
REDWIRE
BUILD ABOVE

WHO ARE WE?

VISION
Decades of flight heritage and innovation of world-class technologies combined with our mission success and focus on customer satisfaction have positioned Redwire Space as a leader in advancing the future of space infrastructure.

MISSION
Redwire Space is accelerating humanity's expansion into space by delivering reliable, economical, and sustainable infrastructure for future generations.

OUR AREAS OF EXPERTISE and OUR PROJECTS

Redwire Space

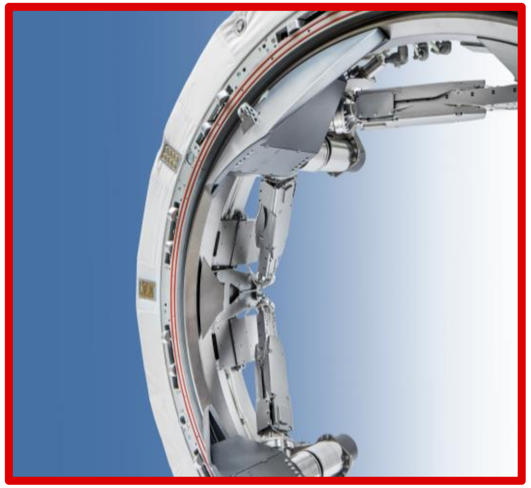
REDWIRE

OUR MAIN AREAS OF EXPERTISE

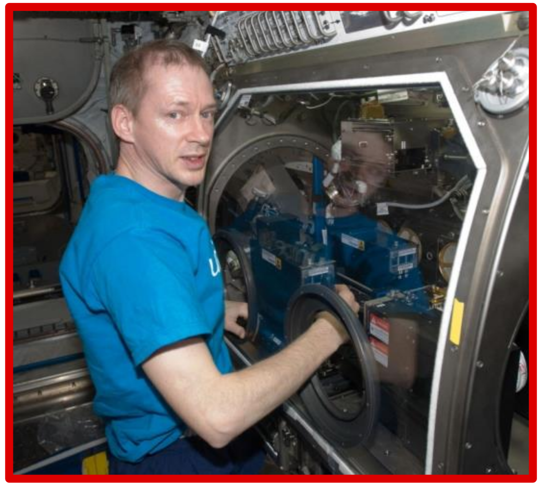
Satellites

- Small satellites and platforms

Science & Exploration



- International Berthing and Docking Mechanism
- Telescope baffles



- Instruments for scientific research in microgravity
- Life support systems

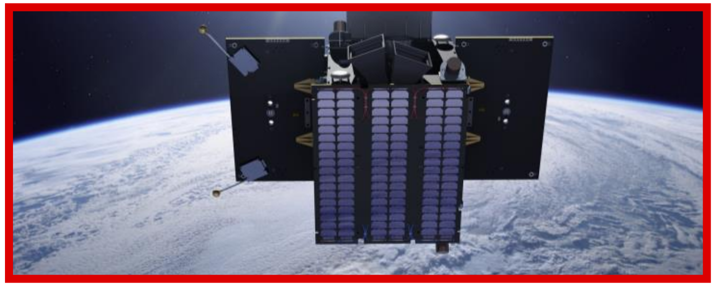
OUR SATELLITES IN SPACE

PROBA – Project for On Board Autonomy

PROBA-1

- ✓ Earth Observation instruments
- ✓ Environment instrument
- ✓ 11 Hardware & software space technologies and sensors
- ✓ On-board operational autonomy

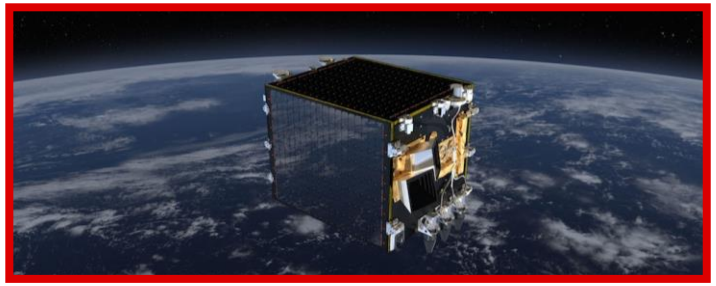
Launch 2001 -> Operational (+23 years)



PROBA-2

- ✓ Solar science experiments
- ✓ Environment instrument
- ✓ 17 hardware & software space technologies and sensors

Launch 2009 -> Operational (+15 years)



