The ORCCAD Control/Command Approach: Application in European Planetary Exploration

K. Kapellos - TRASYS (Belgium)

Journee Bernard Espiau

ESA Planetary Exploration Missions

- Landing humans on planets and returning them safely back to Earth will constitute an effort that present day society finds difficult to fund. Human landings have been postponed for the second half of the next decade.

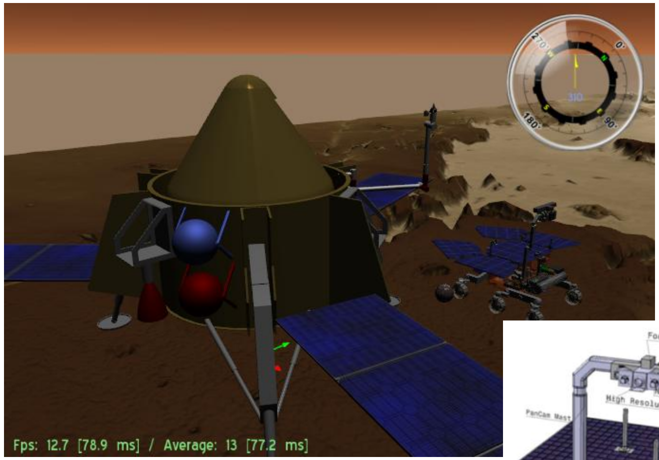
- ESA and NASA have plans to fly joint/coordinated missions to Mars every Earth-Mars opposition until Mars Sample Return (MSR)

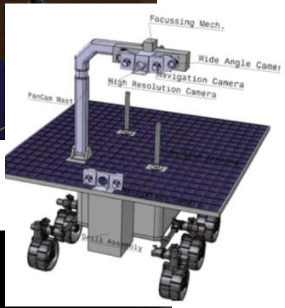
- ExoMars mission
  - 2016 slice: Orbiter + EDL demonstrator
  - 2018 slice: Rover + Pasteur payload

  - Scientific interest
    - Identify and characterise possible hazards to human exploration
    - Enhance the knowledge of the Martian environment
    - Exobiology with in-situ soil sample analysis
  - Technological interest
    - Landing of large payloads on the surface of Mars
    - Mars Surface Mobility

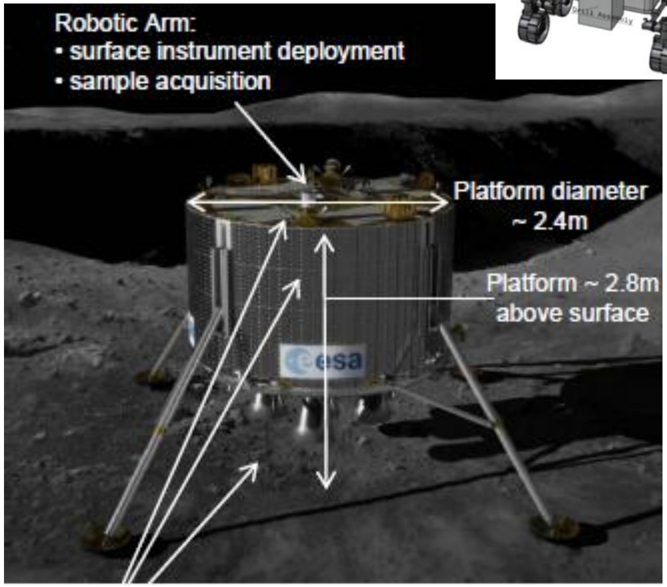
ESA Planetary Exploration Missions

- Mars Sample Return (MSR)
  - ESA develops elements of the mission
    - A Mars Sample Transfer and Manipulation System (MSSTM) providing the function to transfer samples from the acquisition device to the Mars Ascent Vehicle (Prime: ASU)
    - A Sample Fetching Rover (SFR) being capable of traverses of few kilometers (~20km)





- Lunar Mission
  - Not later than 2018
  - Analyzing the structure and composition of lunar dust
  - Characterizing in-situ resources in the form of volatiles



