R&T File Management System
Mass Memory Workshop 2014

WE LOOK AFTER THE EARTH BEAT

17/02/2014

Ref.:

THALES ALENIA SPACE INTERNAL

ThalesAlenia Space
A Thales / Finmeccanica Company

83230352-DOC-TAS-EN-001

2011-2012 study: File Transfer System

On-Board/Ground File Transfer

cnes

ASTRIUM AN EADS COMPANY

ThalesAlenia A Thales / Finmeccanica Company Space

- Parallel TAS & ADS activities
- Analysis & comparison of 2 main protocols:
- PUS Service 13: Large Data Transfer (investigated further by ADS)
- CCSDS File Delivery Protocol (CFDP) (investigated further by TAS)

- PUS LDT service formal proof performed by ADS
- Implementation of CFDP class 2 by TAS (demonstrator)

- Analysis of the impact on the satellite avionics

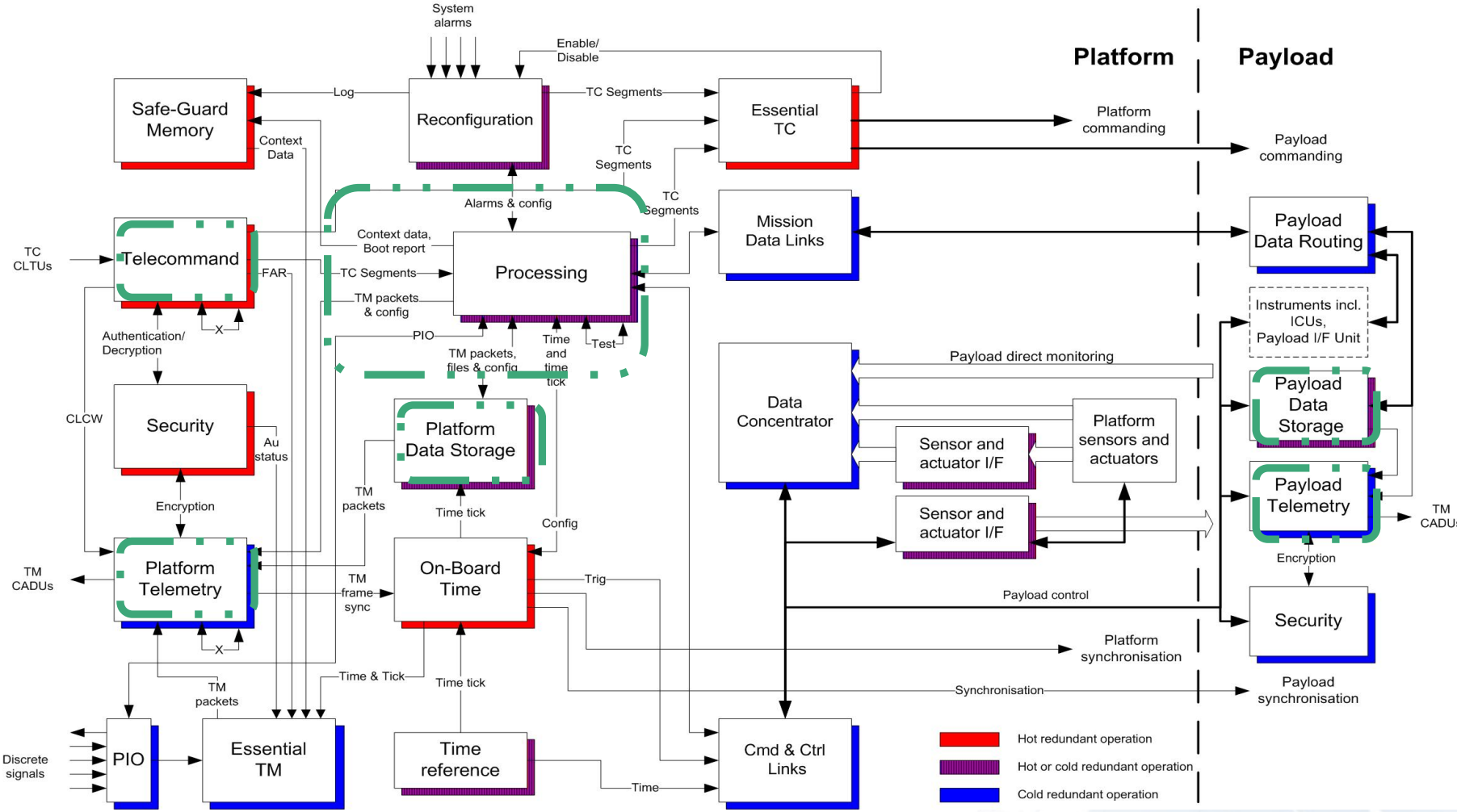
ThalesAlenia A Thales / Finmeccanica Company Space