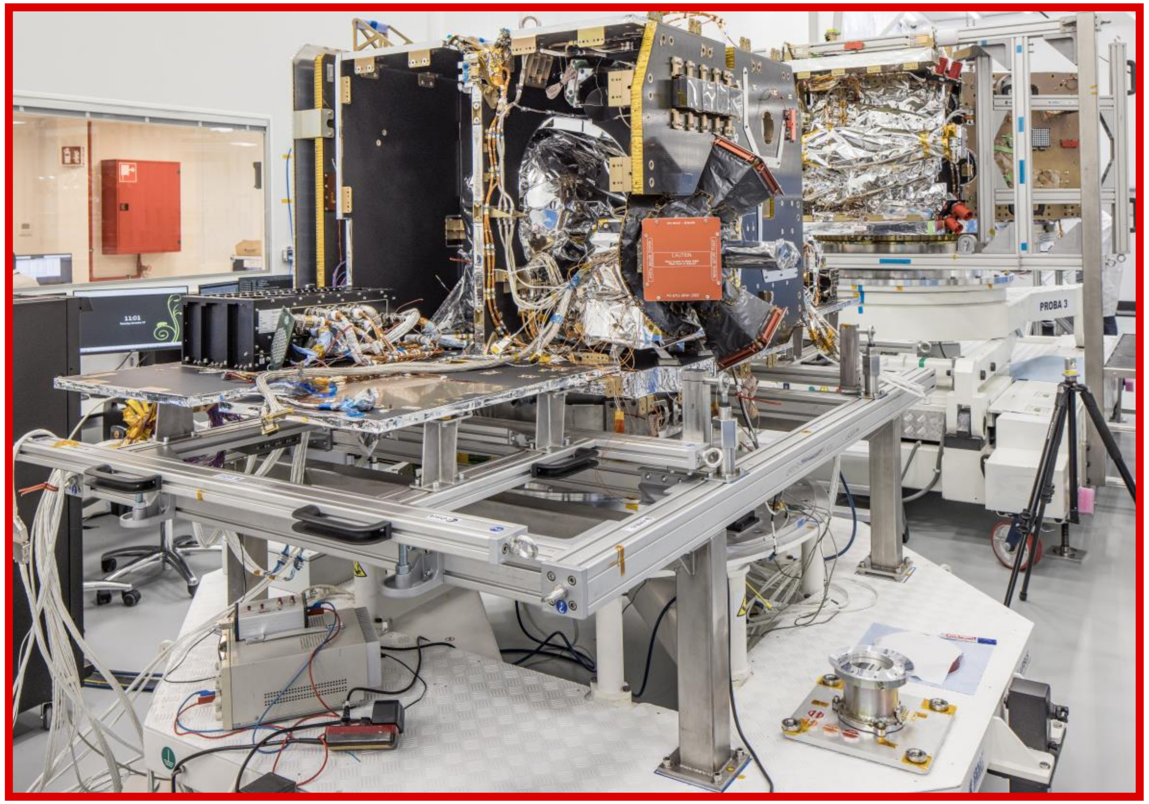
PROBA-V

- ✓ New vegetation monitoring instrument
- ✓ 5 hardware & software space technologies and sensors
- ✓ ADS-B for aircraft monitoring

Launch 2013 -> Operational (+12 years)

