FREE JAVA[™] CNES FLIGHT DYNAMICS TOOLS *Jean-François Goester* Email: jean-francois.goester@cnes.fr



FREE JAVA[™] CNES FLIGHT DYNAMICS TOOLS



PSIMU		👻 Orbit conv	rettor			– п х
Windows Options About		File Conf. A	hout			
		File Colli A	DOUL			
Read context Gave contex	t 🕞 Save results	Orbit *		Pivot		TLE Convert
		Desetivate				The control of
		Deactivate co	onversions			
and an anti-sets and a section of (#1)		Date: *		04/08/1999 1	5h56m59s995ms392us000	ons <u>UIC</u> +/-
Initial Orbit Farth Features Vehicle	Features Forces Scena	Frame:		GCRF	•	
		Type:		Apsis Radius	-	
Earth Potential *		Apsis Ra	dius Parame	ters *		
Potential File Name	GRIM4_S4 💌	rp: *			7020.6955524098	1 <u>km</u>
Maximum degree and order:	69	га: *			7180.4888769503	5 <u>km</u>
Zonal *	8	i: *		[70.647272624606	8 deg
Tesseral *	8	Ω: *			-77.71010097907	1 deg
Third Body *	Active * 🖌	ω: *			113.18533088272	8 deg
Moon *	×	Anom	alv: *	true	-113.18757871726	3 deg
Sun *	V			Soloct	202600 441	2 km43/c42
Venus		μ.		Jeieut	550000.441	0 <u>mm 3/3 2</u>
Mars						
Multiplicative factor Solar Radiation Pressure * Reference distance Reference pressure Multiplicative factor Rediffused Solar Radiation Pressure Ocean Tides Terrestrial Tides Ephemeris type	Active * ☑ 1.4959787 4.5605 Active □ Active □ Active □ @ JPL ○ Meeus ○ ħ	1.0 E11 m E-6 Nm^2 1.0	Attlade Creat	PTFLAX Cestion pilots PTFLAX Cestion pilots PTFLAX Cestion pilots PTFLAX PTFLAX PTFLAX PTFLAX PTFLAX PTFLAX PTFLAX PTFL	Ground track	
TestForGPlotPane Open file Title Cocodito A Name of the file	EPHEM.txt Iblude versus Time ile in the title		-50 -60 -70 -90 -90 -10	-260 -140 -120 -200	-80 40 -40 -20 8 20 40 Longitude (drg)	60 80 100 120 140 1
Abscissa data *	TIME - Ordinate data * A	eodetic Altit	ry Axe 🖂 ude versu	s Time	Add Del Clear	
320 1						

Show Legend 📧 Show Crosshair

Summary

- The old Software suit
- ☆ The choice of the Java[™] language
- ✤ The Java[™] software suit
 - PATRIUS
 - GENIUS
 - GENOPUS
 - PSIMU
 - Other applications
- Distribution
- Conclusion











The old Software Suit

- Mainly coded in FORTRAN
- Originally basic libraries (MSLIB then BIBMS)
- Frameworks:
 - MERCATOR (GEO LEOP/SK)
 - PIMS
 - ✓ MADONA: for data files structure
 - ✓ **GENESIS/DIAMS**: for GUI
 - ✓ XTRACE: for plots
- Basis for:
 - Expert tools
 - Operational FDS





The choice of the Java[™] language

Needs

- Technical capacity of the language: CPU time performance, good numerical precision, robustness, portability …
- Existence of tools: compiler, development environment, quality checks …
- Knowledge of the language in CNES or in contractors working for CNES; training course, …

Criterions

- Numerical performance
- CPU performance
- Portability
- Maintenance and development facilities

Selected languages

- Ada
- C
- ♦ C++
- Fortran (95, 2003, 2008)
- ♦ Java[™]

- Interfaces and interoperability
- Maturity / obsolescence
- Security & reliability
- Treatments sequencing







7th International Conference on Astrodynamics Tools and Techniques, Oberpfaffenhofen, Germany, 6-9 November 2018



PATRIUS

Why?

- Originally based on a Mathematics package corresponding to the Apache Commons Math library and other packages issued from OREKIT
- After several years of evolutions, PATRIUS has been considerably changed with more and more add-ons
 ⇒ since the V4.0 version, it has been decided to break with these links,
- High level of validation (comparison with ZOOM S/W used for POD)
- Used for the next generation of FDS
- Numerous user feedbacks as:
 - Attitude (attitude legs)
 - Maneuvers (on events)
 - Orbits (atmospheric reentry type, orbital parameters notion, …)
 - Vehicle (Assembly, …)
 - New kind of forces (lift component, ...)





Why?

