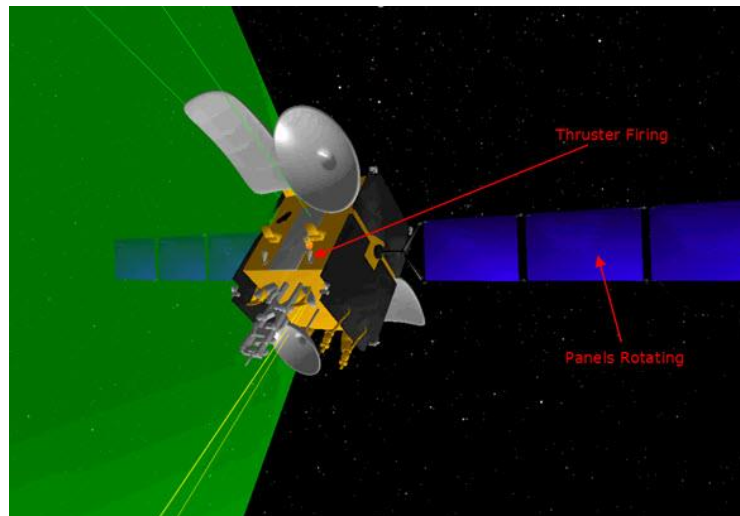


Figure 18: **visualfocus** 3D representation in real time including mobile elements

- Grid sensors: contour lines footprint representing the antenna radiation.

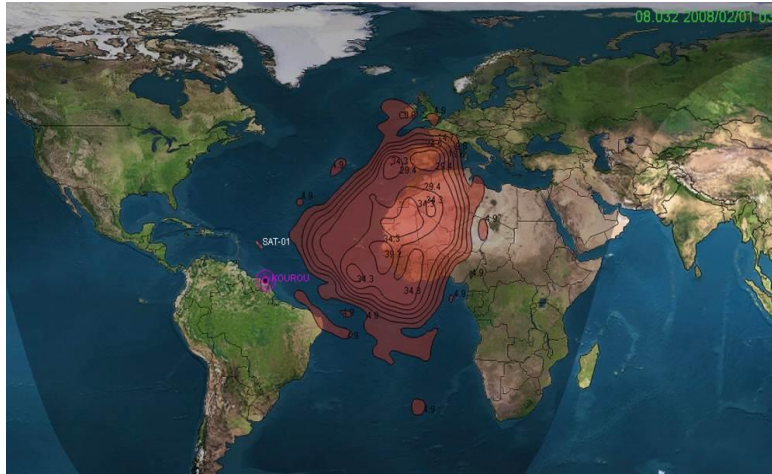


Figure 19: *visualfocus* footprint for antenna radiation representation

- New Interface to retrieve TLE data from a database.

7. FOCUSUITESYNC

GMV offers the **focusuite** servers synchronization tool (**focusuitesync**) to maintain the servers with data aligned. **focusuitesync** is a client-server application:

- The server (**focusuitesyncserver**) schedules and executes backups of **focusuite** data as well as restores of these backups. Its main usage is to synchronize data between different servers, performing the backup on master server and restoring it in the slave server. The same backup can be sent to and restored in several slaves. It can be executed as a daemon.
- The client (**focusuitesyncclient**) is used to configure the server tasks. It provides a Graphical User Interface to configure the tasks to be executed in **focusuitesync** servers. It works as an external application that connects to servers. The following picture is the **focusuitesyncclient** main window, which is comprised of a menu bar, a toolbar and a working area.

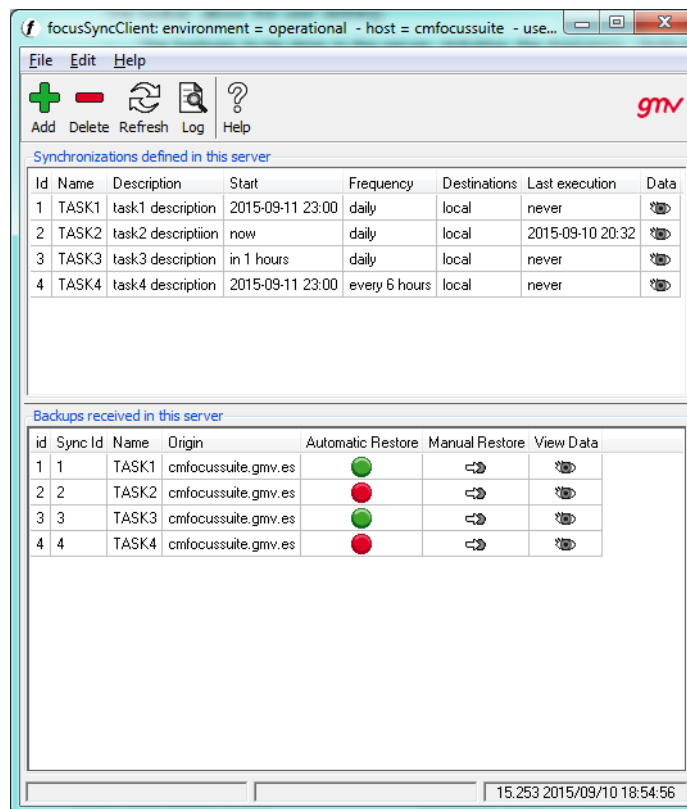


Figure 20: **focusuitesync** Client Main Window

The toolbar allows the user defining:

- The backups to be done in the server, including the frequency, destinations (target servers), time of the last execution and data to be considered.

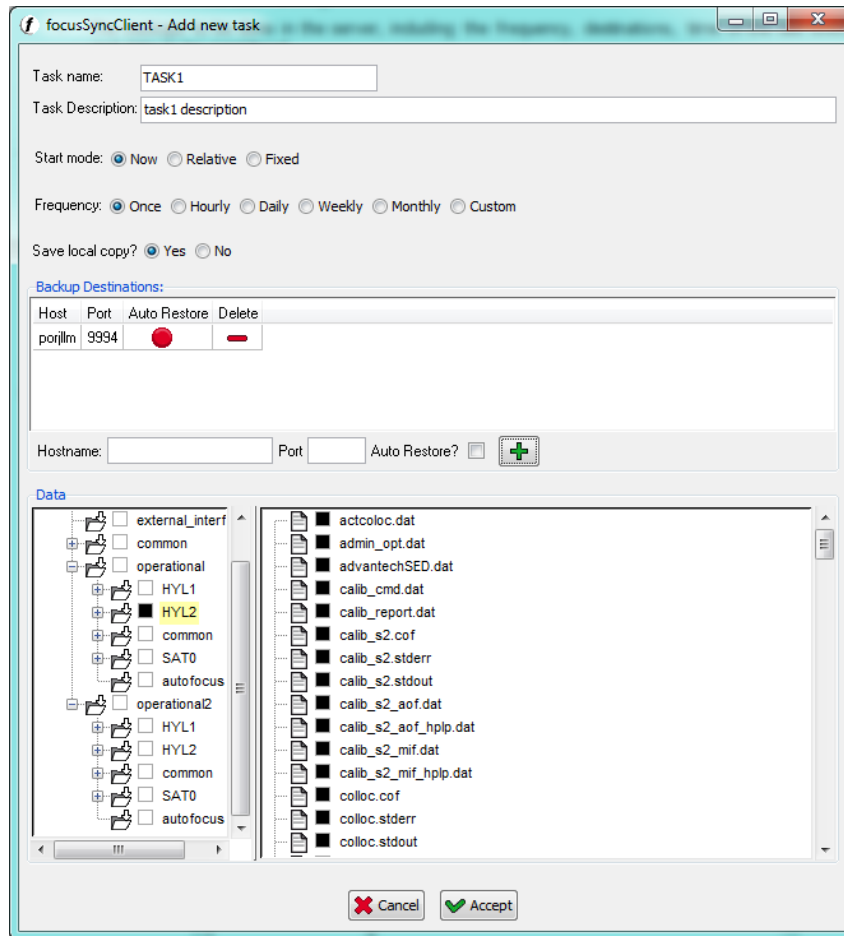


Figure 21: focussuitesync Client Main Window

- The backups received in the server, including the origin, an automatic restore option or manual restore; and the data to be restored.

8. FOCUSCLOSEAP

Due to the intensive activities in the space during the last half century, the **population of man-made space** objects is playing an increasingly relevant role in the space environment. The threat the space debris objects pose on operational satellites is becoming a serious problem.

In order to evaluate the **collision risk** of a satellite against space debris two main analyses are needed. In first place, all **close conjunctions** among the active satellite and all the objects from a catalogue of space debris (one vs. all scenario) must be **detected**. A close conjunction is defined as an encounter in space between a **target**, typically an active satellite, and a **chaser**, normally an object from the debris catalogue, where the **minimum distance** between both objects reaches a value lower than a specified threshold, typically of a few hundreds of metres or a few kilometres. In this process, the evolution of the orbits of all space debris objects and the associated uncertainties are inputs of the system. Once all close encounters have been identified, the second step is to evaluate the collision risk, this is, to compute the **probability of collision** based on the uncertainty associated to the orbit in the form of covariances. For these computations there are several **collision probability algorithms** available that have been validated via Monte-Carlo analyses.

The **conjunction detection** process is a computationally expensive process due to the very large and increasing number of objects in space. In order to speed up this process, a series of consecutive filters and sieves are normally used. During the design of such algorithms, factors like reliability and robustness have to be traded off against the performance of the system. The encounter geometry, time of closest approach and miss distance are an output of this process. Nowadays, there is also an interest on monitoring the evolution of the overall collision and fragmentation risk in space. For this purpose, it is also important to predict collisions among all objects (all vs. all scenarios). The computational needs for such a task are extremely demanding and require special techniques to reduce the time required for such computations to an acceptable level.

On the other hand, the **collision probability computation** is a light process, provided covariance information is available. In this case, the challenge resides on the selection and the validation of the right probability algorithms to represent the actual risk with fidelity.

In order to address the growing demand for tools to control the risk of collision with uncontrolled space objects, GMV has developed a specific **focussuite**-based product called **focuscloseap**, which provides operators of satellites with a cost-efficient solution. It shares many of the general features of **focussuite** products described in this document and includes a computational layer that provides all the required functionalities for close approach predictions.

The **computational layer of focuscloseap** is based on the **NAPEOS technology** and inherits all the capabilities and experience gained by GMV during the development of NAPEOS for ESA (NAPEOS is an ESA software mainly developed by GMV). **focuscloseap** implements the well proven algorithms from CRASS, also used in the new Conjunction Prediction System developed by ESA

8.1. FOCUSCLOSEAP OPERATION MODES

Three different sources of orbital information need to be considered for the collision analyses:

- Two Line Elements (TLEs)
- Conjunction Data Messages (CDMs)
- Operational Orbits including planned manoeuvres of the satellite

From this information, it is possible to define different **operation modes** of the new tool in order to cover completely all possibilities:

- **Automated analysis of TLEs vs. operational orbits:** This mode provides some first estimates about future conjunctions with the manoeuvre planning defined by the operator. This process consists of a several-vs.-all process where all satellites under analysis are screened against the complete TLE catalogue of objects. As part of this process, the maximum probability of collision can be computed due to the lack of covariance information for the TLEs.

- **Automated analysis of CDMs:** CDMs are retrieved automatically from JSpOC and analysed by means of several one-vs.-one analyses. As part of this process, the collision probability is computed based on the orbital information provided in the CDMs. This allows basing the decision-making process in the actual probability of collision.
- **Automated analysis of CDMs vs. operational orbits:** Information provided in the CDMs is compared against the operational orbit in order to take into account the planned manoeuvres. This process allows filtering out spurious CDMs that have been generated without considering the manoeuvre planning of the operator. Moreover, it is also interesting to analyse the CDMs without the manoeuvre planning in order to monitor the effect of the manoeuvres on the conjunctions reported by the CDM.

8.2. CONJUNCTION DETECTION

The Conjunction Detection process consists in the detection of all conjunction among a selected list of objects of interest, called targets, and a selected list of potentially colliding objects, called chasers. The list of targets is normally composed by all satellites controlled by a given operator, whereas the list of chasers is normally a complete catalogue of objects. Any pair target-chaser may give raise to a conjunction within a given analysis period. In order to speed up the detection process some filters are implemented. For **closeap**, the so-called **Smart-Sieve** is considered. This method allows detecting very efficiently all encounters in which the chasers crosses a **safety or screening volume** around the target satellite. Additionally, some given pairs of objects may be filtered out upon user configuration in order to avoid conjunctions between objects whose relative orbit is monitored by other means (colocation of satellites).

The result of the conjunction detection process is a list of conjunctions consisting of the two **objects involved**, the **relative geometry**, the **time of closest approach** and the **miss distance** (relative distance at the time of closest approach).

The orbital information of targets and chasers may be used in the form of tabulated ephemerides, propagated numerically from state information at a given epoch or generated from the TLEs with the SGP4 theory. In a typical scenario, the orbital information of the targets is provided by means of tabulated operational ephemerides including planned manoeuvres, whereas the orbital information of the chasers is provided in the form of TLEs. This scenario corresponds to the operation modes "Automated analysis of TLEs vs. operational orbits". In the case of the other operation modes, which involve the use of CDMs, it is not necessary as the conjunction. All the information usually computed as part of the conjunction detection process, is already provided in the conjunction warning message.

8.3. COLLISION RISK ASSESSMENT

Once a conjunction has been detected, either through a conjunction detection process, or due to the availability of a conjunction warning message, it is possible to evaluate the collision risk, this is, to compute the collision probability associated to the conjunction. The computation of the collision probability usually requires the availability of covariance information. This information is provided through the CDMs, but is not available for the TLEs. For this reason, a separate metric, called maximum collision probability, is computed in the absence of covariance information. This metric provides a maximum value of the collision probability based on the relative geometry of the approach. Two metrics can be computed to evaluate the risk posed by a given conjunction

- **Collision probability** when covariance information is known valid for short and long encounters
- **Maximum collision probability** when covariance information is unknown

Once the probability of each conjunction is computed, it is possible to determine which conjunctions are dangerous, this is, that do pose a real thread to the satellites. This is achieved defining thresholds for the miss distance, the maximum collision probability, and the collision probability.

8.4. RETRIEVAL OF INPUTS

The following information needs to be retrieved from external sources through the Internet:

- Two Line Elements (**TLEs**) for the complete catalogue of objects from space-track.org
- Conjunction Data Messages (**CDMs**) for a specific group of satellites from space-track.org
- But also, the operational orbital information, including planned manoeuvres, of the satellites under consideration are also inputs of the system. These are only available within the Control Centre and thus means to make it available to the application must be found.

Additional auxiliary information that is needed for the handling of different time scales and reference frames and for the use of solar flux information are:

- **Leap seconds** from USNO's publicly available ftp or http server (maia.usno.navy.mil)
- Earth Orientation Parameters (**EOPs**) from USNO's publicly available ftp or http server (maia.usno.navy.mil)
- **Solar flux reports** (RSGAs) from NOAA's publicly available ftp server of (ftp.swpc.noaa.gov)

8.5. GENERATION AND SUBMISSION OF OUTPUTS

Once all the information has been computed and generated, the outputs are created containing all the information relevant to the operator. Two elements are defined:

- **Conjunctions report**, which contains the list of conjunctions detected in the different operation modes of *closeap*
- **Conjunction messages**, which contain detailed information about each of the dangerous conjunctions
- **Conjunction plots**, which provide graphical representation of the conjunction geometry, probability, evolution over time.

