

# The case for an RCOS at ESA

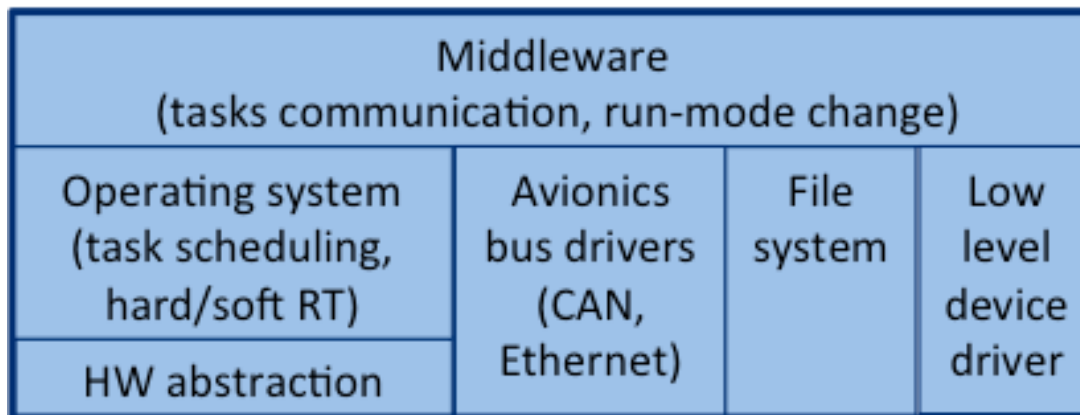
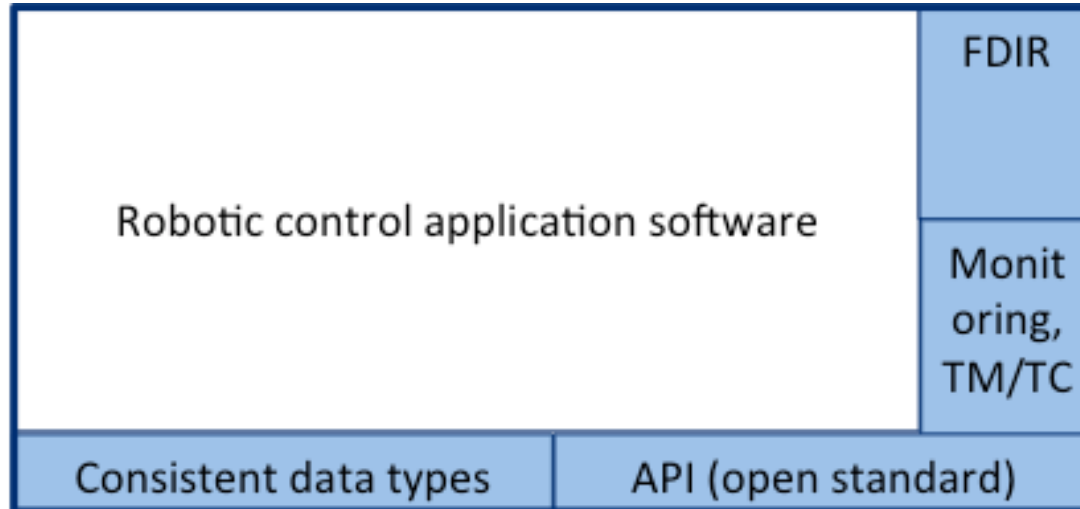
G.Visentin.

Head, Automation and Robotics Section

A Robot Control Operating System is the basic software that allows robot control software to run. A RCOS must provide the following features:

- Hardware abstraction
- Low-level device control
- Scheduling of hard/soft real-time tasks
- Communication and synchronisation between tasks
- Run-mode/run-level/runtime-configuration management and monitoring
- Fault Detection, Isolation and Safing/Recovery
- Filesystem access and filesystem management
- Networking
- Consistent data types across communication, networking and file system operation
- Logging/telemetry generation and command processing
- Application Programming Interface for all the above functions

# What is an RCOS



ESA space robotics missions such as ExoMars or European Robotic Arm (ERA) require significant software engineering effort when compared with other satellite space missions.

This is due to the complexity introduced to space systems by the robotic application (typically requiring far greater level of autonomous decision) together with the additional handicap of having usually low levels of heritage. The result of this is that a high percentage of the total mission cost is dedicated for software development.

# Brief history of robot controller developments at ESA



- ESA has in the past tried to develop standard robot control systems
- Few millions of euros have gone into the development of systems (CAT, CESAR, CIRCUS, CONTEXT) that at the time they were produced were excellent
- However they have never been adopted in missions
- Why?

The why:

1. Obsolescence: from the time they were developed to the time they could be used technology had moved on and, without a a sizable user community, the development stagnated
2. Prohibitive verification cost due to industrial origin: we derived the controllers from industrial grade ones which already at the beginning did not have enough RAMS grade
3. Vendor lock-in: from the RTOS to the application code IPR hanged on the code making not transferrable to new developers
4. Application specificity: we started with robot arms and later went into rovers and the transformation required was great

a *Robot Control Operating System* that overcomes the above underlined issues will need to have the properties of:

- Heritage on an existing body of robot control software
- adequate RAMS attributes for industrial and space use
- support for modularity and re-usability (of modules and/or entire applications)
- an open standard implementation
- a large community of developers and users

# The RCOS product that we seek



***RCOS target:*** a software system that support the creation of robot controllers execution of robot control applications by providing:

- hardware abstraction, low-level device control, scheduling of hard/soft real-time tasks, communication and synchronisation between tasks, run-mode/run-level/runtime-configuration management and monitoring, Fault Detection, Isolation and Safing/Recovery, filesystem access and filesystem management, networking, consistent data types across communication, networking and file system operation, logging/telemetry generation and command processing, Application Programming Interface for all the above functions

***RCOS development and validation system (RDEV):*** a system of software programmes that allows to:

- Develop robot control applications within the RCOS target
  - Test robot control applications within the RCOS target
  - Maintain the robot control applications within the RCOS target
  - Validate the robot control applications within the RCOS target by
- <sup>8</sup> characterising its Reliability, Availability and Safety



An *RCOS Target* is intended to be available in three levels of quality:

- **Lab quality:** this level will be characterised by
  - Easier development of robot controllers
  - Best-effort reliability, availability and safety of the produced controller
- **Hi-Rel quality**
  - More laborious development of robot controller
  - High reliability, availability and safety of the produced controller
- **Space quality**
  - More laborious development of robot controller
  - Adherence to space standards
  - Top reliability, availability and safety of the produced controller
- The *RDEV* is intended to support all levels of quality.

- ESA will issue an ITT [Space Automation & Robotics General cONtroller (SARGON)] for a comparatively small initial development of core specifications and prototype
- In the frame of the EC PERASPERA activity we have proposed to issue a much larger activity that will take care of the full development of the RCOS product

- There are a number of developments that are used by the space/terrestrial communities (e.g. OROCOS, GENOM, ROS, ROCK, MARCO)
- In each of these we can find different approaches and solutions to the RCOS issue
- However we believe no one is the solution
- What is good in each and we should copy?
- What is not good and we must avoid?
- What is actually possible to reuse (the principle? the implementation?)

# Why this workshop



- We want to involve as many of you as possible
- However ESA contracts and EC grants require a team
- So this is also a speed-dating event for you to find possible partners in this important development