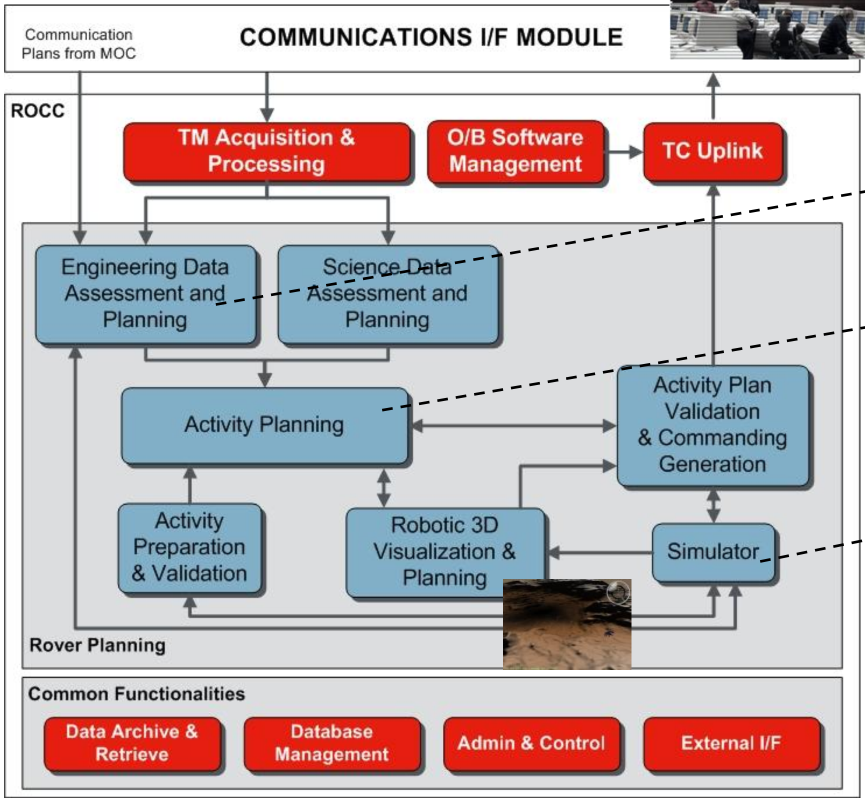
Credits: A. Pradier – ESA

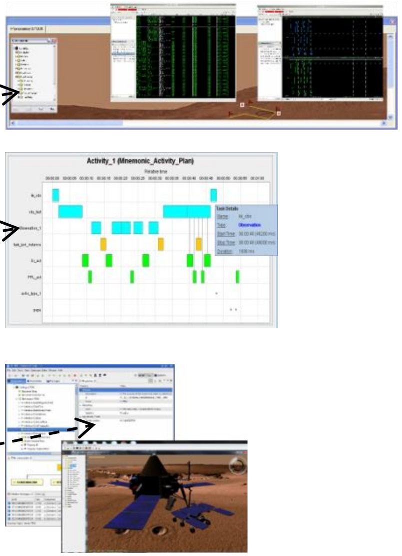
Rover Operations Structure

TM

Activity Plan

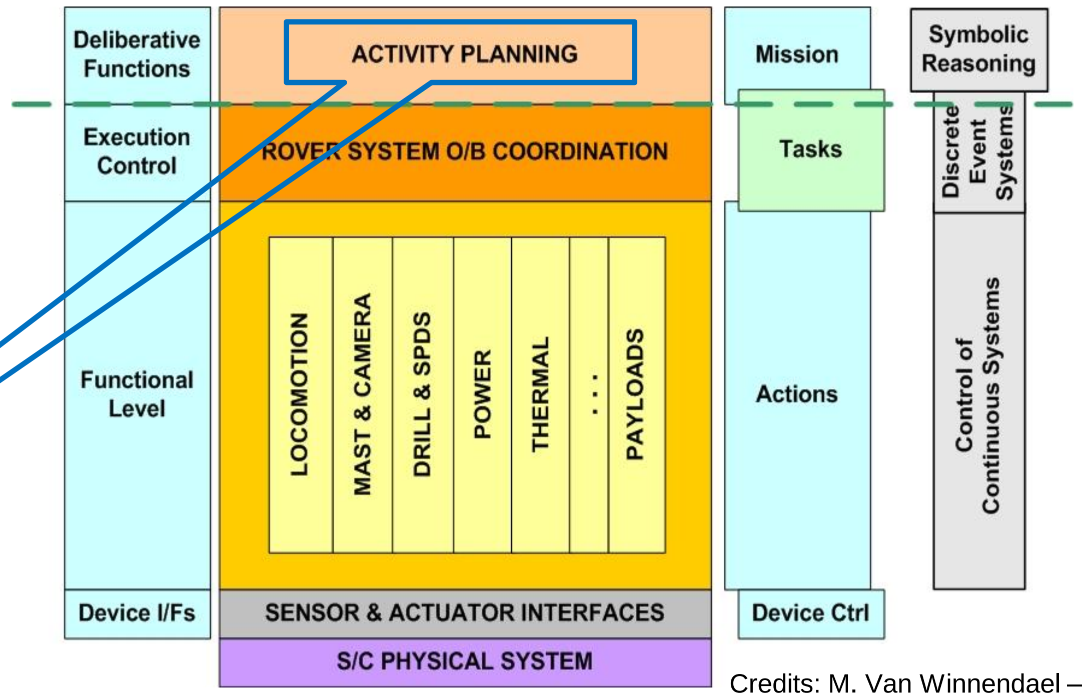
Rover Operations Control Centre



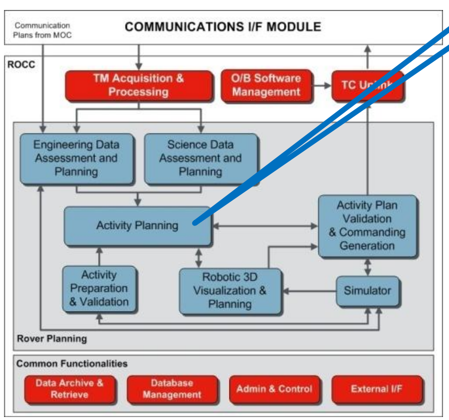


– The concept of 'Activity' is fundamental for the control of the rover operations

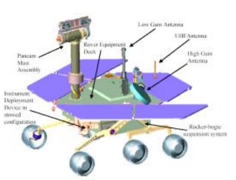
Rover Operations Structure



Credits: M. Van Winnendael – ESA



Rover Operations Control Centre



TRASYS
WE GET IT DONE

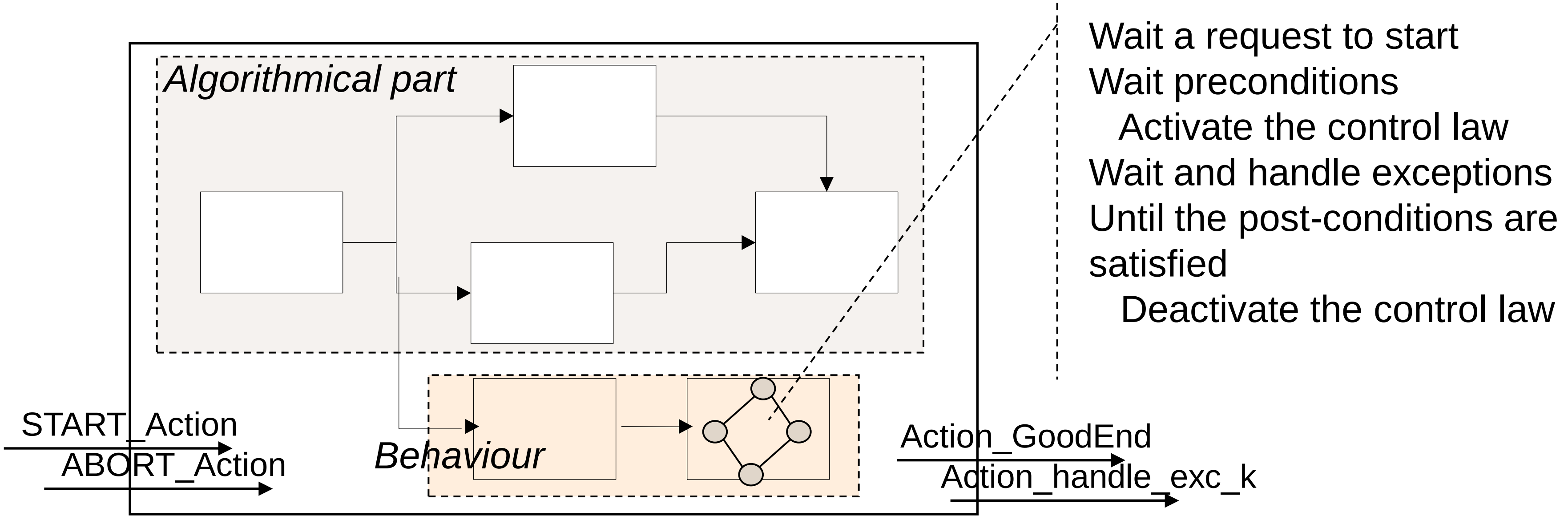
Plan

- Functional layer
  - Action definition
  - Action validation
- Execution layer
  - Task definition
  - Task validation
- Deliberative layer
  - Activity Plan

Functional Layer – Action definition

- An Action is defined as the complete and parameterised specification of:
  - A control law
  - A local behavior rythmed by events
  - Temporal constraints (real-time aspects)

Examples: *move_mast_to(), take_image(), …*



Functional Layer – Action definition

- Actions involved in the Mars Sample Return – Sample Transfer activities

MOVE_TO_JS	Displace activity that moves the robot to a desired joint position **using trajectory tracking control in the joint space**.
MOVE_TO_SE3	Displace activity that moves the robot to a desired cartesian position using **trajectory tracking control in the cartesian space**.
APPROACH	Displace activity that moves the robot to the vicinity of the gof using trajectory tracking control in the cartesian space.
APPROACH_ATTACH	Approach activity that moves the robot to the gof grasping position using **vision based control**.
ATTACH/DETACH	Attach/Detach activity by closing/opening the gripper using **force/torque control**.
LOCK/UNLOCK	Locks/unlocks the item.
EXTRACT_FROM	Extracts an item from its port using force/torque control
INSERT_INTO	Extracts an item into its port using force/torque control
RETRACT	Moves the robot out of the gof proximity area using trajectory tracking control in the cartesian space.

