

Reaching into space  
**TOGETHER**

# FILE MANAGEMENT SERVICES INTERFACE STANDARDISATION

Final Presentation Days  
09.06.2016

Presenter: Arnaud Bourdoux  
ESA TO: Christophe Honvault



# FMSIS FP - Agenda

- Introduction – What, who and why ?
- Study logic
- Study results:
  - Survey of past, present and future mass memories
  - From User, to Technical, to Protocol specification
  - The AAML framework, and its FMSIS dedicated extensions
  - A use case and a demonstrator
  - Lessons learned, and Impacts on Reference Architectures
- Study conclusions
- Steps forward

# Study context / Mass Memories

## Evolution of Mass Memories

- From simple storage areas to a central part of the avionics architecture
- Can hold mission-critical items such as
  - Telecommand timelines
  - Software images
  - OBCPs
  - ...

# Study context / Files

## Introduction of files into operational concepts

- Significant quantity of data to be transferred between space & ground;
- Data naturally structured as files, or aggregated into files
- Availability of reliable file transfer protocols (e.g. CFDP)
- Likely that usage of files becomes common in the near-future missions

# Study context / SAVOIR MASAIS

## SAVOIR-MASAIS Working group constituted

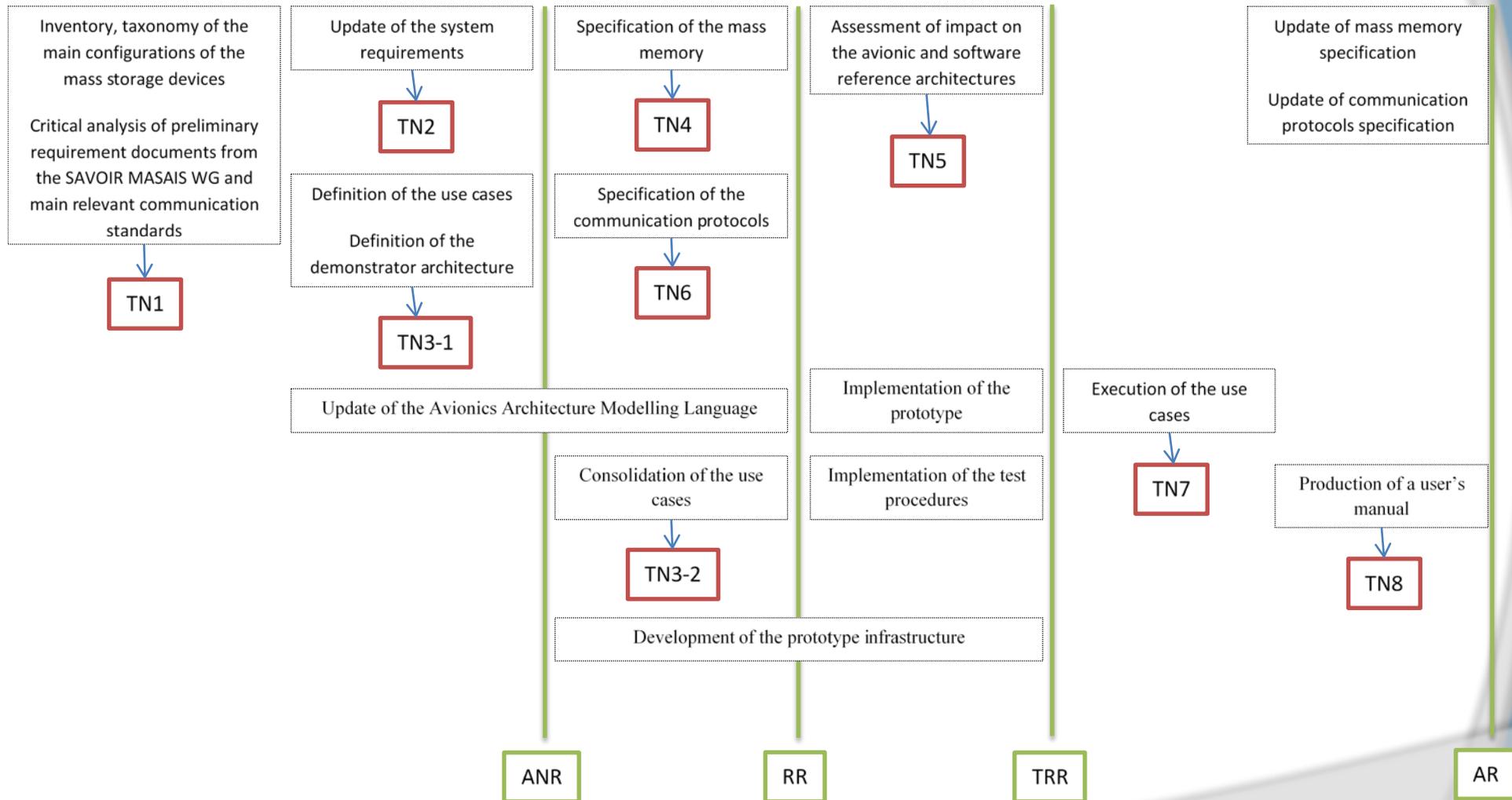
- Stands for SAVOIR Mass Storage Access Interfaces and Services
- Dedicated to the establishment of unified functional and interface requirements for the mass memory function
- FMSIS study in the frame of SAVOIR-MASAIS activities

# Study context / Partners

## FMSIS consortium

- SPACEBEL:
  - Prime contractor
  - Requirements reviewer
  - Demonstrator
- TAS
  - Mass Memory survey
  - User, Technical and Protocol Specification
  - OSRA impact analysis
- GMV
  - AAML tool update

# Study logic



# Study results / Survey / Avionics

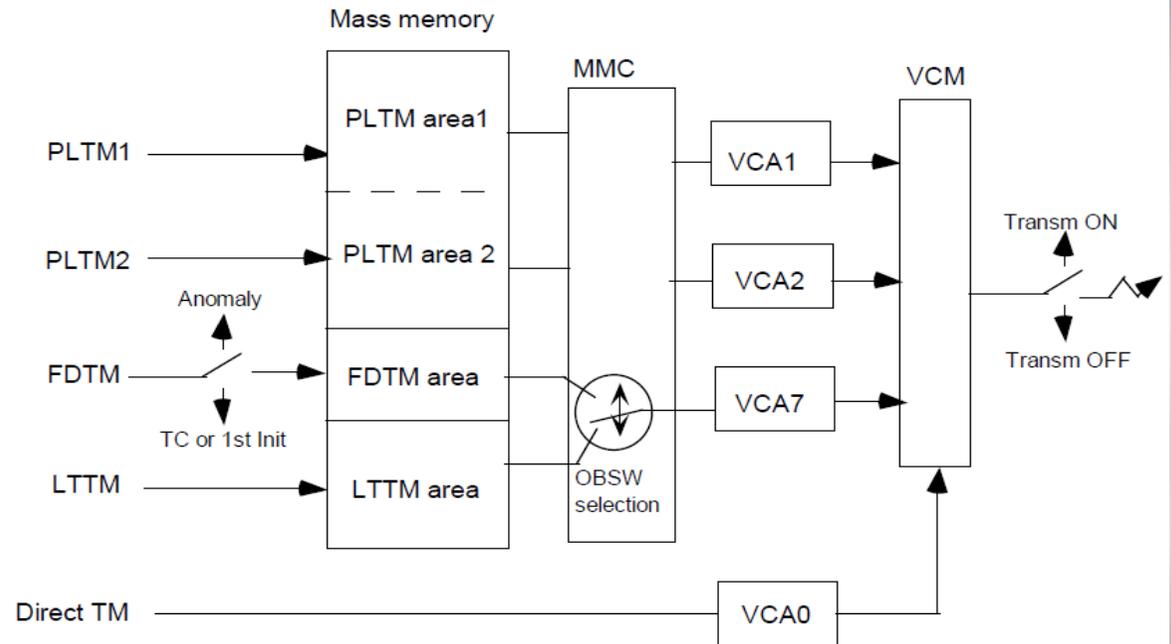
Survey of mass memory architectures on past, present and future missions

- Proteus MK I & II
- Sentinel-3
- Herschel/Planck
- Lisa Pathfinder
- Galileo
- Exomars
- Gaïa
- EUCLID
- JUICE

