m3 SYSTEMS
RESEARCH-CONSULTING-ENGINEERING
We guide you

The TASTE project: Results and lessons learnt

ESA - ESL Day 2011
September 19th, 2011

Marc POLLINA , Yann LECLERC
– M3 Systems

SUMMARY

- **Scope & Overview**
- **TASTE's demonstrator & HW/SW co-design extension**
- **Current results**
- **Lessons learnt & future work**

Critical systems design

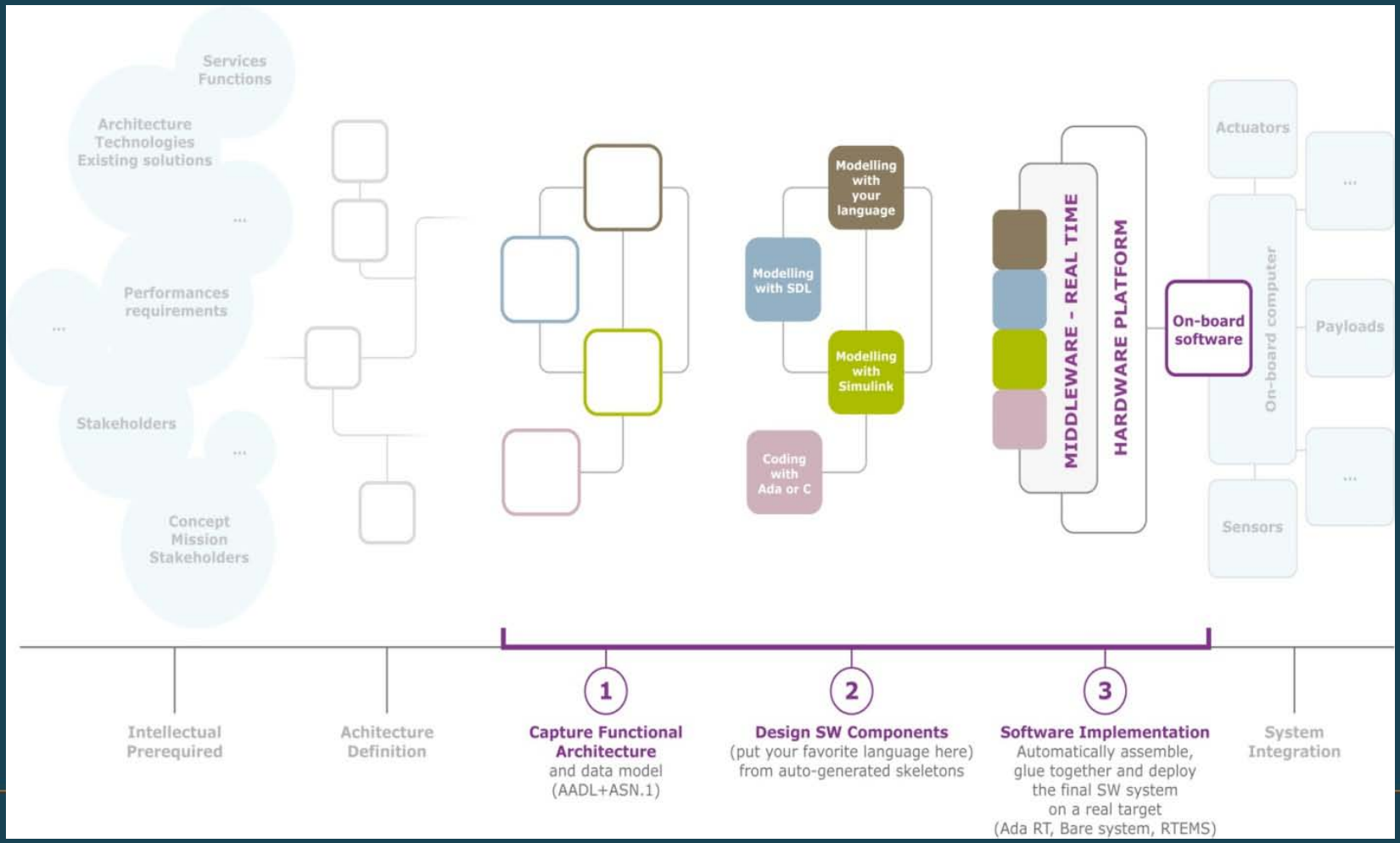
Improving the efficiency of engineering process while increasing functional complexity of critical systems is a well-known challenge and probably one of the keys for competitiveness of space industry.