(*) MSSTM ESA Activity: prime Astrium UK

Functional Layer – Action definition

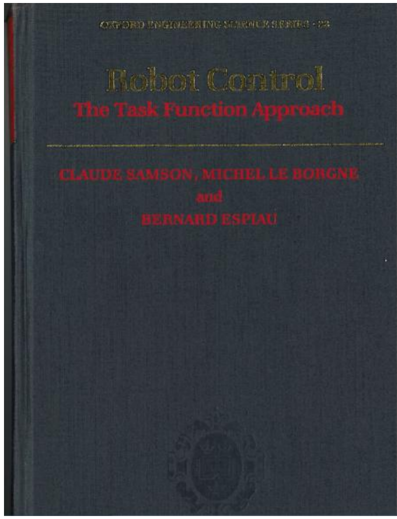
- Designed in the framework of the Task Function approach
  - The user's objective may in general be expressed as the regulation to zero of a n-dimensional C^2 function $e(q,t)$, called task function

$$\Gamma = -\lambda \hat{M} \left(\frac{\partial \hat{e}}{\partial q} \right)^{-1} G \left(\mu D e + \frac{\partial \hat{e}}{\partial q} \dot{q} + \frac{\partial \hat{e}}{\partial t} \right) + \hat{N} - \hat{M} \left(\frac{\partial \hat{e}}{\partial q} \right)^{-1} f \qquad \frac{\partial e}{\partial q} \left(\frac{\partial \hat{e}}{\partial q} \right)^{-1} > 0$$

  - Joint space: $e(q(t)) = q(t) - q_d(t)$
  - Cartesian space: $e(r(t)) = x(t) - x_d(t)$
  - Visual Servoing: $e(r(t)) = C\,(s(r(t)) - s^*)$
    - $s(r(t))$: current value of the visual information
    - s^*: desired value of the visual information
    - C: combination matrix 6xk

$$\dot{e} = C \frac{\partial s}{\partial r} T_q$$

$$L_s = \frac{\partial s}{\partial r}$$

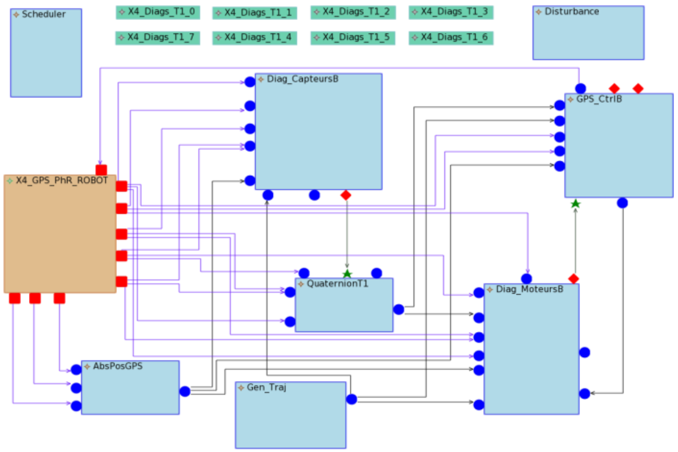


  - Hybrid control: $e = W^+ e_1 + a(I_6 - W^+ W)\, e_2$

TRASYS
WE GET IT DONE

Functional Layer – Action definition

- Real-Time aspects
  - Multi-tasking, multirate real-time
  - Hard or weak synchronization (minimize latencies)
  - Data integrity over asynchronous links
  - On-line reconfigurations via exceptions
  - Links with hardware (drivers, sockets, etc.)



  - Control aware real-time

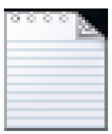
  - Design and implementation of special API for Feedback scheduling:
    - Varying clocks
    - Varying priorities
    - Overrun handlers

  - Automatic code generation

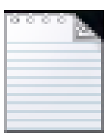
TRASYS
WE GET IT DONE

Functional Layer – Action Validation

- Use of the Siconos simulation platform [INRIA / BIPOP], a free software, dedicated to modeling, simulation and control of Non Smooth Dynamical Systems including mechanical systems with contact, impact and friction
  - Introduction of the kinematics and dynamics data
  - Use of the HuMAns Toolbox for:
    - automatic generation of the robotic arm models and control functions; integration with the Siconos kernel
    - analytical computation of the contacts between the EE and the sample container
  - Integration into the ORCCAD controller

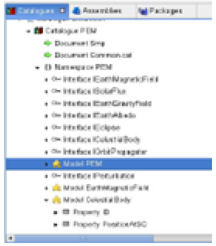


model (in maple)

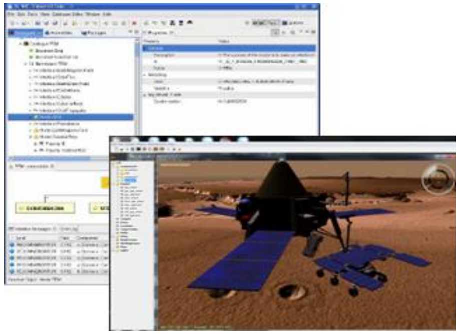


control law (in maple)

'C' code
+
Siconos Kernel



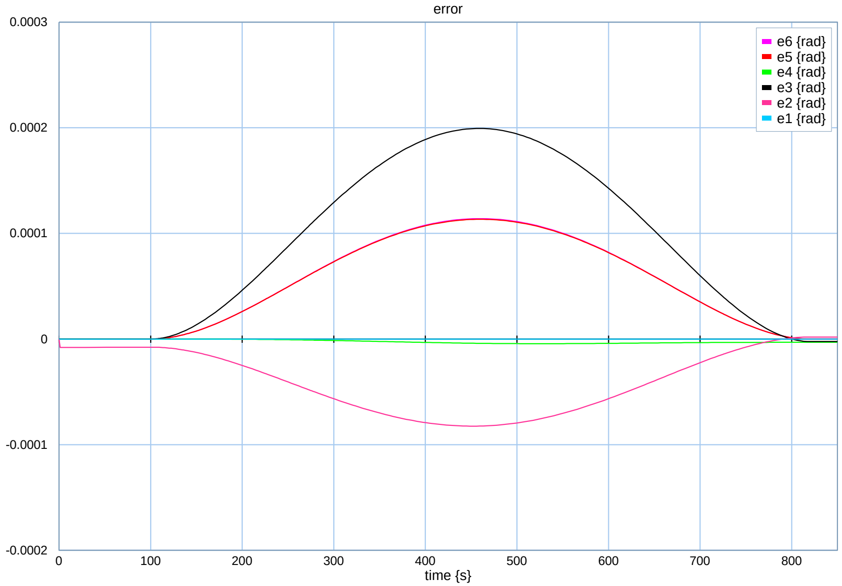
Controller Model (in SIMSAT)