# Study results / Survey / Avionics

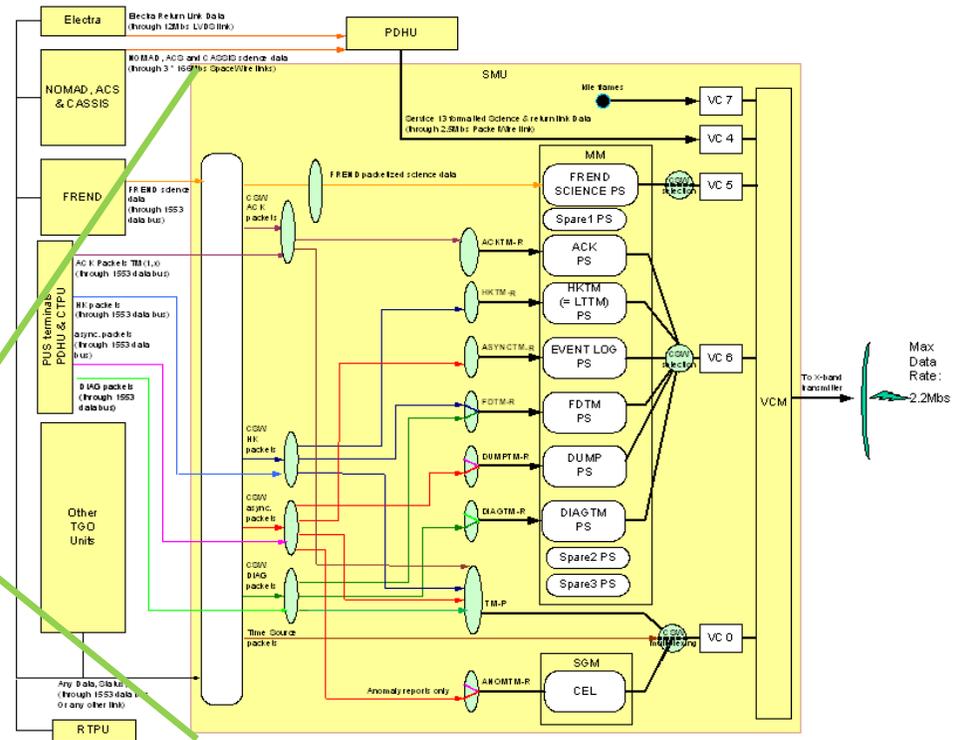
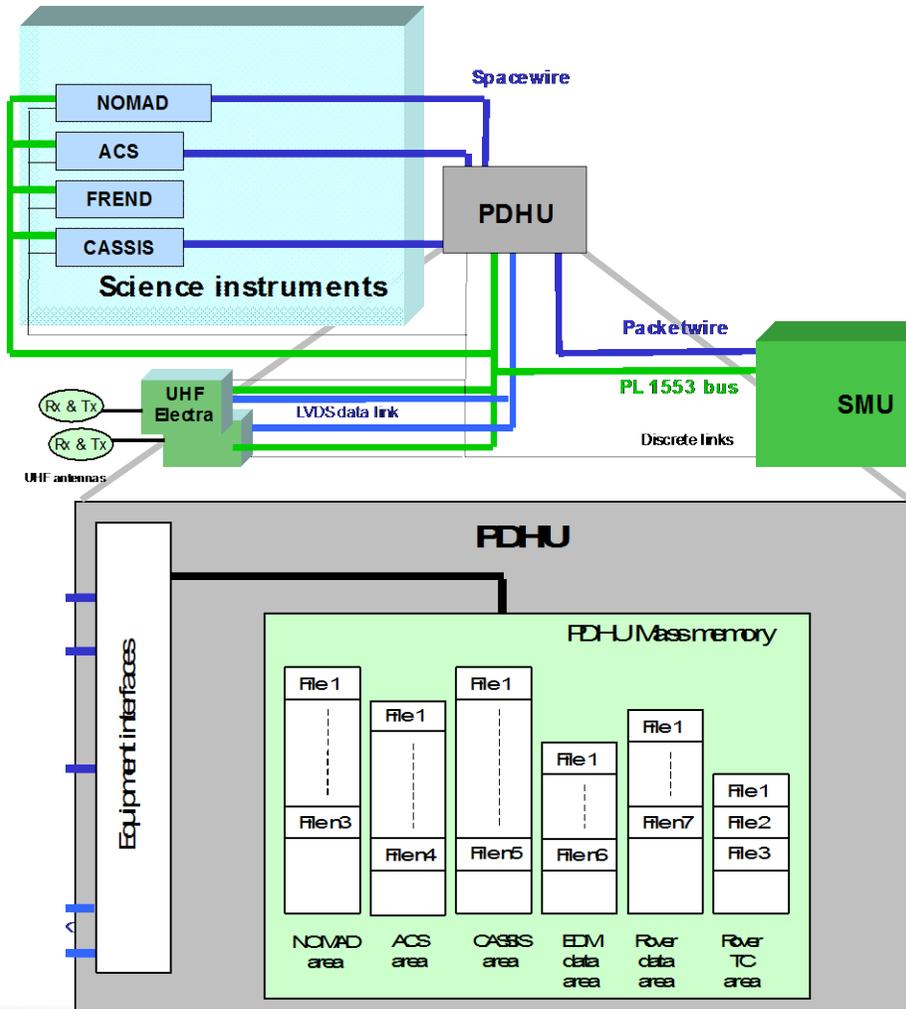
From simple (Proteus Mk I MM)

- Only 4 areas
- Delimited data
- Basic pointer management



# Study results / Survey / Avionics

To complex (Exomars TGO)



# Study results / Survey / Protocols

Many access protocols used or to be used:

- Packet Stores – PUS service 15 widely adopted
- Files:
  - PUS 13 (Large Data Transfer) adapted to files
  - ISIS standardization
  - PUS 23 (File Management) introduced in PUS C
  - SOIS FPSS
  - CFDP
  - Ad-hoc

# Study results / Survey / Protocols

Common denominator to all file accessing protocols:

## Mandatory functions:

- File creation/deletion function: PUS(144), PUS(23), CFDP, SOIS FPSS
- File attributes management function: PUS(144), PUS(23), CFDP
- File listing function: PUS(144), PUS(23)
- File integrity management function: PUS(144)
- File read/write function: PUS(13), CFDP, SOIS FPSS
- File copy/move function: PUS(23), SOIS FPSS

## Optional functions:

- Directory creation/deletion function: PUS(23), CFDP, SOIS FPSS
- Directory attributes management function: PUS(23)
- Directory listing function: PUS(23), CFDP, SOIS FPSS
- File protection management function: PUS(23), SOIS FPSS

# Specification / User

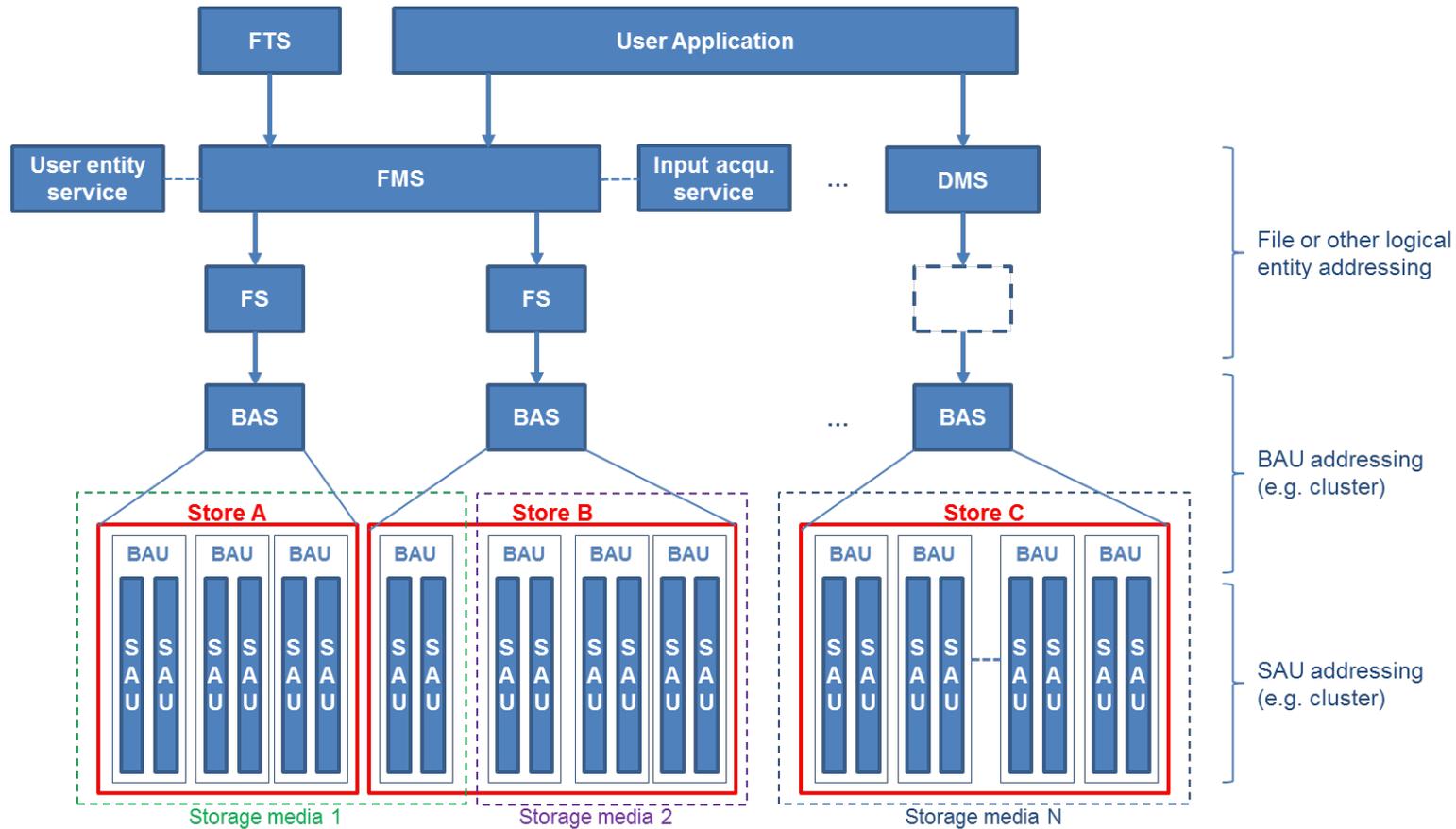
- First output of the study, based on a first iteration provided by the SAVOIR community
- Consolidates needs identified during the MM architecture and protocol survey
- Integrates file and directory awareness at mass memory level
- File abstraction layer has multiple advantages, e.g.
  - On-board autonomy through smart selection of files for communication sessions (according to creation date, data type...)
  - Efficiency of search operations
  - Easier visualisation of available data for operators

# Specification / User

User requirement document addresses several aspects of mass memories:

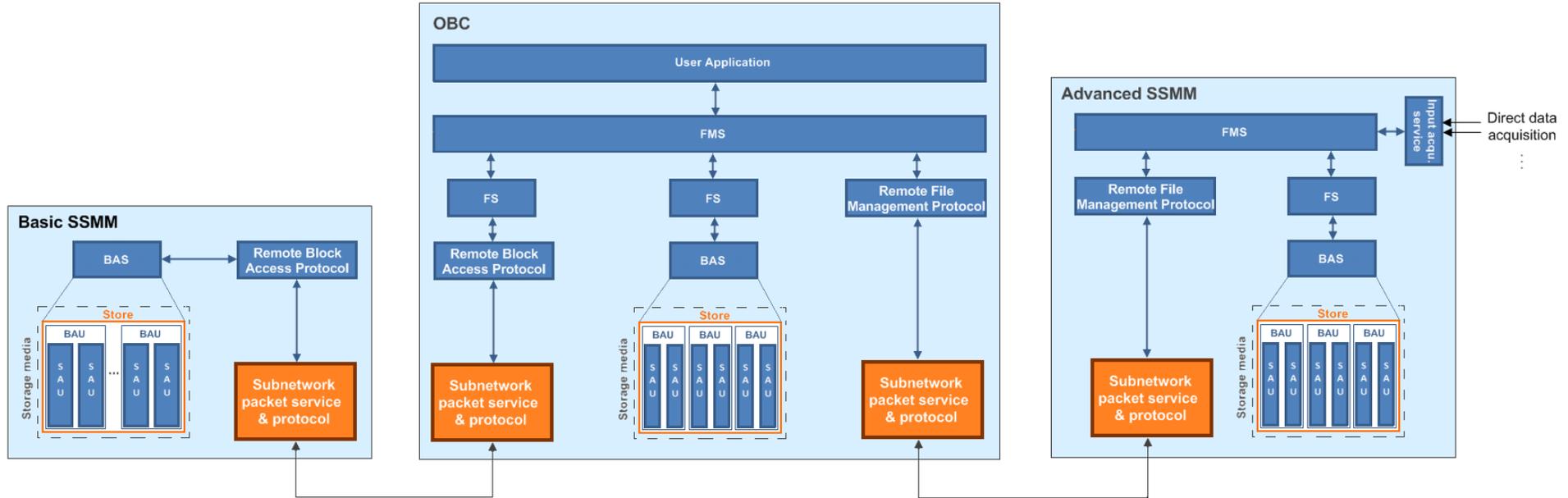
- Generic system requirements (overall functionality)
- Interface requirements (Enforcing standards for different communication link types)
- File Transfer requirement (CFDP based)
- Data organisation requirements:
  - Concept of Store (one data area managed by a block access system)
  - File Store and Packet Store as specialisation of store to specific data
- Performance
- FDIR

# Specification / Data Storage Architecture



# Specification / Deployment example

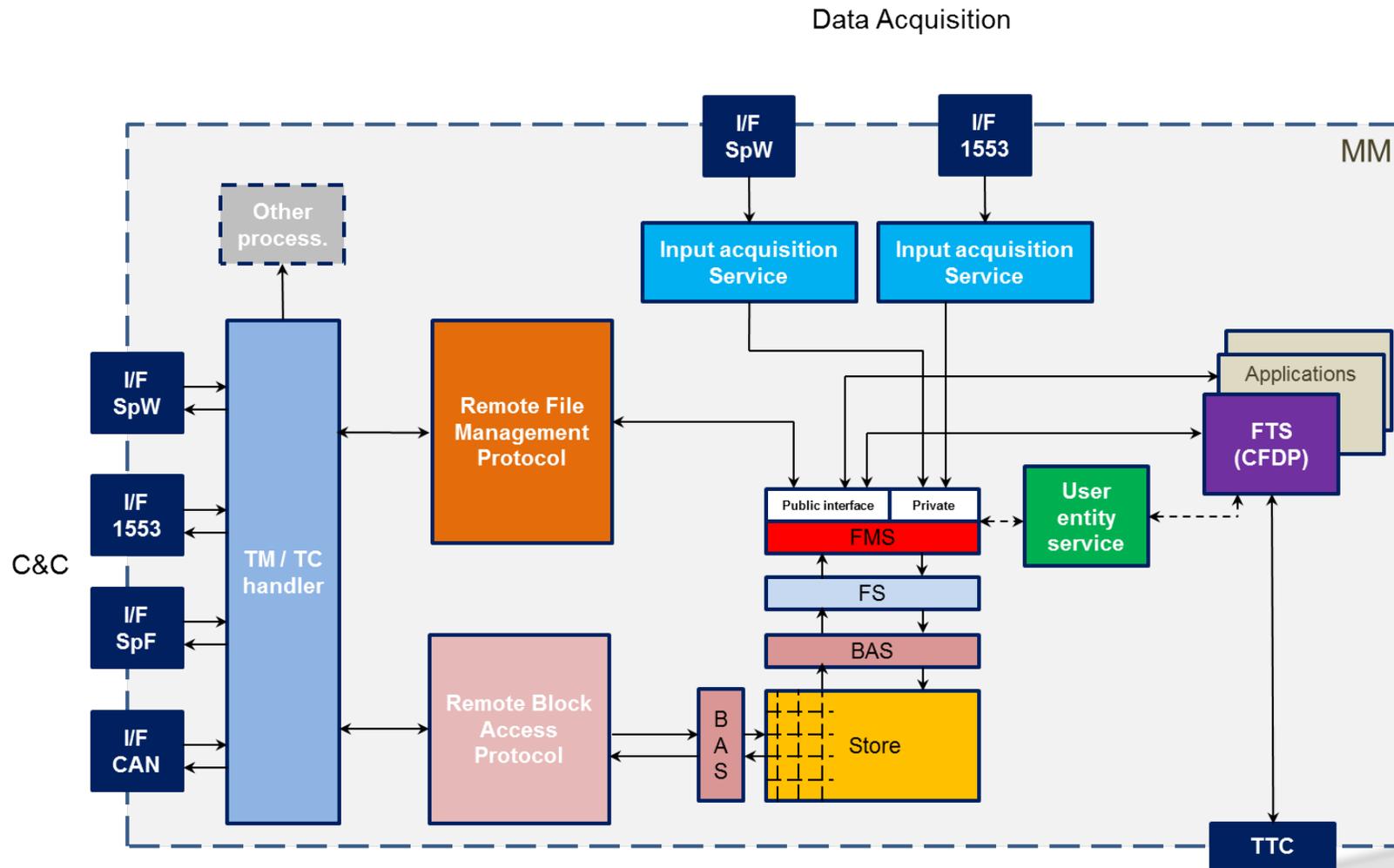
Deployment example used as use case for requirement verification: 3 Mass memories deployed



# Specification / FMSIS specification

- Document derived from user specification (after review by SAVOIR-MASAIS)
- Introduces the generic mass memory functional design
- Specifies all file and directory related directives with their parameters.
  - Usual primitive expected in a file system
  - Specific services related to the mass memory operation heritage

# Specification / Functional design



# Specification / Primitives

In addition to your usual file primitives...

- Set the maximum file size at creation
- Define behaviour upon full file
- Configure autonomous data acquisition to files (map)
- Copy & Move between file stores
- Control (suspend, resume, abort, get status) copies
- Manage custom attributes (statuses and metadata)
- Request file synchronisation (buffering issues)
- Reporting of file store events

# Specification / Opening Schemes

Access Type	Full File Action	Packet Storage	Behavior
Read-only	N/A	N/A	Pointer at beginning, Read, Seek
Read-write	N/A	N/A	Pointer at beginning, Read, Write, Seek. Excess data at end is discarded
Append	Close	No	Pointer at end. Write. Excess data discarded and file closed
Append	Close & Create Next	No	Pointer at end. Write. File closed on excess data, new file opened to store this data (split).
Append	Close File	Yes	Pointer at end. Write. All data discarded on excess data and file closed
Append	Close & Create Next	Yes	Pointer at end. Write. File closed on excess data, new file opened to store this data (atomic)

# Specification / Map

Autonomous acquisition into files:

- Builds upon the append opening options;
- Allows mapping data flowing into data acquisition service according to selection criteria:
  - Exclusive interface mapping
  - Protocol specific address
  - Custom packet fields
  - PUS specifics (APID, type, subtype)

# Specification / File events

Notification of events happening in the file system

- Allows maintaining a ground image of the file system
- Completes the autonomous closing and opening of subsequent files
- Three monitored events:
  - File creation
  - File autoclosing
  - File deletion

# Specification / Protocol

Last specification output from the FMSIS project

- Defines the Remote File Access and Remote Block Access Protocols
- Directly derives the FMSIS technical specification

FMSIS specification and protocol definition prototyped into a demonstrator to verify implementability

# AAML / Overview

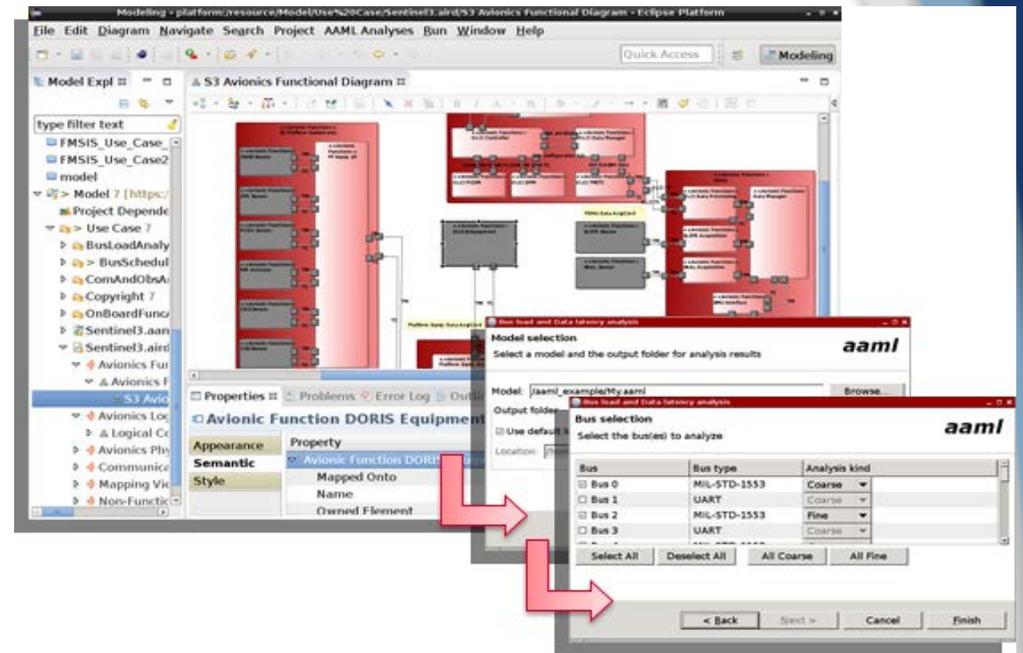
- ❑ The **AAML (Avionics Architecture Modelling Language)** framework provides a toolchain to advance the avionics engineering practices towards a model-based approach.

