

Nanospace, an open source tool to help concurrent engineering teaming in cubesat preliminary design

Thibault Gateau, Lucien Senaneuch



OSCW 2020 - December 11, 2020

Phase 0/A: Preliminary Design

Specialized skills

- Sub-systems budgets
- Sub-systems architectures

Team work

- Communication
- Management



Eyesat synaptic view

Expectations vs reality



<https://www.kerbalspaceprogram.com/> - January 2018

Expectations vs reality

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
1	Spacecraft mission "NSS TEST CUBESAT"																System dissep							
2	Preliminary implementation of the cubesat 1U mission with the basic sensor as a payload without a propulsion system.						Flight profile				Motion & orientation control						Without margin							
3							Orbit type	LEO			On-board computer						Power	x						
9							Argument of Perigee (deg)	180			Product name						Nanodock DMC-3	Payload	x					
10	Mission start date		20/06/2020			RAAN (Omega) (deg)	7.13462			Weight						0.051	kg	Total dissipated power without any margin	x					
11	Mission end date		20/07/2020			Mean Anomaly (M)	0			Power consumption						-	W	Total dissipated power without system margin	x					
12	Operation time		30.00			Period	3429.24			Sensor								Total dissipated system power margin	x					
13	Mission Launch Segment						On-time per orbit				Sensor								Total dissipated power including system margins	x				
14	Launch date		20/06/2020			480				Sensor type						magnetometer								
15	Launch vehicle name		Soyuz			Communication				Product name						HMR2300								
16	Launch site name		Baikonur			Communication system				Weight						0.096		kg						
17	Spacecraft Name						Product name				Power consumption						0.45		W					
18	Payload						Weight				ADCs actuators													
19	Type						Energy consumption				Type						Reaction wheel							
20	Required power		10			Elevation Angle (deg)				Product name						NanoTorque C500-600								
21	Generated data		0			Slew Range (deg)				Weight						0.13		kg						
22	Weight		0.2			Uplink Frequency Choice				Power consumption						2.5		W						
23	Construction						Downlink Frequency Choice				Max Torque						0.019		N.m.s					
24	Structure type						Data rate				Max Angular Momentum						0.002		N.m.s					
25	Structure height						Data budget				Link budget													
26	Structure width						Subsystem				Telemetry type						Number							
27	Structure depth						Voltage				Sample rate, Hz						Word size, bit		Transmission speed, Type					
28	Structure mass						Current				Status						8		0.96		Orbit altitude			
29	Total mass						Time				Accelerometer						32		128		margin, UP			
30	Inx		0.00307			Magnetometer				Gyroscope						16		192		EH_No_Down				
31	Iyy		0.00267			Light sensor, BGRW				Alternative						16		192		margin, Down				
32	Izz		0.00307													16		192		margin, Down				
33	Mass Budget																16		192		margin, Down			
34	Without margin, kg		Margin, %		Margin, kg		Including margin, kg		% of total		Power budget				Propulsion system									
35	Structure		0.1		20 %		0.020		0.120		6.25 %		System mode - NOMINAL				Type				x			
36	Power		0.658		13.61 %		0.090		0.748		41.13 %		Power				Product name				x			
37	Communication		0.3566		5 %		0.005		0.115		6.84 %		Communication				Weight				0			
38	OSC		0.075		10 %		0.008		0.083		4.69 %		OSC				Power consumption				0			

Preliminary 1U design by Anton Poltoradnev - ISAE-Supaero

Software Tools and libraries

Candidate Software List
Specification for Nanostar**Power budget :**

- IDM-CIC
- Home made scripts

Structure :

- IDM-CIC
- **Catia** [X]

Mission analysis :

- GMAT
- Satorb
- DOCKS
- libs :
Celestlab
Poliastro
Orekit

**Visualization Tools :**

- Home made
- VTS
- Celestia
- Cosmographia

**Dynamic database :**

- IDM-CIC (ECSS Standard)
- **Valispace** database
- Home made database

VALISPACE

**LOS :**

- Stella

**Thermal :**

- **Systema thermica**
- Home made scripts

Link Budget :

