AEROFAST: Functional and Real-Time Simulation for Aerocapture GNC Assessment

M. Hagenfeldt, A. Mafficini, V. Fernández, S. Gómez, C. Valle, P. Palomo, A. Latorre

DEIMOS Space, S.L.U.

Aerospace Engineering Business Unit

SESP 2012, Noordwijk, 25-27 September

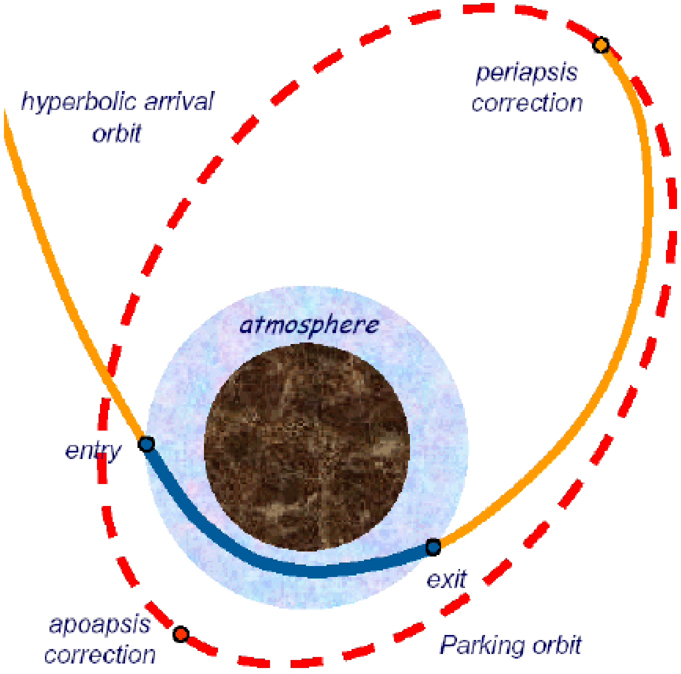
Introduction

The AEROFAST Project

- *AEROFAST: AEROcapture for Future SpAce Transportation*
- Collaborative project funded by the EC (FP7)
- Aimed at raising the maturity of aerocapture technology in Europe (from TRL 2-3 to TRL 3-4)
- Aerocapture allows for a large mass saving at launch (30 %)
- In preparation of future human exploration missions
- Industrial consortium: 12 partners coordinated by Astrium ST
- Ended successfully in September 2011

The AEROFAST Demonstration Mission

- Insertion from a hyperbolic fly-by into Martian orbit, through the atmosphere
- Three phases
  - Pre-aerocapture: hyperbolic path to the entry interface point (EIP)
  - Main aerocapture: from EIP to atmosphere exit
  - Post-aerocapture: transfer to a parking orbit (sun-synchronous low orbit)



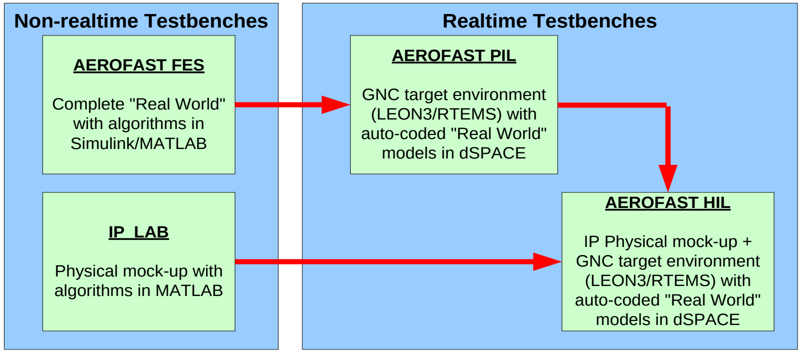
AEROFAST Objectives

- OBJ1: Preliminary definition of the aerocapture demonstration
- OBJ2: Make a significant progress by increasing the TRL of the planetary relative navigation and the aerocapture algorithm up to 5
- ***OBJ3**: Build a breadboard to test in real time the pre-aerocapture and aerocapture GNC algorithms*
- OBJ4: Demonstrate/prototype the thermal protection system for such a mission
- OBJ5: Define on-board instrumentation for aerocapture phase recovery