- ❑ **Main elements:**

- AAML Metamodel.
- Graphical Editor at three different levels: functional, logical and physical.
- A set of analyses to refine/verify the avionics architecture.

- ❑ **Technologies:**

- Developed as a set of Eclipse plug-ins that configure a design and analysis environment integrated into the Eclipse platform (Eclipse Luna).
  - The AAML Metamodel is based on *Ecore* and developed using *EMF*. It is inspired in the Space Component Model (SCM) of OSRA.
  - The Graphical Editor is based on Sirius.



# AAML / FMSIS and AAML toolchain

❑ In the frame of FMSIS project, **new capabilities have been added to the AAML Framework** in order to:

- Address FMSIS requirements concerning Mass Memories and Communication Protocols.
- Support the demonstration of FMSIS Use Case.

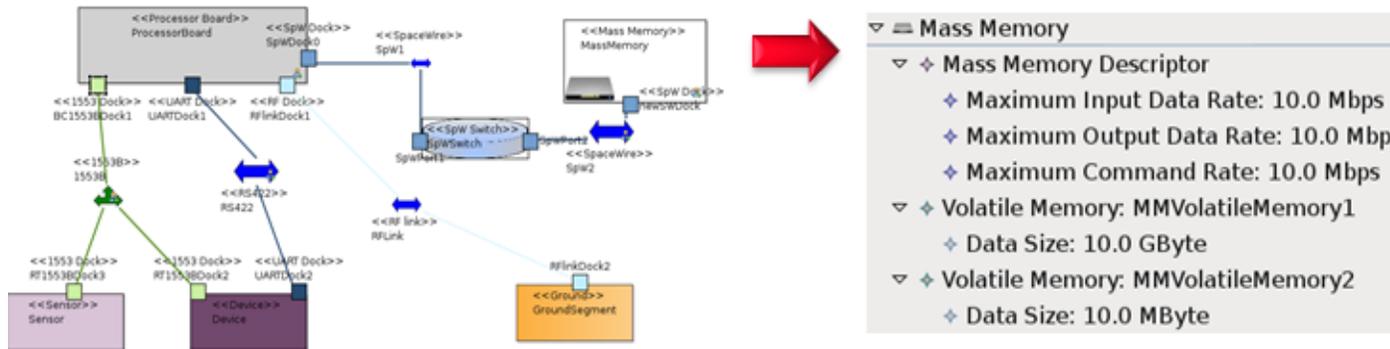
❑ These extensions are gathered in three **main groups**:

1. Mass Memories.
2. SpaceWire.
3. Standards/Protocols.

Next slides list the main updates. Their implementations impact the AAML Component Model, the Graphical Editor and the Analysis Tools.

# AAML / Extensions / Mass Memories

- Definition of **SAVOIR Mass Memories (MM)** and their configuration.



- MM organization characteristics: **Data Storage Systems (DSS)** supporting both file and packet systems.

	Kind	Physical Memory	Endiann	Initial Adr	Final Adr	SAU Val	SAU Uni	SALU Va	SALU Un						
Log. Comp.: Control_Management															
Store: Store1	PayloadData	Volatile Memory MMVolatile													
Memory Area: Area1			BIG_ENDIAN												
Memory Area: Area2			BIG_ENDIAN												
Store: Store2	PlatformData	Volatile Memory MMVolatile													
Memory Area: Area1			BIG_ENDIAN												
Memory Area: Area2			BIG_ENDIAN												
Log. Comp: Control_Management															
Store: Store1															
FS Organization															
Directory															
File															
File															
Store: Store2															
PS Organization															
Packet															
Packet															
RootDirectory															
cont.h															
readme.txt															
Packet1															
Packet2															

- Mass Memory Analysis.

# AAML / Extensions / SpaceWire, Standards and Protocols

## □ SpaceWire:

- Support of SpaceWire communication links and their configuration:
  - SpaceWire Link with its maximum throughput and maximum load margin.
  - SpaceWire Nodes with its core frequency, data rate and logical address.
  - SpaceWire Routing Switches with its type, maximum speed and physical ports.
- The analyses include the “SpaceWire load” through SpaceWire links, nodes and routing switches.

```
SpaceWire 'SpW1'
Requirements - Mode 'Operational':
  Maximum Throughput: 300.0 Mbps
  Maximum Load: 50.0 %

Results - Mode 'Operational':
  Worst case bus load: 0.5248 %
  Worst case data throughput: 262400 bps
  Best case bus load: 0.2624 %
  Best case data throughput: 131200 bps
```

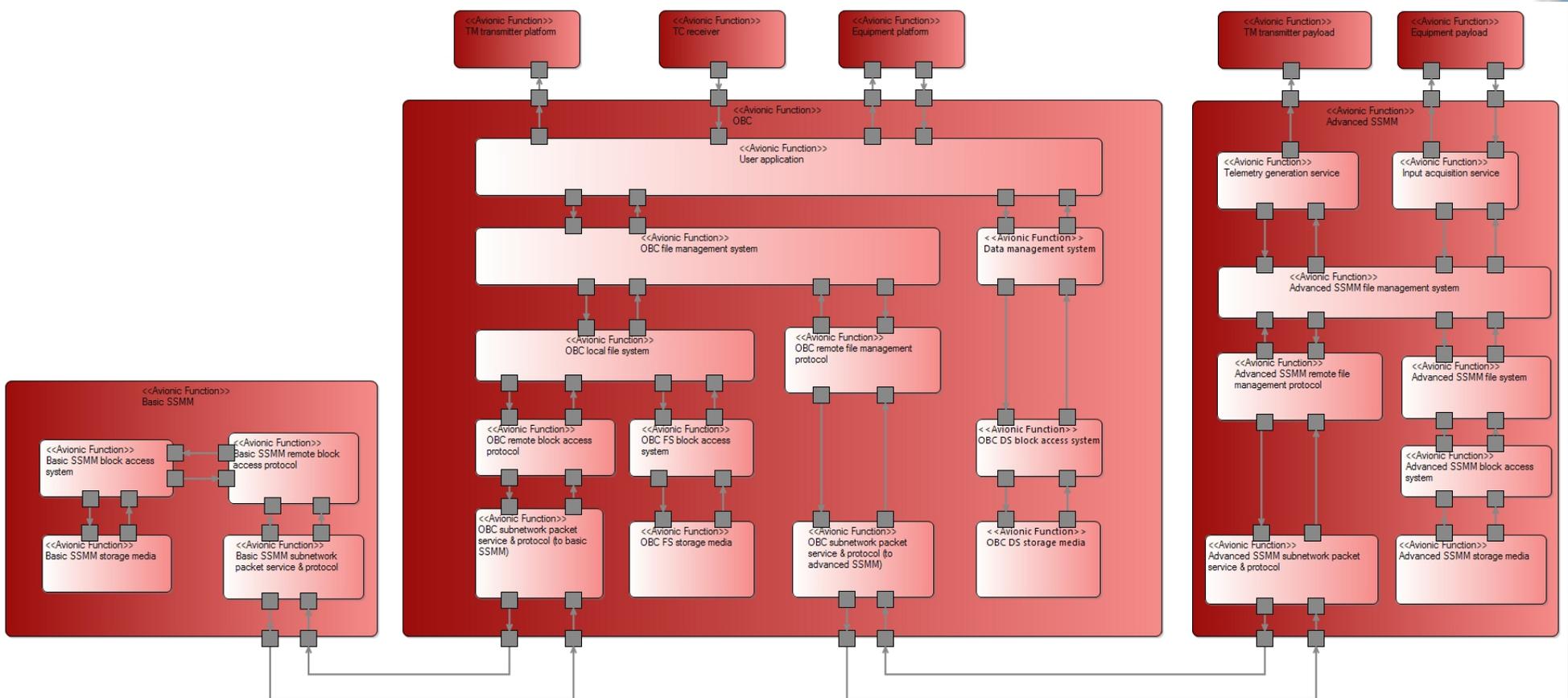
## □ Standards and Protocols:

- Standards: CCSDS Space Packets, PUS Packets and Native Packets.
- Protocols: CCSDS File Delivery Protocol (CFDP) - Class 1 and Class 2.