Answering this challenge is a hard work. One can be faced with wide-ranging issues related to maturity of tools and technologies, but also companies' organisations, cultural convictions of stakeholders,…

Yet, our conviction is that progress is possible, and that step by step industrial implementation of **innovative model based engineering process is feasible today**.

Engineering process

Operationnal needs, interfaces definition, software & hardware design



TASTE project (2010/2011)

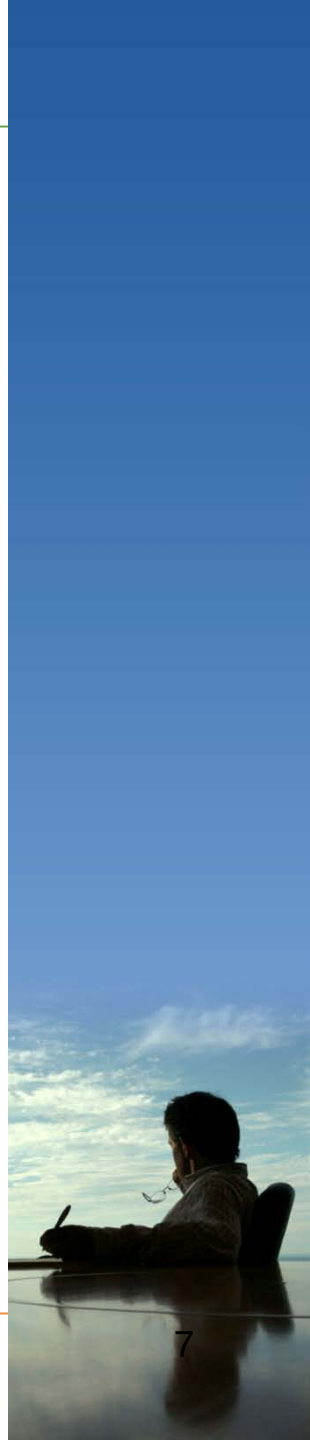
- **Exposing TASTE to industrial experts** and real cases, by opening the way for new concepts to be integrated in the methodology and toolset.

- **Demonstrate a study case**, representative of a satellite architecture

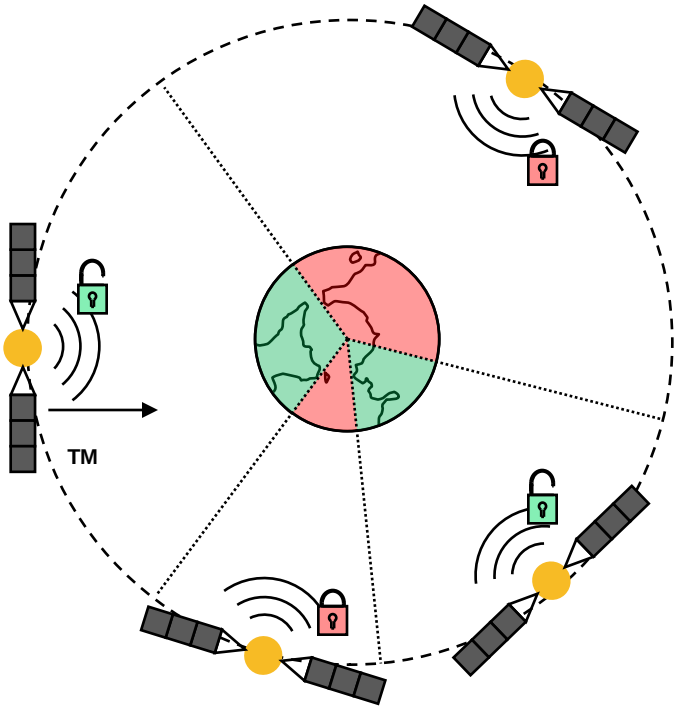
- **Extension of the toolset capabilities**, by integrating hardware/software **codesign** capabilities into TASTE