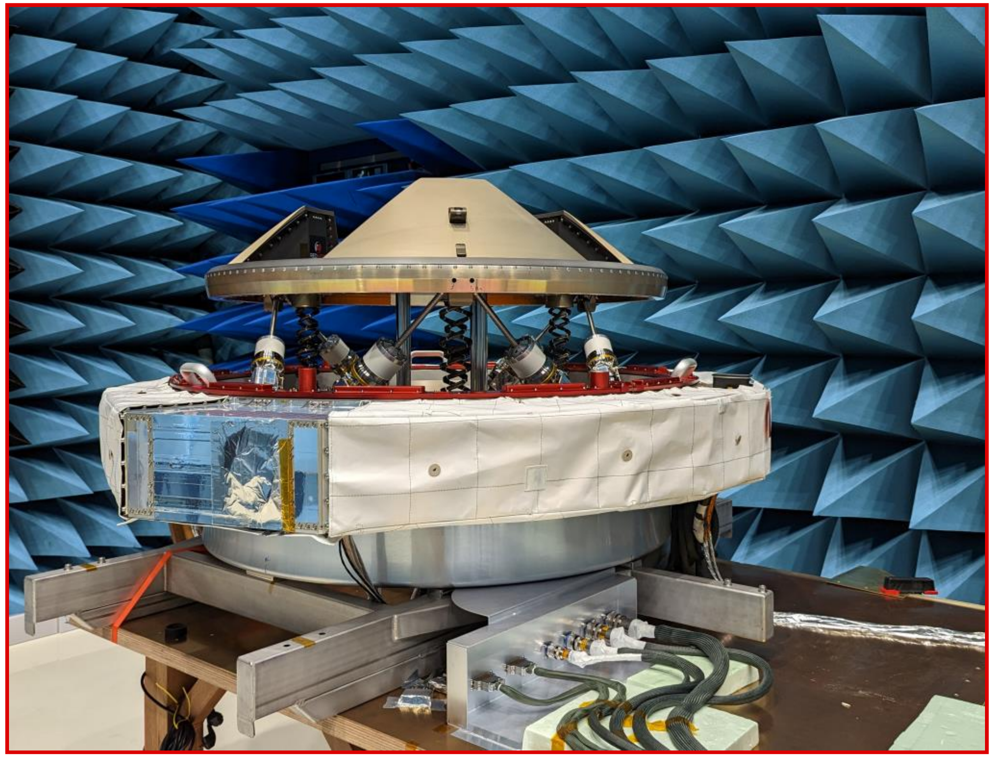
SATELLITE PROJECTS IN PROGRESS

- **PROBA-3** – ESA's first precise formation satellites (launched December 2024)
- **HERA (launched October 2024) & COMET INTERCEPTOR**– ESA's Planetary Defense missions
- **ALTIUS** – ESA's Ozone layer research satellite
- **IOD** – In orbit demonstration technologies for the European Commission



INTERNATIONAL BERTHING AND DOCKING MECHANISM

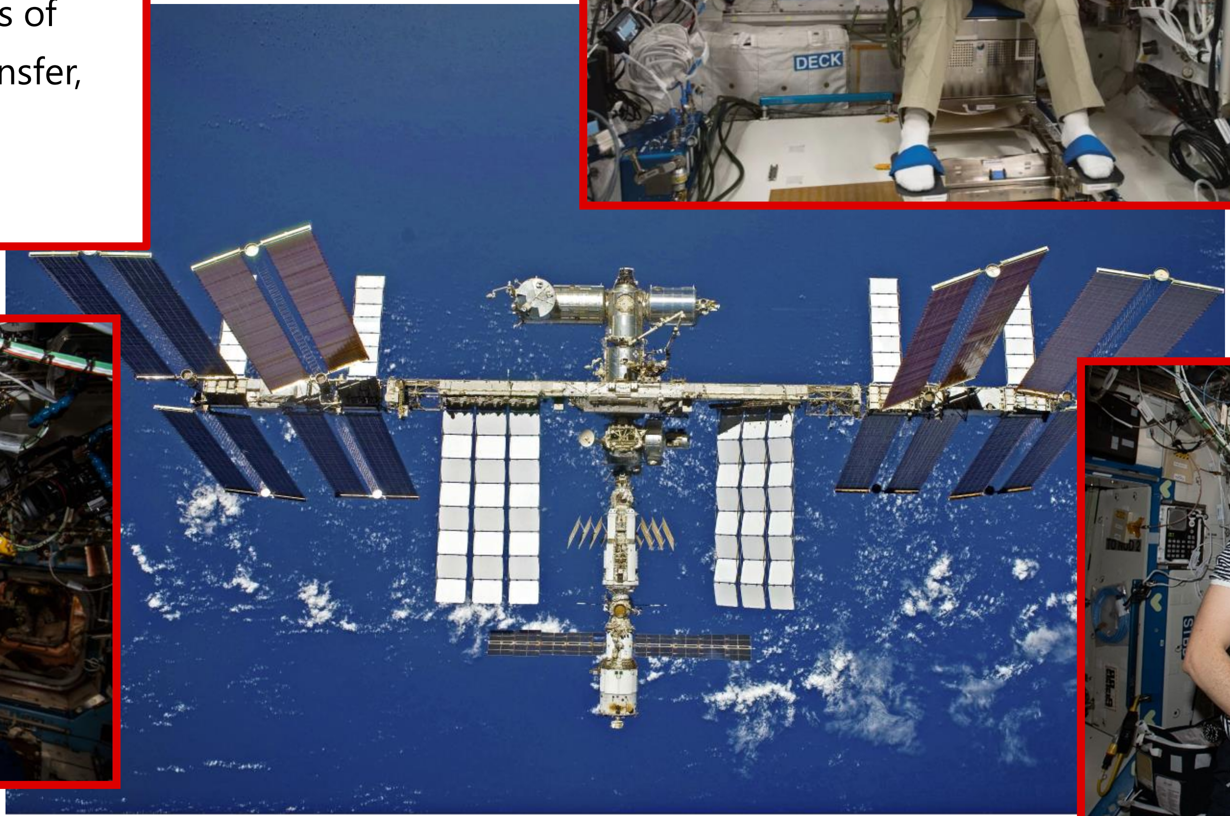
- Autonomous docking of crewed vehicles, cargo vehicles, station modules
- Safe capture and damping in case of vehicle power failure.
- (Re-)berthing of vehicles.
- Resource transfer through automated umbilical mating.