Need of functional and real-time simulation for GNC evaluation

Approach

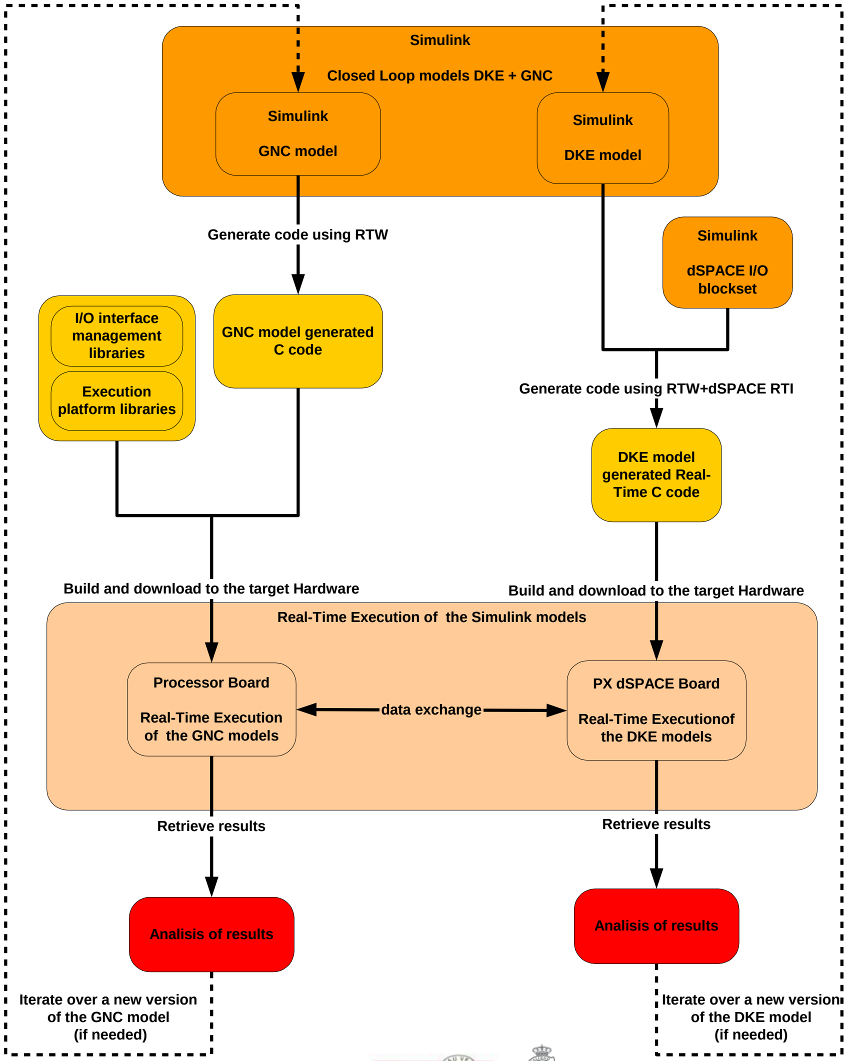
- First choice: to validate separately the pre-aerocapture and aerocapture GNC designs
  - Interest on setting aerocapture initial conditions independently from pre-aerocapture final conditions
- For each phase, two facilities are needed:
  - Non-Realtime: AEROFAST Functional Engineering Simulators
  - Real time: AEROFAST Real-Time Testbenches (PIL and HIL configurations)
- Our approach: **unified architecture**
  - Supports GNC Verification & Validation process
  - Intended for model reuse across facilities (FES → PIL → HIL)