Project team

- **Scope & Overview**
- **TASTE's Demonstrator & HW/SW co-design extension**
- **Current results**
- **Lessons learnt and future work**

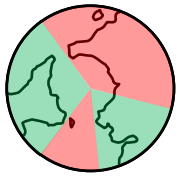


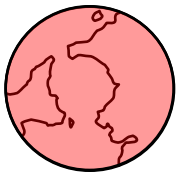
Demonstrator - Mission Concept



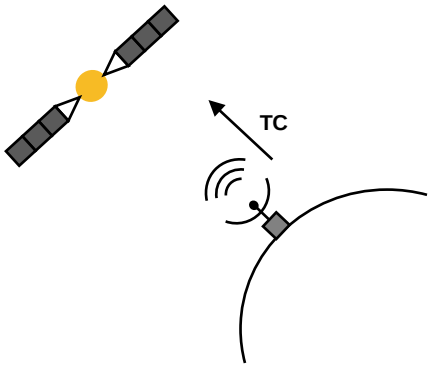
The satellite provided a TM/TC encryption service working on geographical windows, as a function of its position

Demonstrator - Mission Concept





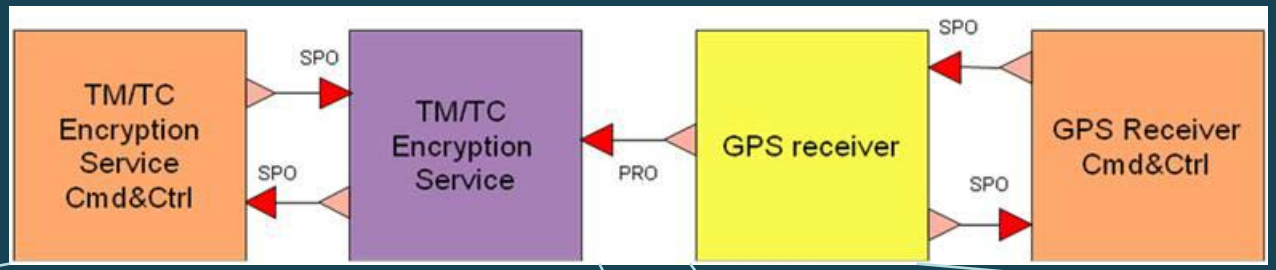
Two functional modes are available : geographical encryption and continuous encryption.