2011-2012 study: File Transfer System

- Communication Performance: CFDP > PUS 13
  - The CFDP overhead is slightly lower than PUS 13 on files > 100kBytes
- Quality of Service: CFDP = LDT 13
  - Basically both services allows to enable packet repetition, CFDP proposes additional acknowledgment methods but the basic functionality is present in both services
- Easiness of Operations: CFDP = LDT 13
  - Both services allow to have an elegant transfer termination in case of error (for instance end of visibility window resulting in loss of communication)
  - The suspend & resume service is native in CFDP but was added in LDT 13 for ExoMars
- Capabilities: CFDP > LDT 13
  - LDT 13 does not currently propose a sub-service to make a relayed file transfer, this could be added in sub-services 128 to 255
- Complexity: LDT 13 > CFDP
  - CFDP relies on the exchange of multiple primitives with the file management system and the file storage system, LDT 13 is more acting as an intermediate communication layer basically segmenting/assembling large data structures (ex.: files)
- Compatibility with PUS: LDT 13 > CFDP
  - Today's EU satellite communication rely a lot on PUS, of course, PUS 13 LDT is designed for this context. Running CFDP on current Space/Ground links require adaptations

2011-2012 study: File Transfer System

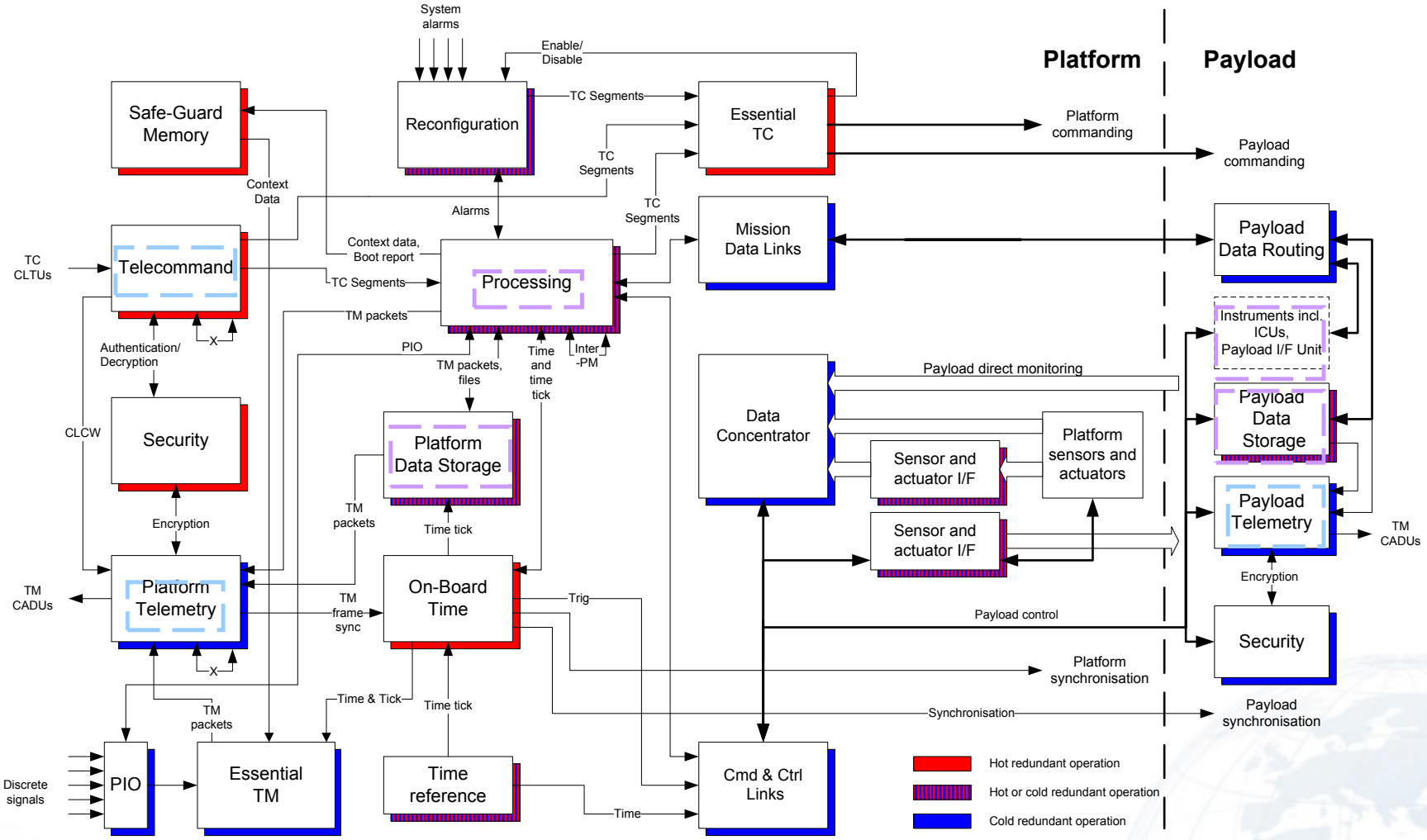
- CFDP relies on a strong coupling between file management system and the file storage device, it could be:
  - In OBC SW
  - In PF MM
  - In PL MM
  - In PL TM
- With implication of:
  - TM/TC for optimization



- *(Picture is taken from ASRA =avionics standard reference architecture)*

2011-2012 study: File Transfer System

- But the full File Management System + File Transfer System can be distributed in a larger scale:



On-Board File Management System:

- Generic FMS specification & prototyping compatible with:
  - PUS Service 13: Large Data Transfer
  - CCSDS File Delivery Protocol (CFDP)
  - PUS Service 25: File Management System
- SAVOIR/ASRA General recommendations for Spacecraft Monitoring & Control
  - Legacy missions
  - Multiple on-board architectures & services

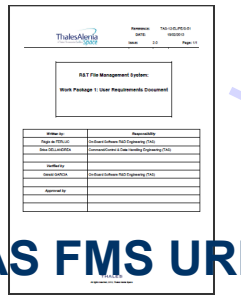
Presentation of the team

- CNES
  - Raymond Soumagne, Technical Officer
    - +33 5 61 27 47 53
    - raymond.soumagne@cnes.fr
- Thales Alenia Space Cannes
  - Regis de FERLUC, Project Manager & SW development – WP 2000 & 4100
    - +33 4 92 28 99 45
    - regis.deferluc@thalesaleniaspace.com
  - Gérald Garcia, Specification support – WP 1100, 3100 & 5000
    - +33 4 92 92 64 17
    - gerald.garcia@thalesaleniaspace.com
  - Brice Dellandrea, Specificiation & evaluation – WP 1100, 3100 et 4100
    - +33 4 92 92 67 04
    - brice.dellandrea@thalesaleniaspace.com
- Airbus Defence & Space Toulouse
  - Jacques Séronie-Vivien, Specificiation & evaluation – WP 1200, 3200 & 4200
    - +33 5 62 19 73 97
    - jacques.seronievivien@ADS.eads.net
  - Alain Corrégé, Specificiation & evaluation – WP 1200, 3200 & 4200
    - +33 5 62 19 56 03
    - alain.correge@ADS.eads.net

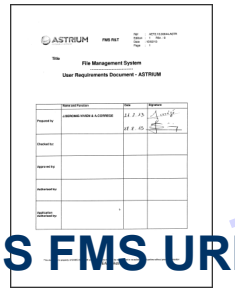
Study Objectives

- Specify a generic file management system for on-board operations compliant with existing reference or applicable documents:
  - ECSS-E-70-41A: PUS Service 13: Large Data Transfer
  - CCSDS File Delivery Protocol (CFDP)
  - ECSS-E-ST-70-41Cdraft: PUS Service 25: File Management System
  - SAVOIR/ASRA General recommendations for Spacecraft Monitoring & Control
  - Legacy & incoming missions or systems, on Thales Alenia Space side:
    - LEONARDO-3G
    - ExoMars
    - GökTürk → *Dynamic FMS*
- Develop a mock-up to evaluate the specification
- Provide a return of experience after testing phase and enrich the specification

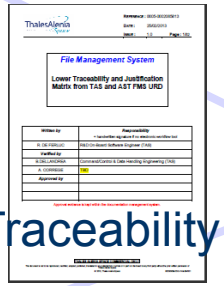
Study Deliveries



TAS FMS URD



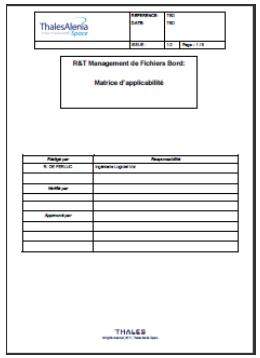
ADS FMS URD



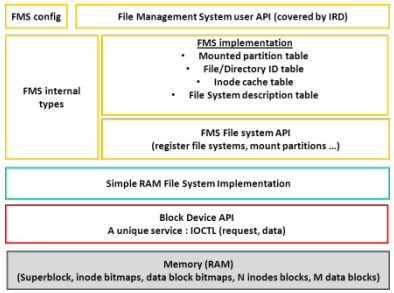
Traceability

FMS IRD

IRD coverage by TAS prototype



PROTOTYPE



TAS FMS Test User Needs

ADS FMS Test User Needs

FMS Test specification

TESTS

TESTS Report

Needs analysis: many potential memory types

- Targeted space applications : On board Memory / Data types and mapping

ADS point of view

Device	Typical Size	Non Volatile
EEPROM	4 Mbytes	Y
RAM	8 – 256 Mbytes	N
SGM EEPROM	128 – 256 Kbytes	Y
SGM RAM	512 – 1024 Kbytes	Y (partial)
MEMORY BANKS	> 256 Mbytes	depend on technology

2.5.2 On board data types

- Software images
- Telecommand files
- OBCP images
- Configuration data
- Context data
- Science data
- Paquet store ?