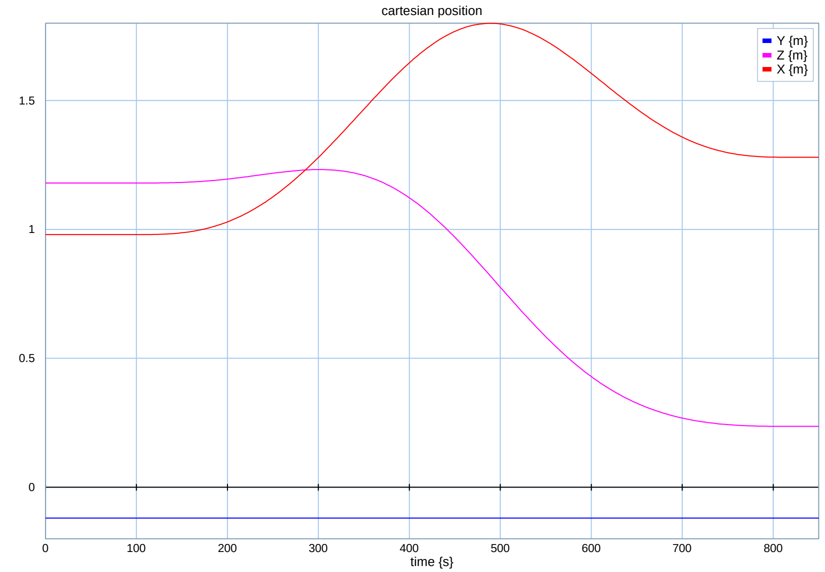


global simulation

Functional Layer – Action Validation

- Continuous Time Simulation



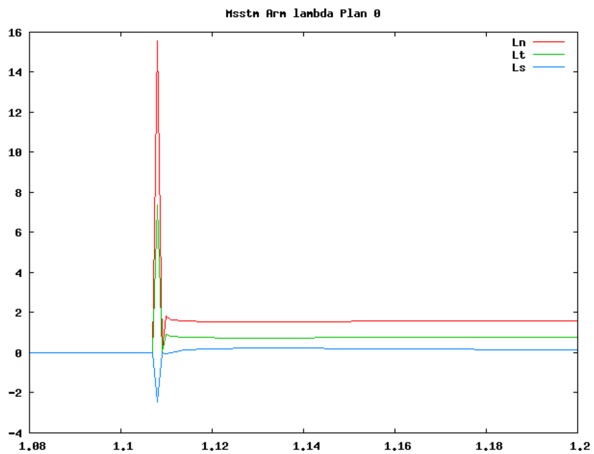


	Q1	*Q2*	*Q3*	*Q4*	*Q5*	*Q6*
Positioning Accuracy (rad)	~e-15	~e-6	~e-6	~e-6	~e-12	~e-16
Max Tracking Error (rad)	~e-11	~e-5	0.0002	~e-6	0.00011	0.00011
Max Joint Torque (Nm)	~e-5	-39.9504	-16.7793	-1.8405	-1.3433	~e-8

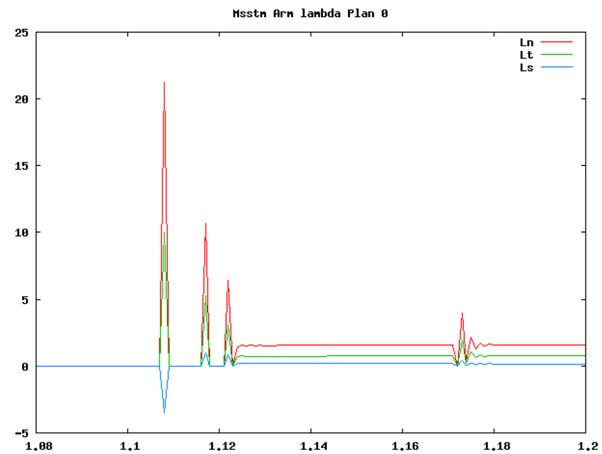
– The positioning accuracy of the control law is high with a very small maximum tracking error. Important torques are applied at the second and the third joints (-39.9504Nm and -16.7793Nm respectively)

- Descretised time simulations

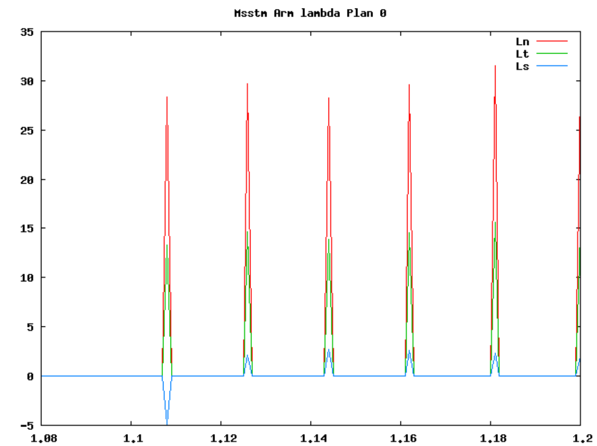
Normal forces (red) and tangential forces (in green)



Restitution = 0.1



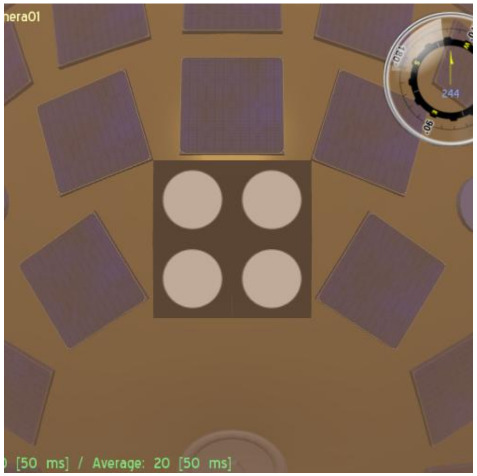
Restitution = 0.3

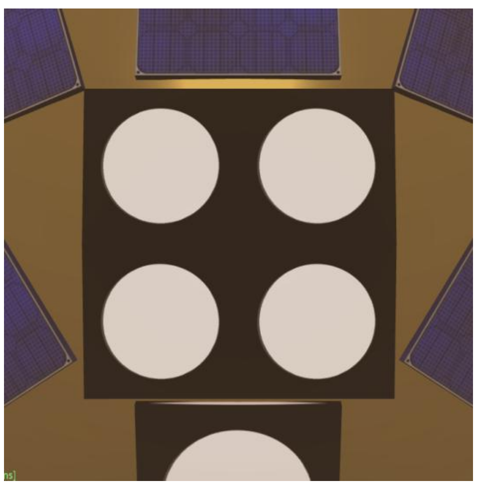


Restitution = 1.0

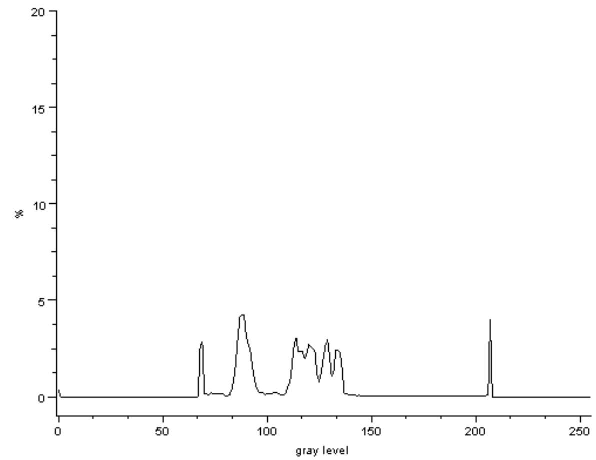
Functional Layer – Action Validation

- Simulation considering environmental conditions





– Grey level histogram



– Mean error per direction

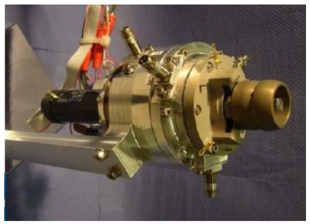
X (m)	Y (m)	Z (m)	Rx (rad)	Ry (rad)	Rz (rad)
0.00065	0.00039	0.00023	0.00359	0.00371	0.00163

– Standard deviation per direction

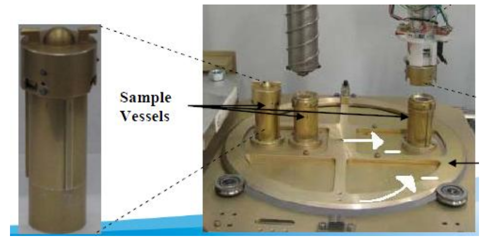
X (m)	Y (m)	Z (m)	Rx (rad)	Ry (rad)	Rz (rad)
0.0002	0.00017	0.00023	0.00286	0.00255	0.00088

Functional Layer – Breadboarding

- Increase the Technology Readiness Level
  - End effector, ...