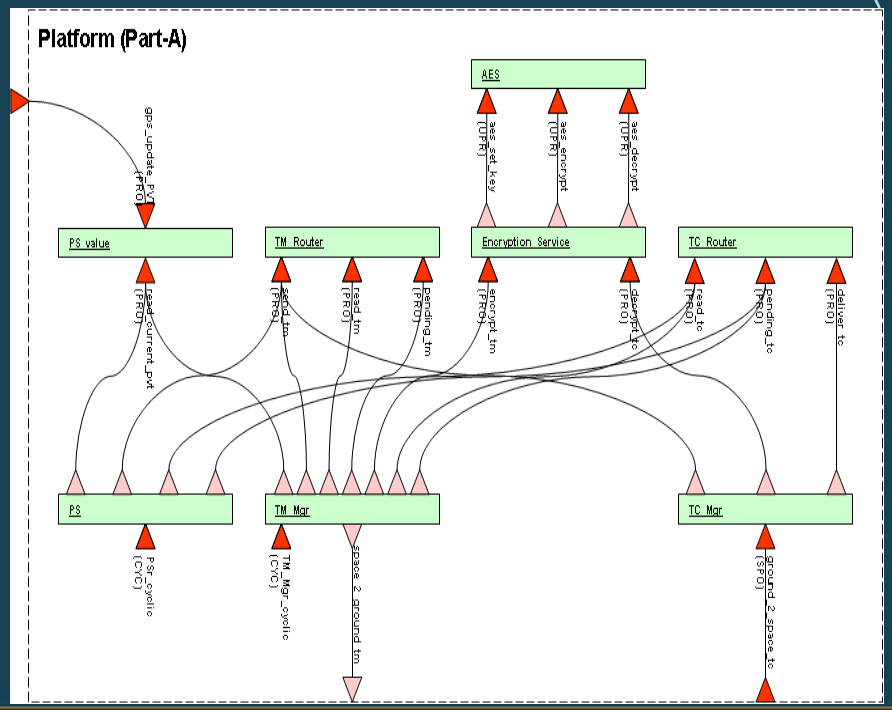


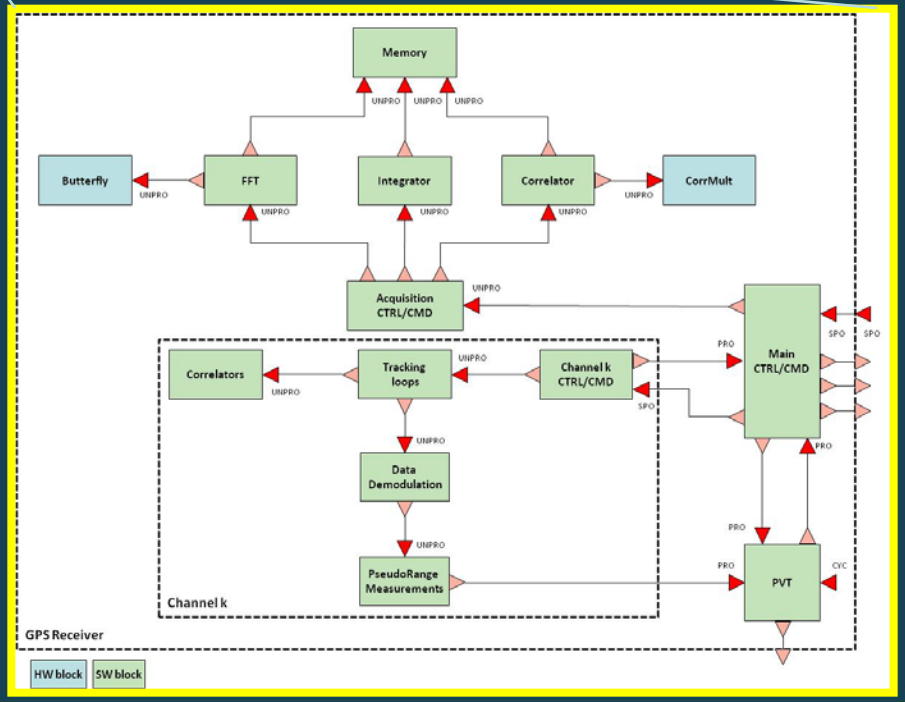
The satellite can be controlled from the ground (mode change, position sensor,…) through encrypted or not TC.

Functional architecture

Modélisation graphique de l'architecture système (AADL)