- AMSAT
- Home made



Trendy way: Concurrent Design Engineering

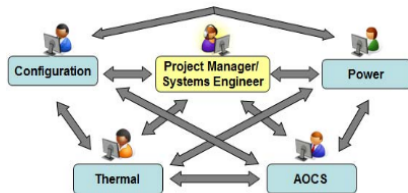
Efficient data management and exchanges

- create data models
- shared common models
- data update should be propagated in each expert tool

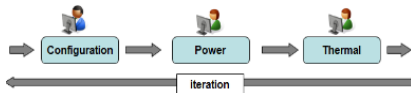
Strategy?

- take an existing CDE?
- adapt from libs?
- adapt from MBSE?

Concurrent Design / Engineering Process



Sequential Engineering (with iterations):



[Di Domizio and Gaudenzi, 2008]

Review [Knoll et al., 2018]

Goal: Concurrent design engineering with some requirements

- Student challenges on cubesats preliminary design
- Multiple access from multiple sites (5 institutions)
- Database rather than spread-sheets [Gordon, 1999]
- Allow users to keep their favorite expert tools
- Student friendly & **Open-source**

Interreg
Sudoe



EUROPEAN UNION

NANO STAR

European Regional Development Fund



TÉCNICO LISBOA



UNIVERSIDADE
BEIRA INTERIOR

uc3m

Universidad
Carlos III
de Madrid



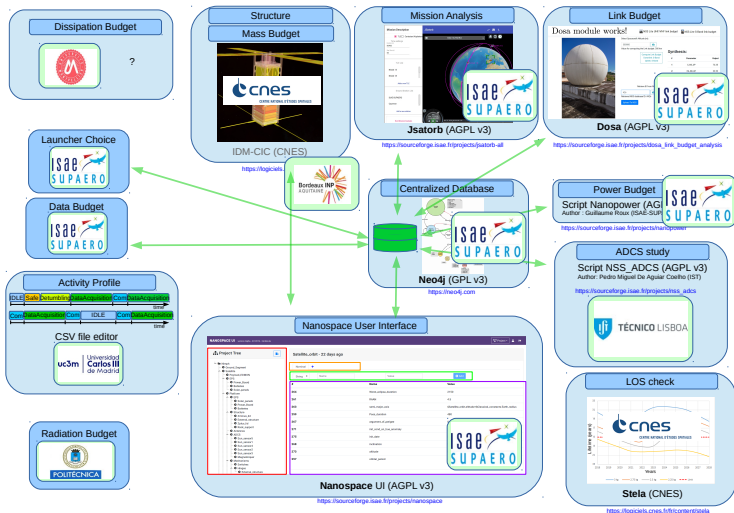
POLITÉCNICA

isae
SUPAERO



Bordeaux INP
AQUITAINE

NSS: Nanostar Software Suite



- 1 Nanospace - Demo
 - Demo: First connection
 - Nanospace-UI
 - Demo: First interactions

- 2 Nanospace - Architecture
 - Technical choices
 - Running Architecture
 - Interaction with third party applications

- 1 Nanospace - Demo
 - Demo: First connection
 - Nanospace-UI
 - Demo: First interactions
- 2 Nanospace - Architecture

Demo: First connection

Using your favorite Web Browser

- Go to <https://dcas-nanostar.isae.fr>
- You can **Subscribe** and **Login**
- Import an example project json project file:
<https://gitlab.isae-supero.fr/nanostar/nanospace/nanospace-user>

The screenshot shows the Nanospace-UI interface. On the left is a 'Project Tree' with a hierarchical view of project components. On the right, the 'OrbitParameter' section displays a table with columns for ID, Name, and Value. A 'Filter' input field is visible above the table.

ID	Name	Value
408	altitude_fm	300
406	altitude	30000.0
445	semi_major_axis_fm	6671
446	eccentricity	0.02
447	inclination	98
448	argument_of_perigee	0
449	mean	0
450	initial_argument_perigee	0
451	initial_mean_longitude	0.0
452	semi_major_axis	6671800.0



Nanospace-UI

The screenshot displays the Nanospace-UI interface. At the top, a dark purple header contains the text "NANOSPACE UI" and "version Alpha - 07/2019 - v16c64c4a" on the left, and "Project" with user and refresh icons on the right.