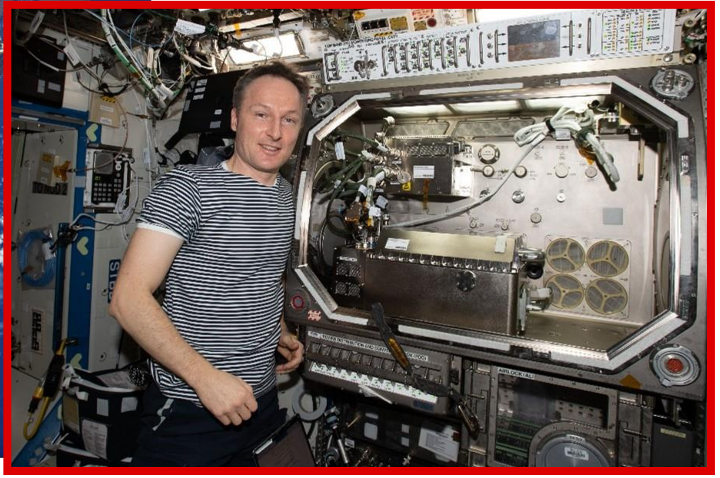
INSTRUMENTS

Instruments flying to the International Space Station for different types of science: new materials, heat transfer, bioprinters, human spaceflight performance, fluid science…

GRIP – studying the effects of weightlessness on the human body



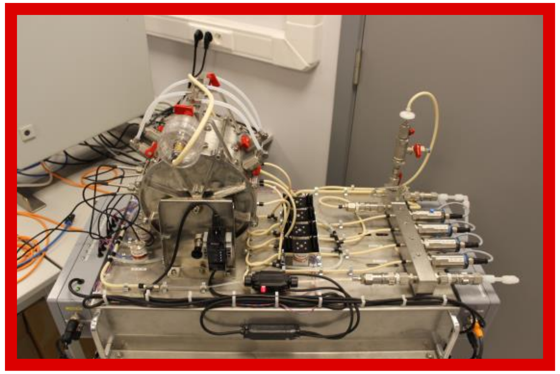
SODI – studying fluid science



Transparent Alloys – studying new materials

LIFE SUPPORT SYSTEMS

Life Support Systems provide everything that astronauts in long-term exploratory missions will need to survive.