Pre-aerocapture development logic

Testbench Development Cycle

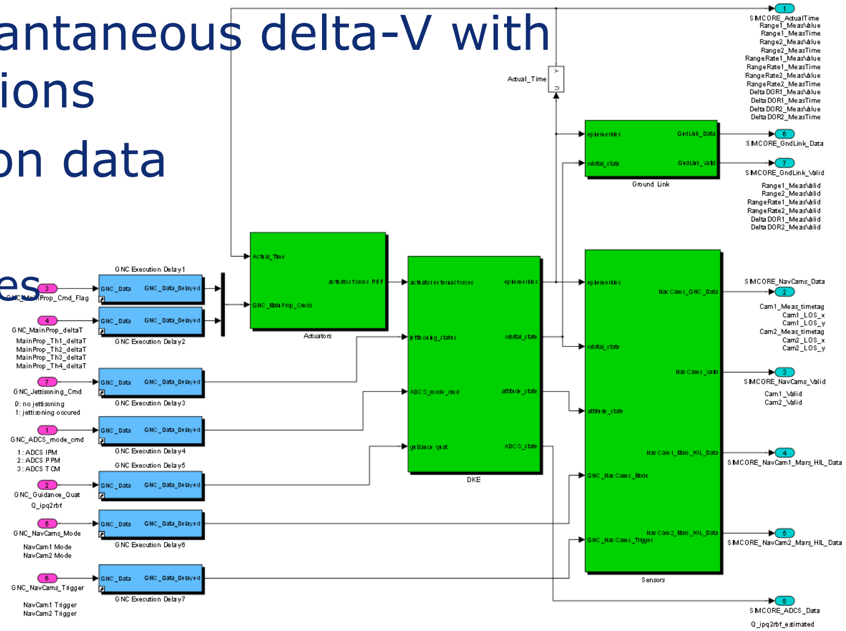
- Two-fold iterative process
  - GNC models
  - DKE, sensor and actuator models
- Two testing flows
  - Sequential
  - Regression



Non-realtime AEROFAST Simulators

Pre-aerocapture FES

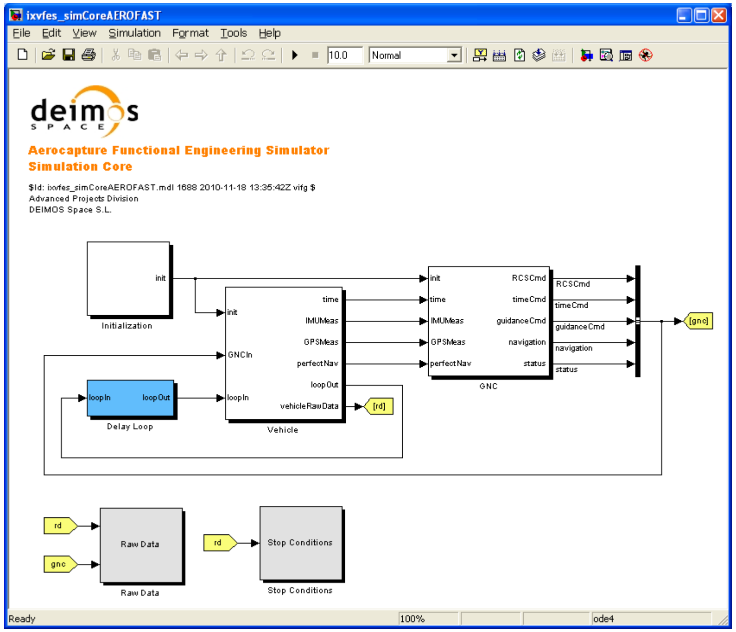
- 4-DOF simulator
- MATLAB/Simulink
- Based on SIMPLAT (in-house simulation infrastructure)
- Trajectory from Earth orbit to entry interface point near Mars
- Main propulsion model: instantaneous delta-V with magnitude/direction dispersions
- Includes models of navigation data
  - Radiometric measurements
  - Navigation camera observables
  - ADCS estimation data