The main interface is divided into two panels:

- Project Tree:** A hierarchical tree view on the left under the heading "Project Tree". The root node is "TutorialNanostar", which is expanded to show several sub-nodes, including "OrbitalParameter" (highlighted in blue), "Refresh", "Synthesis_Global", "Synthesis_ADCS", "Synthesis_Power", "Synthesis_Telecom", "Ground_station", "Platform", "OBC", "Structure", "Antenna", "ADCS", "EPS", "Batteries", "Solar_panels", "Power_board", "Tranceiver", "Payload", "Ground_segment", "S-band-Guyane", and "UHF-TLS".
- OrbitalParameter:** A panel on the right titled "OrbitalParameter". It features a filter bar with "StandBy" (red dot), "Visibility" (red dot), "Survival" (red dot), and "Nominal" (blue plus sign). Below this is a form with a "String" dropdown, a "Name" input field, a "Value" input field, and a blue "Add" button. A table below the form lists orbital parameters with columns for "#", "Name", and "Value".

#	Name	Value
409	altitude_km	300
466	altitude	300000.0
445	semi_major_axis_km	6671
446	eccentricity	0.02
447	inclination	98
448	argument_of_periapsis	0
449	raan	0
450	initial_eccentric_anomaly	0
443	initial_Mean_anomaly	0.0
469	semi_minor_axis	6671000.0

At the bottom left, the text "Nanostar project" is visible. At the bottom right, the copyright notice "© 2019 - AGPL v3 Licence: ISAE-SUPAERO" is displayed.

View of a project in a browser

First interactions

NANOSPACE UI version Alpha - 07/2019 - v16c64c4a Project + 👤 ↔

Project Tree

- TutorialNanostar
 - OrbitalParameter
 - Refresh
 - Synthesis_Global
 - Synthesis_ADCS
 - Synthesis_Power
 - Synthesis_Telecom
 - Ground_station
 - Platform
 - OBC
 - Structure
 - Antenna
 - ADCS
 - Sun_sensors
 - Magnetorquer
 - EPS
 - Batteries
 - Solar_panels
 - Power_board
 - Tranceiever
 - Payload
 - Ground_segment
 - S-band-Guyane
 - UHF-TLS

OrbitalParameter

StandBye ● Visibility ● Survival ● Nominal +

String Add

#	Name	Value
409	altitude_km	300
466	altitude	300000.0
445	semi_major_axis_km	6671
446	eccentricity	0.02
447	inclination	98
448	argument_of_periapsis	0
449	raan	0
450	initial_eccentric_anomaly	0
443	initial_Mean_anomaly	0.0
469	semi_minor_axis	6671000.0

Nanostar project © 2019 - AGPL v3 Licence: ISAE-SUPAERO

View of a project in a browser

First interactions

NANOSPACE UI version Alpha - 07/2019 - v1c64c4a Project + 👤 ↔

Project Tree

- ↳ TutorialNanostar
 - ⚙️ OrbitalParameter
 - ⚙️ Refresh
 - ⚙️ Synthesis_Global
 - ⚙️ Synthesis_ADACS
 - ⚙️ Synthesis_Power
 - ⚙️ Synthesis_Telecom
 - ⚙️ Ground_station
 - ↳ Platform
 - ⚙️ OBC
 - ↳ Structure
 - ⚙️ Antenna
 - ⚙️ ADACS
 - ⚙️ Sun_sensors
 - ⚙️ Magnetorquer
 - ↳ EPS
 - ⚙️ Batteries
 - ⚙️ Solar_panels
 - ⚙️ Power_board
 - ⚙️ Tranceceiver
 - ↳ Payload
 - ⚙️ Ground_segment
 - ⚙️ S-band-Guyane
 - ⚙️ UHF-TLS