Connection	HW Communication Media	Standard	Protocol
Interface Connection conn_Control_Data_Storage_IF_Control_Data_Storage	Space Wire SpW1, Space Wire SpW2	PUS	NONE
Interface Connection conn_Control_Data_Storage_IF_Control_Data_Storage	Space Wire SpW1, Space Wire SpW2	PUS	NONE
Interface Connection conn_Device_IF_Device_IF	MIL STD 1553B 1553B , RS422 RS422	PUS	NONE
Interface Connection conn_Sensor_Data_Acquisition_IF_Sensor_Data_Acquisition	MIL STD 1553B 1553B	PUS	NONE
Interface Connection conn_Ground_IF_Ground_IF	RFLink RFLink	PUS	CFDP_CLASS_1

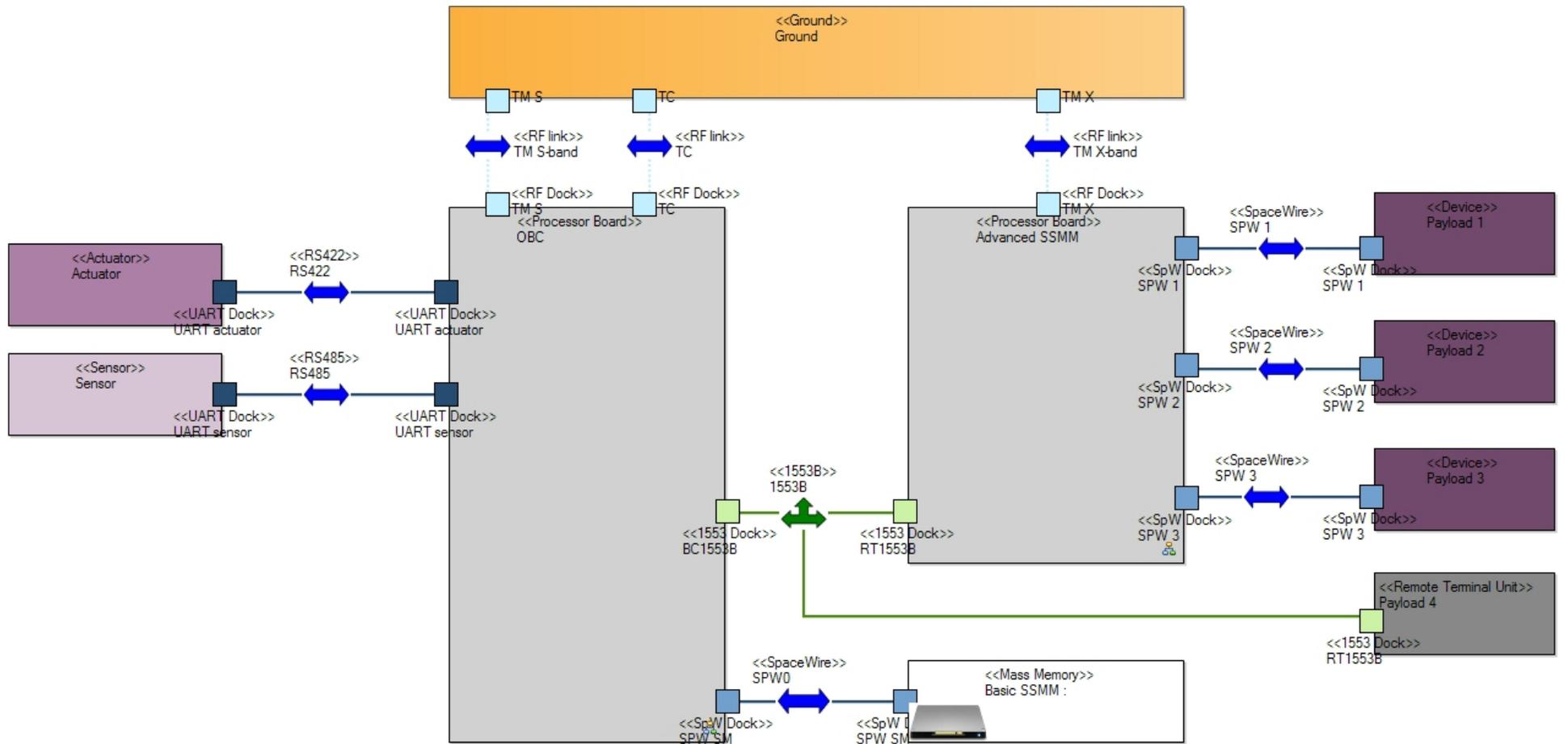
# Demonstrator / Use case

## Use case storage configuration



# Demonstrator / Use case

## Communications configuration



# Demonstrator / Scenarios

## Setup:

- Basic SSMM
  - Equipment platform data files
  - File management from the OBC
- OBC
  - HKTM packet store, CEL packet store, SW configuration
- Advanced SSMM
  - Payload data files
  - File management on the SSMM