- GUI for flight dynamics tools (or, more generally, scientific tools) need most of the time:
 - To enter numerical input data from the screen or the keyboard
 - To read / write these data into files
 - > To execute computation thanks to these data
 - To visualize results
- Other tools are available, but:
 - basic ones, as Swing
 - more elaborated ones do not include such functionalities or, if they have, it will be partially

PSIMU e Windows Options About							-		×
Read context ☐ Save contex	🕞 Save results	Compute	Clear results	🍼 Clear messag	es	0 %		E Qi	uit
oaded context: no confext (*) Initial Orbit Earth Features Vehicle Forces Scenario: * Earth Potential * Potential File Name Maximum degree and order: Zonal * Tesseral * Third Body *	Features Forces Scenario	Maneuvers	Attitude Laws / Integ	rator Évents í G	Dutput Consol	8			
Moon * Sun *									
Venus Mars Jupiter									
Atmospheric Force * Atmospheric Model Solar Activity Type Multiplicative factor	Active * C Exponential ● MSISE200 ● Real Constant By 1.0	0 🔾 US76 🧲 file	DTM2000						
Solar Radiation Pressure * Reference distance Reference pressure Multiplicative factor	Active * 1.4959787E11 4.5605E-6 1.0	<u>m</u> N/m^2							
Rediffused Solar Radiation Pressure	Active								
Terrestrial Tides	Active								
Ephemeris type	● JPL ○ Meeus ○ Meeus	s Stela							



FREE JAVA[™] CNES FLIGHT DYNAMICS TOOLS



The Java[™] software suit



Main advantages

- Numerical widgets with:
 - Units' management
 - Intervals of validity
- Simplified approach, in particular about events management

Distance:

1.0 KF

km

nmi

- Performs very easily conditional display;
- ♦ Process management compatible of all OS (thanks to Java[™])
- Read / write management:
 - Same (or almost same) code for display, read or write
 - No need to code (XML) files methods
- Some higher levels widgets
 - list of widgets
 - 2D plots ... and event ground tracks plots
 - Standard applications frame







- Why?
- To display PATRIUS objects using GENIUS widgets
- To avoid to duplicate same kind of widgets in higher level tools
- To standardize some widgets as:
 - Date with timescale and conversions
 - Inertial and rotating frames configurations
 - Orbit definition (date, frame, parameters) and conversions
 - Maneuvers (impulsive, continuous thrust, sequences)
 - Attitude laws individually or via a sequence of laws;
 - Orbital events (eclipse ...);
 - Vehicle characteristics;
 - Force models (potential, atmosphere, SRP ...)









Why?

- In the old FORTRAN suit, there were no sufficiently high level layers to propose something to propagate easily trajectories ⇒ PSIMU was there to deal with a lack of such level.
- This need has been replaced in **PATRIUS** by the very powerful capacity of the NumericalPropagator class.
- Another need was to propose to users an "on the shelf" tool with its own **GUI**:
 - It has been decided to build a PSIMU Java[™] version, at least with its GUI and batch version (but also with the possibility to use through Java[™] code)
 - This PSIMU version is a tool allowing to propagate trajectories around the Earth. These trajectories may be:
 - ✓ Elliptical ones (all kind of orbits, from LEO to GEO through MEO or HEO),
 - ✓ Hyperbolic ones,
 - ✓ Atmospheric ones (in particular in case of debris reentries).

I card context	Save context	Save results	Commute	Clear results	Clear messanes	0.5	80
a cost contexts		and the second sec	P Compare		o cital incodigeo		
aded context: no cont	ext (*)						
Initial Orbit Earth Fe	atures Vehicle Featur	es Forces Scenario	Maneuvers	Attitude Laws Inte	grator Events Out	out Console	
Reference Date:							
Date mode:	custom external						
External date:	01/01	/2018 00h00m00s	rc +/-				
Attitude Laws:							
Set attitude laws							
Mode: *	Switch Simple						
Switches:							
Amount of switches	2						
Amount of switches	2	Manua el					
Amount of switches ' Number of the switch		tems + -					
Amount of switches ' Number of the switch		tems + -					
Amount of switches ' Number of the switch Switch1		toms +/-	7				
Amount of switches ' Number of the switch Switch1	2 < 1	Items +i-	7				
Amount of switches * Number of the switch Switch1 Law before * Law after *		tems +/-]				
Amount of switches * Number of the switch Switch1 Law before * Lew after * Event Detertor *	< 1 LVLH SunPoir	v teens +/-					
Amount of switches ' Number of the switch Switch1 Law before ' Law after ' Event Detector ' True '	< 1 LVLH SunPor	ted v					
Amount of switches ' Number of the switch Switch1 Law before * Law after * Event Detector * Type *	LVLH Eclipse	ted v					
Amount of switches ' Number of the switch Switch1 Law before ' Law after ' Event Detector ' Type ' Display event of	< 1 LVLH Eclipse	ted v					
Amount of switches ' Number of the switch Switch1 Law before ' Law after ' Event Detector ' Type ' Display event of Eclipse Detector '	< 1 LVLH SunPor Eclipse	terns +i-					
Amount of switches Number of the switch Switch1 Law before * Law after * Event Detector * Type * Eclipse Detector * Event type: *	2) C 1 LVLH SunPidr Eclipse ontig	Berns +/- ted v					
Amount of switches Humber of the switch Switch Law before * Law after * Event Detector * Type * © Osplay event of Eclipse Detector * Event type: *	LVLH Eclipse (* Entry Extra	v Rams +					
Amount of switches Number of the switch Switch1 Law after * Event Detector * Type * Classa Detector * Event type: *	Clipse onfig	Rams +L					
Amount of switches Number of the switch Switch Law before * Law after * Event Detector * Type * Eclaps Detector * Event type: *	LVLH LVLH Eclipse onfig * Entry - Extr - Extr - Sant -	 Hams +f. tool w w a a b b b c a b c a b a a b a a b a a					
Amount of switches : Number of the switch Switch1 Law before * Law after * Event Detector * Type * © Display event of Eclass Detector * Event type: *	LVLH Eclipse onfig * Entr Both Sanger	e # Multiple					
Amount of switches : Number of the switch Switch1 Law before * Law after * Event Obector * Type * Event Obector * Eclaps Detector * Event type: *	LVLH LVLH SumPor Eclipse ontig Eclipse SumPor Sum SumPor Sum SumPor Sum SumPor Sum SumPor Sum Sum Sum Sum Sum Sum Sum Sum	e Mettiple					
Amount of switches : Number of the switch Switch1 Law before * Law after * Event Detector * Type * Chipspay event Eclipse Detector * Event type: * Triggering: *	LVLH LVLH Eclipse onfig (* Entry Est Both Sing From * Every * Utilit	Bams +.					