Both the Conjunctions report and the Conjunction messages are sent via email to a configurable list of users. However, the Conjunction plots are not submitted via email to the operator, but are available through the **focussuite HMI** for the operator to access them when necessary.

The E-mail Conjunction Report includes the following information:

- TCA and age of the orbit data
- Miss distance
- Radial, Along-Track and Cross-Track distance
- Probability of Collision
- Maximum probability of Collision

CONJUNCTION DETECTION REPORT

Execution Time: 2015/05/18-10:39:09.239
 Report Interval Start: 2015/05/18-00:00:00.000
 Report Interval End: 2015/05/28-10:39:09.239
 Number of detected conjunctions: 8 Conjunctions

Id	Target	Chaser	Target Src/Age	Chaser Src/Age	TCA	Miss Dist (km)	Collision Prob	Max Collision Prob	Rad Dist (km)	Along Dist (km)	Cross Dist (km)	High Risk
520	SAT1	SL-12 R/B(2)	OPS/ 1.9	TLE/ 1.1	2015/05/20 -16:46:07	28.562	1.014 E-022	2.965 E-004	18.771	-21.087	-4.333	
523	SAT3	ARIANE 44LP DEB	OPS/ 1.9	TLE/ 3.0	2015/05/23 -21:16:20	1.384	2.650 E-009	6.119 E-003	1.348	0.157	0.273	X
			OPS/ 1.9	CDM/ 0.7	2015/05/23 -21:16:23	1.760	1.405 E-052	2.078 E-003	-1.705	-0.191	-0.395	X
			CDM/ 0.7	CDM/ 0.7	2015/05/23 -21:16:22	1.959	2.827 E-070	6.030 E-004	-1.934	-0.255	-0.179	X
524	SAT5	ARIANE 44LP DEB	OPS/ 1.9	TLE/ 3.0	2015/05/23 -21:16:38	23.061	1.579 E-009	3.672 E-004	20.607	1.504	10.242	
			OPS/ 1.9	CDM/ 0.7	2015/05/23 -21:16:41	20.015	0.000 E+000	1.827 E-004	17.539	1.149	9.575	
525	SAT2	RADUGA 21	CDM/ 1.1	CDM/ 1.1	2015/05/27 -12:06:02	1.500	2.144 E-006	1.545 E-003	-1.289	-0.722	-0.262	X
223	SAT4	RADUGA 17	OPS/ 0.2	TLE/ 0.4	2015/05/28 -01:16:22	2.630	1.760 E-009	3.220 E-003	2.515	0.263	0.726	X

Figure 22: *closeap* Conjunction detection report

9. AUTOMATION (**AUTOFOCUS**)

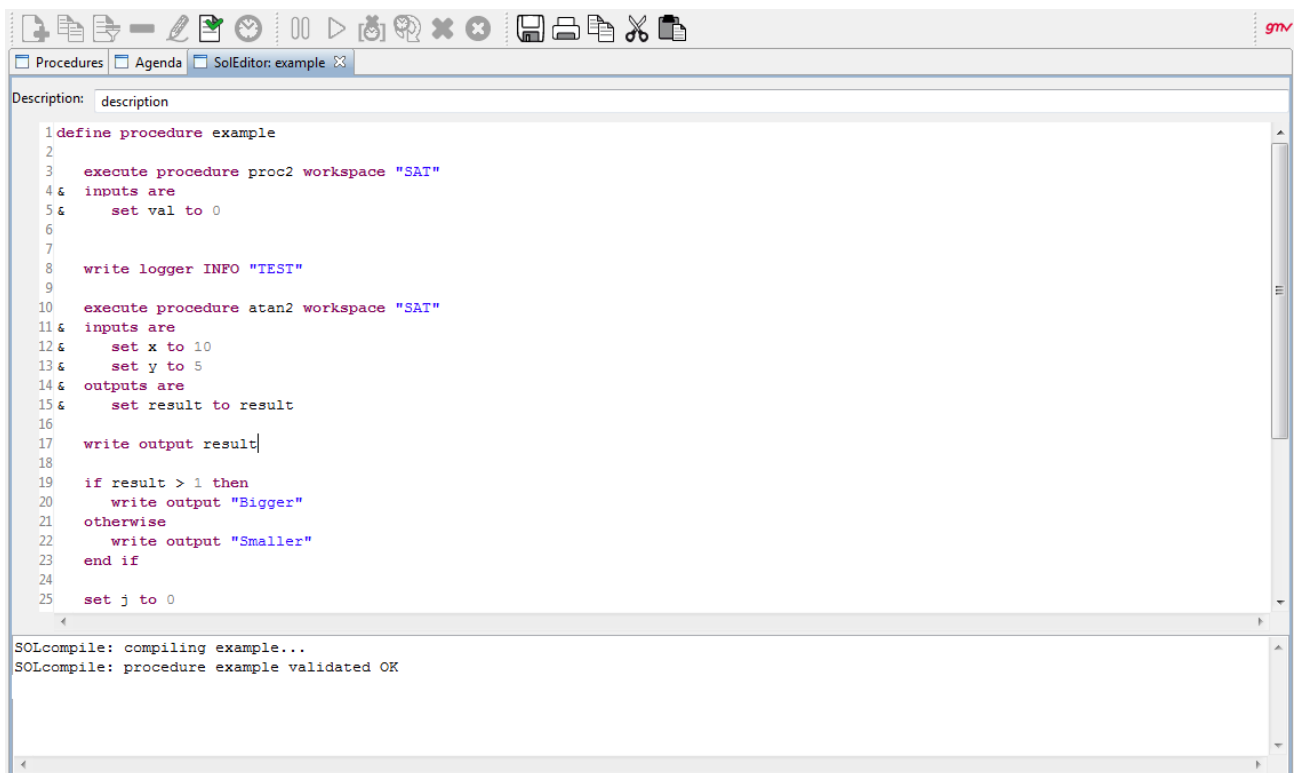
autofocus is an advanced component that delivers full automation support for hands-off operations to the entire **focussuite**. The solution is based on an integrated set of tools for the definition, scheduling, execution, monitoring, and control of high-level user-defined operational procedures.

autofocus is fully compatible with today's operations based on procedures, but assists the human operator by means of an agent that handles procedures written in **SOL**, the spacecraft operations language. **autofocus** provides two user environments,

- one for procedure definition (the **SOL** factory),
- another one for procedure execution (the procedure agenda).

9.1. SOL: SPACECRAFT OPERATIONS LANGUAGE

SOL is a GMV automation language specifically designed to support spacecraft operations. Oriented towards spacecraft operators (not programmers), **SOL** is a very high-level language with natural language-like syntax in which the number of language elements has been minimised.



The screenshot shows the SOL Editor window with a toolbar at the top and a text area containing the following SOL code:

```

1 define procedure example
2
3   execute procedure proc2 workspace "SAT"
4 & inputs are
5 &   set val to 0
6
7
8   write logger INFO "TEST"
9
10  execute procedure atan2 workspace "SAT"
11 & inputs are
12 &   set x to 10
13 &   set y to 5
14 & outputs are
15 &   set result to result
16
17  write output result
18
19  if result > 1 then
20    write output "Bigger"
21  otherwise
22    write output "Smaller"
23  end if
24
25  set j to 0
  
```

Below the code editor, a status bar shows the compilation output:

```

SOLcompile: compiling example...
SOLcompile: procedure example validated OK
  
```

Figure 23: autofocus SOL Editor

SOL is a procedural language which features numeric, text, Boolean and date (relative and absolute) data as well as list handling.

A rich set of mathematical (trigonometric and hyperbolic functions, logarithms, power, etc.) and logical expressions is available in **SOL**. Date expressions and arithmetic operations are also supported, such as in

SOL features the ability to call procedures from within other procedures. Procedure call, via the execute construct, allows to explicitly enumerating the inputs and outputs of a procedure for improved legibility:

SOL control structures include branching via `if/otherwise/end` if and flexible looping: repeat a block of instructions a fixed number of times (`repeat for/end repeat`), repeat while a condition is true (`repeat while/end repeat`) and iterate over the elements of a list (`repeat for each/end repeat`). For example:

SOL supports the whole **focusuite** for full **focusuite** framework automation. It is possible, for example, to activate external **focusuite** components from a **SOL** procedure, collect its output and process it afterwards.

9.2. AUTOFOCUS PROCEDURE DEFINITIONS

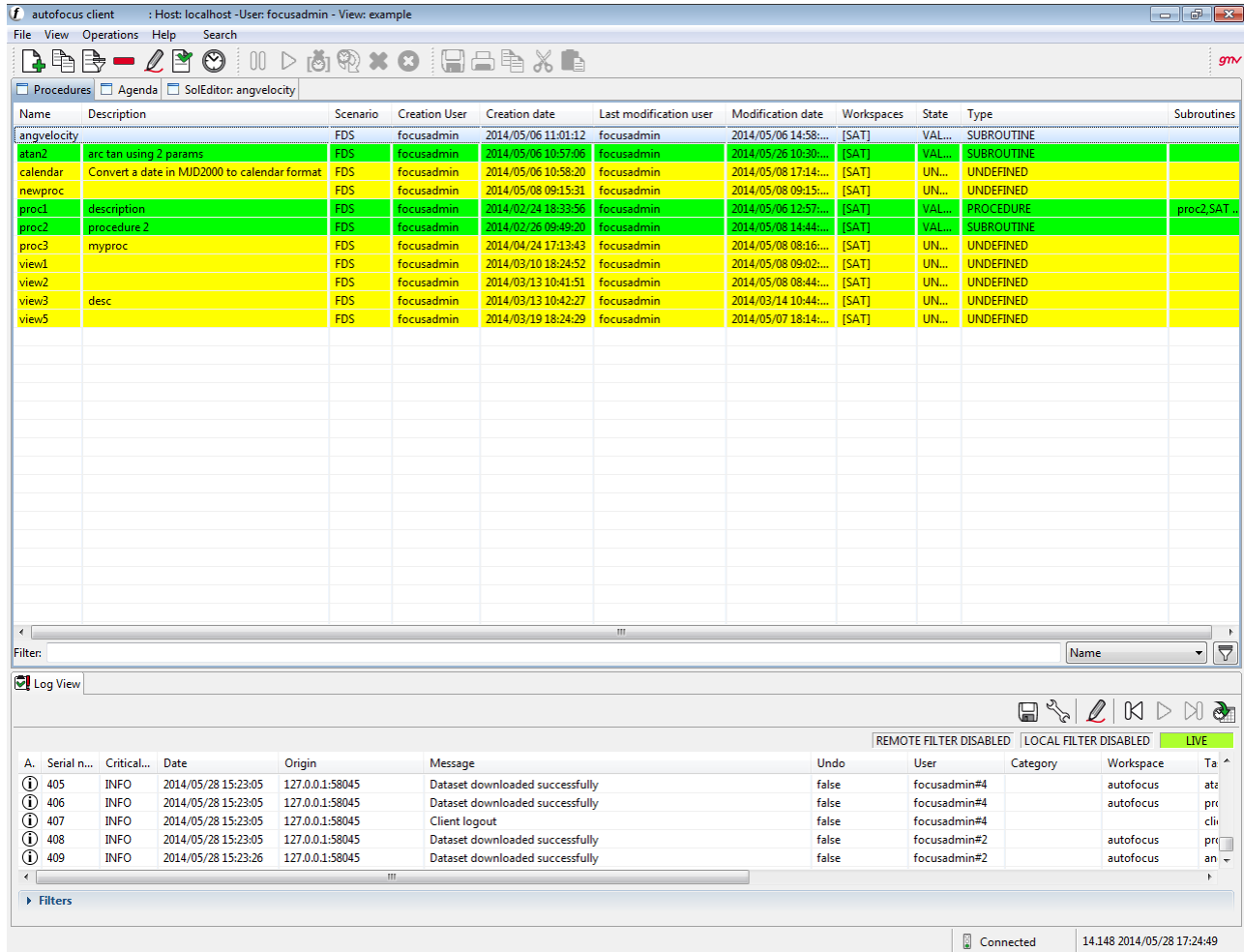
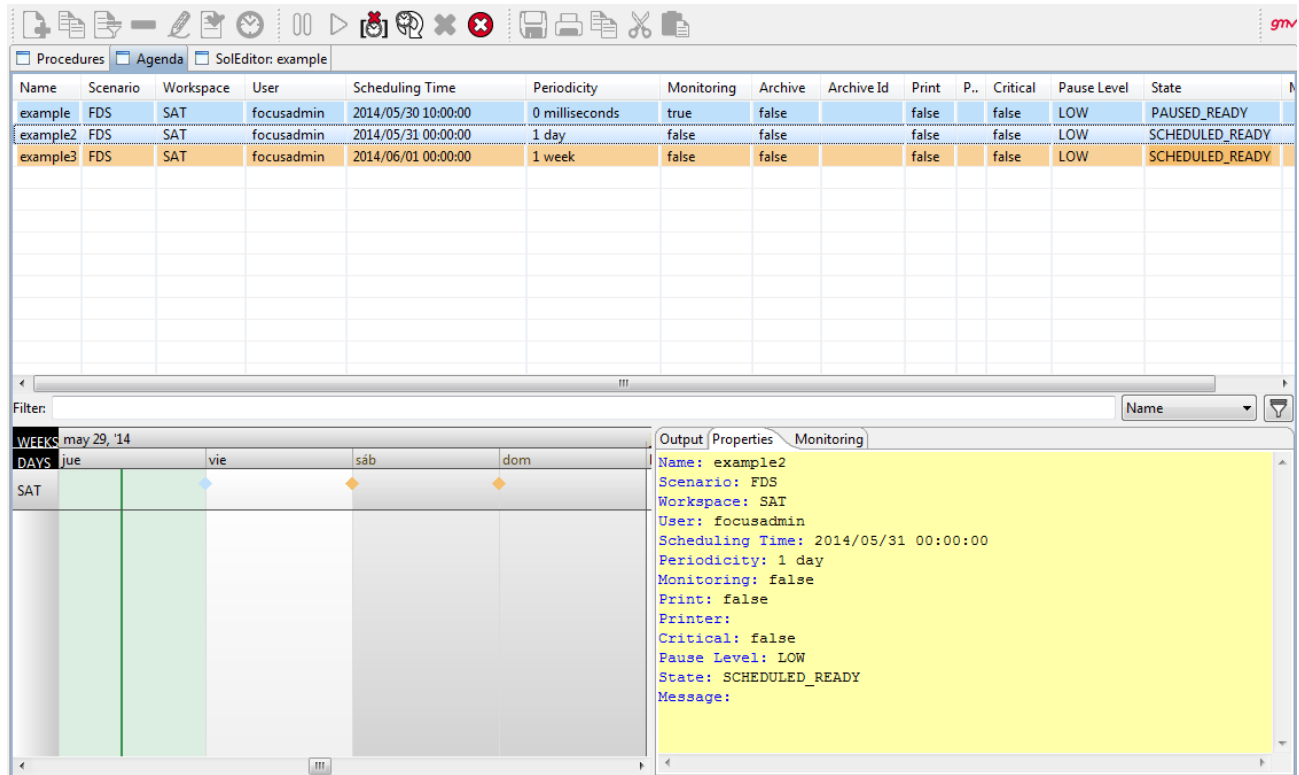


Figure 24: autofocus procedures editor window

The **autofocus** integrated procedure definition environment allows the user to write and validate procedures at syntactic and semantic levels.