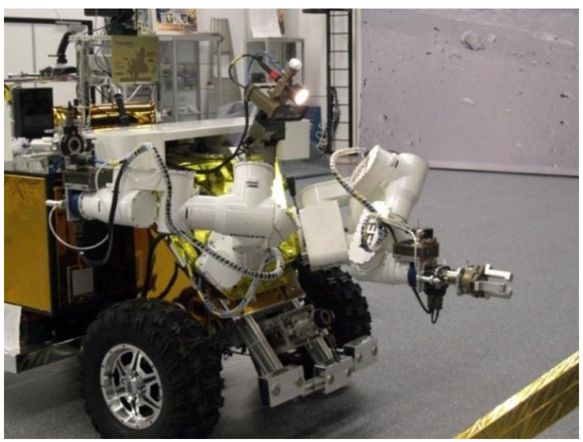


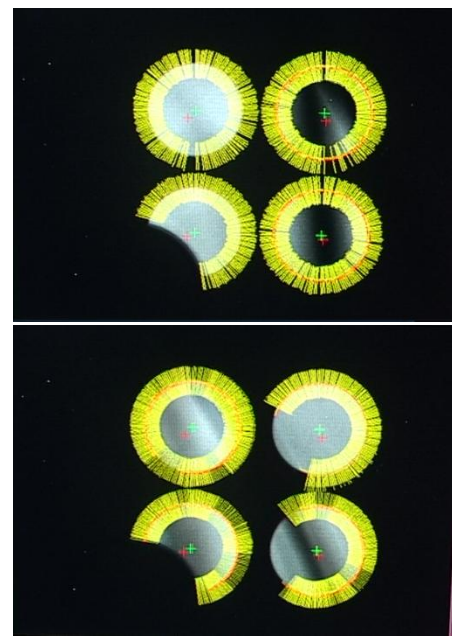

Credits: RUAG



Credits: Selex - GA

  - Vision Based Control

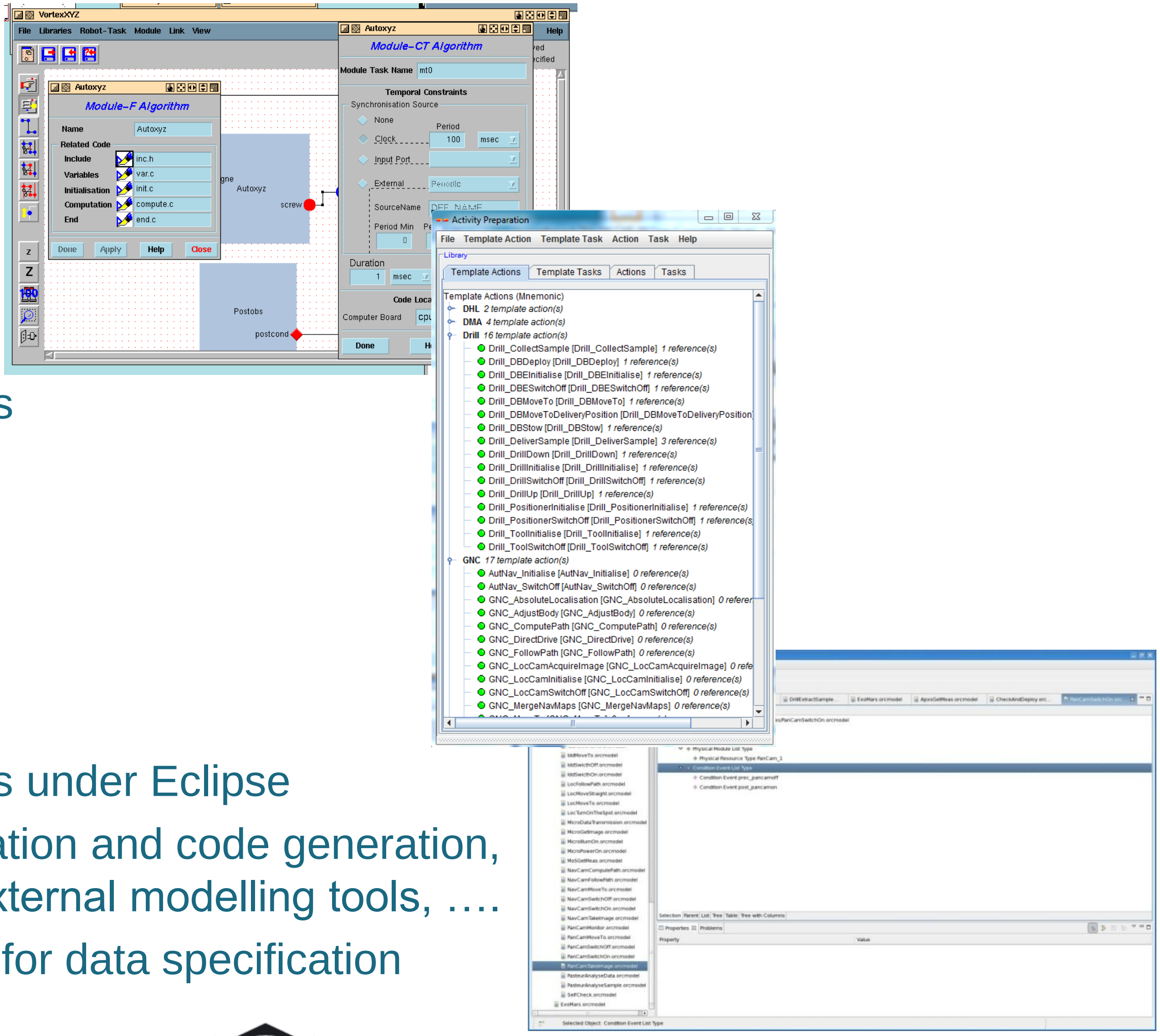




    - Use of LAGADIC – ViSP s/w

Functional Layer – Tools

- ILOG Based MMI



- Simplified environments

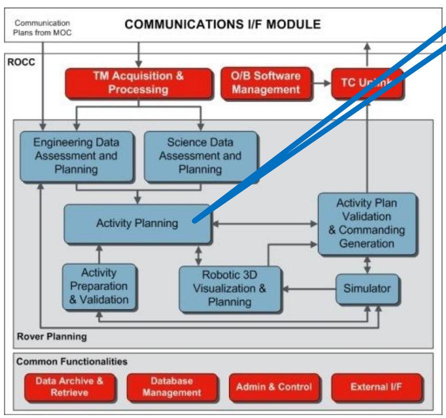
- On-going developments under Eclipse
  - Plug-in for specification and code generation, connections with external modelling tools, ….
  - Use of XML format for data specification

