

## **ADCSS 2012 conclusions and SAVOIR mass memory requirements**

G.Furano, J. Iltad, G. Magistrati, C. Taylor  
ESA-ESTEC TEC-ED



# Agenda



- ADCSS 2012 - day 3 ( "Mass Memories for P/L applications and file based operations" ): conclusions
- SAVOIR
- CFDP and Mass Memory requirements in SAVOIR documents



# ADCSS2012 Day 3 - Programme



<http://congrexprojects.com/2012-events/12c25/introduction>

TEC-EDD/EDP/ED | SMMD 2014 | ESTEC | 14/02/2014 | Pag. 3

ESA UNCLASSIFIED – For Official Use



# ADCSS2012 Day 3 – Summary (1 of 2)



- **The three sessions identified future requirements and trends in relation to:**
  - Mission requirements
  - Mission operations and technology concerning on-board mass memory,
  - Reliable data transfer requirements when using files
- **From an ops viewpoint:**
  - A strong belief in the use of files was expressed on the premise that it will simplify and stream-line operations
  - The use of CFDP on the downlink was fully supported but added to this was the use of CFDP on the uplink
- **From the project viewpoint:**
  - Use of Ka-band and optical links to cope with increasing payload data rates will require more frequent retransmission
  - File based operation facilitates fast science data availability
  - Involve all parties, including ESRIN, to optimise the end-to-end system
- **Use of files and a standard file delivery protocol:**
  - Promoted as a way forward particularly for Euclid where CFDP is now baseline for the ITT



# ADCSS2012 Day 3 – Summary (2 of 2)



- **Some concerns where raised:**

- The need for files versus retrieving sets of packets from packet stores
- Overall consensus that both have their place to place in future systems

- **On-going developments:**

- Several on-going developments that support the use of files and file transfer.
- No particular concern was expressed on the difficulties to implement files or CFDP

- **Implementation considerations:**

- Several options in which CFDP is implemented in the flight avionics
- Trade-off is required on how SSMM access is performed
- Updates to Savoir documentation to reflect this will be required

- **Way forward:**

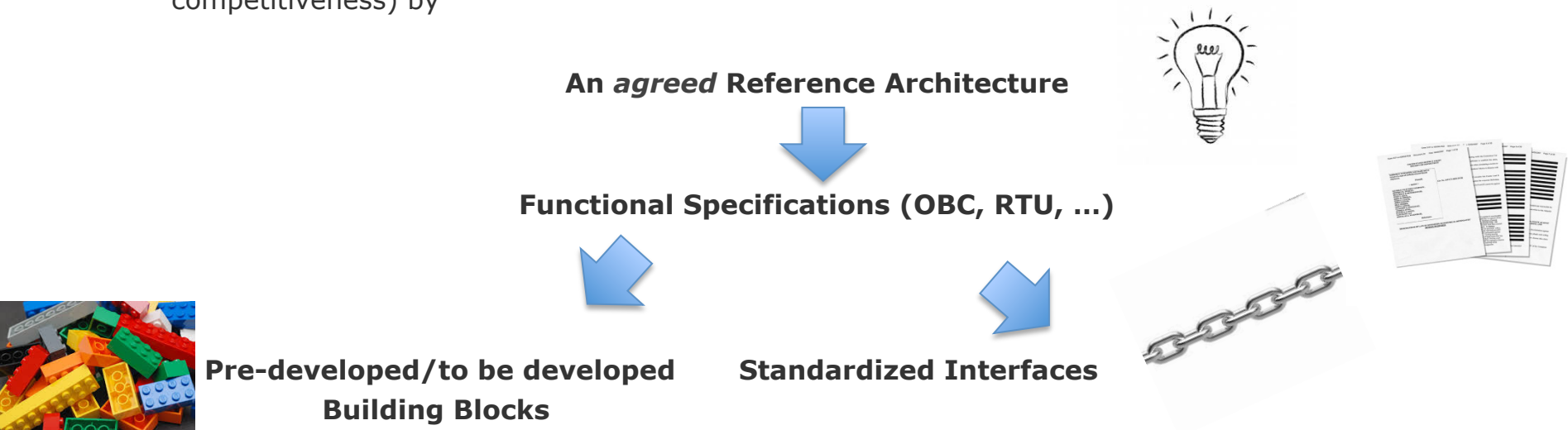
- Emphasis must be put on how contents of files are used onboard
- Effort to be put on defining the way in which the content of files are used onboard and this will eventually require updates to PUS

- The presentation from NASA on LRO and the successful use of files/CFDP added to the confidence that file based operations is a solid way forward