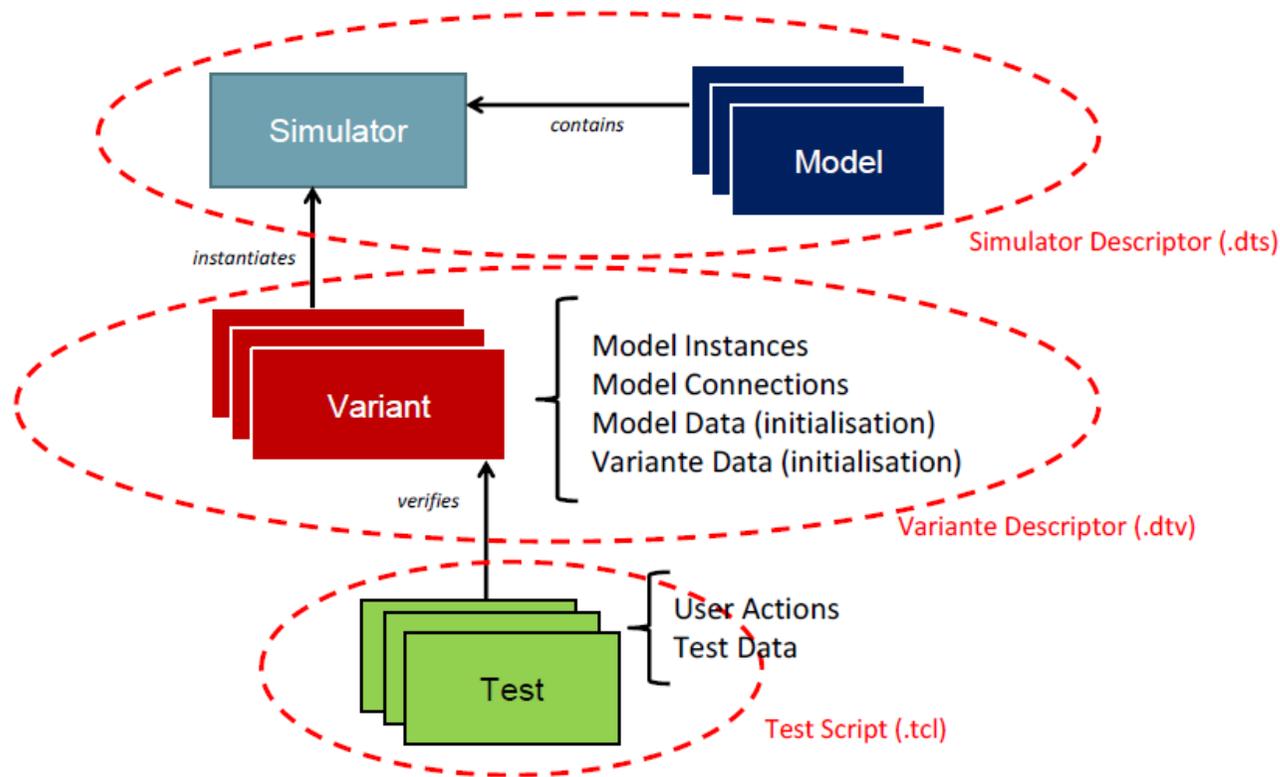
# Demonstrator / Scenarios

## Scenarios:

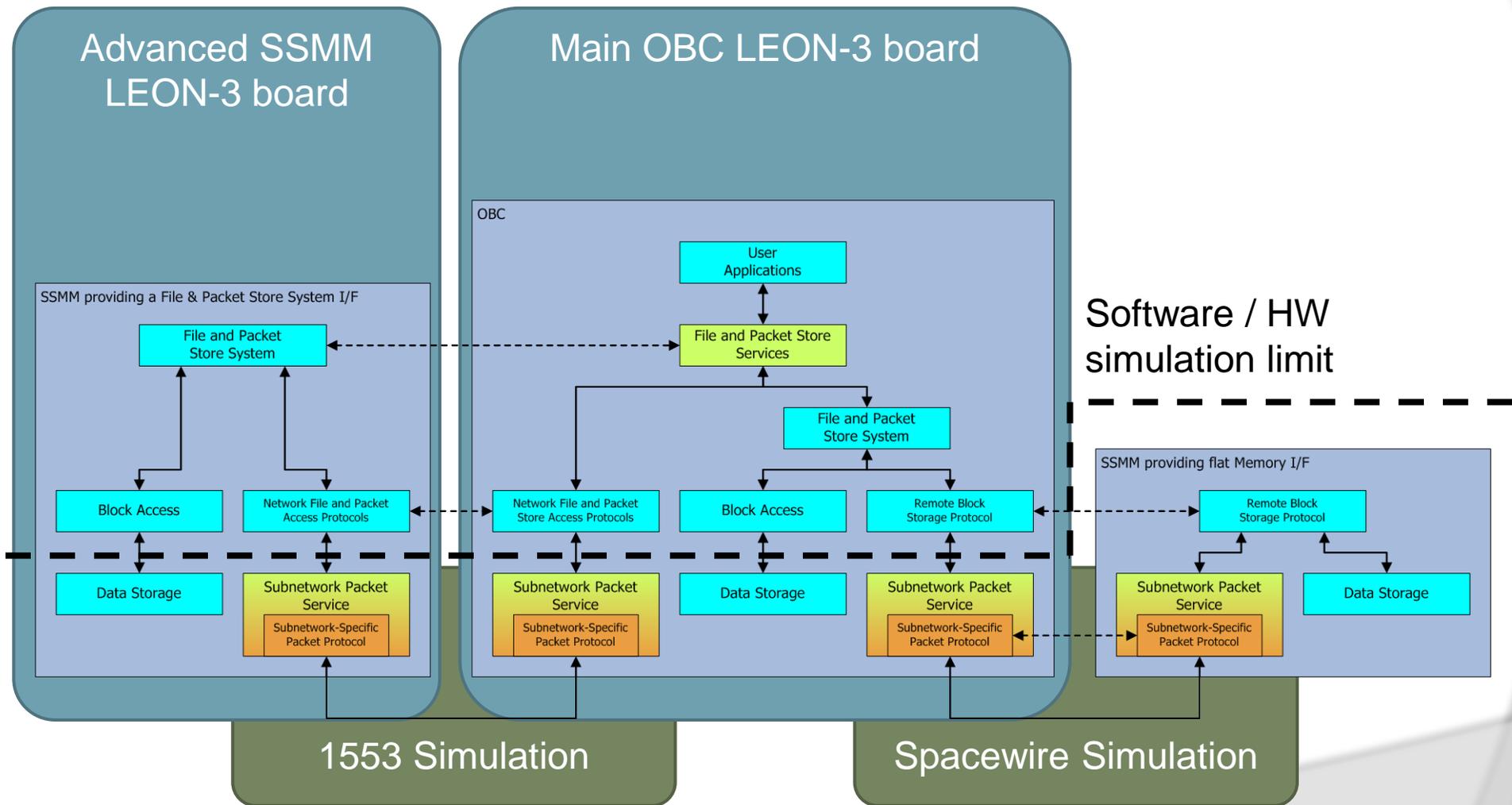
- SC1: Record without visibility
- SC2: Download in visibility
- SC3: Reconfigure recording capabilities
- SC4: Uplink files
- SC5: Manage directories and files
- SC6: Perform maintenance operations
- SC7: Perform FDIR operations

# Demonstrator / Architecture

## Demonstrator based on the BASILES architecture



# Demonstrator / Architecture





# Demonstrator / Results

ESTEC's CFDP ground entity integrated in simulation and verification environment

The image shows a simulation environment with two main windows. The left window is a terminal window titled '/cygdrive/i/WIN\_server' showing the execution of a program. The right window is a network traffic monitor showing upload and download statistics.

```
ocn@l-pcocrn /cygdrive/i
$ ls
20151206_153534_DSC_3050.JPG  file1copy  logging      tests
CFDP_Temp                    Leon3NL.zip small_file.txt WIN_server

ocn@l-pcocrn /cygdrive/i
$ cd WIN_server/

ocn@l-pcocrn /cygdrive/i/WIN_server
$ ./WIN_server.exe
MSG |WIN_server: -----
MSG |WIN_server: Killing the ground CFDP entity
ERROR: The process "CFDP_SVF.exe" not found.
MSG |WIN_server: Wait for 5 seconds to kill the ground CFDP entity. Please wait
...
1 dir(s) moved.
MSG |WIN_server: Waiting for a connection from SVF...
MSG |SRV10091TC: Server is now waiting for a connection...
MSG |WIN_server: Waiting for messages from SVF...
MSG |WIN_server: starting new entity
MSG |WIN_server: result of command <cmd /C "I:\CFDP_Temp\CFDP_SVF.exe" &
> : 0
MSG |WIN_server: Waiting for messages from SVF...
```

The network traffic monitor window shows the following data:

UPLOAD									
Size	Dest I...	Next Ho...	Status	Progre...	Bytes S...	Duration	Rate		
DOWNLOAD									
Size	Dest I...	Next Ho...	Status	Progre...	Bytes R...	Duration	Rate		
02_00	3:/pltf_eqpt_data	I:\tests\pltf_eqpt_d...	Termin...	100	25088	0H:0M:3s	66.27 K...		
02_01	3:/pltf_eqpt_data.00	I:\tests\pltf_eqpt_d...	Termin...	100	24576	0H:0M:2s	98.60 K...		

11:30:00 RCV Transaction[02\_01] Terminated ! [ACKNOWLEDGED TRANSMISSION] SND Transactions : 0 Receiving Transactions : 0

# Demonstrator / Results

Requirement prototyping activity results:

- Successful execution of scenarios
- Feedback to specification
  - Necessity of an offset parameter in read & write requests (performance over time slotted buses like 1553)
  - Missing protocol messages related to the continuous storage of the « continuous storage » feature, and removal of pointer management implementation constraint (effect on other file descriptors)

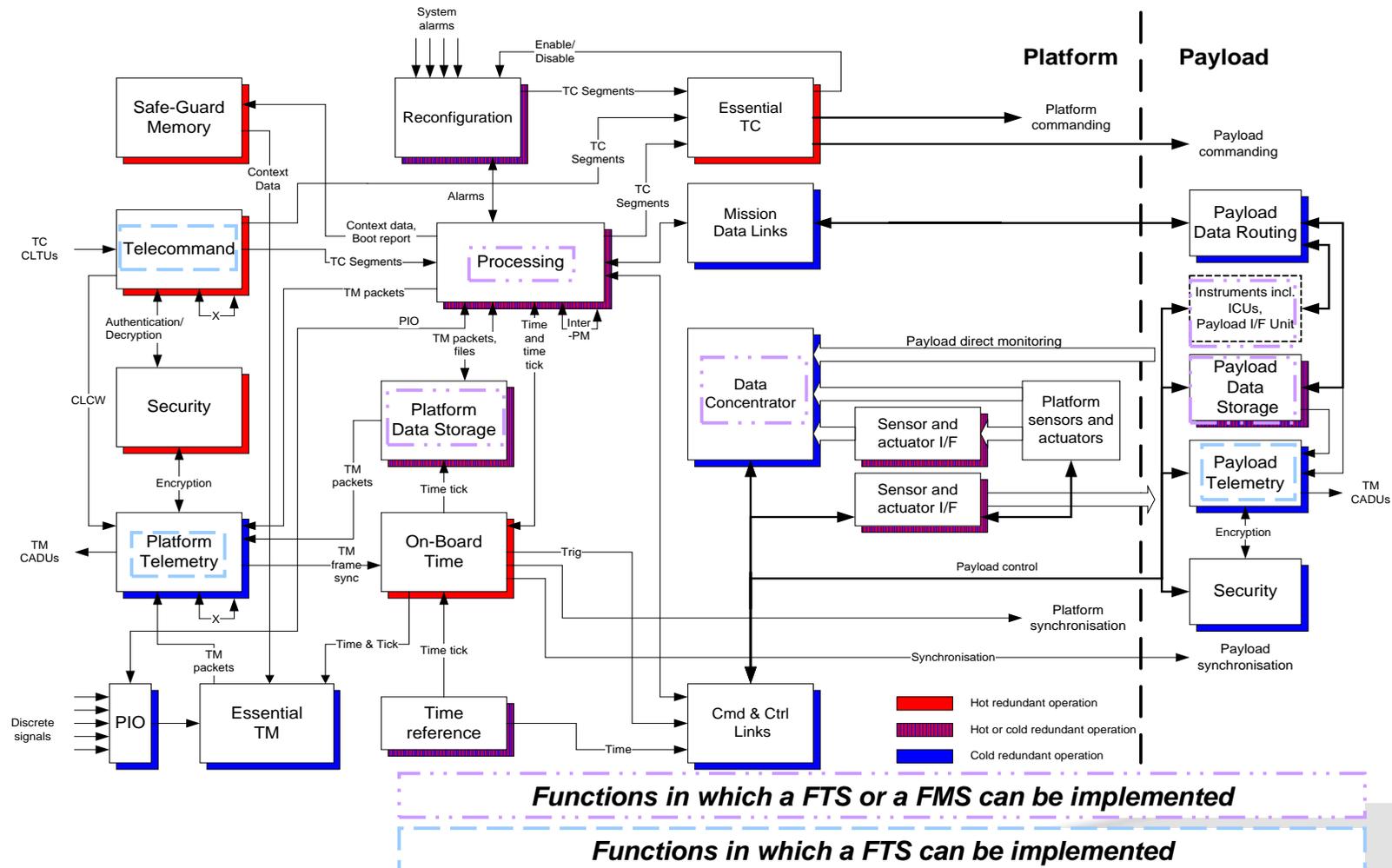
# Demonstrator / Results

Requirement prototyping activity results:

- Feedback to operation / TMTC standards
  - Expose file mapping capabilities to PUS
  - Extend the concept of S23 file attributes
    - File Statuses (Inherent to FS: size, lock status, ...)
    - File Metadata (User-Modifiable: Downlink priority, datatype...)
- First attempt at Service 15 (packet storage) implementation using files