TAS point of view

Storage Area	Memory Type	PF/PL	Typical Size	Satellite	Data type	Need for FMS
BOOT	PROM or EEPROM with write protection	PF & smart PL	40 kBytes	All (LEO, MEO, GEO, HEO, Lagrange, Inter-planetary)	BOOT SW	Potentially
RAM	SRAM	PF & smart PL	2 x 4MBytes	All (LEO, MEO, GEO, HEO, Lagrange, Inter-planetary)	OBSW Variables	No
NVRM	EEPROM, Flash	PF & smart PL	2 x 2 MBytes	All (LEO, MEO, GEO, HEO, Lagrange, Inter-planetary)	OBCP MTL & OPS TC files Event TC SW patches	Yes
SGM	Combination of EEPROM and SRAM	PF only	2 x 512 (SRAM) and 2 x 256 kBbytes (EEPROM)	All (LEO, MEO, GEO, HEO, Lagrange, Inter-planetary)	Conf registers Event logs SW patches	Potentially
MM	DRAM, SDRAM, DDR, Flash or EEPROM	PF or PL	PF: Between 4 and 16Gbits PL: up to 3000Gbits	LEO, MEO, HEO, Lagrange, Inter-planetary	HKTM Science TM	Yes
					Packet Stores	Potentially(*)

(*): Packet Stores are a priori out of scope of FMS as owning very specific services

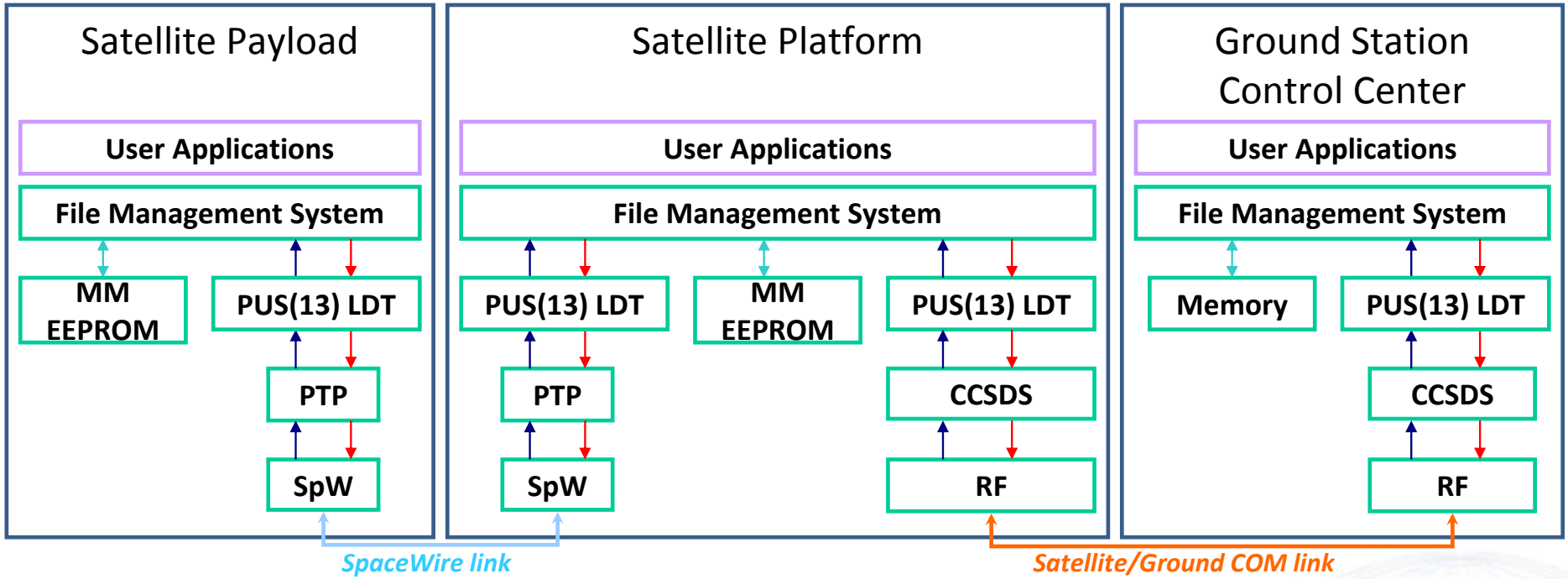
Needs analysis: many potential data types

- Data types :
  - Telecommands
    - Mission Time-Line (MTL) and Orbit Position Scheduling (OPS) → OK for files
    - On-Board Control Procedures (OBCP) → OK for files ?
      → Use of files would be very interesting for storing these more complex TC sets.
    - OBSW and BOOT→ OK for files, even if tricky for BOOT (basic FMS in first BOOT)
    - Event triggered TC → OK for files?
      - PUS service 19 enables ground operators to add to an on-board event-action list some telecommands that are triggered by these on-board events. Telecommands and Events used by the event-action service shall be entirely instantiated in the system database.
  - Telemetries
    - Platform TM (Diagnostics/Dump/Events logs) → OK for files ?
      - Usualy stored in circular or bounded packet stores, which is similar to a file storage. This kind of non critical data downloaded on Ground request or in failure cases is well adapted to a storage in a file system.
    - Payload/Science TM, divided in 3 main categories:
      - Payload/Science TM formatted into CCSDS/PUS packets by instruments → PS better than files
      - Payload/Science TM generated in continuous flow of measurements by instruments: in that frame, the board is responsible for splitting data in usable data units, which could be files → OK for files
      - Payload/Science TM generated as homogenous bursts of data: The instruments produce data blocks which have operational integrity (for instance observation sequence, image, …). The data length has no direct relation with a packet length and can be downloaded only on completion of the data block. → OK for files

    - ***Special case: Packet Stores (concatenated structure made of already formatted packets) → It has been decided to use FMS functionalities to manage PS also***