OrbitalParameter

StandBye 🔴 Visibility 🔴 Survival 🔴 Nominal 🔴 +

➕ Add

#	Name	Value
409	altitude_km	300
466	altitude	300000.0
445	semi_major_axis_km	6671
446	eccentricity	0.02
447	inclination	98
448	argument_of_periapsis	0
449	raan	0
450	initial_eccentric_anomaly	0
443	initial_Mean_anomaly	0.0
469	semi_minor_axis	6671000.0

Nanostar project © 2019 - AGPL v3 Licence: ISAE-SUPAERO

View of a project in a browser

First interactions

NANOSPACE UI version Alpha - 07/2019 - v16c64c4a Project + 👤 ↔

Project Tree

- TutorialNanostar
 - OrbitalParameter
 - Refresh
 - Synthesis_Global
 - Synthesis_ADCS
 - Synthesis_Power
 - Synthesis_Telecom
 - Ground_station
 - Platform
 - OBC
 - Structure
 - Antenna
 - ADCS
 - Sun_sensors
 - Magnetorquer
 - EPS
 - Batteries
 - Solar_panels
 - Power_board
 - Tranceiver
 - Payload
 - Ground_segment
 - S-band-Guyane
 - UHF-TLS

OrbitalParameter

StandBy Visibility Survival Nominal +

String *	Name	Value	Add
#	Name	Value	
409	altitude_km	300	
466	altitude	300000.0	
445	semi_major_axis_km	6671	
446	eccentricity	0.02	
447	inclination	98	
448	argument_of_periapsis	0	
449	raan	0	
450	initial_eccentric_anomaly	0	
443	initial_Mean_anomaly	0.0	
469	semi_minor_axis	6671000.0	

Nanostar project

© 2019 - AGPL v3 Licence: ISAE-SUPAERO

View of a project in a browser

First interactions

NANOSPACE UI version Alpha - 07/2019 - vfc64c4a Project + 👤 ↔

Project Tree

- TutorialNanostar
 - OrbitalParameter**
 - Refresh
 - Synthesis_Global
 - Synthesis_ADACS
 - Synthesis_Power
 - Synthesis_Telecom
 - Ground_station
 - Platform
 - OBC
 - Structure
 - Antenna
 - ADACS
 - Sun_sensors
 - Magnetorquer
 - EPS
 - Batteries
 - Solar_panels
 - Power_board
 - Tranceiver
 - Payload
 - Ground_segment
 - S-band-Guyane
 - UHF-TLS

OrbitalParameter

StandBy Visibility Survival Nominal +

String *	Name	Value	Add
#	Name	Value	
409	altitude_km	300	
466	altitude	300000.0	
445	semi_major_axis_km	6671	
446	eccentricity	0.02	
447	inclination	98	
448	argument_of_periapsis	0	
449	raan	0	
450	initial_eccentric_anomaly	0	
443	initial_Mean_anomaly	0.0	
469	semi_minor_axis	6671000.0	

Nanostar project

© 2019 - AGPL v3 Licence: ISAE-SUPAERO

View of a project in a browser

First interactions

NANOSPACE UI version Alpha - 07/2019 - v16c64c4a Project + 👤 ↔

Project Tree

- TutorialNanostar
 - OrbitalParameter
 - Refresh
 - Synthesis_Global
 - Synthesis_ADCS
 - Synthesis_Power
 - Synthesis_Telecom
 - Ground_station
 - Platform
 - OBC
 - Structure
 - Antenna
 - ADCS
 - Sun_sensors
 - Magnetorquer
 - EPS
 - Batteries
 - Solar_panels
 - Power_board
 - Tranceiver
 - Payload
 - Ground_segment
 - S-band-Guyane
 - UHF-TLS

OrbitalParameter

StandBye 🔴 Visibility 🔴 Survival 🔴 Nominal +

String * Name Value ➕ Add

#	Name	Value
409	altitude_km	300
466	altitude	300000.0
445	semi_major_axis_km	6671
446	eccentricity	0.02
447	inclination	98
448	argument_of_periapsis	0
449	raan	0
450	initial_eccentric_anomaly	0
443	initial_Mean_anomaly	0.0
469	semi_minor_axis	6671000.0