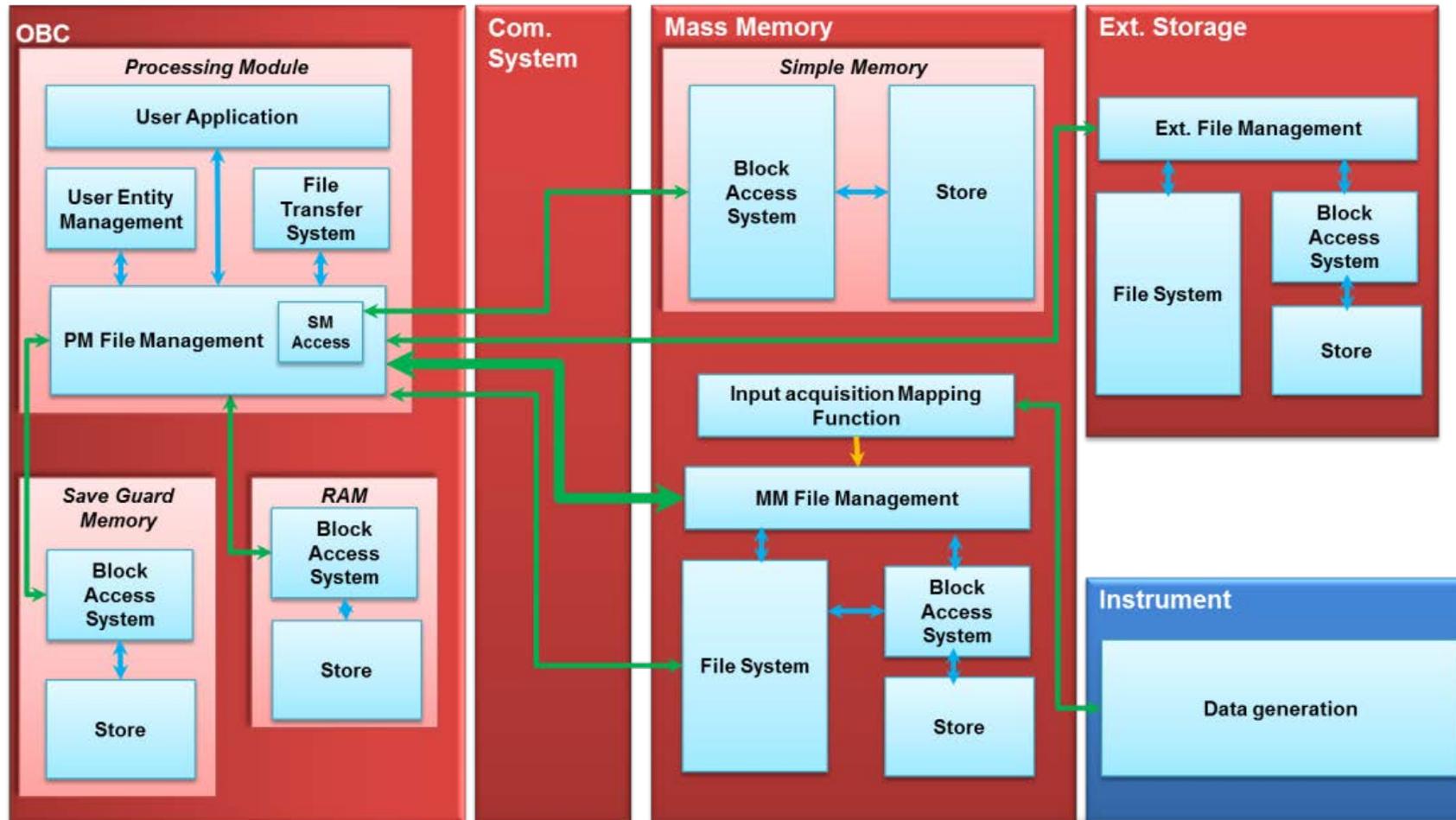
# Lessons learned and ASRA impacts

## Possible insertions of FMS & FTS functions in ASRA:



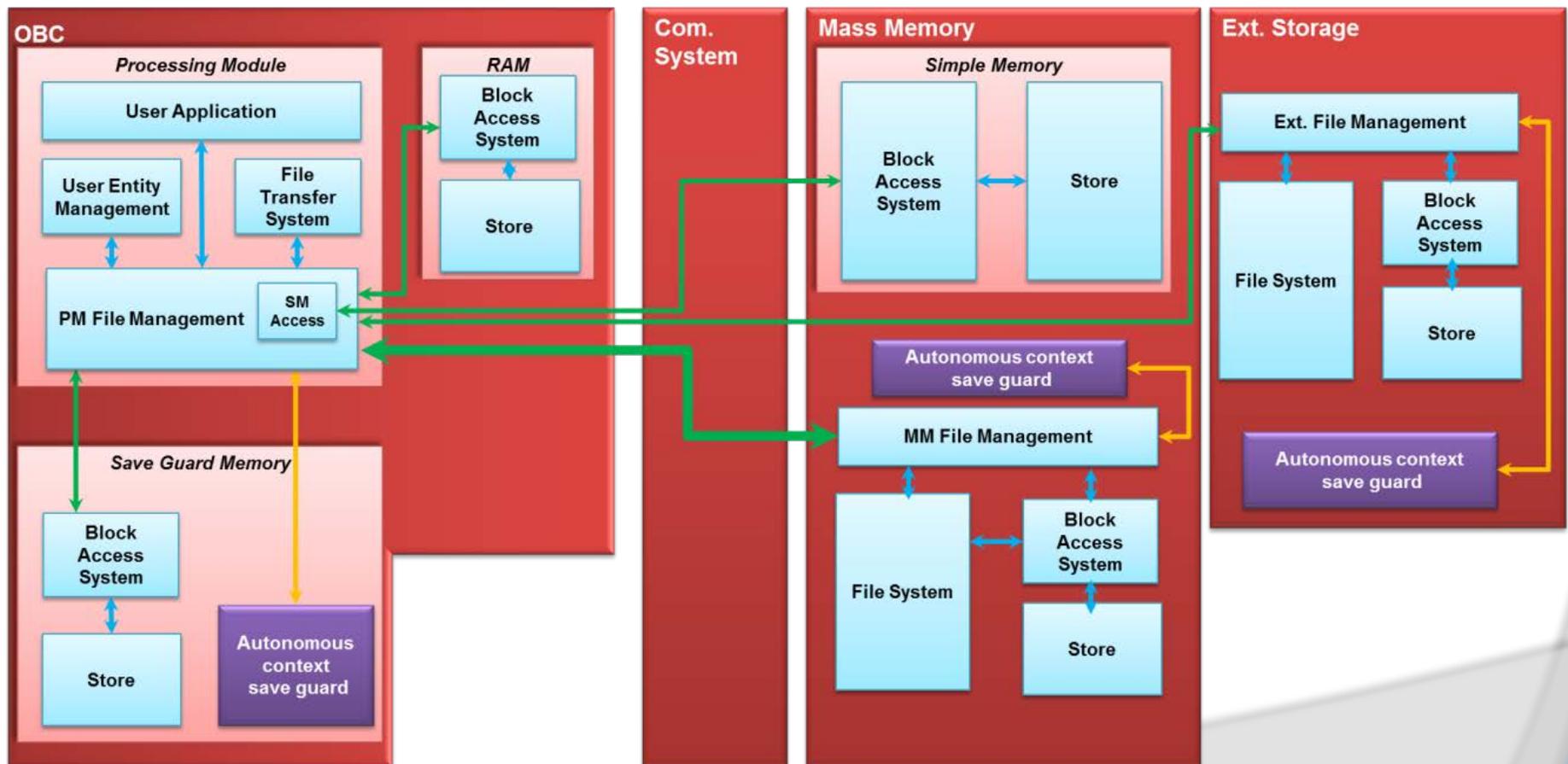
# Lessons learned and ASRA impacts

Interaction between OBC FMS and storage devices:



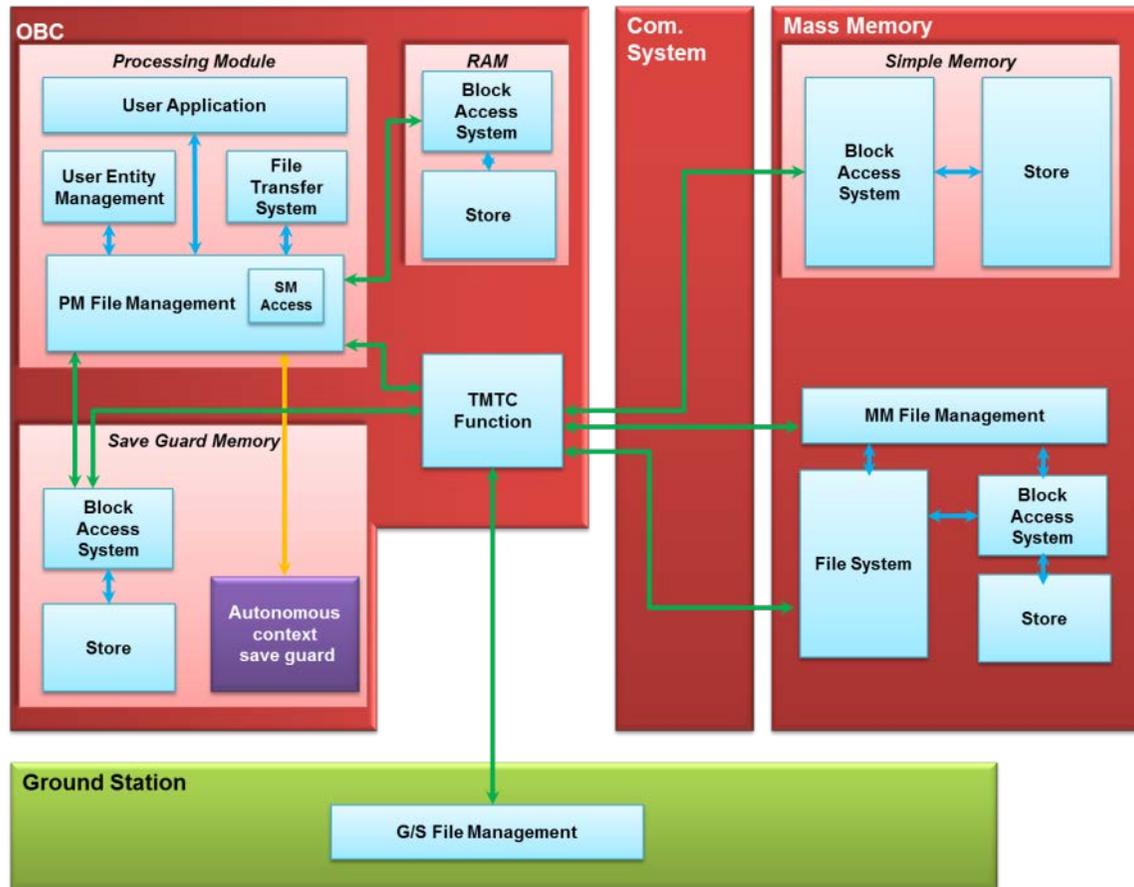
# Lessons learned and ASRA impacts

Autonomous storage for memory organisation (store creation, BAU / SAU allocation...)