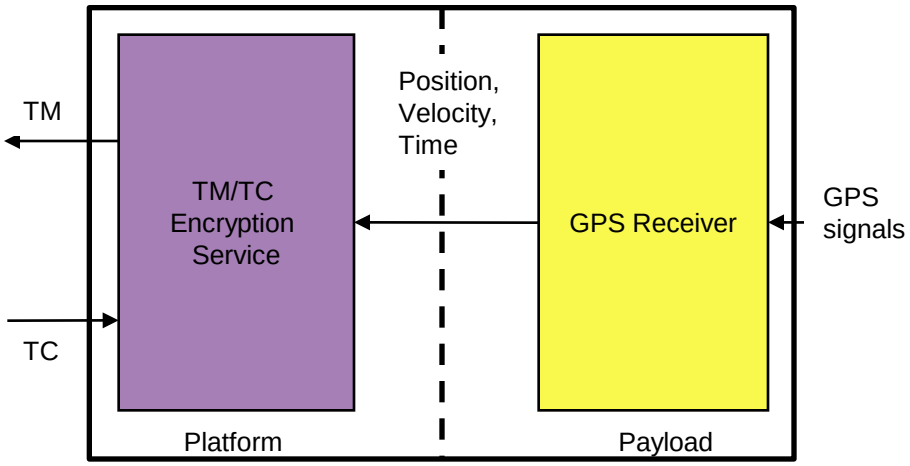






Demonstrator - Satellite Architecture

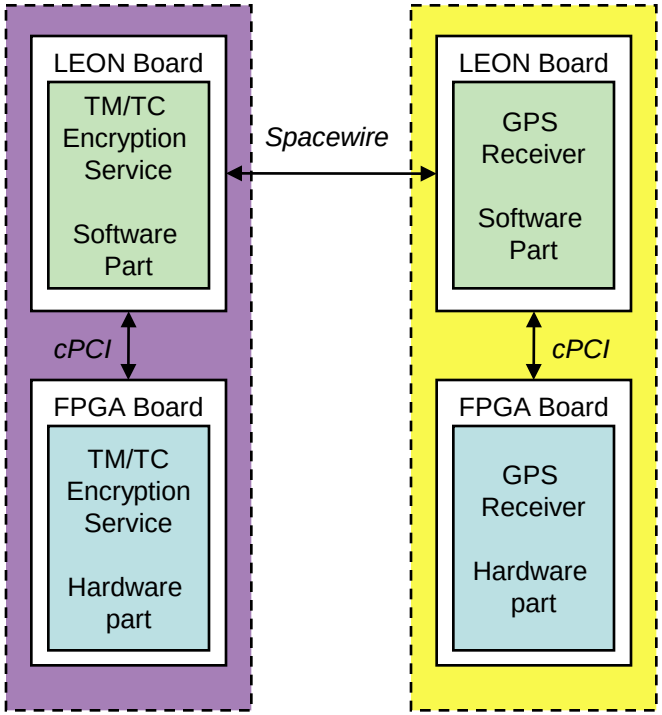
The platform computes the geographical area flown over, based on the position sent by the payload. It sends encrypted or not TMs, and processes encrypted or not incoming TCs.



The payload is a GPS receiver, computing satellite's position from GPS signals, and providing the results to the platform.

Demonstrator - Implementation Target

Both platform and payload are split into software and hardware parts.



LEON2FT ASIC processor is the final target processor of the software parts. **FPGA board** (Virtex 4) are used for the hardware parts.