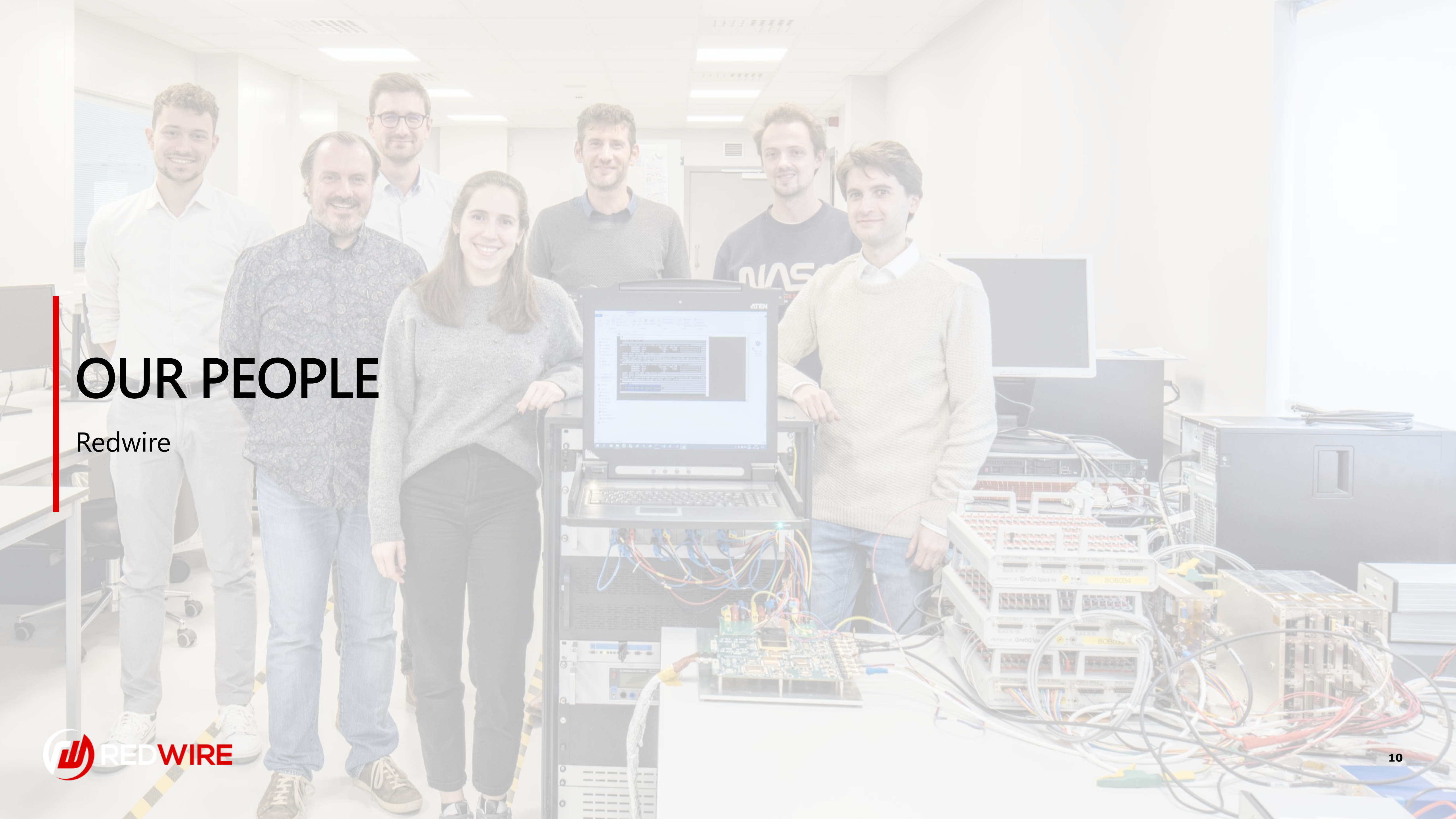


FOOD!

WATER!

FUEL!

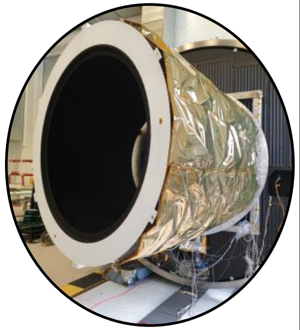
OXYGEN!



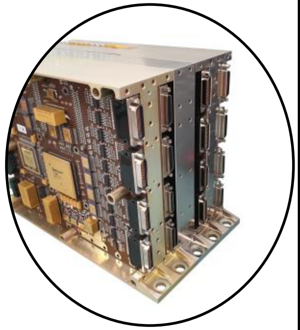
OUR PEOPLE

Redwire

OUR EXPERTISE



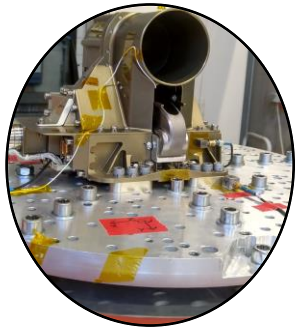
Mechanical & Thermal Engineering



Electrical & Electronics Engineering

Software Engineering

Mechatronics Engineering



Validation & Test

Materials Engineering

Quality/Safety/ Reliability

Project & risk management -Systems engineering - Design & development - Problem solving

EN/AS9100 + ISO9001 Certified

Software in Redwire Space Belgium

Electronic & Software Engineering (ESE) department

	HW	FPGA[1]	SW	Test
Architect / System Engineer	Electronic System Engineer	FPGA Architect	SW Architect	-
Design Engineer	Electronic Design Engineer	FPGA Design Engineer	SW Design Engineer	Electronic Test Engineer

Architect / System Engineer : someone who keeps the overview

Design Engineer : someone who does the detailed design/coding

[1] FPGA stands for: Field Programmable Gate Array

Software in Redwire Space Belgium

Software key words

- ECSS (European Cooperation for Space Standardization) compliant design (incl documentation)
- C-language
- GUI design
- Space instruments (International Space Station – Life sciences):
  - SW Drivers
  - Application SW
- Space equipment (satellites)
  - Bootloader SW
  - SW Drivers
  - RTEMS (Real-Time Executive for Multiprocessor Systems)
  - Control Software
  - Independent Software Validation and Verification (ISVV)

Software in Redwire Space Belgium

ESA – ECSS requirements

- Development is done inline with:
  - ECSS-E-ST-40C: Engineering requirements
  - ECSS-Q-ST-80C: Product Assurance requirements