Non-realtime AEROFAST Simulators

Aerocapture FES

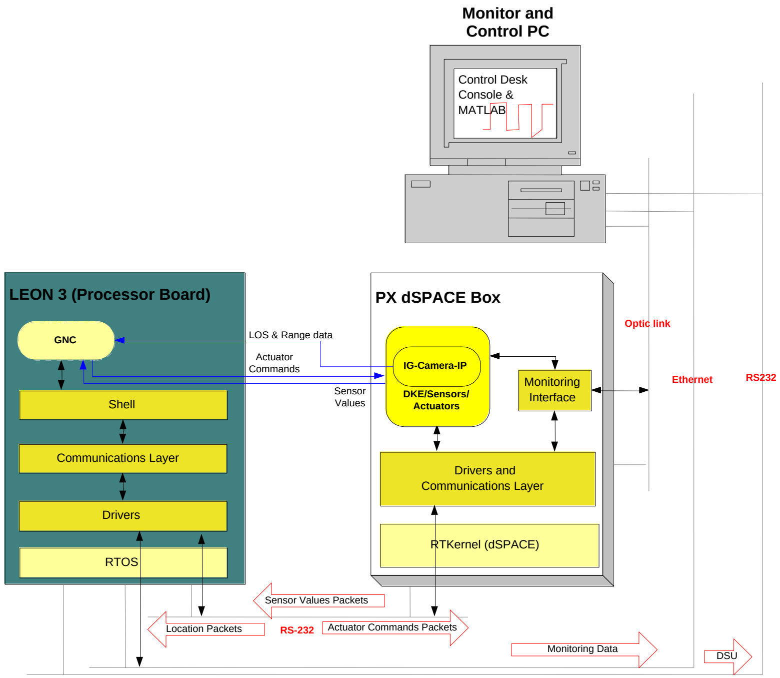
- 6-DOF re-entry simulator (IXV-FES reuse)
- MATLAB/Simulink
- Based on SIMPLAT (in-house simulation infrastructure)
- Customized with Mars environment and biconic shape capsule



Real-Time AEROFAST Simulators

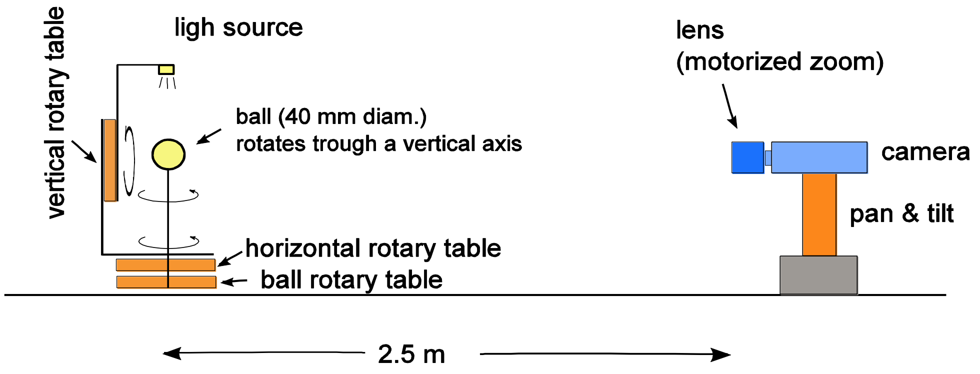
Real-time Testbench (Processor-In-the-Loop configuration)

- PX dSPACE Box runs DKE, sensor and actuator models
- LEON 3 processor runs the GNC model
- Control Desk computer controls the experiment