Nanostar project © 2019 - AGPL v3 Licence: ISAE-SUPAERO

View of a project in a browser

First interactions

NANOSPACE UI version Alpha - 07/2019 - v16c54c4a

Project + [User] [Refresh]

Project Tree

- TutorialNanostar
 - OrbitalParameter
 - Refresh
 - Synthesis_Global
 - Synthesis_ADCS
 - Synthesis_Power
 - Synthesis_Telecom
 - Ground_station
 - Platform
 - OBC
 - Structure
 - Antenna
 - ADCS
 - Sun_sensors
 - Magnetorquer
 - EPS
 - Batteries
 - Solar_panels
 - Power_board
 - Tranceiver
 - Payload
 - Ground_segment
 - S-band-Guyane
 - UHF-TLS

OrbitalParameter

StandBy Visibility Survival Nominal +

String Name Value

#	Name	Value
409	altitude_km	300
466	altitude	300000.0
445	semi_major_axis_km	6671
446	eccentricity	0.02
447	inclination	98
448	argument_of_periapsis	0
449	raan	0
450	initial_eccentric_anomaly	0
443	initial_Mean_anomaly	0.0
469	semi_minor_axis	6671000.0

Nanostar project

© 2019 - AGPL v3 Licence: ISAE-SUPAERO

View of a project in a browser

- 1 Nanospace - Demo
- 2 Nanospace - Architecture
 - Technical choices
 - Running Architecture
 - Interaction with third party applications

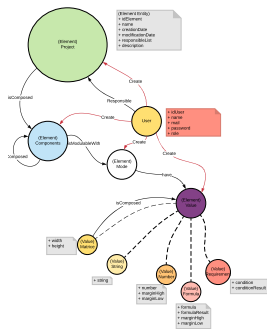
Requirements

- Web App
<platform independent>
- REST API
<ease third party connection>
- ACID¹ property
<concurrent access>

Technical choices

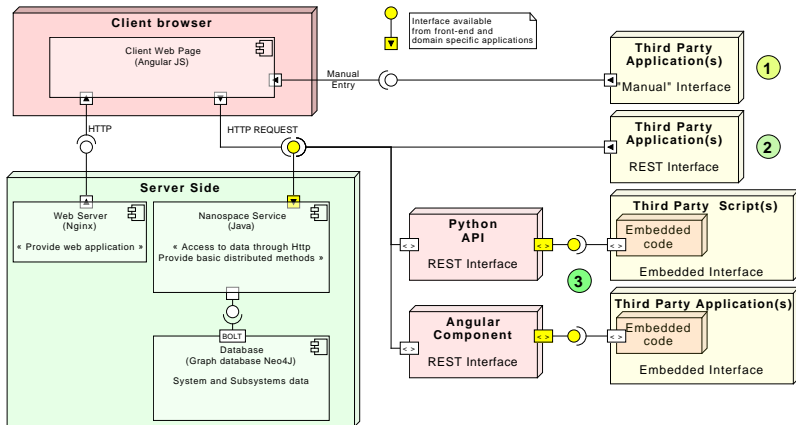
- Database: Neo4j
- Front-end: Angular
- Back-end: Spring Boot
(Neo4j direct compatibility)

1: Atomicity, Consistency, Isolation, Durability



Database model prototype

Running Architecture



REST interface

Easy to run your own scripts

- Direct connection to the REST interface
- Python API provided (nanospace.py)
- You can use your favorite libs (poliastro, orekit, celestlab...)