ECSS-E-ST-40C
6 March 2009

EUROPEAN COOPERATION
FOR SPACE STANDARDIZATION

Space engineering

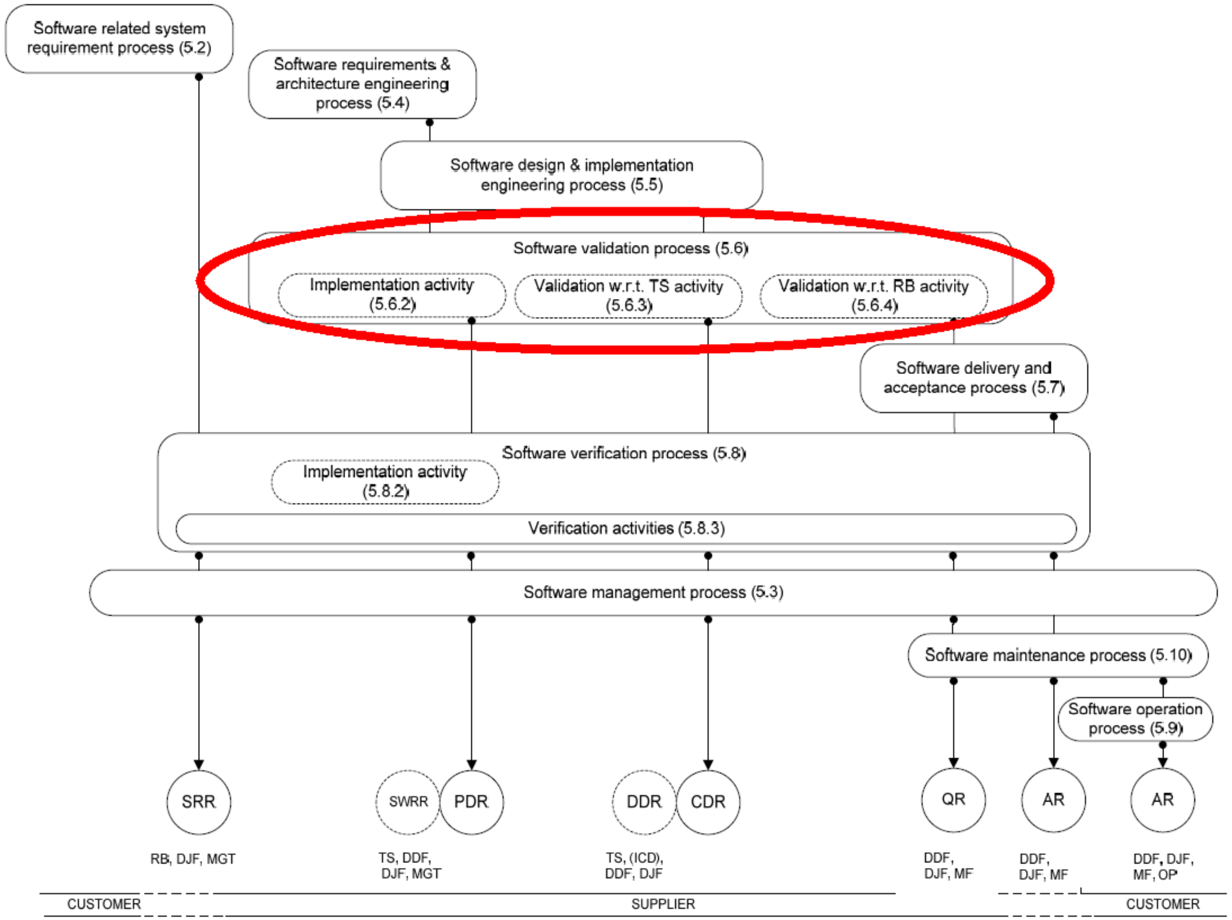
Software

ECSS-Q-ST-80C Rev.1
15 February 2017

EUROPEAN COOPERATION
FOR SPACE STANDARDIZATION

Space product assurance

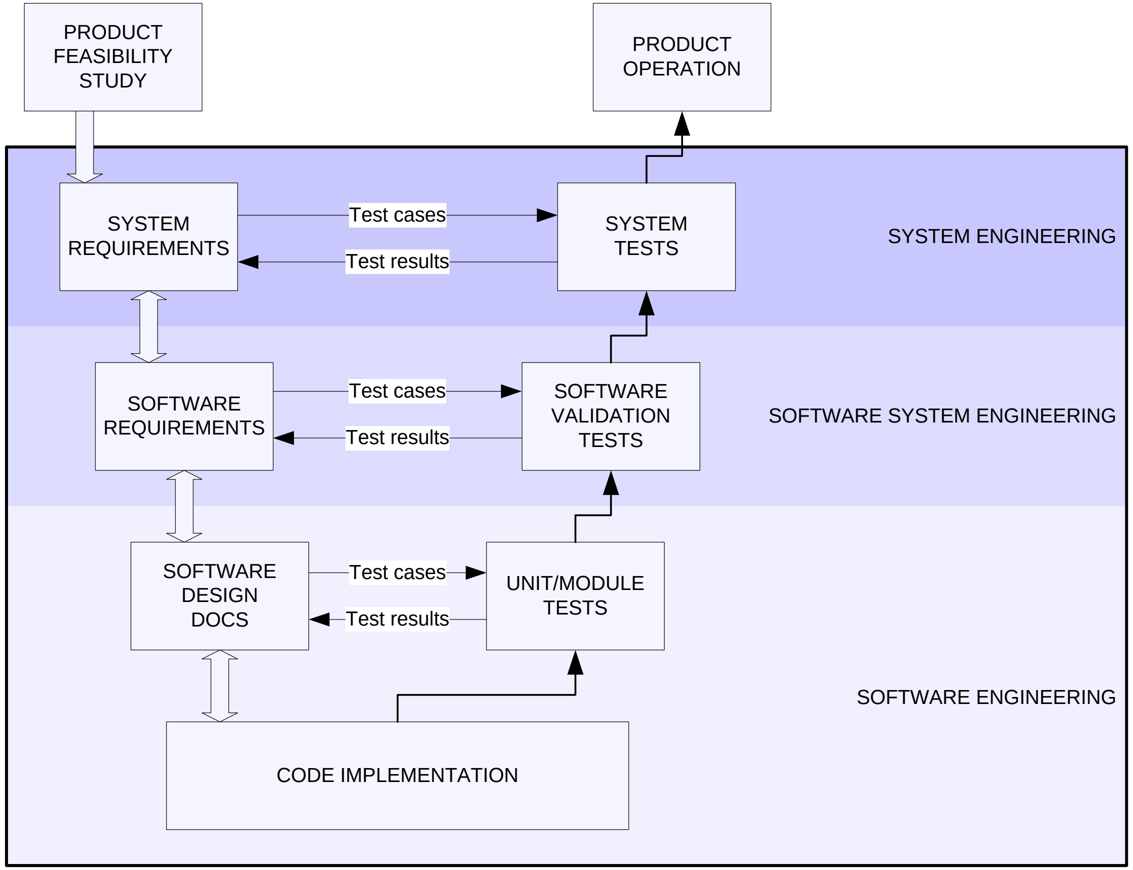
Software product assurance



Software in Redwire Space Belgium

Life-Cycle Model[1]

1. **Sequential Model (Waterfall and V-Model)**
2. Iterative Model
    1. Incremental
    2. Evolutionary
    3. Spiral
    4. Agile
3. Multi-level (nested) Model



[1] More information about different Life-Cycle models: [ECSS-E-HB-40A]

Software in Redwire Space Belgium

SW release scheme

Release	Description
V0.minor	An early development release that implements the basic functions. It has been covered by sanity testing and is functionally working. It has not yet undergone all formal unit, integration or acceptance testing.
V1.minor	An intermediate development release and a possible flight software candidate. This version contains all software components of the detailed design and technical budgets are confirmed. It has been formally covered by all UT, IT and VT on the **QM** avionics.
V2.minor	The V2 version contains corrections of anomalies detected on the **QM** avionics and gets first retested on the QM avionics if changes from V1 to V2 were made. The V2 version will be integrated on the **FM** avionics for final acceptance testing.

Software in Redwire Space Belgium

SW Criticality

Name	Level	Type of Consequences	
		Dependability	Safety
Catastrophic	1	Failure propagation	• Loss of life • Loss of system • …
Critical	2	Loss of mission	• Temporarily disabling but not life-threatening injury • Major damage to an interfacing flight system • …
Major	3	Major mission degradation	
Minor/Negligible	4	Minor mission degradation	

Software in Redwire Space Belgium

SW Criticality

- Investigation on dependability and safety aspects
  - The criticality of a function implemented in the space system shall be assigned in accordance with the severity of identified hazardous events it can cause
  - The highest identified severity of hazardous events shall define the criticality of the function
  - Taking into account the overall system design (do compensating provisions exist which can prevent software-caused

Function	Software Criticality
Catastrophic (I)	**A**, if the SW product is the only means to implement the function **B**, if on top, compensating provisions are available (HW, SW, procedure)
Critical (II)	**B**, if the SW product is the only means to implement the function **C**, if on top, compensating provisions are available (HW, SW, procedure)
Remaining	**D** (mainly for test software)