9.3. AUTOFOCUS PROCEDURE EXECUTIONS

The **autofocus** agenda allows scheduling, monitoring and controlling **SOL** procedures.



Name	Scenario	Workspace	User	Scheduling Time	Periodicity	Monitoring	Archive	Archive Id	Print	P..	Critical	Pause Level	State
example	FDS	SAT	focusadmin	2014/05/30 10:00:00	0 milliseconds	true	false		false		false	LOW	PAUSED_READY
example2	FDS	SAT	focusadmin	2014/05/31 00:00:00	1 day	false	false		false		false	LOW	SCHEDULED_READY
example3	FDS	SAT	focusadmin	2014/06/01 00:00:00	1 week	false	false		false		false	LOW	SCHEDULED_READY

```

Output Properties Monitoring
Name: example2
Scenario: FDS
Workspace: SAT
User: focusadmin
Scheduling Time: 2014/05/31 00:00:00
Periodicity: 1 day
Monitoring: false
Print: false
Printer:
Critical: false
Pause Level: LOW
State: SCHEDULED_READY
Message:
  
```

Figure 25: autofocus procedures agenda window

Validated procedures can be scheduled immediately, at a given time (either absolute or relative) or by means of links to other procedures (soon before, late after, right before, etc.). The **autofocus** scheduler takes care of the execution of the procedures when their activation condition becomes true. The list of scheduled procedures can be viewed in both a full mode, where all scheduling details are available, or in a Gantt chart via an integrated **ebro** component.

10. COMPUTATION LAYER

The family of *focussuite* products is composed of:

- *focusgeo* for on-station phase of GEO satellites
- *focusleop* for GTO phase of GEO satellites
- *focusleo* for low Earth orbit satellites
- *focuscn* for satellite constellations

There exists an extension of the system, called *autofocus* that allows fully automating flight dynamics operations.

Each of the *focussuite* family members (*focusgeo*, *focusleop*, *focusleo*, and *focuscn*) includes its own computational layer providing the required flight dynamics functions for the corresponding mission (GEO, LEOP, LEO or satellite constellations).

10.1. FOCUSGEO COMPUTATIONAL LAYER

focusgeo computation layer is based on the extensive reuse of operational Flight Dynamics Systems that GMV has extended and upgraded for different satellite operators over years. It is originally based on **ESA's PEPSONC** technology. The functionalities embedded in the package are divided in two groups:

The computational modules in *focusgeo* can be divided into two categories:

- **Mission-independent modules:** Modules which do not change from mission to mission. These are generally not customized for a given mission.
- **Mission-specific modules:** Modules which are generally customized for each mission, based on the mission-specific and platform-specific requirements. To avoid providing proprietary information from the manufacturer, only a high level description of the main mission-specific functions is provided here. Full compliance with the Ground Software Requirements Document provided by the manufacturer will be demonstrated as part of the validation.

A description of the *focusgeo* functions is done here below:

10.1.1. MISSION-INDEPENDENT MODULES

PREREAD: Tracking Data Pre-Processing

PREREAD reads the raw tracking data stored in an ASCII history file, smoothes them, and reduces them. The tracking data are converted to physical measurements of distance and angles and appended to the Pre-processed Tracking File where they can be processed by the Orbit Determination process. PREREAD can be used to condense raw measurements or to change the format of the condensed measurements into the *focusgeo* internal format.

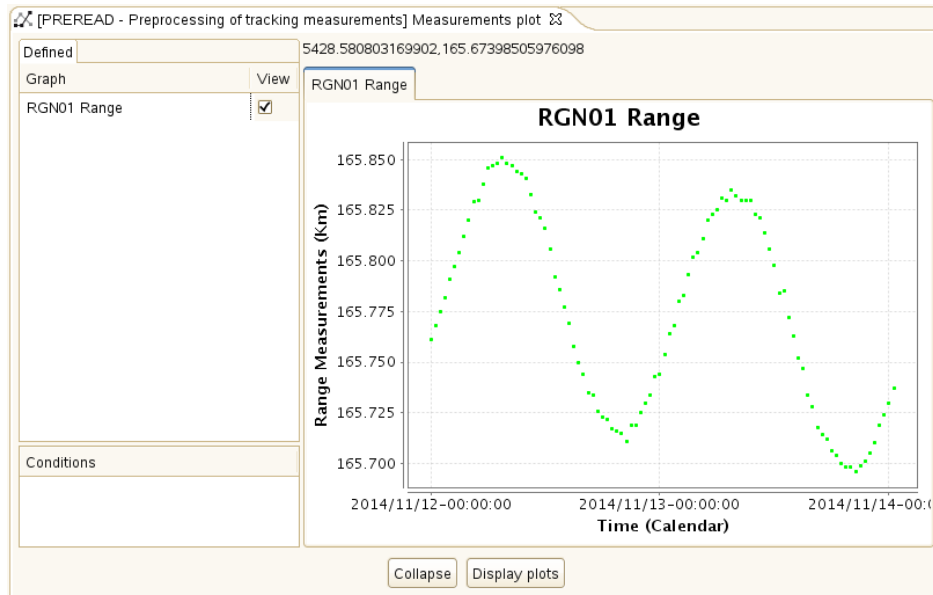


Figure 26: Preread example of graphical output for condensed range measurements

GORDAM: Batch Least-Squares Orbit Determination

GORDAM determines the orbit by applying an iterated least squares fit of a numerically integrated orbit to the pre-processed tracking data. The determined orbit is represented by six orbit elements at an epoch at the beginning of or inside the tracking interval. The residuals are obtained by propagating the orbit by numerical integration of the equations of motion in inertial Cartesian co-ordinates using an Adams-Bashforth/Adams-Moulton prediction-correction method of order 8. The measurement residuals used in the orbit determination can be plotted.

GORDAM processes measurements from multiple sites, as available in the measurements file, computing their best fit in order to adjust the orbital parameters needed to perform a short-term orbit prediction. GORDAM reads the station coordinates and other station characteristics from the Station Database. This module also calibrates manoeuvres already performed based on tracking data. Any manoeuvres that are in the manoeuvre files and fall within the time interval specified will be read by the process and used in the orbit determination. There is an option for determining the size of one or more performed manoeuvre burns in GORDAM, by setting a flag on the burns in question the results are stored in the manoeuvres file and can be used for the calibration of future manoeuvres.

An option is provided in the input allowing the user to request the determination of the solar radiation pressure coefficient in the form of a correction factor. Additional estimation options include transponder delay and ground station bias.

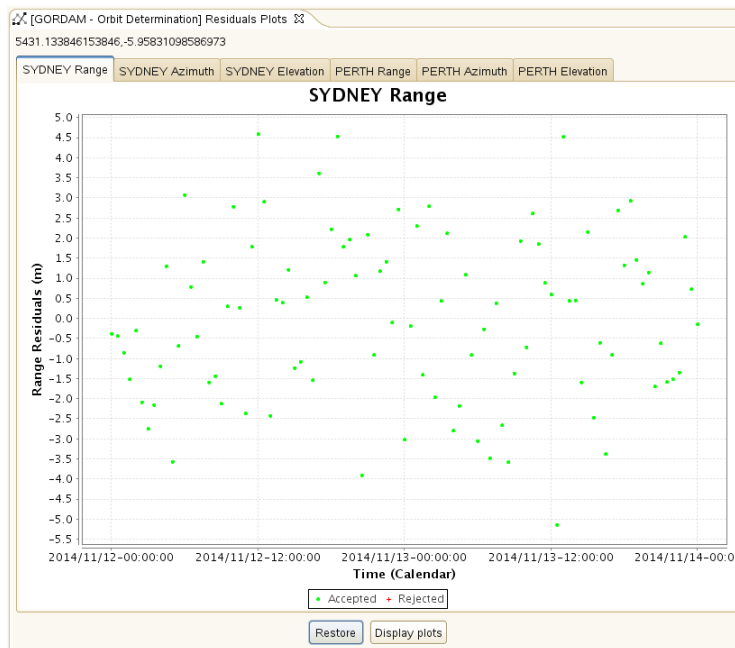


Figure 27: GORDAM Example of Range residual plot

SORBAM: Orbit Propagation

SORBAM computes the evolution of a geosynchronous orbit forward and backward in time. Planned impulsive manoeuvres will be taken into account in the prediction. Half-hourly elements are written to an ephemeris file, and graphical displays are available to show the orbit evolution.

The orbit is evolved in time by numerical integration of the equations of motion in inertial Cartesian coordinates. The time step is half an hour, and an Adams-Bashforth/Adams-Moulton prediction correction method of order 8 is used, together with a Runge-Kutta of order 8 also for initialization purposes, after each discontinuity.

As well as the classical two-body force model, the equation of motion includes also the (perturbing) forces due to the Earth’s sphericity, the Sun and Moon’s gravity, Coriolis from precession of the mean system of date, and the force due to solar radiation pressure.

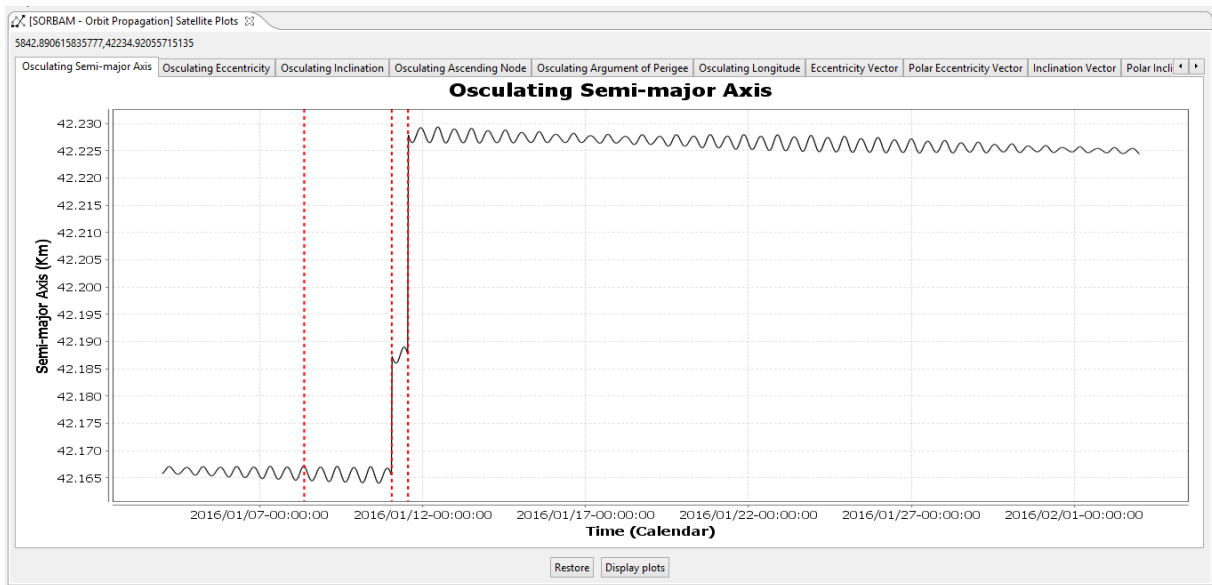


Figure 28 SORBAM Orbital evolution Plots (Osculating semi-major axis example)

LEGO: Long-Term Evolution of Geostationary Orbits

LEGO is a specific orbit propagator for long-term predictions of geostationary and near-geostationary orbits based on averaging techniques. It takes into account the perturbations due to the geopotential, the luni-solar gravitational attraction and the solar radiation pressure. Compared to the full integration of the movement equations taking into account the short-period terms, this orbit propagator gives a quite accurate orbit evolution within a reasonable computer-time and is therefore particularly suitable for parametric study.

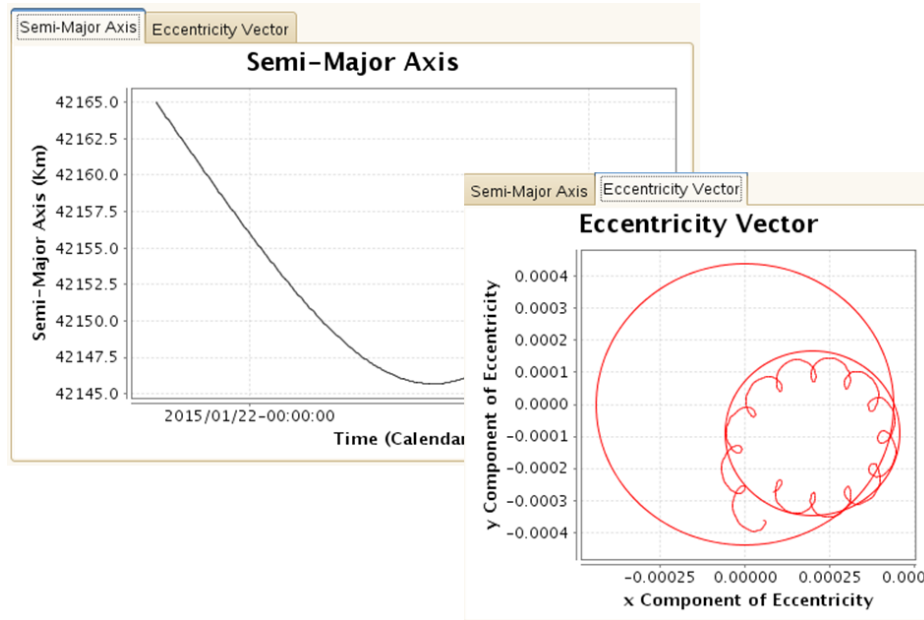


Figure 29: focusgeo: LEGO plots showing long term propagation

SOLONG: East-West Station-Keeping Manoeuvre Planning

SOLONG calculates one, two, or three burn longitude manoeuvres for longitude station keeping with eccentricity control. SOLONG uses the ephemeris file and calculates the execution time and size of the tangential burn according to the options specified in the input file. If multiple manoeuvres are planned then the first and second manoeuvres will be separated by odd multiples of 12 hours. SOLONG includes the capability of supporting longitude manoeuvre planning for collocated satellites.

Two station keeping strategies are available: Sun-pointing perigee and Minimum eccentricity, and follow the control requirements (longitude box, eccentricity control circle), which are defined in the Station Keeping Database. Additional strategies can be implemented. End of life manoeuvre can also be computed by SOLONG for the final de-orbiting manoeuvres to move the satellite to the "graveyard" orbit.