Needs analysis: many potential avionics types

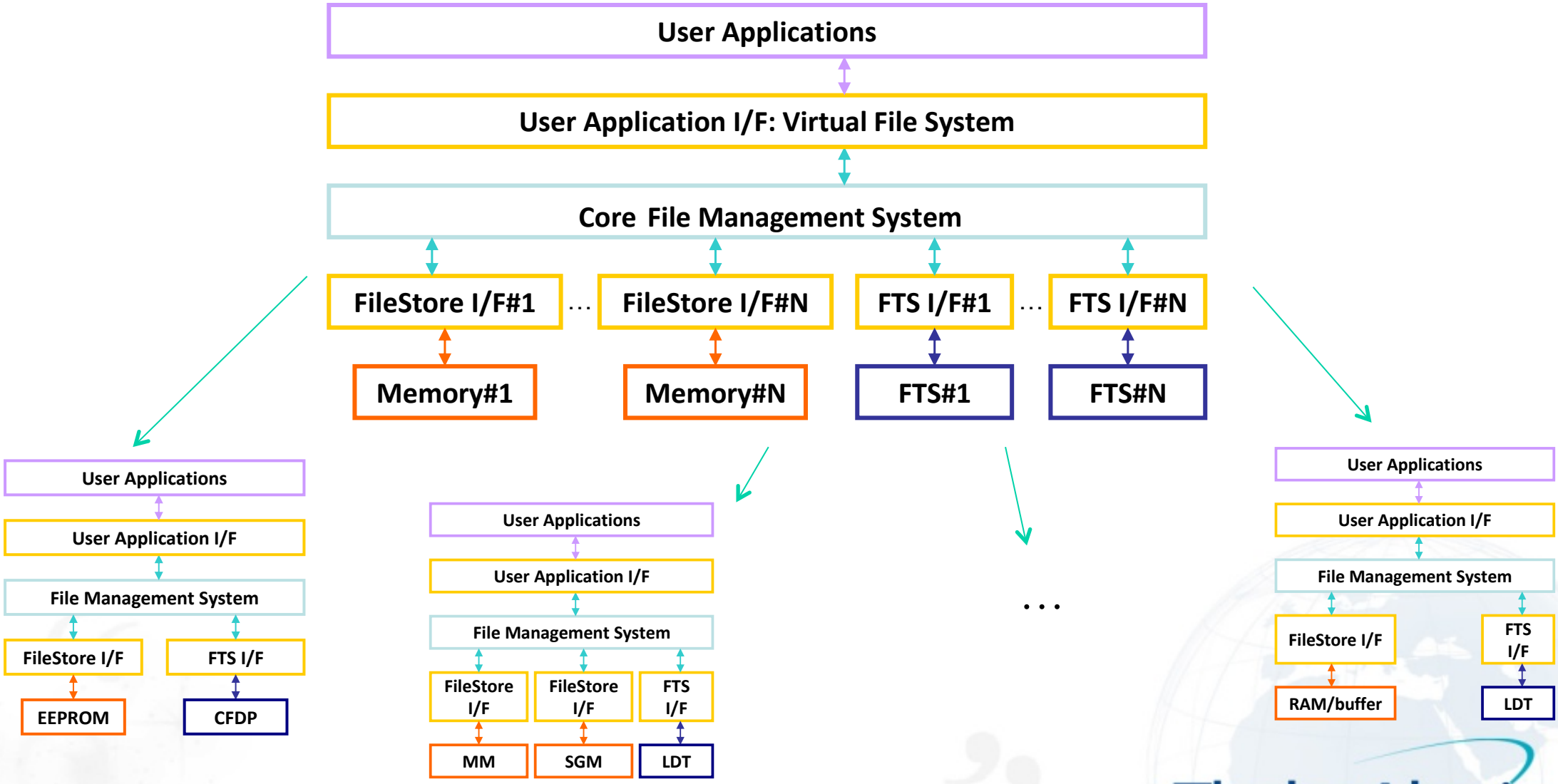
- Fictive example: File transfer from Payload MM to Platform MM using SpW & PUS 13 LDT then download from Platform MM to GS using PUS 13 LDT or CFDP:



- Many potential applications:
  - FMS in OBSW managing a remote MM with basic primitives or PUS services
  - Autonomous MM with embedded FMS & FTS
  - FMS in OBSW managing a local MM or SGM or RAM etc…

Needs analysis: conclusion

- Required rationalization to define a FMS core and distinguish common & specific functions:

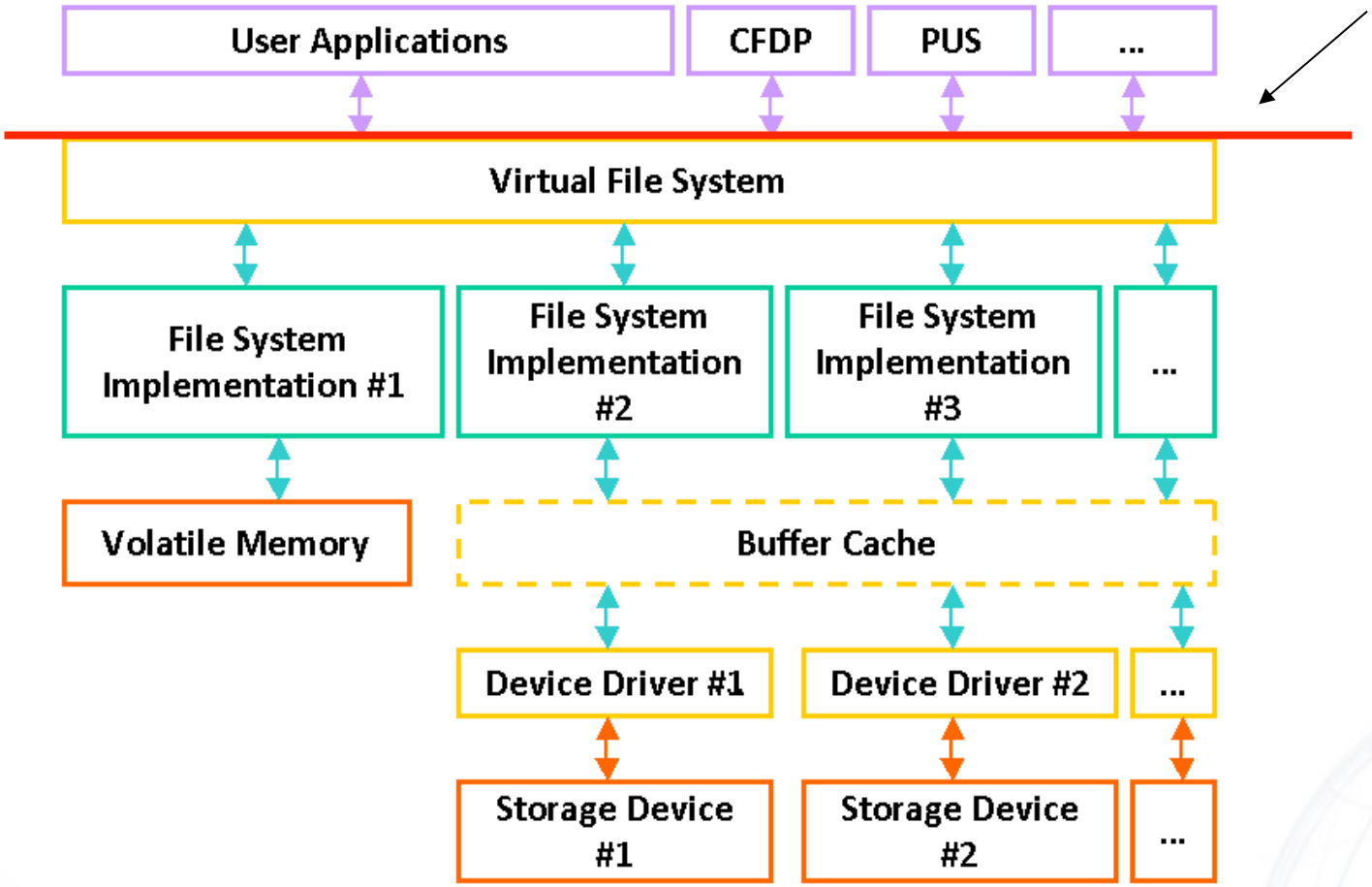


Document d'interface : FMS Interface Requirement Document (IRD)

- An IRD has been defined from TAS & ADS URD with standard services:
  - File System Services (detailed): functions & formalism based on POSIX&ARINC
    - File creation : CREATE_FILE
    - File Open : OPEN_FILE
    - File deletion: DELETE_FILE
    - File read: READ_FILE
    - File write: WRITE_FILE
    - File synchronization with storage media: SYNC_FILE
    - File seek position : SEEK_FILE
    - File and directory rename : RENAME
    - File and directory copy : COPY
    - File get attributes : GET_FILE _ATTRIBUTES
    - File set attribute : SET_FILE _ATTRIBUTES
    - Partition get attributes : GET_PARTITION _ATTRIBUTES
    - Directory creation: CREATE_DIR
    - Directory deletion: DELETE_ DIR
    - Directory open: OPEN_DIR
    - Directory close: CLOSE_DIR
    - Directory read: READ_DIR
    - Directory rewind: REWIND_DIR
    - Directory get attributes: GET_DIRECTORY_ATTRIBUTES
    - Directory set attributes: SET_DIRECTORY_ATTRIBUTE

FMS Interface Requirement Document (IRD)