Software in Redwire Space Belgium

SW Validation

- The process is to confirm that the requirements are correctly and completely implemented in the final product.
- Validation can happen through:
  - (Static) Analysis
  - Review of design
  - Inspection
  - **Testing**
- Validation against Requirements Baseline (RB at system level) and against Technical Specification (TS at subsystem level)

Software in Redwire Space Belgium

SW Validation - Testing

1. Unit Test (UT)
   - To check the correctness of the implementation of a software unit (at source code level) with respect to its definition
   - They are run as early as possible in order to gain sufficient confidence in the source code
   - Including 100% statement and decision coverage (on target)
     - Preferably by test (repeatable)
     - If not possible (defensive programming techniques, coding rules,…): by analysis (manual)
2. Integration Test (IT)
   - To combine the individual software items and applies tests to those assemblies and delivers the integrated software ready for validation testing
   - Including 100% interface coverage (on target)
3. Validation Test (VT)
   - Including 100% requirements coverage (on target but without instrumentation); covering both RB and TS

Software in Redwire Space Belgium

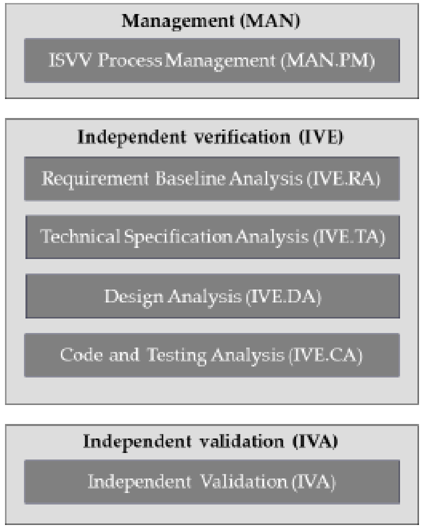
SW Verification

- The process is to confirm that adequate specifications and inputs exist for any activity, and that the outputs of the activities are correct and consistent with the specifications and inputs.
- The execution of test cases will be conducted based on:
  i. Software test specification (UT, IT and VT)
  ii. Software test procedure (UT, IT and VT)
  iii. Software test report (UT, IT and VT)
- But also other aspects will be verified:
  i. Requirements management
  ii. Adherence to selected coding standard
  iii. Software revision and release control
  iv. Software reviews, inspection and documentation

Software in Redwire Space Belgium

Independent Software Validation & Verification (ISVV)

- Objective is to find faults and to increase the confidence in the software
- Applicability is dependent on two factors:
  - Software criticality category
  - Error potential (dependent on development organization, development process, characteristics of the software)
- ISVV requirements can be tailored dependent on Criticality, Complexity and Reusability
- ISVV is executed by independent SW organization
- Independent Verification:
  - Review of requirements baseline (completeness,…)
  - Review of technical specification
  - Review of Architectural Design
  - Review of source code
- Independent Validation: Independent organization creates an additional set of tests to cover requirements



Software in Redwire Space Belgium

Secure Coding

- Independent software Product Assurance (PA) does the reporting of the verification process
- Coding rules to be applied (based on MISRA-C)
- Cyclomatic Complexity is monitored & checked
- Risk management is part of the software development process
- Continuous Integration flow to avoid regression
- Defensive programming techniques to be applied
- Disabling compiler object code optimization (if processing performance can be sustained).
- Investigate compiler warnings

Software in Redwire Space Belgium

Failure Mode and Effects Analysis (FMEA) and Failure Mode, Effects and Criticality Analysis (FMECA)

- Performed to systematically identify potential failures in products or processes
- Failure modes identified in the FMEA analysis are classified according to their criticality leading to FMECA (combined measure of the severity of a failure mode and its probability of occurrence).
- Bottom-up analysis
- Multi-disciplinary activity
- SW: functional approach (based on functions and its effects)
- Analysis of SW reactions to HW failures is covered by dedicated Hardware-Software Interaction Analysis (HSIA)
- FMEA/FMECA provides information and support to FDIR (Failure Detection, Isolation and Recovery)

Software in Redwire Space Belgium

FDIR (Failure Detection, Isolation and Recovery)

- Certain failures need to be detected in housekeeping or by detection of off-nominal behavior
- For each of these failures, the following will be detailed:
  - Observable parameter
  - Corresponding expected value, range or delta-check
  - Response time (monitoring interval x repeat count)
  - Required action
- In addition, it will be specified if the monitoring and/or action has to be done by:
  - Software (for time critical events)
  - FPGA (only monitoring of parameters, no recovery action decided by FPGA)
  - Hosting vehicle/crew/ground (for non-critical events)

Unlock Humanity's Future in Space

Explore Career Opportunities at:

www.redwirespace.com/careers