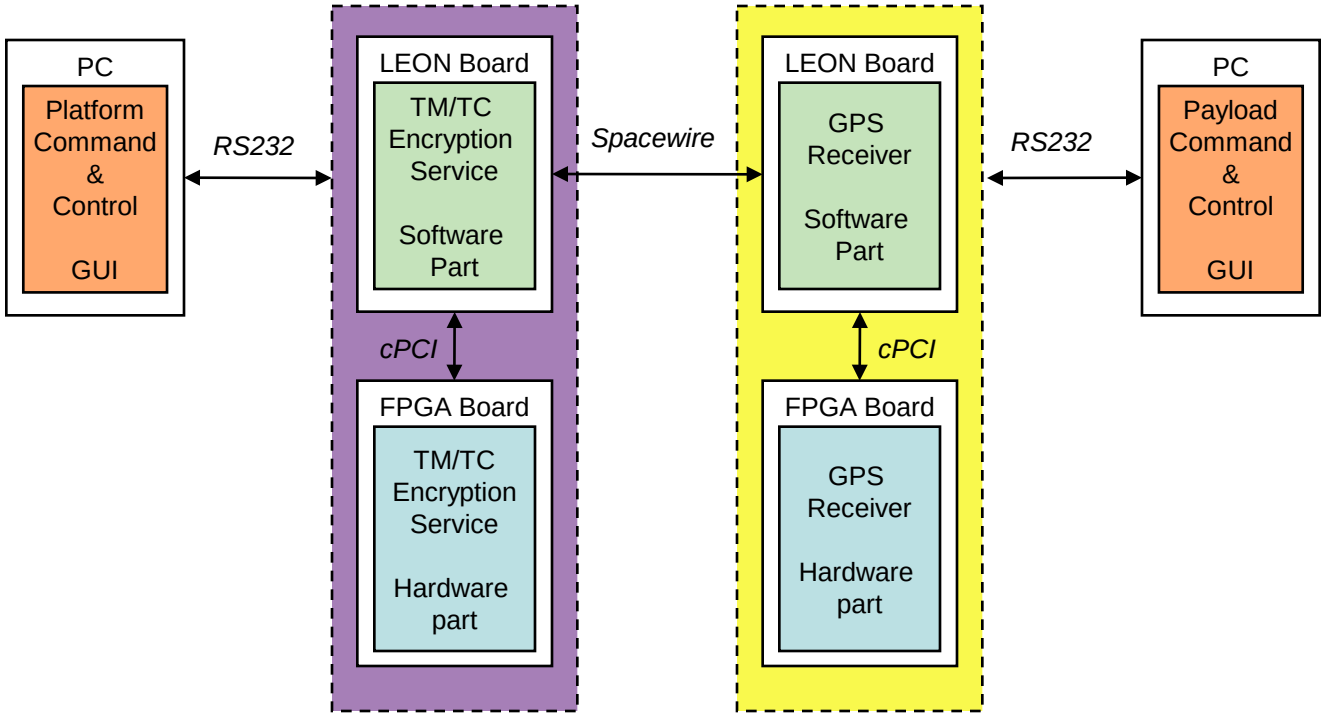
LEON and FPGA board are plugged into a **cPCI backplane**. Payload and platform are linked through **SpaceWire**.

Demonstrator - Implementation Target

Two GR-RASTA racks are used for the demonstrator

Demonstrator - Implementation Target

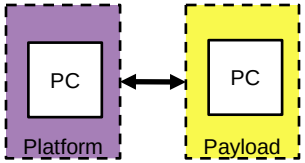
Ground stations are simulated by PCs



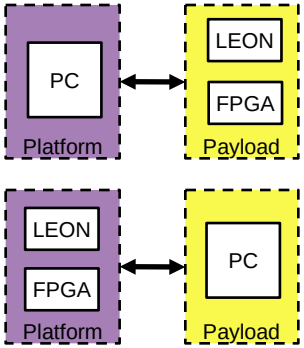
Demonstrator – Development Process

This development process mimics real development process.

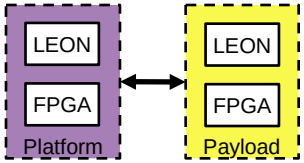
Demonstrator's reliability is strengthen by the gradual integration/validation.



Step 1: Platform and payload are simulated by PCs



Step 2: Simulated platform and payload are used to develop real functions.



Step 3: Platform and payload are integrated into the demonstrator.

TASTE aims to make this process more efficient and reliable.

HW/SW codesign extension

TASTE had mainly been designed to cope with software systems.

However, the **ASSERT methodology**, and the overall construct of TASTE are **very general**, designed to deal with various programming languages/tools, and can be extended to hardware systems.

One objective of the project was to add hardware/software codesign capabilities to TASTE.