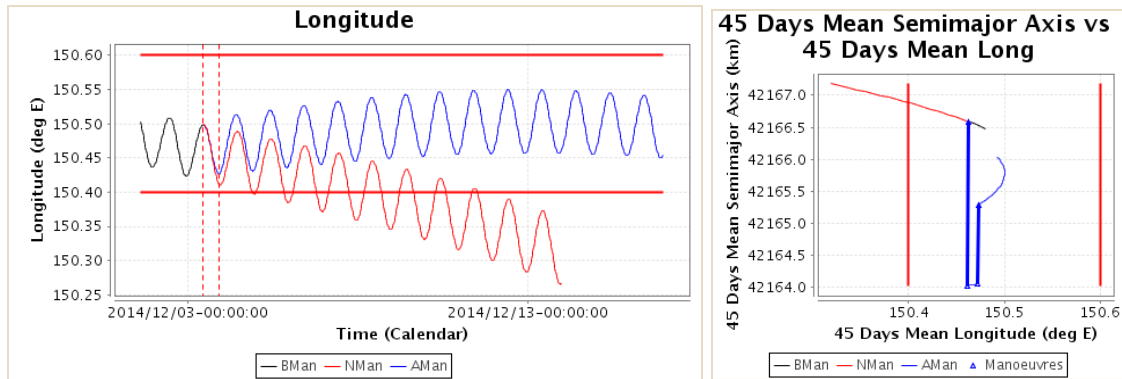


Figure 30: SOLONG Manoeuvre plots

INCLON: Inclination Station-Keeping Manoeuvre Planning

INCLON calculates a North or South manoeuvre thrust for inclination station keeping. INCLON uses the ephemeris file and calculates the execution time and size of the North/South thrust according to the options specified by the user, including the capability of supporting inclination manoeuvre planning for collocated satellites. This capability is accomplished by allowing the user to define the centre of an offset inclination control circle (IX, IY) to be used in the different North/South station keeping strategies supported by INCLON.

There are currently three modes supported for computing the North/South manoeuvre: Target Pole, Pole Motion Direction, and Inclination Symmetry. Additional strategies can be implemented.

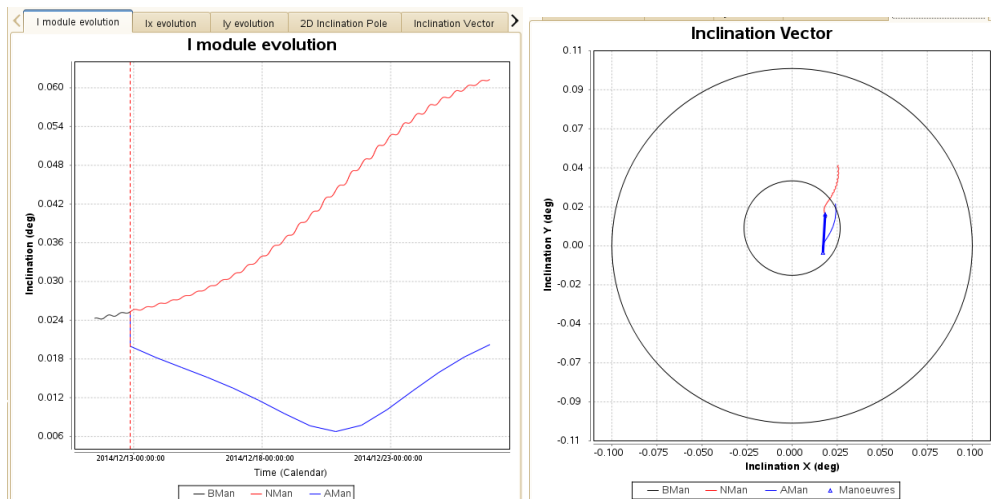


Figure 31: INCLON Manoeuvre plots

INCLEC: Inclination and Eccentricity control

INCLEC computes the N/S continuous manoeuvres required to control the S/C inclination and eccentricity (if required). The main objective of INCLEC is to correct the inclination secular drift to keep the satellite latitude inside the control box. Therefore, based on the user inputs, it computes the required planned manoeuvres (delta-V and midpoint epochs) to be calibrated afterwards by a manoeuvre calibration program. The tool can also compute E/W manoeuvres to control the S/C longitude.

Note that this module is only provided in case of ionic propulsion to satisfy a simultaneous inclination and eccentricity control.

PRODGEN: Products Generation

PRODGEN is the orbital events prediction and products generation function. Using an orbital ephemeris file, an attitude ephemeris file and a manoeuvres file it is able to provide the satellite operator with information concerning:

- satellite position data
- manoeuvre summary
- eclipse details
- apses crossings
- nodes crossings
- IRES sensor blinding times
- Earth collinearity
- station collinearity
- quadrature and orthogonality
- sub-window transitions
- expected range and antenna angles from the ground stations

The events will be displayed in text format, XML, and a Gantt diagram.

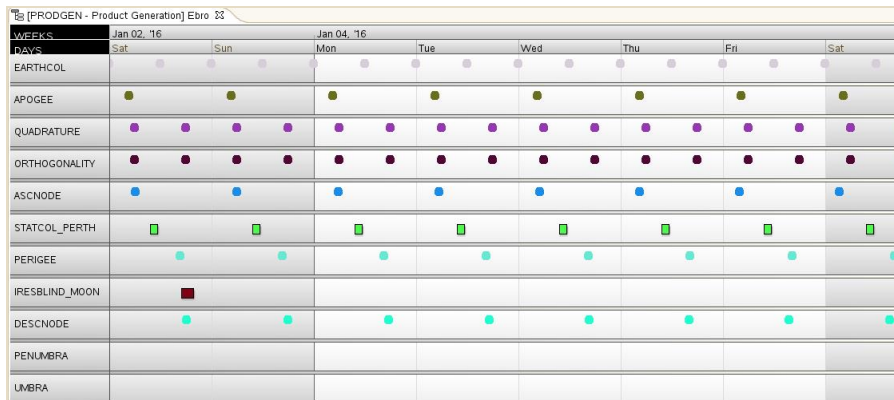


Figure 32: PRODGEN Gantt diagram

[PRODGEN - Product Generation] Xml

TKF/DAT Export:
 (EPOCH) 2015/11/24-23:59:59.000
 (EPOCH) 2015/12/10-01:00:01.000
 (MENU) ORBFILE
 TSS

AOS/LOS_0_RIO_1	2015/11/25-00:00:00.000000	2015/12/10-01:00:00.000000	RIO_1_1
EARTHCOL	2015/11/25-02:14:54.000000	2015/11/25-06:14:17.000000	
ASCNODE	2015/11/25-04:38:17.000000	2015/11/25-04:38:17.000000	
APOGEE	2015/11/25-09:06:56.000000	2015/11/25-09:06:56.000000	
QUADRATURE	2015/11/25-10:14:49.000000	2015/11/25-10:14:49.000000	
ORTHOGONALITY	2015/11/25-10:21:14.000000	2015/11/25-10:21:14.000000	1.50000
EARTHCOL	2015/11/25-14:15:16.000000	2015/11/25-18:14:39.000000	
DESCNODE	2015/11/25-16:35:17.000000	2015/11/25-16:35:17.000000	
PERIGEE	2015/11/25-19:59:18.000000	2015/11/25-19:59:18.000000	
QUADRATURE	2015/11/25-22:14:55.000000	2015/11/25-22:14:55.000000	
ORTHOGONALITY	2015/11/25-22:21:20.000000	2015/11/25-22:21:20.000000	1.50000
EARTHCOL	2015/11/26-02:15:16.000000	2015/11/26-06:14:39.000000	
ASCNODE	2015/11/26-04:31:52.000000	2015/11/26-04:31:52.000000	
APOGEE	2015/11/26-09:19:46.000000	2015/11/26-09:19:46.000000	
QUADRATURE	2015/11/26-10:15:11.000000	2015/11/26-10:15:11.000000	
ORTHOGONALITY	2015/11/26-10:21:37.000000	2015/11/26-10:21:37.000000	1.50000
EARTHCOL	2015/11/26-14:15:40.000000	2015/11/26-18:15:03.000000	
DESCNODE	2015/11/26-16:30:44.000000	2015/11/26-16:30:44.000000	
PERIGEE	2015/11/26-20:06:39.000000	2015/11/26-20:06:39.000000	
QUADRATURE	2015/11/26-22:15:18.000000	2015/11/26-22:15:18.000000	
ORTHOGONALITY	2015/11/26-22:21:44.000000	2015/11/26-22:21:44.000000	1.50000
EARTHCOL	2015/11/27-02:15:39.000000	2015/11/27-06:15:02.000000	

Figure 33: Events XML format

ISATPAR: INTELSAT 11 Parameters Generation

ISATPAR is the INTELSAT 11 elements pointing data generation function. This process produces pointing elements for steering a ground antenna azimuth and elevation automatically. The pointing elements are represented as 11 coefficients, valid for up to 170 hours or until a manoeuvre occurs, and written to a file, which can optionally be transmitted to the station computers.

ISATPAR Interface File

```

DATE
YEAR MONTH DAY HOUR MINUTE SECOND
2012 11 19 00 00 00

THE EPHEMERIS VALUES ARE:

LM0          LM1          LM2
DEG. E       DEG/DAY      DEG/DAY/DAY
41.9175      0.0037       0.000027

LONC         LONC1        LONS         LONS1
DEG. E       DEG/DAY      DEG. E       DEG/DAY
-0.0020     0.0001      -0.0327     -0.0005

LATC         LATC1        LATS         LATS1
DEG. N       DEG/DAY      DEG. N       DEG/DAY
0.0693      0.0012      0.0034      0.0011

THE NOMINAL ORBITAL LOCATION FOR THIS SATELLITE IS 57.0000 DEG. E

THE PREDICTED SATELLITE LONGITUDE AND LATITUDE AT 170 HOURS AFTER
EPOCH ARE 41.9222 DEG. E. AND 0.0689 DEG. N.

```

Figure 34: focusgeo ISATPAR Interface File

COLLOC: Collocation Monitoring

COLLOC is the collocation strategy and proximity monitoring and manoeuvre sensitivity assessment function. The process performs a proximity check for all satellites within a collocation group for the purposes of monitoring the inter-satellite separations in terms of distance and angles and for the purpose of warning against the risk of too close an approach that might cause radio signal interference or a collision.

COLLOC reads the appropriate orbit files and prints the inter-satellite separations for all combinations of pairs of satellites with a warning message if they become too close (when distance falls below specified threshold values). Both text and graphical output is produced including:

- Inter-satellite distance for all combinations of pairs of satellites
- Inter-satellite angular separations for all combinations of pairs of satellites

In addition, the process can assess the impact on the inter-satellite separation (distance and angle) of a planned manoeuvre, either under-performing and/or over-performing by factors specified by the user.

In case the minimum distance threshold is violated a collision avoidance manoeuvre will be computed in the following way:

- Eccentricity manoeuvre will be applied to accomplish the condition that the delta-e – delta-i angle is 0. This is done in such a way that the East/West manoeuvre will achieve an eccentricity target that will be the minimum that avoid a collision. Delta-e and delta-i values are obtained by the differences between the centres of eccentricity and inclination control circles.
- The longitude drift will be kept within bounds by proposing double manoeuvres with exactly the same delta-V in opposite directions.
- Once the collision avoidance manoeuvre has been applied internally, a check if the satellite is kept in the longitude control window is performed. The number of days for longitude window control is taken from the COLLOC main input file. If the manoeuvres cause the satellite to drift out of the window then COLLOC will modify the longitude drift by re-planning the collision avoidance manoeuvres, maintaining the eccentricity target.

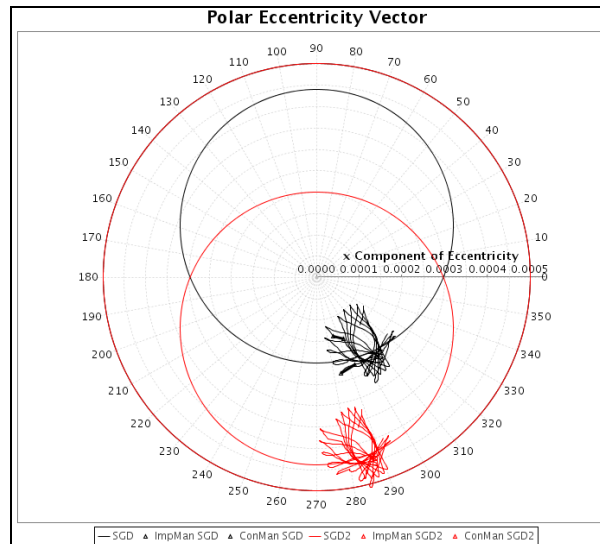


Figure 35: COLLOC Orbit Plots: Eccentricity Vector

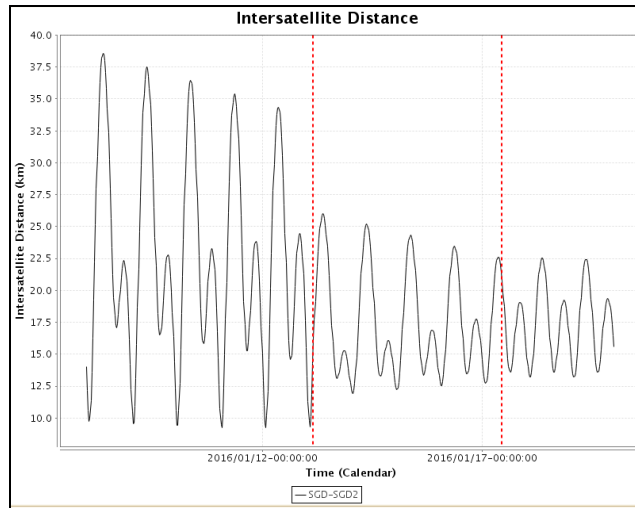


Figure 36: focusgeo COLLOC Inter-satellite Distance Plot

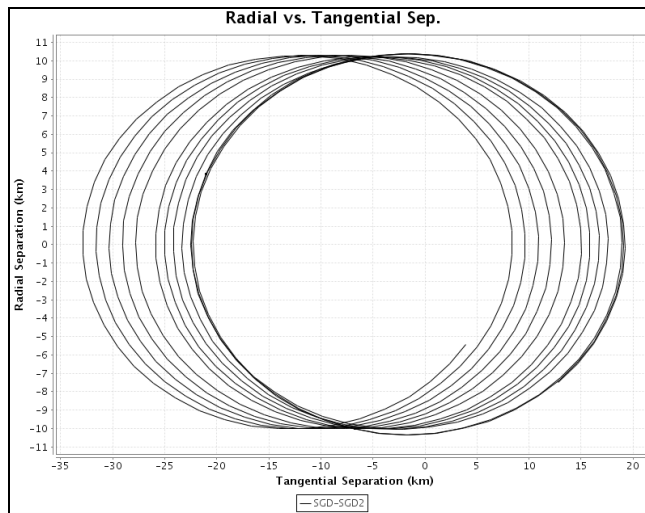


Figure 37: focusgeo COLLOC Collocation Plots

INICOL: Collocation Initialization and Station Acquisition Support

INICOL is the collocation strategy initialization function of **focusgeo**. Three different scenarios are supported:

- New satellite addition to an existing collocation group.
- Satellite removal from an existing collocation group.
- Satellite return to the group from a non nominal situation.

The INICOL program calculates the sequence of manoeuvres to insert a satellite in its on-station window, collocated with other satellites or alone. INICOL computes the manoeuvres to reach the on-station window, avoiding other satellite windows in the path. INICOL also initializes the first cycle of station keeping manoeuvre (East/West and North/South). Finally, there is an option to update the operational manoeuvre file with the manoeuvres proposed by INICOL.