# Lessons learned and ASRA impacts

## Ground – board communication



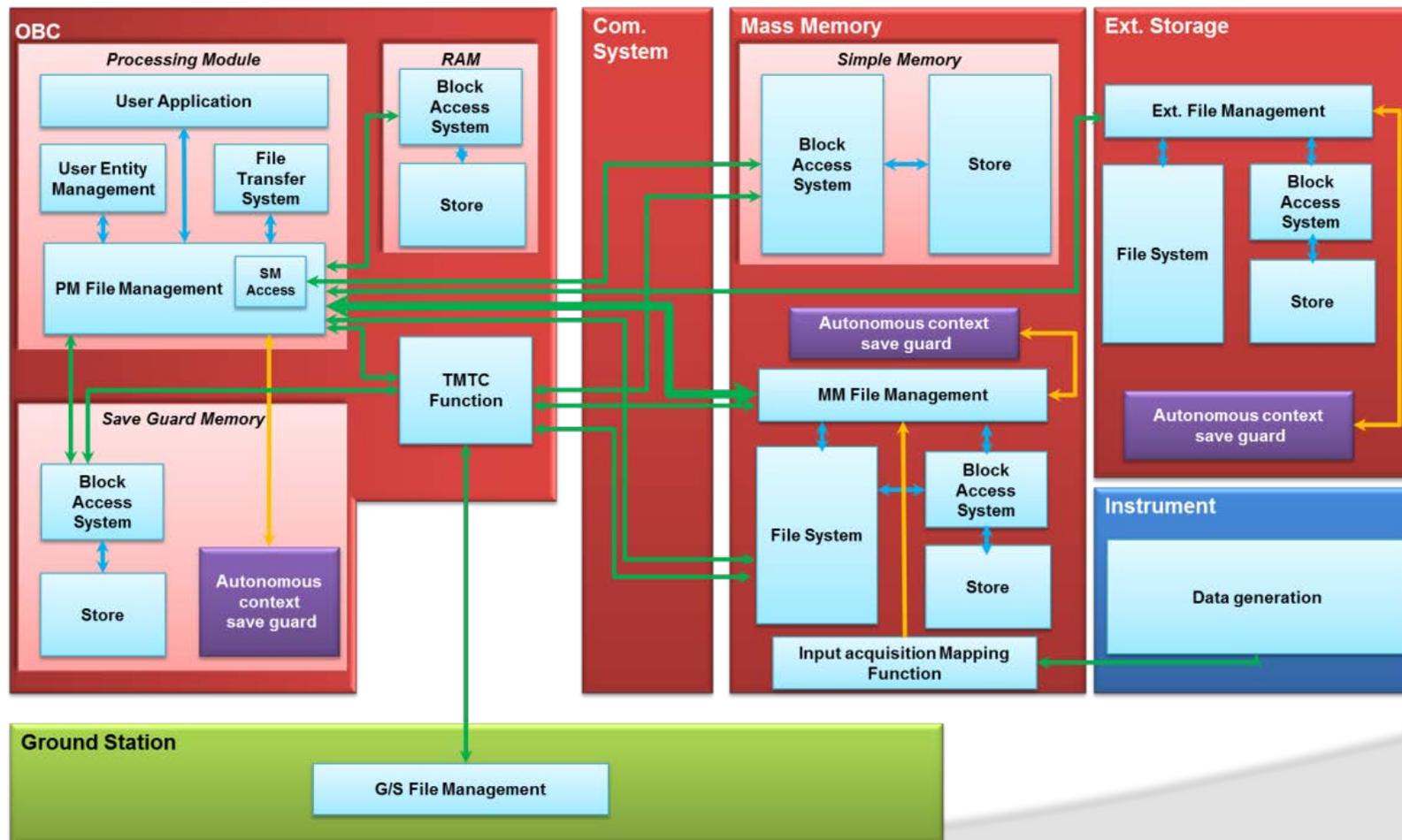
Shared link =>

Segregation:

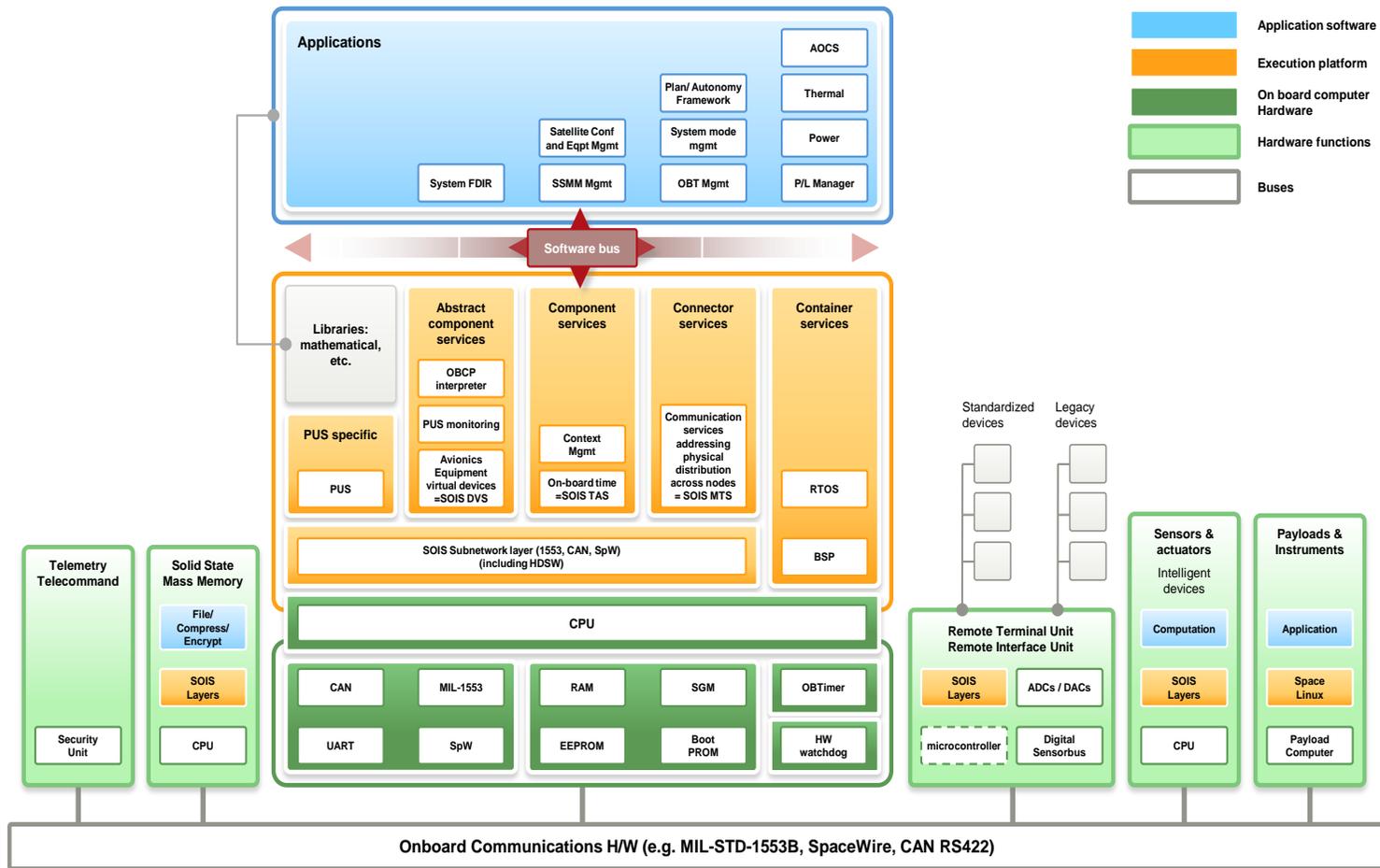
- HW: TC/MAPID  
TM/VCID
- SW: TC&TM/APIID

# Lessons learned and ASRA impacts

## Overall recommended architecture



# OSRA impacts for SW FMS/FTS



# Conclusions

- Generic user, technical, and protocol requirements produced for mass memories, integrating file awareness;
- Implementability of these requirements demonstrated thanks to a flexible demonstrator based on a software simulator;
- Impacts of introducing the file transfer and file management services into the ASRA & OSRA examined

# Steps forward

With the FMSIS study completed, several tasks are still at hand:

- Review by SAVOIR-MASAIS
- Implementation of comments and transfer to a SAVOIR document
- Review by SAVOIR Advisory Group
- Public review

Reaching into space  
**TOGETHER**

**QUESTIONS ?**

Thank you for your attention