Python example

```
2 from nanospace import Nanospace
3 nanospace = Nanospace(srvAddr, usr, pw)
4 altitude = getNanospaceString(nanospace, altitudeID)
5 d = getMarginDown(altitudeKm, dataJsonFile)
6 EB_N0 = round(d['Eb_N0_Down'], 1)
7 nanospace.update_string_value(Eb_N0_DownID, 'EB N0', str(EB_N0)+" in [dB]")
```

Adding an Angular component

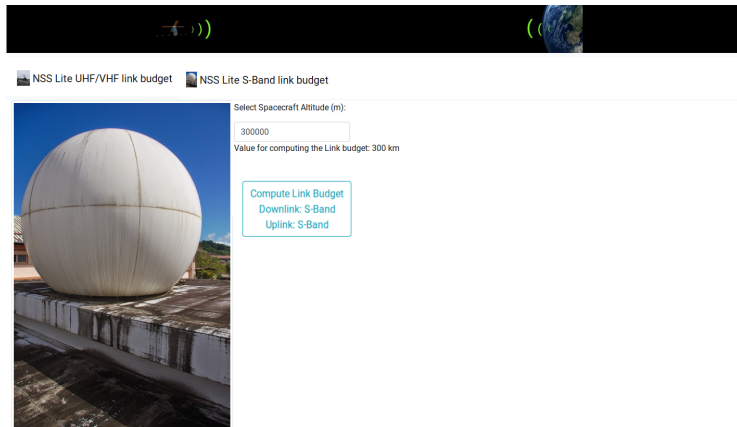
Angular component example available on-line

- Angular Component (3 lines of code)
- Available online:
www.npmjs.com/package/ngx-nanospace-client-lib

Embedded component example

```
/*app.component.html*/  
<nano-input-id [(ngModel)]="idImported"></nano-input-id>  
<nano-input-value [(ngModel)]="valueImported"></nano-input-value>  
<nano-import-export-value [(ngModel)]="valueImported"></nano-import-export-value>
```

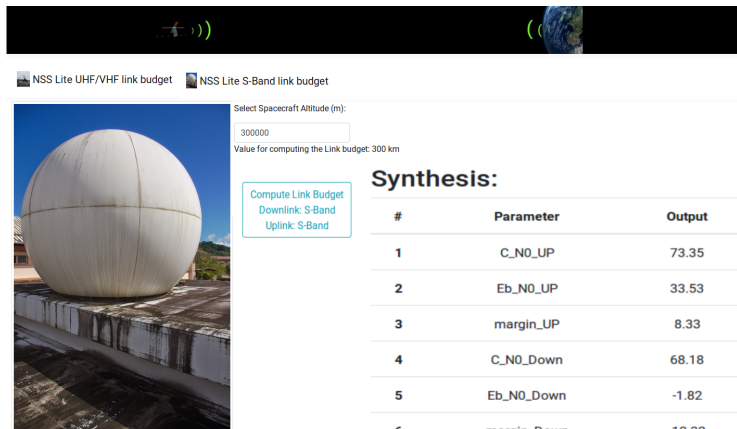
Example with Angular component



The screenshot shows a web application interface with two tabs: "NSS Lite UHF/VHF link budget" and "NSS Lite S-Band link budget". The "NSS Lite S-Band link budget" tab is active. On the left, there is a photograph of a large white spherical antenna on a rooftop. To the right of the photo, the text "Select Spacecraft Altitude (m):" is followed by a text input field containing "300000". Below the input field, it says "Value for computing the Link budget: 300 km". A blue-bordered button labeled "Compute Link Budget" is present, with two sub-links: "Downlink: S-Band" and "Uplink: S-Band".

Pedagogical link budget module view

Example with Angular component



Select Spacecraft Altitude (m):

Value for computing the Link budget: 300 km

Compute Link Budget
Downlink: S-Band
Uplink: S-Band

Synthesis:

#	Parameter	Output
1	C_N0_UP	73.35
2	Eb_N0_UP	33.53
3	margin_UP	8.33
4	C_N0_Down	68.18
5	Eb_N0_Down	-1.82
6	margin_Down	-12.32

Pedagogical link budget module view

Example with Angular component

Select Spacecraft Altitude (m):
300000
Value for computing the Link budget: 300 km