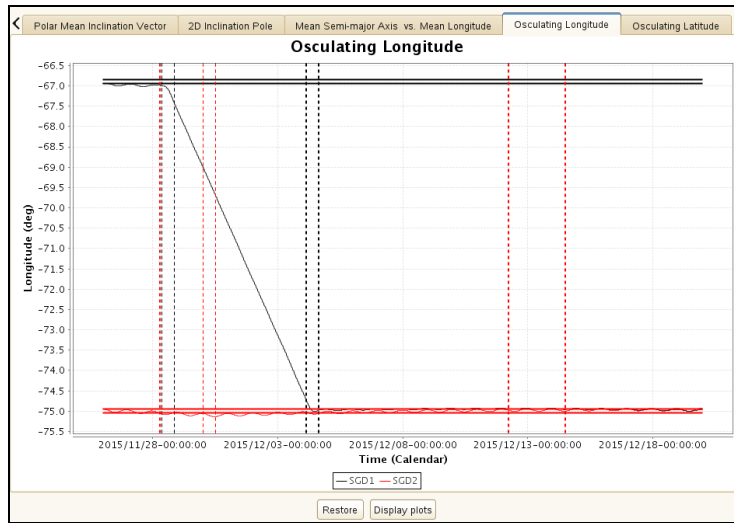


Figure 38: focusgeo INICOL Collocation Insertion Plot (Osculating longitude)

10.1.2. MISSION-DEPENDENT MODULES

Mission dependent modules are platform specific modules, therefore they will be directly based on platform requirements, and nevertheless there are common functionalities that are share for all of them as are described below:

CALIB: Manoeuvre Implementation

CALIB is used to analytically predict the thrusters activity that the satellite on-board system will initiate for a given manoeuvre midpoint time and direction. CALIB models the performance and fuel flow rate of the thrusters based on telemetered measurements and mathematical expressions provided by the manufacturer.

CALIB is run following execution of one of the station keeping functions (INCLON, INCLEC, SOLONG or INICOL). It determines how the thrusters must be fired to reach a delta-v target set by the station keeping function. CALIB generates commanding information including required total impulse in jet seconds, starting epoch for the manoeuvre, and the thrusters involved. At the end of a CALIB execution, the user will have all necessary commanding information to generate a manoeuvre sheet or to send commands directly to the Real Time System through the external interfaces.

ISKM: Ionic SK Manoeuvre Calibration and Preparation

This module is in charge of the computation of the continuous thrust durations and manoeuvre start times as a function of the Delta-V requirements and satellite/equipment configuration.

After having run INCLEC to plan the theoretical N/S manoeuvres (manoeuvre mid-point epoch and delta-v components), ISKM shall be run to compute the manoeuvre start times, durations and other relevant parameters to be commanded to the S/C.

Note that this module is only provided in case of satellite is using ionic propulsion.

MASSEVO: Mass Book-Keeping

MASSEVO is used to compute the propellant mass consumed by satellite on-board and to maintain the fuel book-keeping. MASSEVO will normally be run immediately following a manoeuvre to calculate the amount of fuel mass consumed so that a current estimate of the overall spacecraft mass and mass distribution may be maintained. Given a starting time and a finite interval, MASSEVO uses telemetered burn time totals for the thrusters in combination with the same fuel flow rate model used in CALIB to determine the total fuel consumption over the interval, and updates the stored spacecraft mass estimate to reflect the loss of fuel mass. It also uses the telemetered burn times to generate an estimate of the actual delta-v achieved during the maneuver. It updates a mass evolution file, which is used by other functions in the system to get the current mass of the satellite.

FDSEXPOR: Export to RTS

FDSEXPOR is the FDS program in charge of generating the interface files needed by RTS in order to update certain operational procedures that needs FDS data. After the interface files are generated by FDSEXPOR, they are automatically copied to a designated folder on the RTS server.

DBUPDATE: Database Update

DBUPDATE is the function that updates the historical databases (which contain the information for the entire life of the satellite) using the information contained in the temporary files (which contain the information generated during the last few days of operations, such as the manoeuvres of the station-keeping cycle, the last initial state vector estimated, etc). The DBUPDATE program updates the Measurements File, the Initial State Vector, the Impulsive Manoeuvre File and the Mass Evolution File, by taking the data from these file(s) and transferring it to their prospective database(s). Once the transfers are complete, and the database(s) are updated, the temporary

files are emptied (except for the state vector file). DBUPDATE gives the option to select which file(s) to update. The consistency of the new data is checked to ensure temporal coherence in terms of date and duplication of records.

DBREPORTS: Database Reports

DBREPORTS is intended for the purpose of generating database reports. DBREPORTS generates plots using the historical information stored in the database, for all the selected satellites in a collocation group. The database reports generated are:

- Orbit Plots
- Mass Evolution Plots
- Measurements Plots
- Manoeuvre Plots
- Collocation Plots
- Latitude vs. Longitude Plots
- Various Histograms

Other modules that are usually included as they are often required by satellite manufacturers could be:

MAPREP: Manoeuvre Sheet Preparation

This function generates reports in a flexible way using the information on manoeuvres existing on databases.

HPLP: Satellite Remaining Lifetime Estimation

HPLP is the function of **focusgeo** in charge of predicting precisely the lifetime of the satellites. This prediction is based on regular station-keeping cycles, specifying the execution of both inclination and drift manoeuvres. Additionally, this program provides the capability of introducing relocation and eccentricity manoeuvres. This program simulates the manoeuvres that a geostationary satellite should perform during its life, in order to calculate the propellant mass consumption and, hence, predict the spacecraft lifetime.

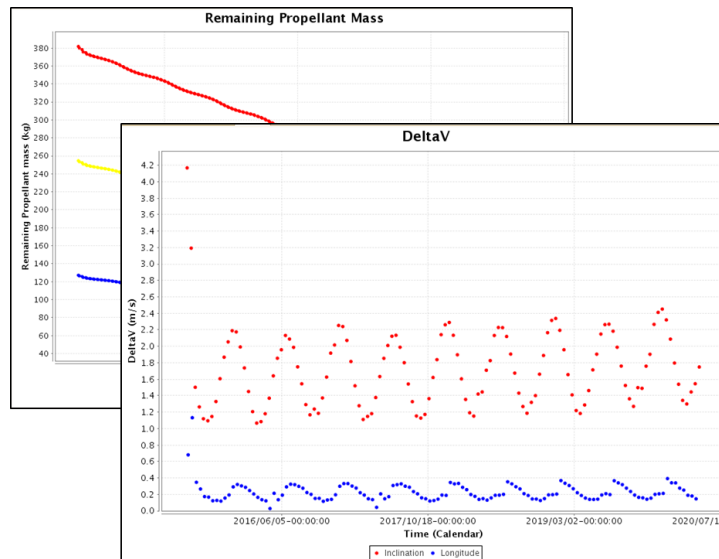


Figure 39: focusgeo: HPLP plots example

PVT: Remaining Fuel Computation

PVT is another function to estimate mass consumption. It is used as a checking tool for MASSEVO. While MASSEVO uses a mass-bookkeeping method of thruster data to calculate mass evolution, PVT uses pressure, density, and temperature to determine current volume and mass of the remaining fuel in the fuel tank.

OBPU: On Board Propagator Update

This module is in charge of computing the required information for updating the inputs used by the on board propagator.

10.2. FOCUSLEO, FOCUSLEOP AND FOCUSCN COMPUTATIONAL LAYER

Their computational layer is based on **ESA's NAPEOS** technology.

10.2.1. METHODS AND MODELS

10.2.1.1. Force Models

The forces that affect the motion of a near-Earth satellite can be separated into gravitational forces and surface forces. In the following, the models implemented in NAPEOS are briefly described. It is to be noted that the algorithms and models have been implemented to make the system compliant with the latest recommendations of the International Earth Rotation Service (IERS) standards.

Gravitational perturbations	
Gravitational Potential of the Earth	<p>The potential function for the Earth is generally expressed in terms of a spherical harmonic expansion and is referred to as the geopotential.</p> <p>The harmonic coefficients are a function of the Earth's mass distribution and are commonly obtained from observing the motion of artificial satellites. In practice, the number of series expansion are taken to a finite number, N, where N may be chosen as a function of the individual satellite orbit, and the required orbit determination accuracy.</p> <p>A significant improvement has been achieved over the last years in the area of the gravity field. Several models have been developed, some of them are: GEM-T1, GEM-T2, GEM-T3, TEG-2, PGS-4591, PGS-4988, JGM, JGM-1, JGM-1S, JGM-2, JGM-3, EGM-96, TEG-3, etc. The latest models are high degree and order models (70x70, even 360x360 of EGM96), and they account for temporal variations in the low order coefficients and the effect of the permanent deformation of the Earth.</p>
Solid Earth Tides	<p>The Earth, which is not perfectly rigid, deforms under the solar and lunar gravitational attractions. These deformations, associated with the redistribution of mass, can be conveniently defined using the Love numbers introduced by A. Love in 1909.</p> <p>The deformation can be expressed as a change in the coefficients defining the external geopotential.</p>
N-Body	<p>The perturbing forces of the Sun, Moon and other planets, namely, Mercury, Venus, Mars, Jupiter, Saturn, Uranus, Neptune and Pluto, can be approximated with sufficient accuracy as point masses.</p> <p>The values of planet ephemeris are implemented in NAPEOS from the Jet Propulsion Laboratory Development Ephemeris-200 (JPL DE-200 and DE-405).</p>
Ocean Tides	<p>The dynamic contribution of ocean tides due to the gravitational attraction of the Sun and Moon can be accurately formulated in terms of time-varying geopotential coefficient corrections. NAPEOS implements the ocean tide model developed by the University of Texas, Centre for Space Research: CSR3.0.</p>
General Relativity	<p>The dynamic effect of general relativistic satellite perturbations on an Earth satellite is small but detectable. This effect is generated by the curvature in the space-time induced by the presence of a heavy body. For satellites closely orbiting the Earth, only the effect of the Earth needs to be considered</p>
Indirect Moon-J2 Interaction	<p>The effect of the Moon on the Earth bulge generates a deviation in the computation of the third body perturbation as considering Earth and Moon as point masses. NAPEOS can take this effect into account for accurate data analysis.</p>
Non gravitational Perturbations	
Atmospheric Drag	<p>The dominant feature of atmospheric resistance for most satellites is a drag force in the direction opposite to the relative wind. The drag force is considerably larger than lift forces that, if they exist, would be perpendicular to relative wind velocity. NAPEOS implements drag, lift and lateral forces.</p>

		<p>Atmospheric forces mainly depend on three factors:</p> <ul style="list-style-type: none"> ■ Air density. This can be computed using an accurate enough model. For precise data analysis these models are normally not accurate enough and temporal scale factors have to be estimated. NAPEOS implements the latest version of the thermospheric atmosphere MSIS-90. ■ Solar activity: This influences the air density. Unexpected geomagnetic storms can induce rapid variations of air density that cannot be measured accurately enough. Measured and predicted data are retrieved by NAPEOS from NOAA. This is the main source of inaccuracy in the computation of aerodynamic forces ■ Satellite geometry: NAPEOS can provide the effective cross and lateral areas, as function of the satellite attitude. This is just the spin-off from a sophisticated program to analyse the behaviour of a spacecraft under the aerodynamic influence of the free molecular flow.
Solar Pressure	Radiation	<p>Radiation from the Sun produces a small force by transferring momentum from particles streaming out from the Sun to the satellite. This force for a spherical symmetric satellite is a function of the momentum flux due to the Sun, the cross sectional area of the satellite normal to the Sun, its mass and the reflectivity coefficient. Reflectivity values vary between 0 and 1. Eclipse effects are also properly taken into account.</p> <p>As for the air drag forces modelling, NAPEOS can produce cross section area tables to account for the time variations of this perturbation.</p>
Earth Pressure	Radiation	<p>The Sun light reflected by the Earth (albedo) and the infrared radiation of the Earth produce the same effect as the Sun radiation pressure, only much smaller. NAPEOS implements simplified models to account for these perturbations, which are accurate enough given the extremely small absolute value of these accelerations.</p>

10.2.1.2. Observation Models

NAPEOS supports the processing of several tracking types, including preprocessing of raw observations and reconstitution of physical observable magnitudes. Without entering deeply in the details of the tracking processing, NAPEOS implements the following algorithms in the processing of observables and the conversion between reference frames (considered as part of the observation reconstitution from an inertial orbit file):

- Precession/nutation (IERS)
- Detailed Earth orientation (polar motion and Earth rotation)
- Station tectonic plate motion
- Solid tide and ocean loading
- Ocean tide (altimetry only)
- Antenna phase correction
- Tropospheric/ionospheric delay
- Relativistic delay

The preprocessing of the following tracking formats is supported by NAPEOS:

- SLR (MERIT-II)
- PRARE (GFZ)
- DORIS (CNES)
- GPS (RINEX)
- Altimetry (PO-DACC, ESOC)
- MPTS-MK3

The reconstitution of the following observation types is supported by NAPEOS:

- Range
- Range Rate (Doppler)
- Azimuth/Elevation
- Pseudo-range
- Double differences in phase and pseudo-range
- Altimetry
- Inertial and Earth fixed positions

Particular interest has the modelling of altimetric observations. During the years of ERS missions, accurate models of the ocean topography have been developed in ESOC. These models, together with the advances in this area performed by other centres around the world, have been incorporated in NAPEOS. ENVISAT is primarily an altimetric mission whose accuracy depends strongly in the accuracy of these models.

10.2.1.3. Databases

Underneath the processing and the sophisticated algorithms implemented by NAPEOS there is a set of databases holding the information required by the different components. The NAPEOS database consists of a set of tables that configure the properties of the Central Body, Physical Constants, Tracking Systems, Tracking Stations, Tracking Sites, Satellites, Transponders and Thrusters. These databases are fully configurable from the user interface, and the system allows the simultaneous use of tables designated by mission and scenario.

Together with the software, NAPEOS has been delivered with a quite complete database supporting most launched satellites, tracking stations and tracking types (DORIS, PRARE, GPS, etc).

10.2.2. FOCUSLEO, FOCUSLEOP AND FOCUSCN COMMON FUNCTIONALITIES

The following functionalities are shared between *focusleo*, *focusleop* and *focuscn*:

- Tracking data preprocessing
- Orbit Determination
- Orbit Prediction
- Event Generation
- Orbit and attitude tools

10.2.3. FOCUSLEO SPECIFIC FUNCTIONALITIES

focusleo implement specific orbit control functionalities needed to fulfil the Low Earth Orbit satellites orbit control requirements.

10.2.4. FOCUSLEOP SPECIFIC FUNCTIONALITIES

focusleop addresses the flight dynamics needs for launch and early orbit phases of geostationary missions. It provides an engine firing optimization software supporting GTO and SSTO transfer strategies. Apart from that, the following functionalities are integrated:

- Automatic Ingestion of satellite parameter book data
- Tracking data preprocessing
- Orbit Determination
- Orbit Prediction
- Event Generation
- Antenna Pointing
- Manoeuvre optimisation (impulsive (chemical) manoeuvres and low thrust (electrical) manoeuvres) and calibration
- Mass evolution
- Graphical capabilities based on **visualfocus**

The following platforms are already fully supported: Astrium Eurostar 3000 and Thales Alenia Space Spacebus 4000.

focusleop allows to define an **strategy tree** (set of nominal and backup strategies for performing a geostationary satellite transfer orbit) as shown in the following picture. In the following, **strategy profile** means each of the branches of a strategy tree, corresponding to the nominal or backup strategies.