Interface described By the IRD



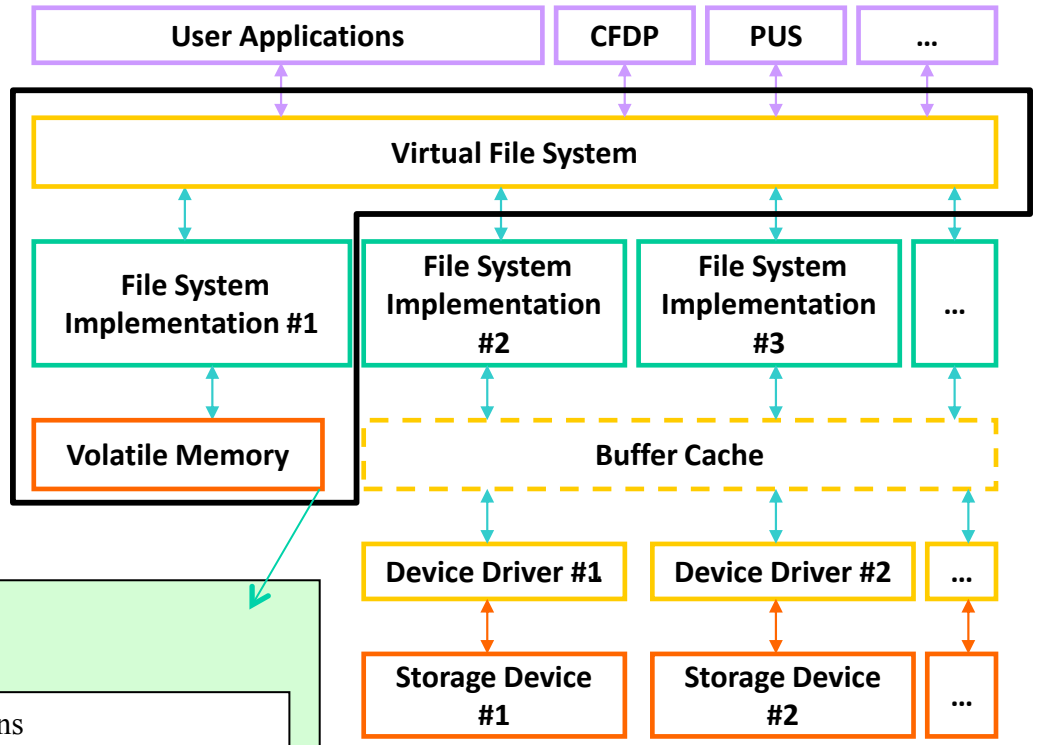
FMS Interface Requirement Document (IRD)

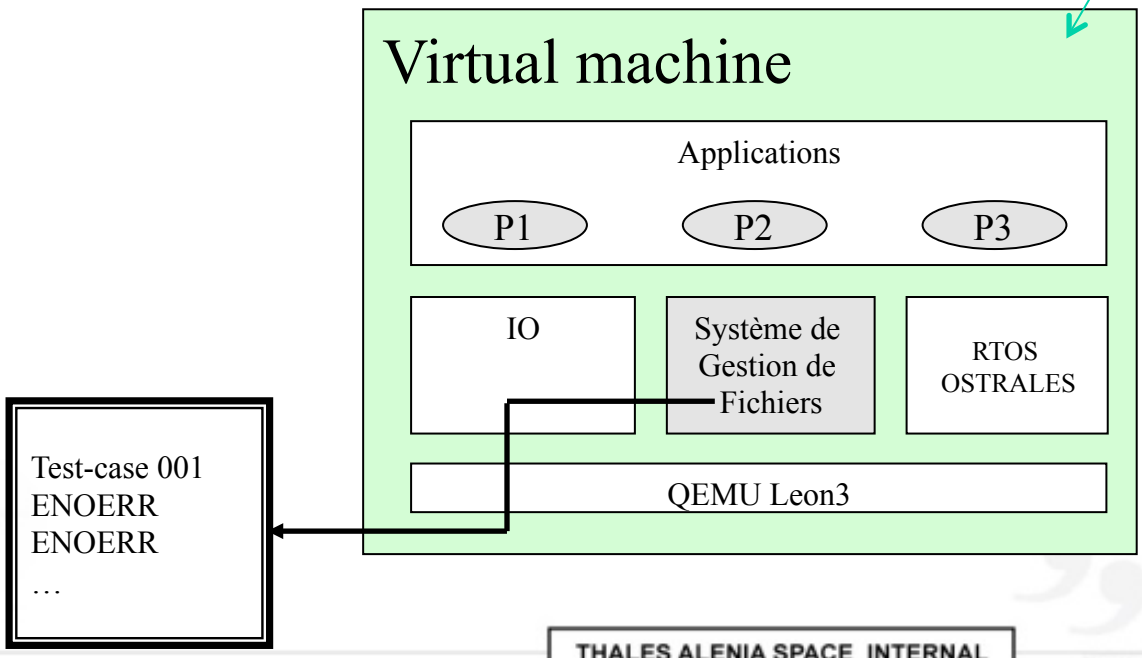
- Several issues treated (when generic) or left opened (when specific) in the IRD:
  - Access rights (per application IDs) → solution proposed
  - Concurrent access management (R/W & multiple R) → solution proposed
  - Concurrent operations handling (delete while write-opened, etc…) → solution proposed
  - Bounded & unbounded files management at creation & update → solution proposed
  - Files & Directory attributes (generic & extended attributes lists) → solution proposed
  - FMS configurability at compilation time → solution proposed
    - Selection of sub-services
    - Tuning of parameters (authorized parallel opened files, path depth)
  - Cache management for remote MM management → opened per implementations
  - Robustness/reliability: Redundant files? Mirror partitions? Context recovery? → opened per implementations
  - File search in a partition → not implemented in the FMS core
  - Operation abort → not implemented in the FMS core (atomic calls)
  - Symbolic link service: useful to hide the real path to the file to facilitate the application reuse → not implemented in the FMS core (atomic calls)
- All issues, trade-offs and opened points are listed in annex of the IRD