HW/SW codesign extension

Extension to a new language: VHDL

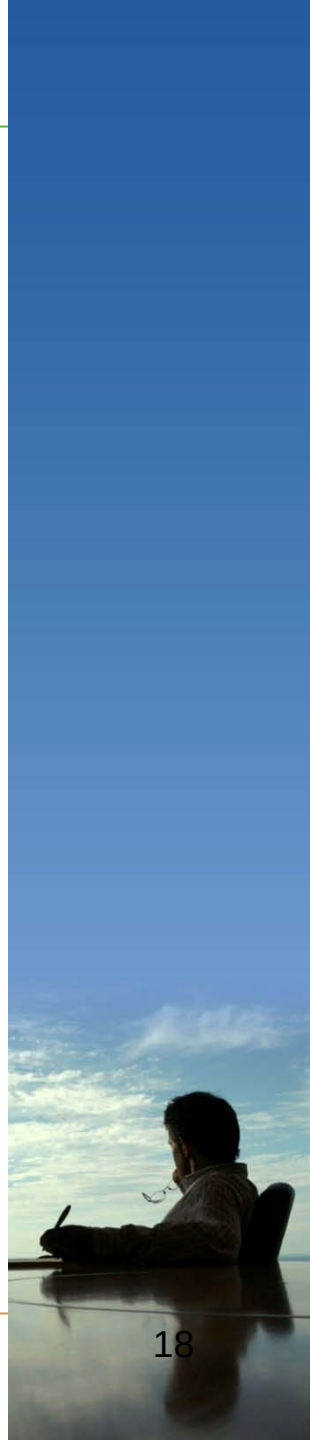
VHDL and SystemC can be selected as programming languages when modeling a system. VHDL and SystemC code skeletons, that have to be filled by system designers, are automatically generated by TASTE.

Addition of a new target: FPGA, via PCI bus

All codes required for communication with hardware components (“glue” codes) are automatically generated by TASTE: a driver on software side, PCI interface, bus, wrappers, and registers on hardware side.

This evolution gives TASTE basic HW/SW capabilities.

- **Scope & Overview**
- **TASTE's Demonstrator & HW/SW co-design extension**
- **Current results**
- **Lessons learnt and future work**



Current Result - Codesign extension

TASTE is now supporting **VHDL** has **programming languages**.

The **GR-CPCI-XC4V** board (RASTA's FPGA board) has been added as a **new target** of TASTE.

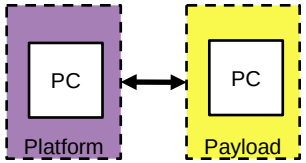
A **generic hardware architecture** (VHDL) targeting the GR-CPCI-XC4V board has been designed. This architecture is made of: a PCI interface, a bus, a template VHDL IP with communication registers. The VHDL template is the "**code skeleton**" to be filled by the HW designer**.**

A **driver** targeting the generic hardware architecture has been designed.

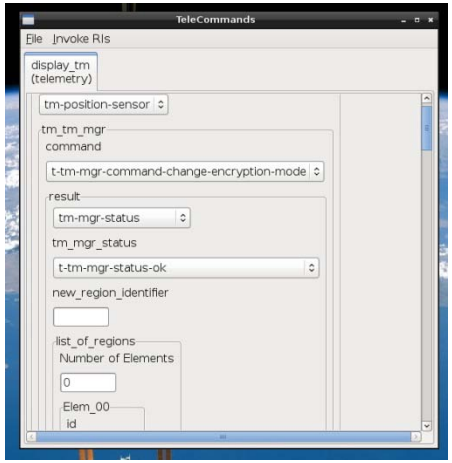
Generic hardware architecture and driver are **automatically generated** by TASTE based on high level specification.

This codesign extension has been successfully tested on basic applications.