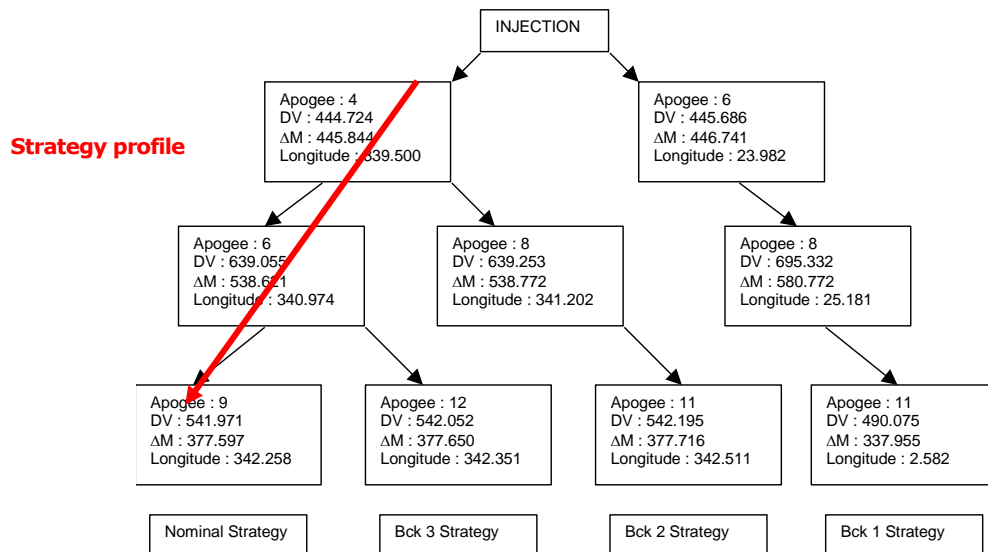


Figure 40: Strategy tree and strategy profile

10.2.4.1. **focusleop** for mission analysis

The following data have to be setup in **focusleop** for initialising a mission analysis:

- TT&C ground station network in the Stations Database
- State vector at separation
- On-station position in the station keeping Database
- Mission constraints and satellite constraints
- Satellite characteristics in the satellite Database, Main Engine database, Cross coupling database and Station Acquisition Thrusters database.
- Sensor characteristics in the Sensor Database
- Transponder characteristics in the Transponder Database

The sequence of modules to be executed for doing a mission analysis is the following:

- **EOP** ingests the files from the IERS with data about the Earth Orientation Parameters and Leap seconds and stores the data in an internal format used for reference frame changes and time conversions throughout **focusleop**. The following two files have to be downloaded from the internet and stored in a specific directory in the **focusleop** workstation:
 - Earth Orientation parameters: <ftp://maia.usno.navy.mil/ser7/finals.daily> (it is updated on daily basis)
 - Leap seconds: <ftp://maia.usno.navy.mil/ser7/leapsec.dat> (it is updated on daily basis)
- **SOLMAG** updates the local solar flux file and the solar flux history file if desired, by parsing the NOAA 3-day Solar and Geophysical Activity Report (Bulletin A), taking into account: the operational flux file, the new values from Bulletin A, and a prediction based on the combination of these two. The solar flux information is needed for the air density calculation used in the aerodynamic forces modelling. The following file has to be downloaded from the internet and stored in a specific directory in the **focusleop** workstation (it is updated on daily basis): <http://www.swpc.noaa.gov/ftplib/latest/RSGA.txt>
- **INITGUESS:**

INITGUESS allows defining the separation state vector in several formats (covering the format provided by most of the commercial launchers) and the satellite properties and constraints to be taken into account for searching the possible transfer strategies.

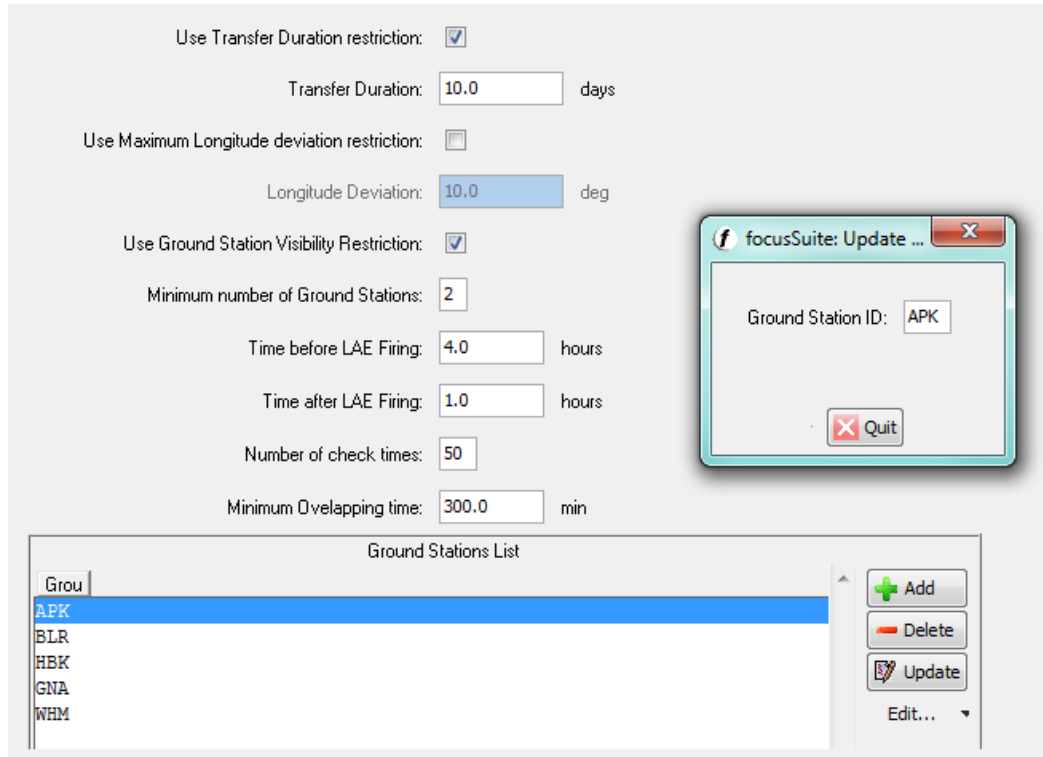


Figure 41: Example of INITGUESS constraints options panel - Transfer Constrains Panel

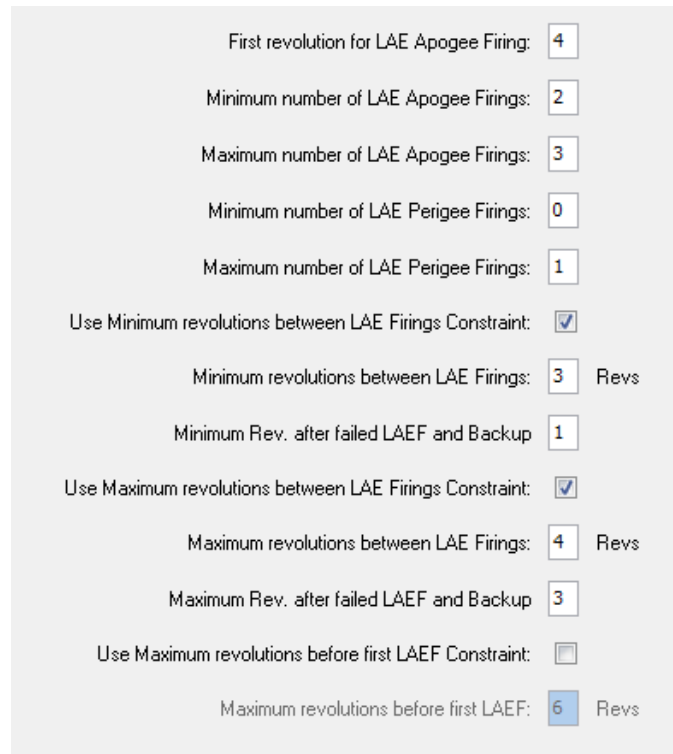


Figure 42: Example of INITGUESS Constraints Options Panel - Sequencer Constrains

- First execution in sequencer mode to generate possible tree strategies, it means nominal and associated backup strategies. The following figure depicts an example of a strategy file containing several tree strategies.

```

Output: Initguess Strategy File
=====
STRATEGY NUMBER      1 =====
NOMINAL SEQUENCE
Sequence Data | End Epoch = 11/01/2014 09:12:42 | Mass = 2408.521 kg | STACQ Mass = 10.172 kg
Man. Data  Man Rev Type      Epoch      Duration[s]  Delta-V[m/s]  Mass[kg]  Drift[°/orb]  Long.[°]
Manoeuvre  1   5   A  04/01/2014 03:48:42  10373.23     568.28    1460.02     93.75    109.48
Manoeuvre  2   8   A  06/01/2014 09:26:52  6551.08     427.12     922.06      2.51    30.73
Manoeuvre  3  13   A  11/01/2014 09:10:46  115.64       8.16      16.28      -0.02    43.29
-----
BACKUP SEQUENCE      1
Sequence Data | End Epoch = 14/01/2014 09:46:38 | Mass = 2402.290 kg | STACQ Mass = 12.978 kg
Man. Data  Man Rev Type      Epoch      Duration[s]  Delta-V[m/s]  Mass[kg]  Drift[°/orb]  Long.[°]
Manoeuvre  1   9   A  06/01/2014 11:43:20  1358.97      70.31     191.27     149.10    7.36
Manoeuvre  2  15   A  09/01/2014 22:51:33  8522.85     473.45    1199.58     97.41   181.93
Manoeuvre  3  21   A  14/01/2014 07:48:24  7093.92     459.81     998.46     -0.02    46.40
-----
BACKUP SEQUENCE      2
Sequence Data | End Epoch = 14/01/2014 08:52:30 | Mass = 2409.421 kg | STACQ Mass = 13.306 kg
Man. Data  Man Rev Type      Epoch      Duration[s]  Delta-V[m/s]  Mass[kg]  Drift[°/orb]  Long.[°]
Manoeuvre  1   5   A  04/01/2014 03:48:42  10373.23     568.28    1460.02     93.75   109.48
Manoeuvre  2  11   A  08/01/2014 14:38:53  5868.15     380.46     825.93     15.81   311.97
Manoeuvre  3  17   A  14/01/2014 08:39:28  782.69       54.82     110.16     -0.02    46.78
-----
BACKUP SEQUENCE      3
Sequence Data | End Epoch = 12/01/2014 08:58:45 | Mass = 2410.765 kg | STACQ Mass = 12.415 kg
Man. Data  Man Rev Type      Epoch      Duration[s]  Delta-V[m/s]  Mass[kg]  Drift[°/orb]  Long.[°]
Manoeuvre  1   5   A  04/01/2014 03:48:42  10373.23     568.28    1460.02     93.75   109.48
Manoeuvre  2   8   A  06/01/2014 09:26:52  6551.08     427.12     922.06      2.51    30.73
Manoeuvre  3  14   A  12/01/2014 08:56:49  115.64       8.16      16.28      -0.02    45.80
=====
STRATEGY NUMBER      2 =====
NOMINAL SEQUENCE
Sequence Data | End Epoch = 11/01/2014 09:12:42 | Mass = 2408.521 kg | STACQ Mass = 10.172 kg
Man. Data  Man Rev Type      Epoch      Duration[s]  Delta-V[m/s]  Mass[kg]  Drift[°/orb]  Long.[°]
  
```

Figure 43: INITGUESS output strategy file

- Second execution in analyser mode, selecting the desired strategy tree to be analysed, it generates a file (initial guess file) tree strategy data in a format suitable for ingesting them in STAN and AEFOS for further optimisation.

- **STAN** performs a first analysis of the strategy profile given as a set of apses and using a full numerical propagator for the nominal and backup profiles of the selected tree strategy (to be executed in separate workspaces)
- **AEFOS** performs a full optimization of the strategy profile for the nominal and backup profiles of the selected tree strategy (to be executed in separate workspaces).

The problem solved by AEFOS is mathematically regarded as a constrained non-linear local optimization (non-linear programming), using a full dynamic model and implementing equality and inequality constraints. It takes in to account strategy constraints defined by the user such as eclipses duration, station visibilities, maximum duration of manoeuvres, minimum number of orbital revolutions between apogee/perigee manoeuvres, maximum duration of the transfer orbit, etc... It is stated as:

- Minimize $F(\mathbf{x})$

- Subject to $g_i(\mathbf{x}) > 0, \quad i = 1, \dots, m_i$
 $h_i(\mathbf{x}) = 0, \quad i = 1, \dots, m_e$

Where $\mathbf{x} = (x_1, \dots, x_n)$ is the vector formed by the n optimization variables, $F(\mathbf{x})$ is the non-linear objective function to be minimized (the propellant consumed by the manoeuvres), $g_i(\mathbf{x})$ are the non-linear inequality constraints that must be satisfied (the mission constraints), $h_i(\mathbf{x})$ are the non-linear equality constraints that must be satisfied (the target to be achieved), m_i is the number of inequality constraints and m_e is the number of equality constraints.