FMS demonstrator description

- Language: ADA 2005
- Inode mgt of files & directories
- Compilation GNAT on Leon 3 target
- Metrics of the core FMS modules:

Total	SLOC	CLOC	Blank	Subroutines
6992	4340	1230	1422	275

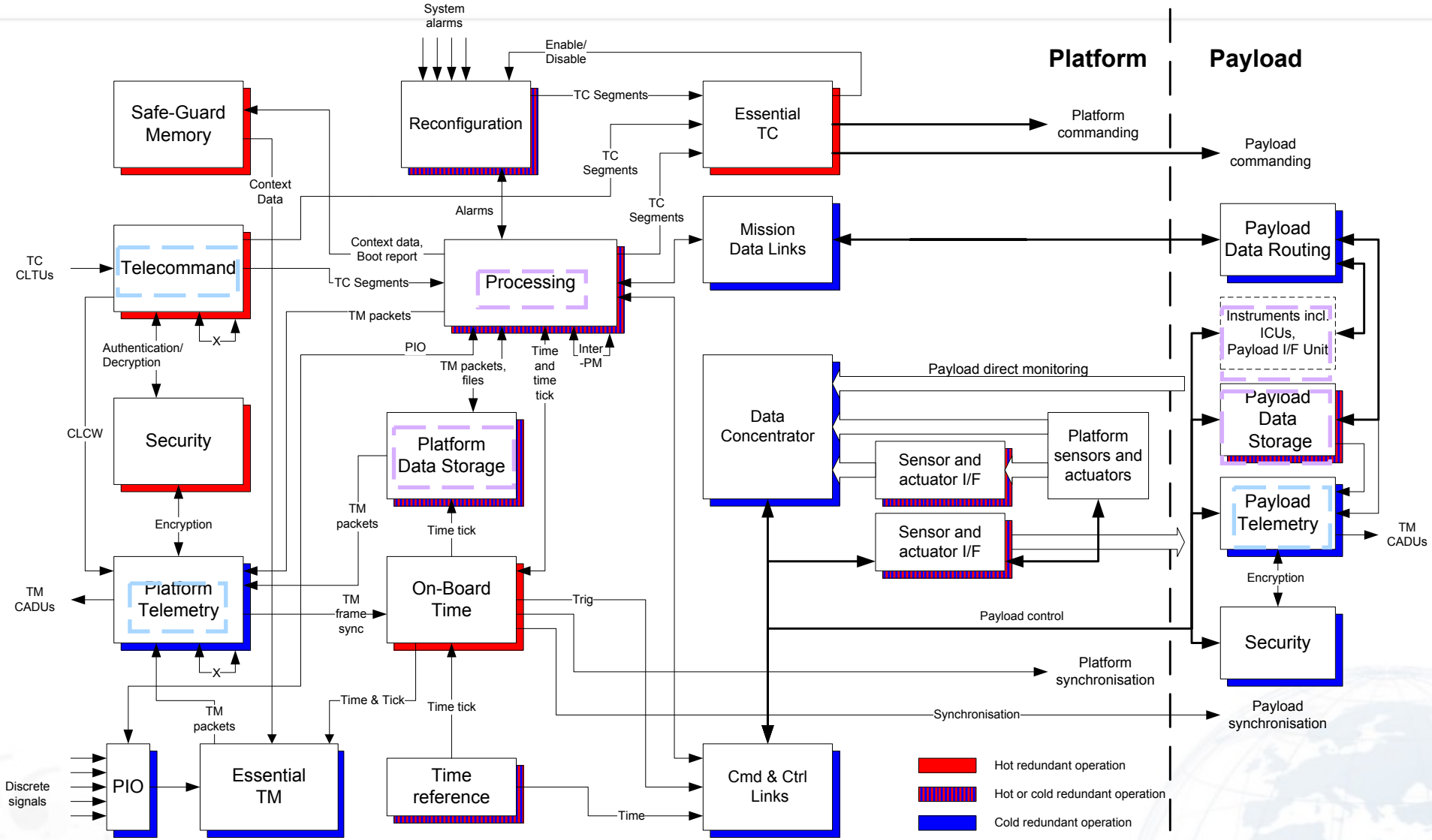




Conclusion

- CNES study performed by Thales Alenia Space with participation of Airbus Defence & Space resulting in a common IRD for a File Management System from *non-conflicting URDs*
- Virtualization of FMS services resulting in a generic Virtual FMS extendable to many missions (vs minimal ad hoc current systems)
- Mission-specific features (ex.: specific memory drivers) are tackled to lower layers to keep the core's genericity
- List of points a user has to assess when specifying a FMS & identification of potential issues

!!! Thanks for your attention !!!



Functions potentially impacted by a FMS and/or FTS

Functions potentially impacted by a FTS

THALES ALENIA SPACE INTERNAL

ThalesAlenia Space
A Thales / Finmeccanica Company