How ...

- Its initialization is made by:
 - The initial orbital parameters definition (epoch, frame, coordinates)
 - The vehicle modelling with shapes, dimensions, aerodynamic, radiative pressure and propulsive characteristics (engines, tanks)
 - > A maneuver sequence (optional) including impulsive maneuvers and/or continuous ones
 - A sequence of attitude laws (optional)
 - Numerical integrator parametrization (Runge Kutta or Dormand Price).
 - Choice of various force models
- PSIMU also allows to set its output data within several tens of variables.
- A graphical interface for plots is also integrated (next version with a dedicated plot with ground tracks).



Other applications

- ◆ Thanks to PATRIUS, GENIUS and GENOPUS but also sometimes PSIMU, several other expert tools have been (re)developed in Java[™]:
 - ➤ MIPELEC: may be one of the most ancient tool to be freely delivered by CNES but only via its source code to be recompiled sometimes with some difficulties. So, it has been decided to propose, always freely, a new JavaTM version with its own GUI
 - Then, several other tools have been also recoded (but not available outside CNES) as:
 - CRASH for guided reentry trajectories
 - DOORS for deorbit strategies
 - OSCAR / DRAGON for phasing & rendezvous strategies





Other applications (ct'd)

- At last, FSOA tools:
 - STELA for Semi-analytic Tool for End of Life Analysis
 - DEBRISK for evaluating the survivability of fragments from a satellite entering the Earth's atmosphere
 - ELECTRA for precise quantification of the risks involved in the launch and the re-entry of a spacecraft



cnes









DISTRIBUTION

Why?

- To be more easily adopted by our contractors and then to get a better efficiency
- A good mean for cooperation
- Support to education (universities but even high schools)
- To become a reference (for example in the FSOA context)
- Making our tools more and more robust by increasing the amount of users.

How?

- ✤ For Basic libraries ⇒ « Open Source » distribution with an Apache 2.0 license
- ✤ For expert tools ⇒ « Freeware » distribution with a « Stela like » distribution
- ✤ For FSOA tools ⇒ specific licenses linked to FSOA use (except for STELA)



DISTRIBUTION

How?

- CNES dedicated Web site (not only for Flight Dynamics tools): <u>https://logiciels.cnes.fr</u>
- Wikis:
 - GENIUS (<u>http://genius.cnes.fr</u>)
 - GENOPUS (<u>http://genopus.cnes.fr</u>)
 - PATRIUS (<u>http://patrius.cnes.fr</u>)
 - PSIMU (<u>http://psimu.cnes.fr</u>)
- Training courses for GENIUS and PATRIUS





WELCOME TO THE PSIMU WIKI

WELCOME TO THE PSIMU WIKI

Conclusion



Since 2010, most of all the developments of CNES flight dynamics software use the Java[™] language based on basic libraries as PATRIUS for algorithms and GENIUS for Graphical User Interface (others use the Scilab[™] language with links between both).

☆ Thanks to the Java[™] (and Scilab[™]) portability on about all the existing platforms and Operating System, it becomes easier to distribute these tools outside CNES

- It has been decided to propose:
 - > an Open Source approach using Apache 2.0 license (or equivalent for Scilab[™] toolboxes) for basic libraries
 - freeware conditions with a "Stela like" license for higher level expert tools as PSIMU in their binary versions including their own GUI.
 - Specific licenses for FSOA tools (except STELA)
- Tools downloadable via the CNES dedicated Website (https://logiciels.cnes.fr)
- More information are available through several Wikis



THANK YOU FOR YOUR ATTENTION



7th International Conference on Astrodynamics Tools and Techniques, Oberpfaffenhofen, Germany, 6-9 November 2018