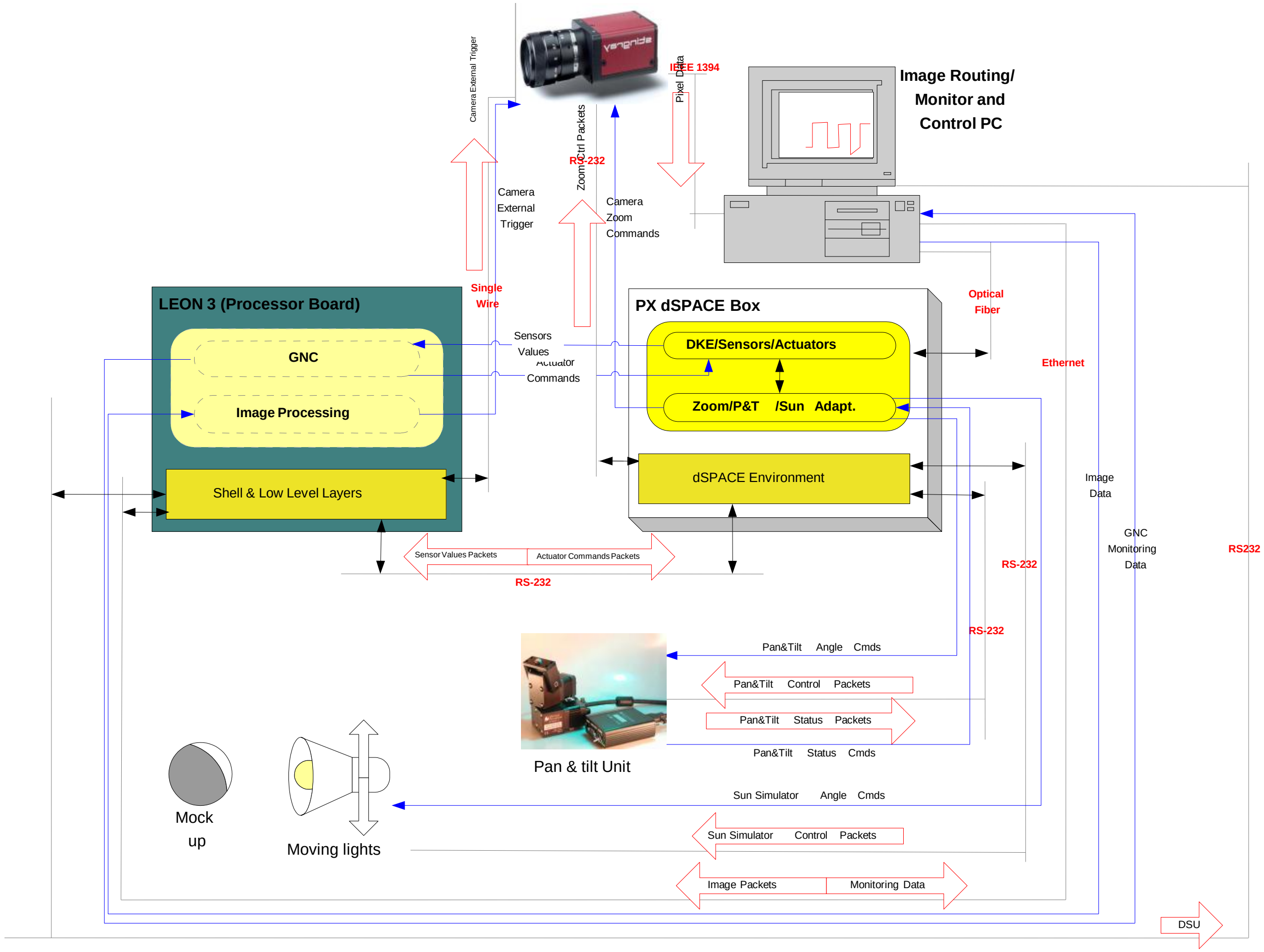
Real-time Testbench (Hardware-In-the-Loop configuration)

- Mock-up of the optical navigation system, with spherical targets, lights and camera with zoom on pan-and-tilt platform
- Image processing (IP) routines are executed in real-time on images grabbed from the camera
- PX dSPACE Box runs DKE, actuator and sensor models, now using image generation chain HW in the loop and controlling the pan/tilt platform
- LEON 3 processor runs the GNC model and the image processing SW
- Control Desk computer controls the experiment, records camera images and sends them to the LEON 3 processor