The optimisation variables are the following for each manoeuvre to be optimized:

- Manoeuvre Midpoint
- Manoeuvre Duration
- Manoeuvre Right Ascension (inertial)
- Manoeuvre Declination (inertial)

Please note that the mass consumption to be minimised is defined as: $F(x) = \Delta m_{LAEFs} + \Phi$

Where Δm_{LAEFs} corresponds to the mass consumption corresponding to the transfer manoeuvres (apogee and perigee manoeuvres)

Φ is a function modelling the station acquisition mass consumption correction of residual inclination, residual eccentricity and rendezvous)

The following picture depicts a snapshot of part of the AEFOS output file.

```

STATION ACQUISITION -----
Station Acquisition Computation
Apogee radius = 42258.442488 km
Perigee radius = 41765.743924 km
Drift After LAEFs = 1.950106 °/day
Minimum drift = -0.596743 °/day
Maximum drift = 2.566661 °/day
Minimum longitude = 355.593964 °
Maximum longitude = 27.228006 °
Inclination correction - dv= 0.009622 m/s
                       - dm= 0.008583 kg
                       - dt= 0.125317 s
Apo/perigee correction - dv= 9.014518 m/s
                       - dm= 8.028748 kg
                       - dt= 234.439446 s
Station rendez-vous   - dv= 0.000000 m/s
                       - dm= 0.000000 kg
                       - dt= 0.000000 s

Station-acquisition fuel cost (kg) : 8.037332
-----
RESULTING FUEL COST -----
LAE firings + mass losses (kg) : 1531.971832
Station acquisition (kg) : 8.037332
Total (LAEF + STACQ) (kg) : 1540.009163
APPLIED MASS LOSS CORRECTIONS
Sun acquisition after separation (kg) : 0.000000
Sun acquisition after eclipses (kg) : 0.000000
Earth acquisition and gyro calibration before burns (kg) : 0.000000
Attitude maintenance between burns (kg) : 0.000000
Sun acquisition after burns (kg) : 0.000000
  
```

Final Cost

Figure 44: Example of part of the AEFOS output file

The following picture depicts an example of the report generated by AEFOS for an optimized strategy profile:

	MAN1	MAN2	MAN3
DV	660. 701714	498. 007338	328. 453692
Delta mass (kg)	769. 806679	482. 835381	279. 330202
THRUST			
RA (°)	118. 582039	119. 543412	119. 320111
Decl. (°)	-6. 590024	-6. 618866	-5. 880408
Duration (s)	5518. 672397	3515. 955238	2095. 731976
LQ start epoch	2009/05/17-10:30:59.559	2009/05/19-03:24:41.071	2009/05/21-09:55:55.178
a (km)	24442. 154245	28824. 900347	35029. 423314
ecc.	0. 728784	0. 465972	0. 206378
incl. (°)	6. 002961	2. 349995	0. 733444
RAAN (°)	30. 485177	31. 300572	33. 617537
arg. perigee (°)	179. 113206	178. 551164	176. 165696
True anom. (°)	173. 877330	174. 650701	176. 245944
Longitude (°)	350. 509328	96. 401193	357. 875018
Drift (°/day)	456. 906506	277. 650565	115. 726020
Drift (°/orb)	201. 110063	156. 511957	87. 393219
Mass (kg)	4086. 400000	3316. 593464	2833. 758227
Main. eng. start	2009/05/17-10:33:29.559	2009/05/19-03:27:11.071	2009/05/21-09:58:25.178
a (km)	24443. 846338	28829. 048692	35038. 267930
ecc.	0. 728665	0. 465761	0. 206074
incl. (°)	6. 000161	2. 348219	0. 732192
RAAN (°)	30. 488392	31. 305563	33. 630103
arg. perigee (°)	179. 111874	178. 548400	176. 156724
True anom. (°)	174. 210336	175. 108191	176. 799374
Longitude (°)	350. 215703	96. 233820	357. 805286
Drift (°/day)	456. 821582	277. 512726	115. 545529
Drift (°/orb)	201. 093563	156. 468028	87. 289966
Mass (kg)	4085. 875634	3316. 069098	2833. 233861
Main. eng. end	2009/05/17-12:02:58.232	2009/05/19-04:23:17.027	2009/05/21-10:33:50.910
a (km)	28824. 522573	35028. 627314	42012. 093206
ecc.	0. 465890	0. 206332	0. 005864
incl. (°)	2. 348542	0. 729328	0. 049944
RAAN (°)	31. 420308	33. 695861	89. 804542
arg. perigee (°)	178. 285343	175. 984103	118. 963568
True anom. (°)	188. 118079	186. 638016	185. 460724
Longitude (°)	341. 756528	93. 520937	356. 565893
Drift (°/day)	277. 663120	115. 742269	1. 961830
Drift (°/orb)	156. 515957	87. 402511	1. 945898
Mass (kg)	3316. 593464	2833. 758227	2554. 428168

Figure 45: Example of report generated by AEFOS

- **LAUNCH** computes the launch window for the transfer strategy profile optimised by AEFOS, it calculates the range of time in which it is possible to launch a satellite fulfilling all the platform and transfer strategy constraints. It allows the user to define sequence of satellite operations (timeline of operations) and the corresponding constraints to be fulfilled during the transfer phase. It supports some predefined satellite attitude operational modes (others can be added), these include: Earth pointing, Sun pointing, Inertial navigation with a defined attitude,... It provides as output the beginning and end epoch of the launch window per day.
- **GLOBLW** is in charge of merging the launch windows computed by the LAUNCH module for each transfer strategy (nominal and backups strategies), in order to compute the global launch window to be taken into account for the decision of the launch day and epoch.
- **ORBUPD** updates the operational orbit file from other orbit files generated in several modules in *focusleop*, in this case, it will generate the operational orbit file from the orbit file generated by AEFOS.

- EVENTS** computes geometrical events related to orbit, ground stations, sensors and transponders, the events to be generated can be selected by the operator.

The following figure depicts the EVENTS orbital events panels where the orbital events can be selected.

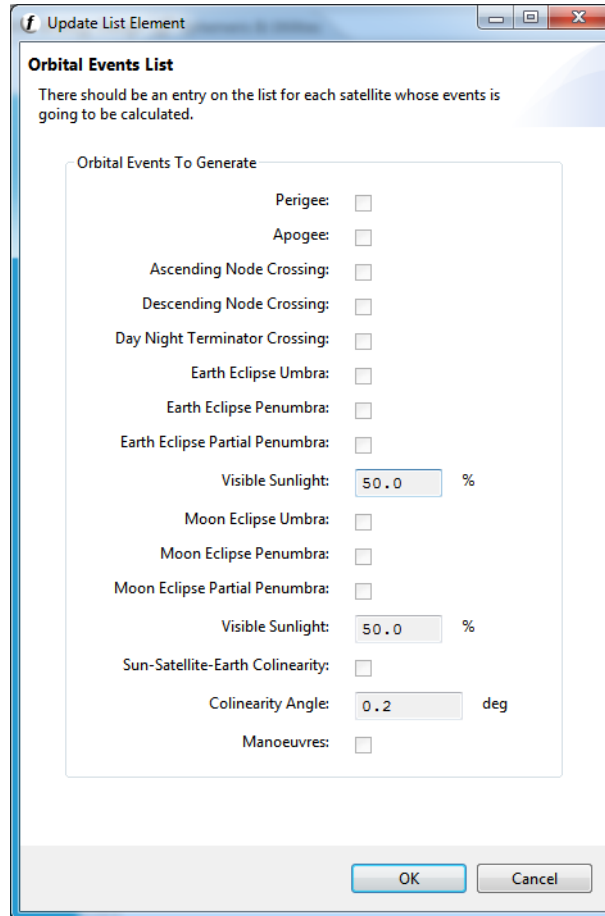


Figure 46: EVENTS Orbital Events panel

The following figure presents an example of a report generated by EVENTS.

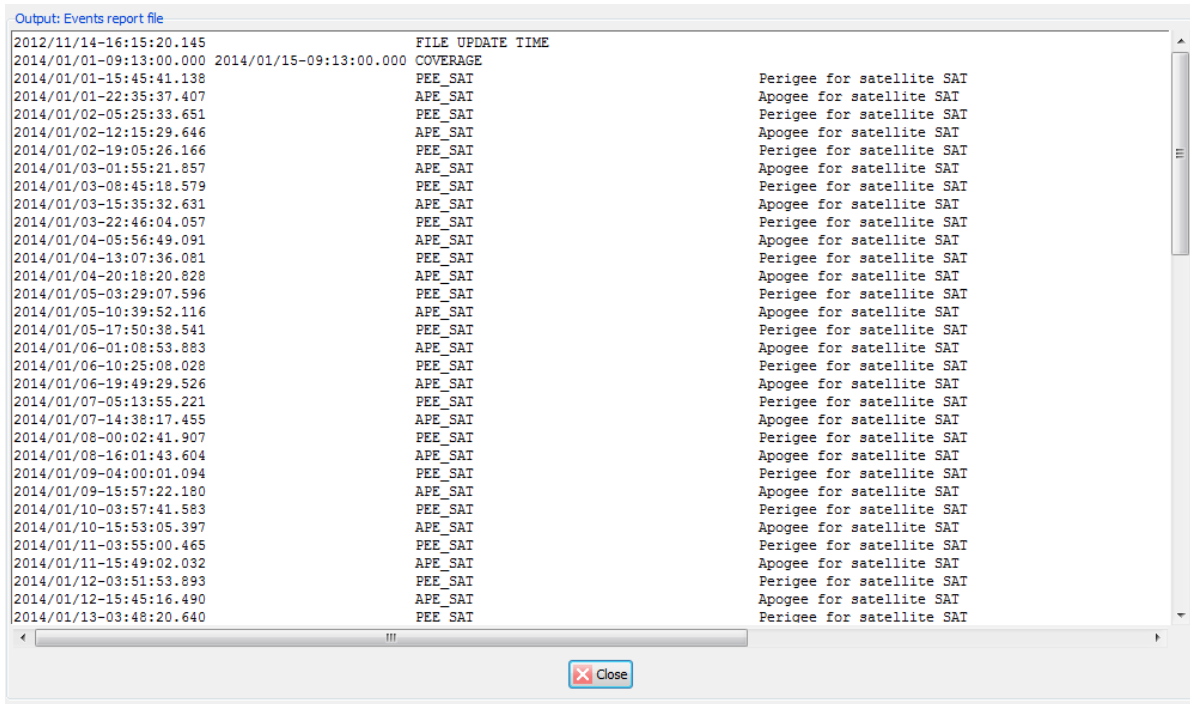


Figure 47: EVENTS Events Report File

The EVENTS report can be browse using a *focusleop* application (EBRO), the following figure depicts an example.

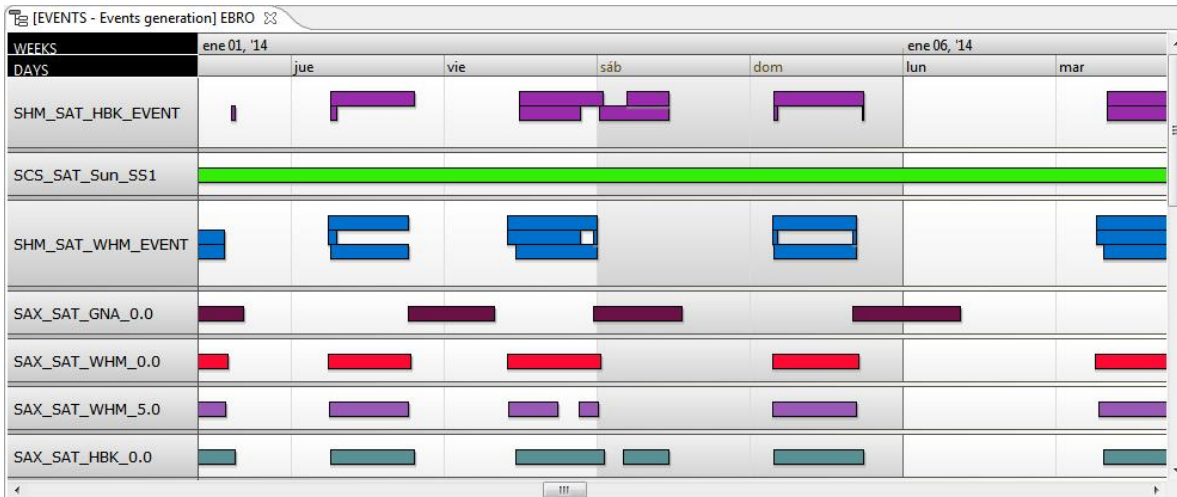


Figure 48: EVENTS EBRO Gantt chart

- **VISIBILITY** computes the visibility between a satellite and a set of ground stations, and plots the azimuth and the elevation of the satellite.

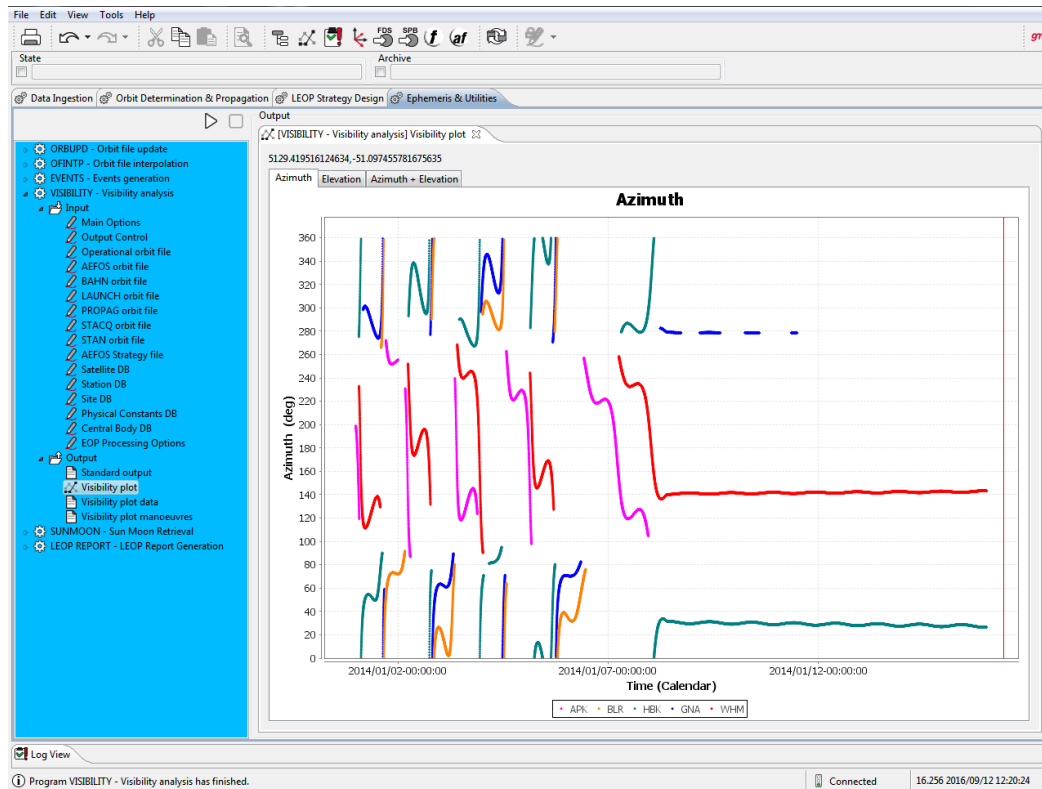


Figure 49: Example of plot generated by VISIBILITY

- **MONTECARLO** simulates errors in the injection orbit, in orbit determination and in the manoeuvre attitude and thrust performance in order to evaluate the robustness of the transfer strategy.