TRASYS
WE GET IT DONE

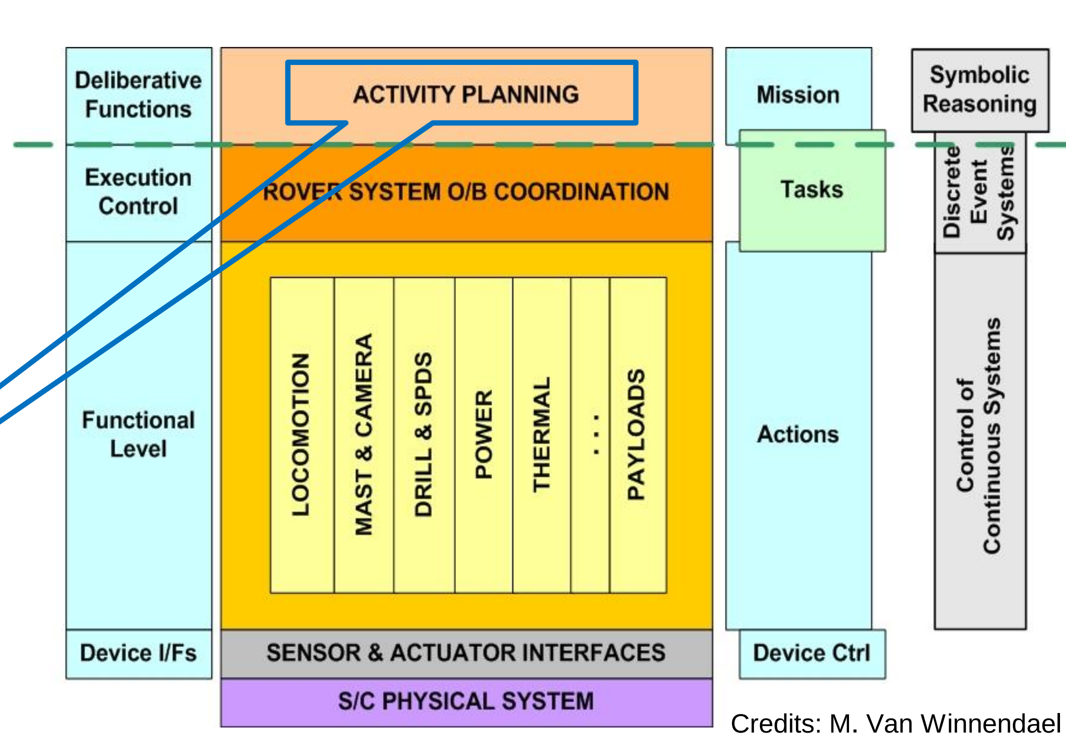
Plan

- Functional layer
  - Action definition
  - Action validation
- Execution layer
  - Task definition
  - Task validation
- Deliberative layer
  - Activity Plan

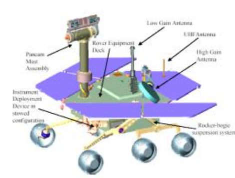
Rover Operations Structure



Rover Operations Control Centre



Credits: M. Van Winnendael – ESA



Execution Layer – Task Definition

- When moving from Actions to Mission objectives there is an intermediary level where the Rover activities (Tasks) are expressed in terms of FSMs

```
do [
  DrillMoveTo ();
  CloseImagerMoveTo ();
  [
     CloseImagerMonitor()
    ||
     DrillExtractSample()
   ]
] watching Alarm do ….
```

  - Examples: *Autonomous Navigation, Travel, Drill, Sample Analysis*

- Tasks shall be formally defined and validated

TRASYS
WE GET IT DONE

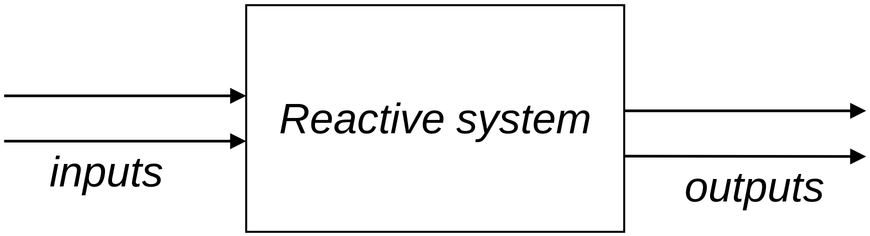
Execution Layer – Task Definition

- A Task is defined as a logical and temporal composition of Actions and other Tasks including logic for making checks and decisions. It is formally defined in its most complete form as:
  - A set of pre-conditions which need to be fulfilled before the main body of the Task starts.
  - A main body, (nominal execution of the Task), composed of Actions, Tasks and conditions which fulfils the goal of the Task.
  - A set of post-conditions that induce the end of the Task.
  - A set of reaction rules to process every exception by a recovery handling body (this is a way to provide optional activities).
  - A pre-defined behavior for the logical co-ordination of the previous items: the main body of the Task is activated after satisfaction of the pre-conditions, and normally ends when the post-conditions are satisfied. If an exception occurs, this nominal execution is aborted and replaced by the specified recovery body.

- Definition adapted to the ExoMars operations
- In line with the ECSS-E-ST-70-01on OBCPs

Execution Layer – Formal Framework

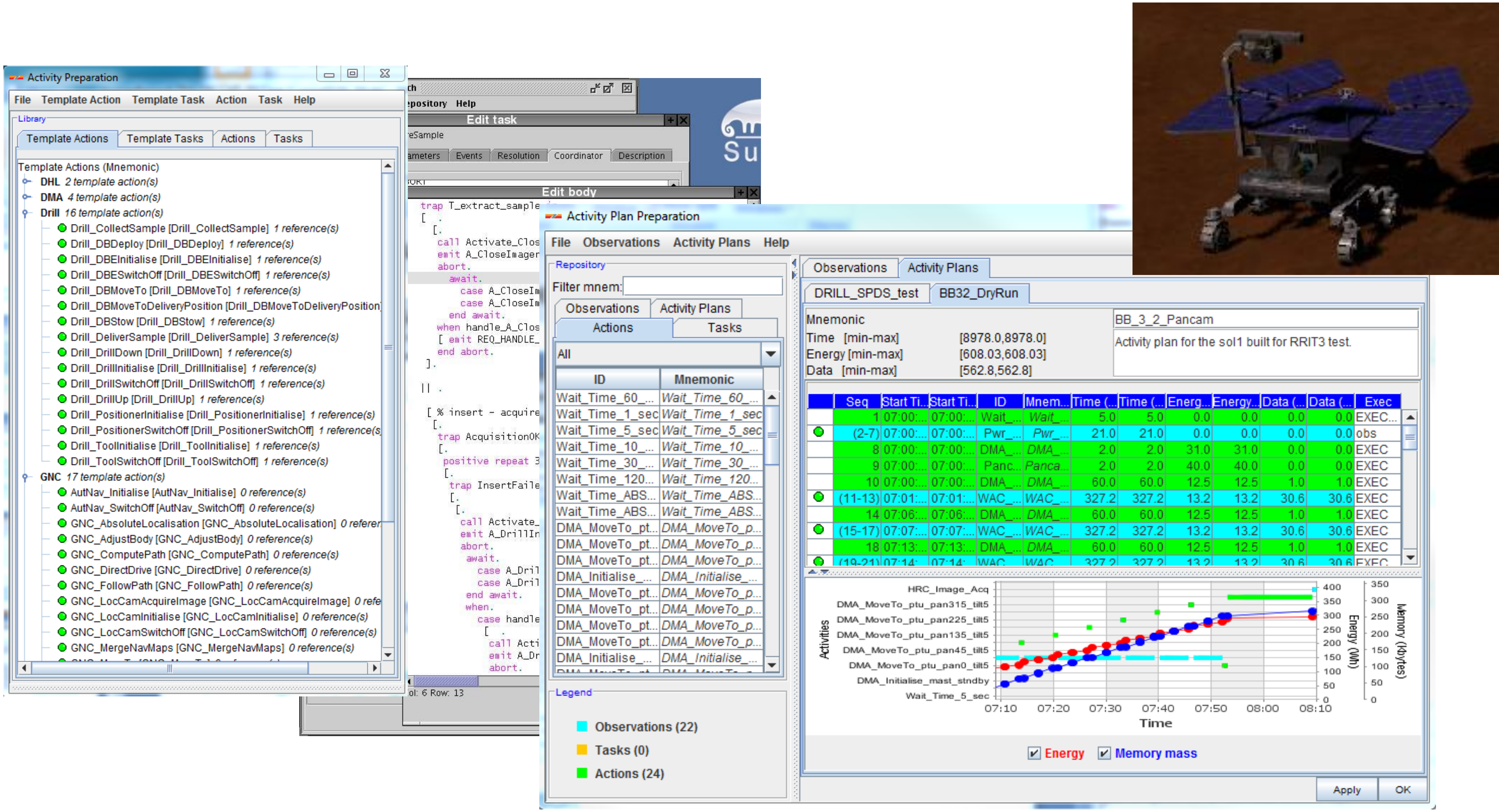
- Reactive systems



- Behavior of the system
  - All allowed sequences of the input/output events
  - Synchronous approach
    - The duration of the system reaction is negligible wrt the input signal occurrences
  - ESTEREL language
    - Dedicated for reactive systems programming
    - Compilation to finite state automata
    - Tool for simulation, verification and code generation