Real-Time AEROFAST Simulators

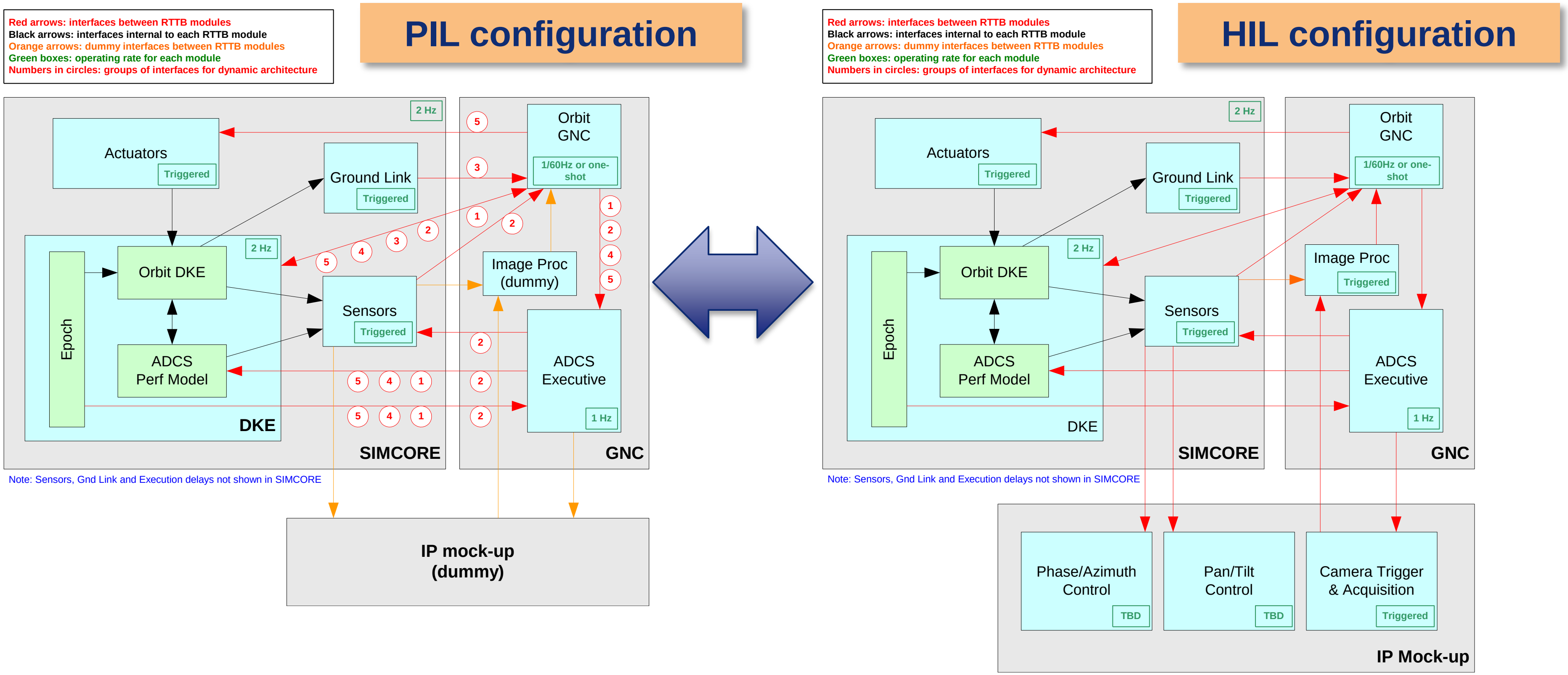
Real-time Testbench (Hardware-In-the-Loop configuration)



Unified Architecture

Features (1/3)