The following figure depicts the injection dispersion errors that can be simulated in this module:

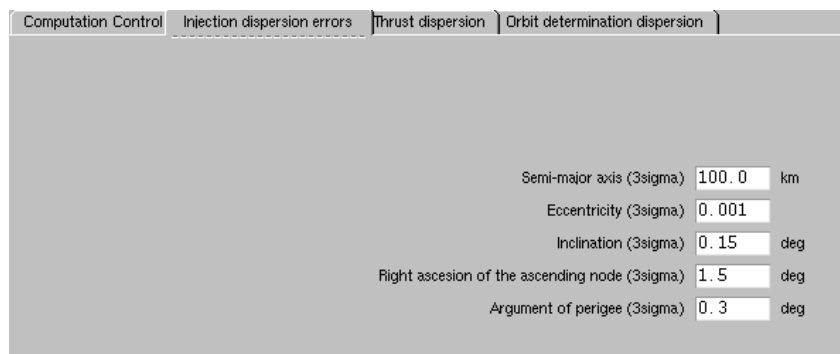


Figure 50: MONTECARLO injection dispersion errors

- **STACQ** computes the in-plane and out-of-plane manoeuvres required during the station acquisition phase to initialise the station keeping cycles. It takes also into account the collocation parameters in case the satellite will be included in a collocation cluster

- **OFINTP** is a tool used to interpolate state vectors from various possible orbit files created by other modules in **focusleop**. This component is in charge of providing output with one or more state vectors read from the orbit file, in any reference frame and vector type that the user specifies. It is proposed as an auxiliary tool

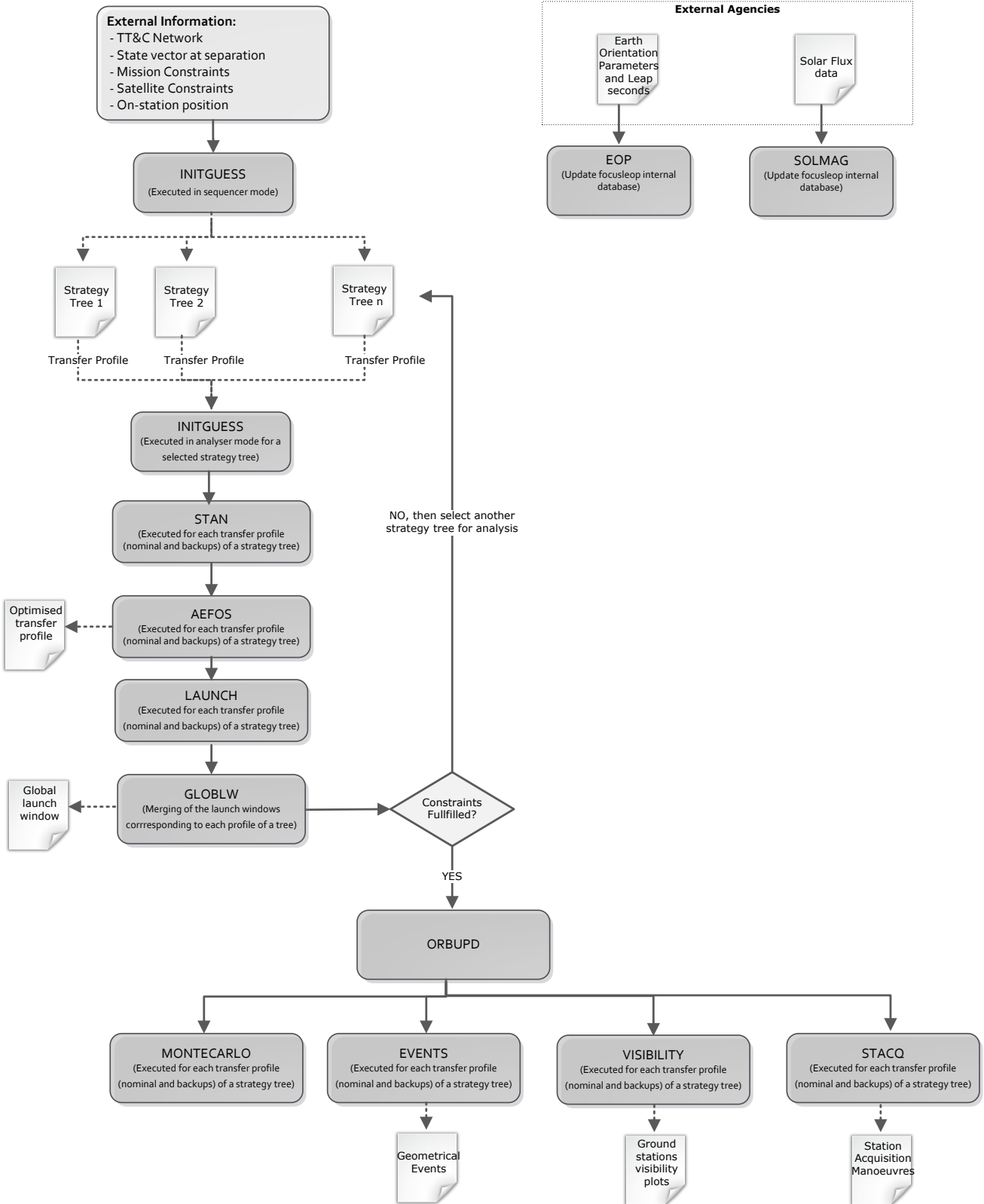


Figure 51: focusleop for mission analysis

10.2.4.2. *focusleop* for generic operations

The sequence of modules to be executed for doing a mission analysis is the following:

- **EOP:** see previous section.
- **SOLMAG:** see previous section.
- **PROPAG:** propagates satellite orbits accounting for multiple dynamical parameters (including manoeuvres) with either a fixed or variable step size in the integration of the equations of motion.
- **BAHN:** Estimates satellite orbit, multiple dynamical parameters (including manoeuvres), ground station, and geophysical parameters by processing a batch of measurements. It is based on an iterative Bayesian least-squares estimation algorithm. It is executed several times after the separation from the launcher and before the manoeuvre optimization.
- **ORBUPD:** see previous section.
- **STAN:** see previous section.
- **AEFOS:** see previous section.
- **BAHN:** it is executed after the manoeuvres execution to estimate them and to compute their performance.
- **ORBUPD:** see previous section.
- **EVENTS:** see previous section.
- **VISIBILITY:** see previous section.
- **STACQ:** see previous section.
- **OFINTP:** It is proposed as an auxiliary tool, see previous section.

Apart from that, additional modules have to be implemented for supporting specific functionalities of each satellite bus.

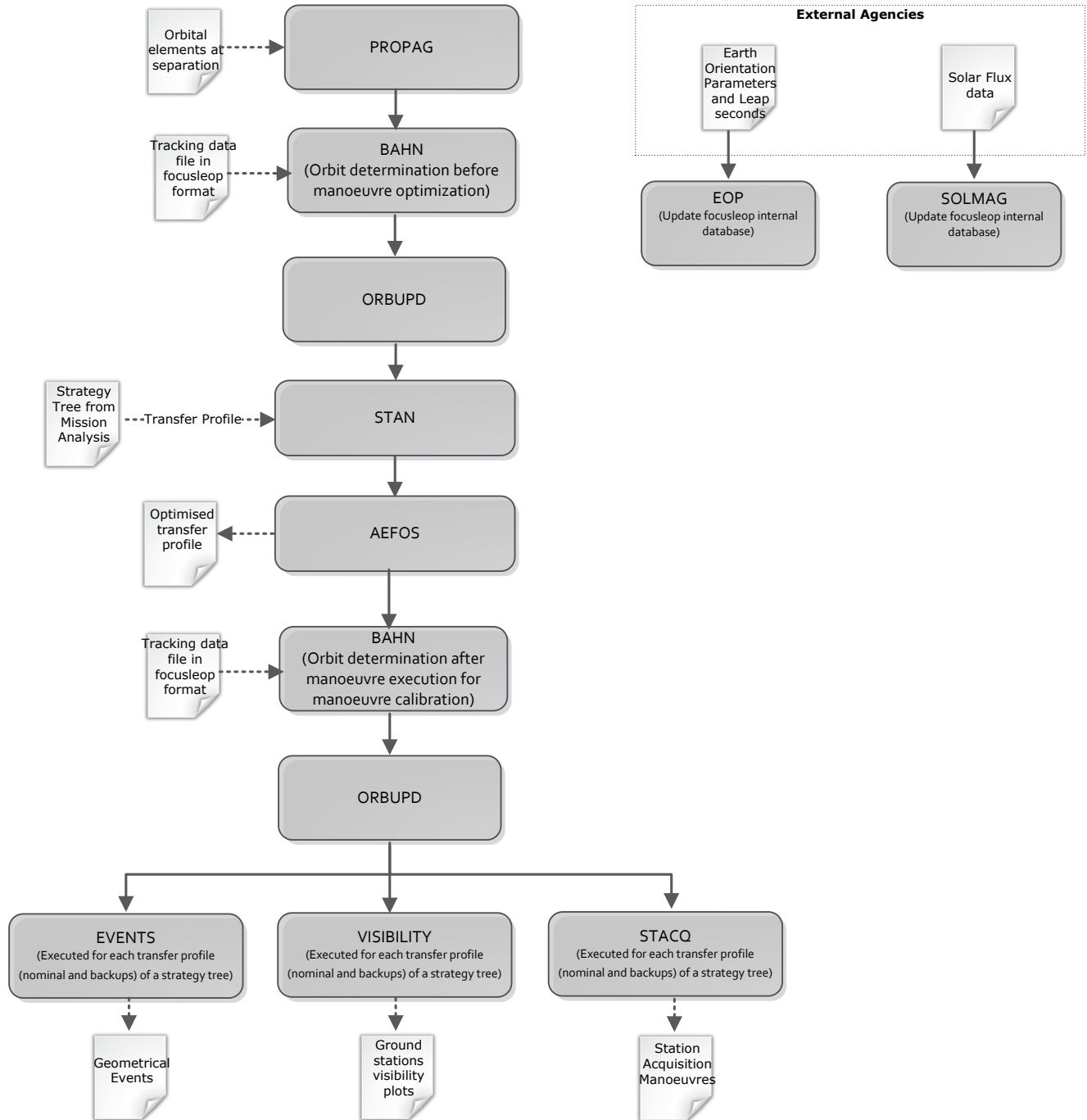


Figure 52: focusloop for real operations

Apart from the modules described above there are dedicated mission-specific modules that depend directly on the satellite platform.

Please note that currently **focusleop** supports two platforms: EUROSTAR-3000 and SPACEBUS-4000 and it would be required to perform dedicated development to support new platforms.

In general terms the following functions are considered Mission-specific:

- **Manoeuvre Command generation:** using the theoretical delta-V, simulates the manoeuvre on-board according to manufacturer specifications and generates the command.
- **Manoeuvre calibration:** using the estimated delta-V and the predicted one, computes calibration factors for next manoeuvres.
- **Mass computation:** compute mass consumption from TM data according to manufacturer specifications
- **Manoeuvre attitude computation:** generates the command to be sent to the S/C to acquire the attitude during manoeuvre.

Additionally, the station keeping acquisition would include some common functionality with **focusgeo**, as they are: INCLON/SOLONG/INICOL in case of chemical propulsion or INCLEC/ISKM in case of ionic propulsion.

10.2.5. FOCUSCN SPECIFIC FUNCTIONALITIES

The following constellations are currently supported: Galileo, Globalstar and O3b.

Manoeuvre Planning software for constellations has been developed covering the following functionalities: Insertion, relocation, station acquisition and deorbiting.

The following operational set-up is used in these systems:

- A Constellation workspace to execute all the multisatellite tasks.
- A Satellite workspace per satellite of the constellation, they are devoted to execute nominally all the satellite dependent modules (monosatellite tasks, such that Attitude Determination, Mass Computation and TC data Generation). The non-dependent satellite tasks will be also available allowing to operate a satellite autonomously in its own workspace without any interference with the rest of the constellation. This can be needed in case of contingency in one of the satellites of the constellation.
- Databases are shared between Constellation and satellite workspaces.
- Orbit and attitude data are exchanged between Constellation and satellite workspaces.

The following picture depicts the share of operations and data between the constellation workspace and the satellite workspaces:

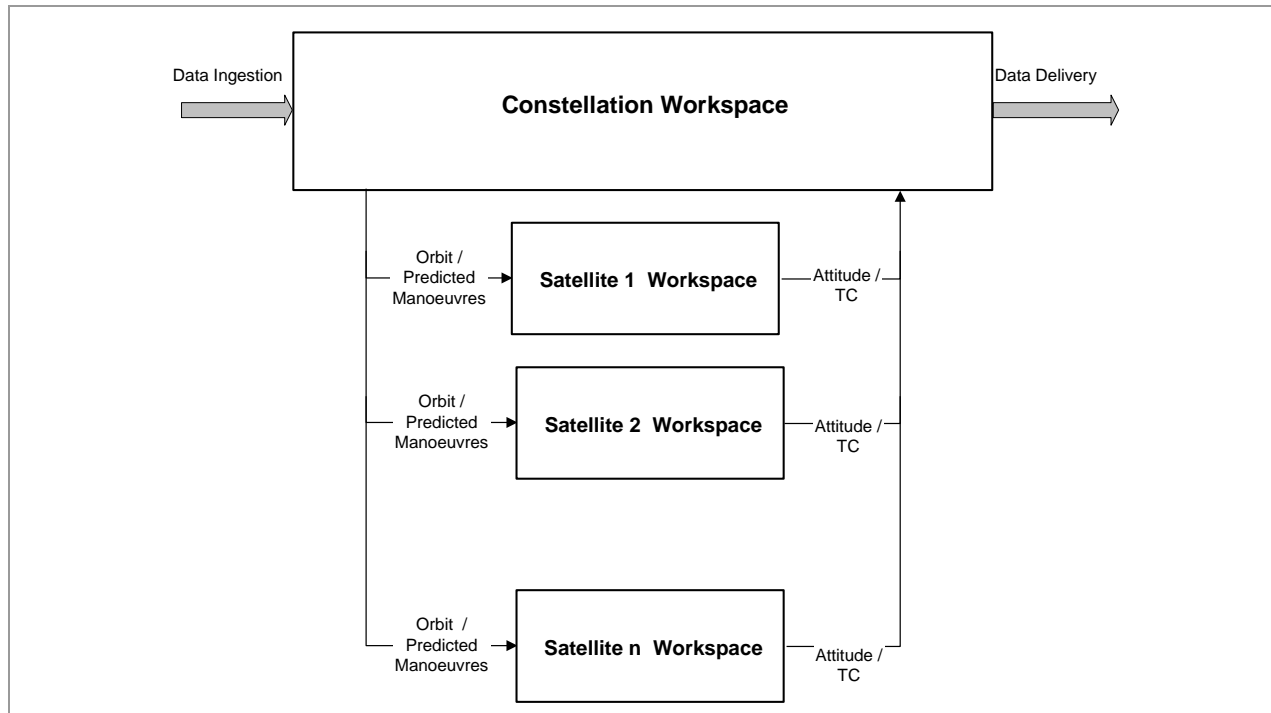








































Figure 53: FDF Scenario and workspace structure

The constellation workspace will perform nominally the following tasks:

- Data Ingestion
- Orbit Determination & Propagation
- Manoeuvre Planning
- Geometrical Events Generation
- Products Generation (TLE, daily plan,...)
- Data Delivery

10.3. FOCUSSUITE SUPPORTED PLATFORMS

MANUFACTURER	PLATFORM	CHEMICAL	HYBRID	IONIC	focusgeo	focusleop	focuscn	focusleo
	STAR-2		X					
	GEOSTAR-3	X	X	X				
	LEOStar-2							
	SPACEBUS-2000	X						
	SPACEBUS-3000	X						
	SPACEBUS-4000	X						
	SATELCOM	X						
	PROTEUS	X						
	ELiTeBus							
	EUROSTAR-2000	X						
	EUROSTAR-2000+	X						
	EUROSTAR-3000	X						
	EUROSTAR-3000pps			X				
	Astrobus							
	BSS-376	X						
	BSS-601	X						
	BSS-702			X				
	SSL-1300	X						
	SSL-1300I		X					
	LUXOR			X				
	DS-2000	X						
	EKSPRESS			X				
	I-3000	X						
	I-2000	X						
	A2100-TR	X	X					



 OPERATIONAL
 UNDER DEVELOPMENT

Table 10-1: focussuite supported platforms (October 2016)



focus
SUITE

Code: GMV-FOCUSSUITE-WP
Date: 15/07/2016
Version: 1.2
Page: **75** of **75**

END OF DOCUMENT