TRASYS
WE GET IT DONE

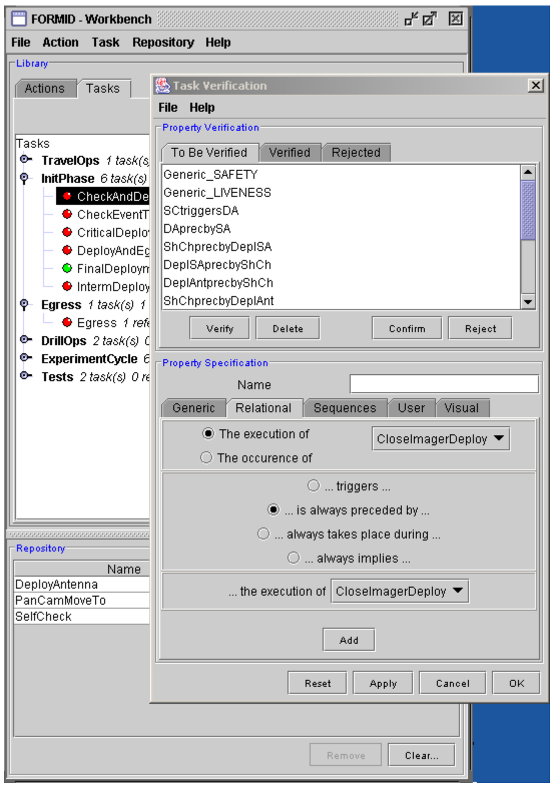
Execution Layer – Task Specification



- ExoMars Activities specification
  - ~120 Actions / 30 Tasks

Execution Layer – Task Formal Verification

- Task Formal Verification
  - Systematisation of the properties to be verified
  - Automatic generation of the 'observers'
  - Use of the Esterel tools



- <u>Relationship</u> between Actions/Events and Actions
  - The execution of the Action/Task

    … triggers …

    … is always preceded by …

    … always takes place during …

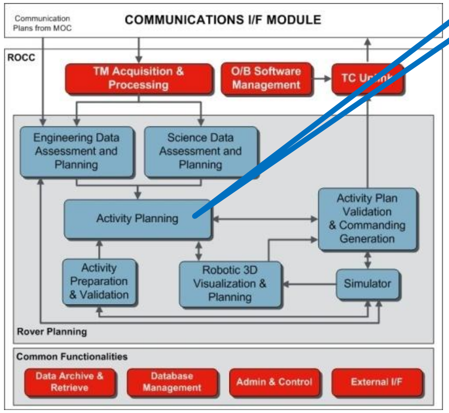
    … always implies …

    the execution of the Action/Task
  - The occurrence of the Event 'event'

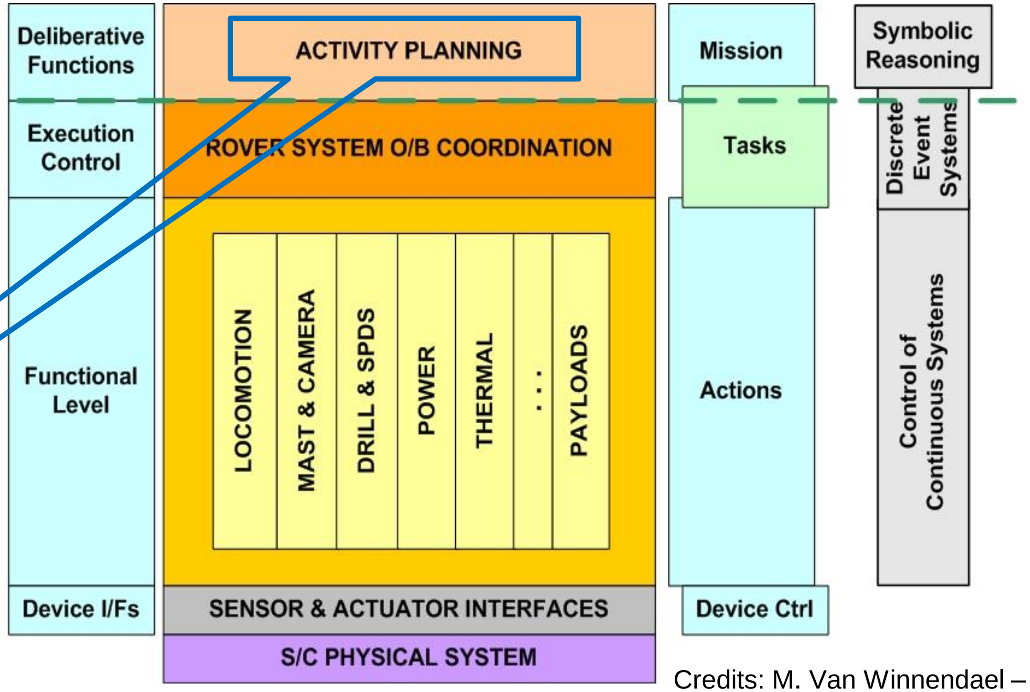
    … triggers …

    the execution of the Action/Task

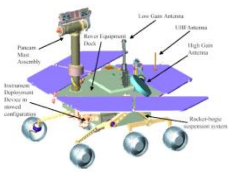
Rover Operations Structure



Rover Operations Control Centre



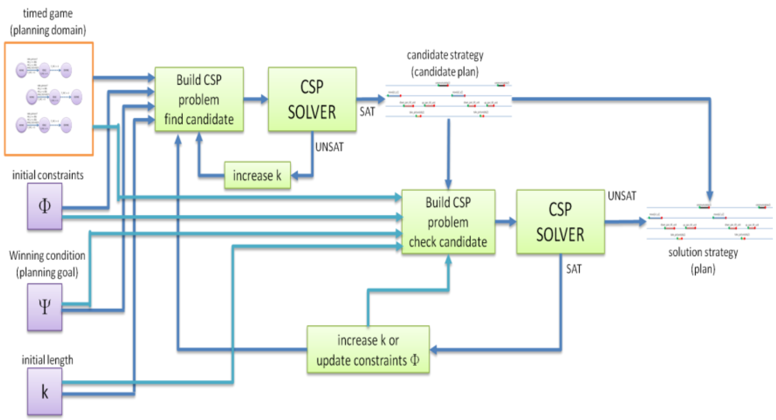
Credits: M. Van Winnendael – ESA

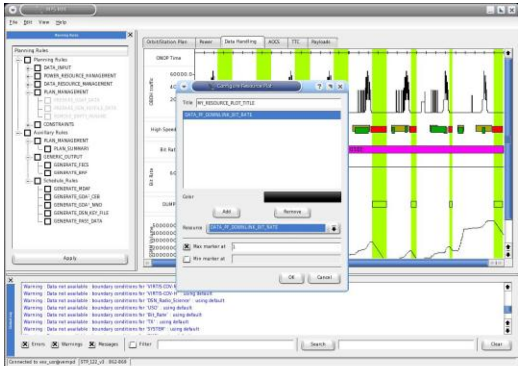


TRASYS
WE GET IT DONE

Deliberative Layer

- IRONCAP considers the problem of generating strong conditional plans which allow for parallel activities (Actions/Tasks), partial observability and take into account uncertainty on the duration of the activities, uncertainties on the ordering of events and uncertainty on resource consumption to guarantee goal achievement
  - Uncertainty is modelled by distinguishing controllable and uncontrollable state and temporal variables