Compute Link Budget
Downlink: S-Band
Uplink: S-Band

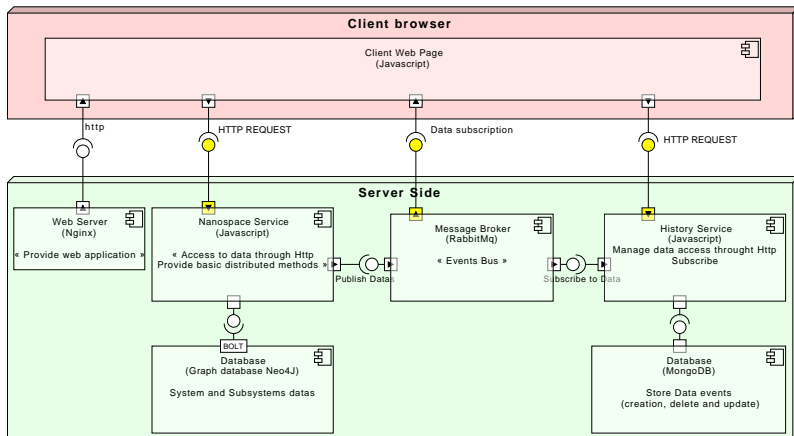
Retrieve ID from NSS database:
426
Retrieved NSS database ID: #426
Upload To NSS

Synthesis:

#	Parameter	Output
1	C_NO_UP	73.35
2	Eb_NO_UP	33.53
3	margin_UP	8.33
4	C_NO_Down	68.18
5	Eb_NO_Down	-1.82
6	margin_Down	-12.32

Pedagogical link budget module view

Targeted Architecture



Take Home Message

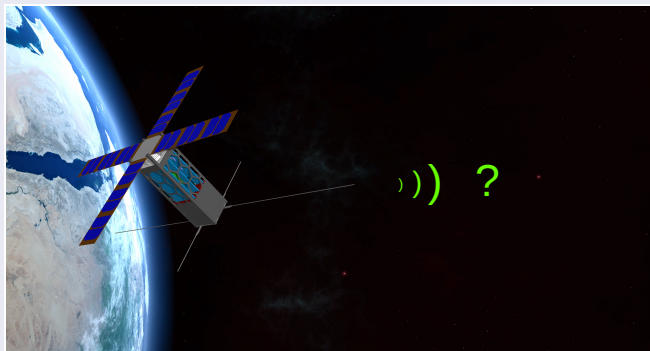
- Easy to integrate to third party application
- Concurrent access, remote-located team context
- Available source code (AGPL v3):
<https://gitlab.isae-superaero.fr/nanostar/nanospace>
- Available test server:
<https://dcas-nanostar.isae.fr/>
- Web-service - Docker Version

Future Works

- Event management
- Formal pipeline management, top bottom approach (MBSE?)
- Life cycle beyond phase 0/A (up to C...)
- Check resilience (interaction with DOCKS? GMAT?)

Thank you for your attention!

Any question ?



Special thanks to:
Marie-Carmen Fauré, Frédéric Fal, Maxime Syidalza,
Jacques Villemur, Ludovic Bosseaux



Di Domizio, D. and Gaudenzi, P. (2008).
A model for preliminary design procedures of satellite systems.
[Concurrent Engineering](#), 16(2):149–159.



DLR (2019).
Virtual satellite.
https://www.dlr.de/sc/en/desktopdefault.aspx/tabid-5135/8645_read-8374/.



ESA (2019).
Ocdt.
<https://ocdt.esa.int>.



Gordon, K. J. (1999).
Spreadsheet or database: Which makes more sense?
[Journal of Computing in Higher Education](#), 10(2):111–116.



Knoll, D., Fortin, C., and Golkar, A. (2018).
Review of concurrent engineering design practice in the space sector: state of the art and future perspectives.
In [2018 IEEE International Systems Engineering Symposium \(ISSE\)](#), pages 1–6. IEEE.



Le Gal, J.-L. and Lopes, P. R. (2016).
Idm-cic.
<https://www.clever-age.com/fr/case-studies/cnes-une-application-de-modelisation-3d/>.



NASA.
GMAT.
<http://gmatcentral.org/>.
Accessed: 2017-03-28.



Rheagroup (2019).

Cdp4.

[https://www.rheagroup.com/fr/news/cdp4-open-source-community-edition.](https://www.rheagroup.com/fr/news/cdp4-open-source-community-edition)