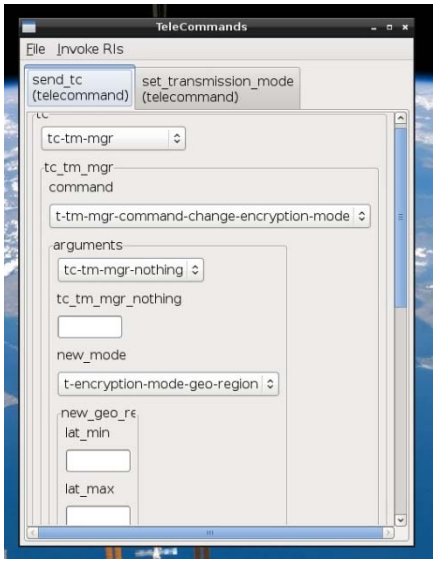
Current Result - Demonstrator



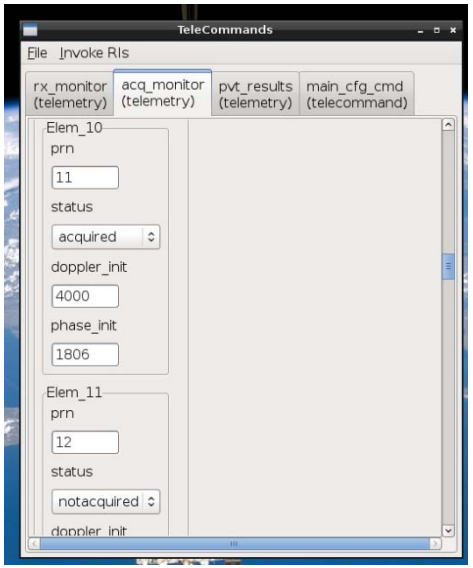
Step 1: Platform and payload are simulated by PCs



Telemetry GUI



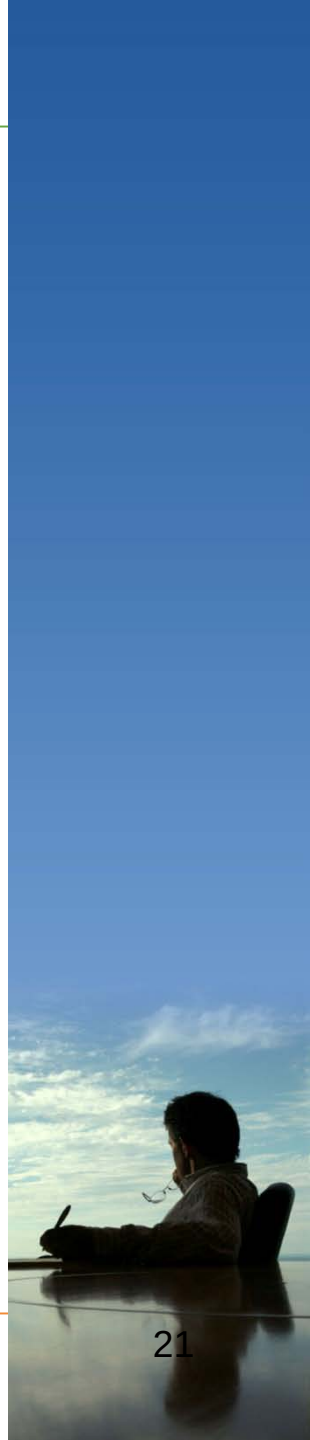
Telecommand GUI



GPS Receiver GUI

Step 1 of the development process has been successful. The whole demonstrator has been successfully tested on various scenario.

- **Scope & Overview**
- **TASTE's Demonstrator & HW/SW co-design extension**
- **Current results**
- **Lessons learnt and future work**



Lessons learnt and future work

TASTE methodology is a great asset when designing a complex system: the designer benefits of all the **concepts f**ollowed by TASTE (Ravenscar Computational Model, Component Based design,…).

The TASTE toolset is a **prototype**, and as such is not very well suited for the development of complex systems. Nevertheless, we've seen that a designer can benefits a lot from such a toolset, which enforced, by design, all the concepts followed by the methodology, and simplify the integration work.

Lessons learnt and future work

This feasibility of the “missing link” between high level system description and equipments design has been demonstrated. Yet, all difficulties have not been overcome, and significant work has to be done:

- **End-to end Methodology** : definition & engineers training, focus on key concept
- **Modular Toolset** : breakdown into mature components, to facilitate validation & adoption
- **Industrial constraints** : implementing a new process is not for free, industrial evolutions needs to be facilitated.
- **Implement prototype projects** : get a technical result & improve the process

Thanks for your attention!

Marc POLLINA ,
pollina@m3systems.net
+33 (0)5 62 23 10 80