  - The planning problem formally represented as the problem of finding a winning strategy in timed/hybrid game

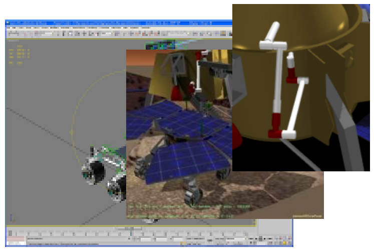
(*) IRONAP: ESA on-going Activity: VEGA / FBK (A. Cimatti) / TRA

TRASYS
WE GET IT DONE

ORCCAD Integrated into the 3DROV Simulator

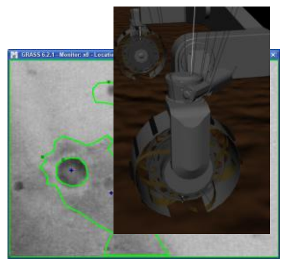
Physical s/s

- Robotic Arm
- Rover, ….
- Power, Thermal



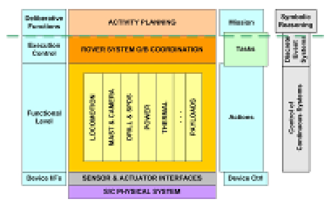
Environment

- Atmosphere
- Orbiter & Timekeeping
- Terrain



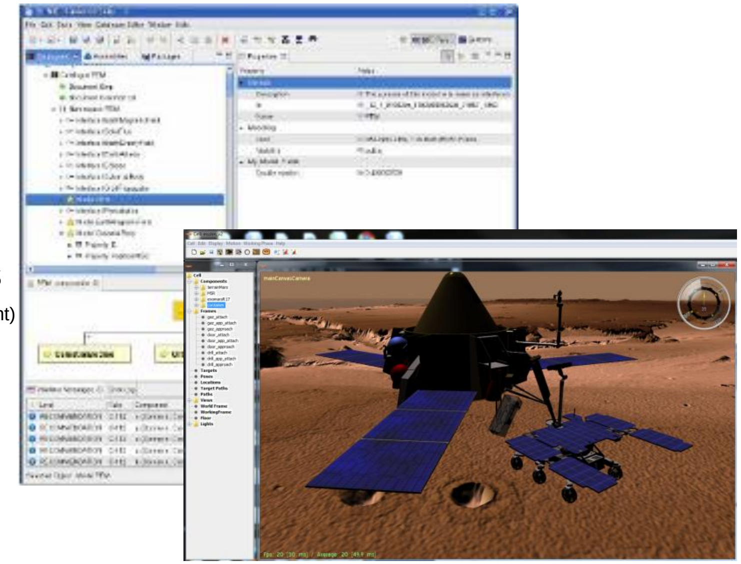
ORCCAD Controller

- Control laws
- Actions, Tasks
- Real-time impl.



Models
(SMP2 compliant)

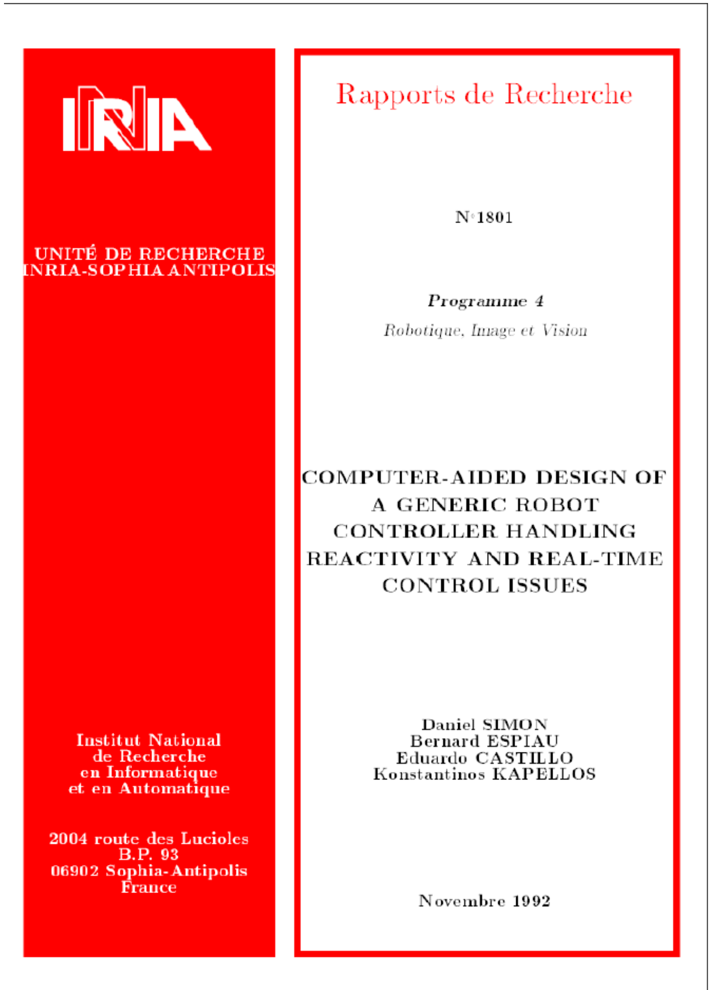
<u>Simulation</u> (SIMSAT)



Visualisation Tool

- Main components
  - Physical s/s models
  - Controller model
  - Environment model
  - 3D Visualisation component
  - SIMSAT framework (ESA/ESOC Tool)

ORCCAD


UNITÉ DE RECHERCHE
INRIA-SOPHIA ANTIPOLIS

Institut National
de Recherche
en Informatique
et en Automatique

2004 route des Lucioles
B.P. 93
06902 Sophia-Antipolis
France

Rapports de Recherche

N°1801

Programme 4

Robotique, Image et Vision

COMPUTER-AIDED DESIGN OF
A GENERIC ROBOT
CONTROLLER HANDLING
REACTIVITY AND REAL-TIME
CONTROL ISSUES

Daniel SIMON
Bernard ESPIAU
Eduardo CASTILLO
Konstantinos KAPELLOS

Novembre 1992

- Bernard
- Daniel
- Jean-Jacques
- Antoine
- Eve
- Eduardo
- Konstantinos
- Roger
- Soraya
- Nicolas
- Eric
- Olivier
- Florine
- …

Contacts

K. Kapellos
TRASYS
Terhulpsesteenweg, 6C, 1560 Hoeilaart
Phone: +32 2 893 15 08
Fax: +32 2 893 14 00
E-mail: Konstantinos.Kapellos@trasys.be

L. Joudrier
ESA/ESTEC
Keplerlaan 1, 2201AZ Noordwijk, The Netherlands
Tel: +31-71-565-8294
Fax: +31-71-565-5419
E-mail: luc.joudrier@esa.int