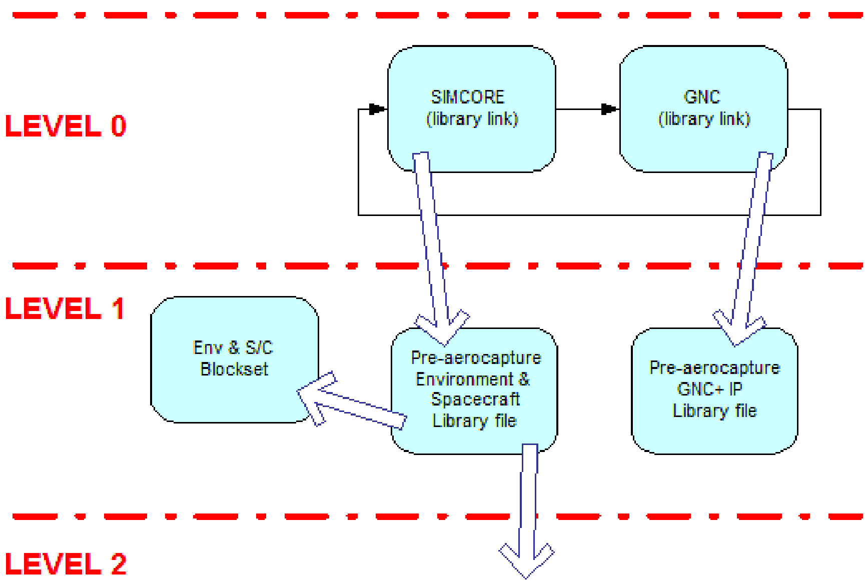
- Model variants with well-defined interfaces



Features (2/3)

- Modular and layered model architecture
  - DKE models and GNC models placed in separate Simulink libraries
  - Allows set-up of model variants at level-2

Pre-aerocapture model architecture



Features (3/3)

- Modular model setup scripts
  - Real-time testbenches reuse as much as possible the FES functionalities
  - Model variant setup scripts are shared by FES and real-time testbenches
  - The Real-Time Testbenches use additional scripts for building the GNC code
- Simulation models, GNC C code and precompiled RTEMS binaries are stored in a shared SW repository

Real-Time Validation Issues

Problem

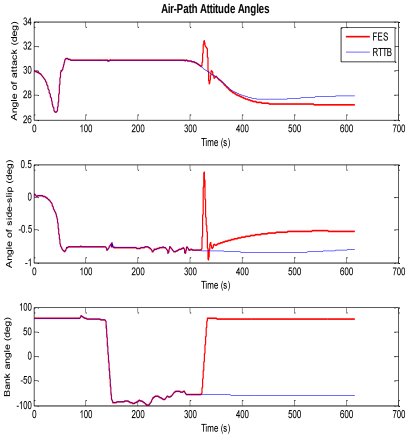
- GNC validation must take into account model uncertainties and random perturbations
- Functional validation (non-realtime) makes use of Monte Carlo simulations to account for off-nominal conditions
- What about real-time validation? How can we cope with random dispersions, since real-time Monte Carlo is unfeasible?

Approach

- Real-time worst cases were selected from Monte Carlo (non-realtime) simulations by identifying the longest execution cases
- Worst case delays for GNC code were found by profiling the GNC software during non-realtime execution

Need of Regression Testing

- The comparison of non-realtime and real-time aerocapture simulations revealed different GNC behaviour
- The origin of such differences was identified to be the GNC computational delays
- The differences disappeared when computational delays were modelled in the Aerocapture FES simulator
- The new iteration of the validation process (regression testing) was eased by the unified architecture



Conclusions and Lessons Learned

elecnor deimos

Benefits of a unified architecture

- Efficient reuse of simulation models and features
- Easy execution of regression tests, to achieve functional revalidation when changes were required to simulator and/or GNC models

Lessons learned

- It is essential that the unified architecture is designed at development kick-off, to allow proper sharing of model library and database files
- Start real-time simulations as soon as possible for early identification of constraints imposed by the RT S/W implementation on functional design

Acknowledgements

The authors wish to thank

AEROFAST Team at Astrium ST for their support and contributions to the design and development of the AEROFAST simulation framework

Thank you

www.deimos-space.com