# SAVOIR mission

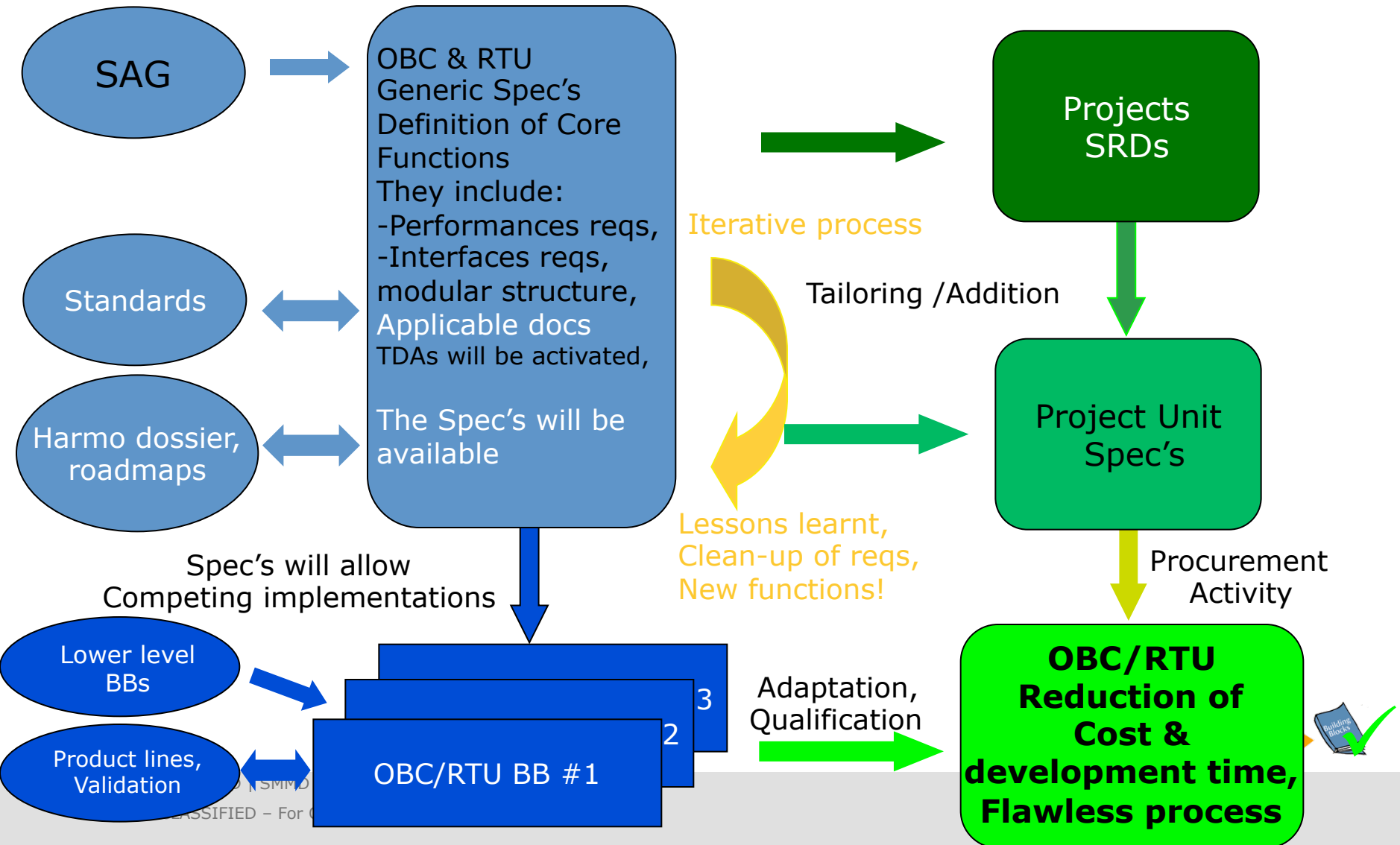
- SAVOIR means **S**pace **AV**ionics **O**pen **I**nterface **aR**chitecture.
- It is an initiative to federate the space avionics community and to work together in order to improve the way that the European Space community builds avionics subsystems.
- Improve the way we deliver Space Systems (risk & schedule, and therefore cost, and industry competitiveness) by



- If, in ITTs, Customers use *agreed* mission specification and System Integrators use *agreed* product specification, then Suppliers should be in a position to have product lines, and System Integrators should have easier integration phases



# Generic Specification for OBCs and RTUs



# OBC Generic Specification



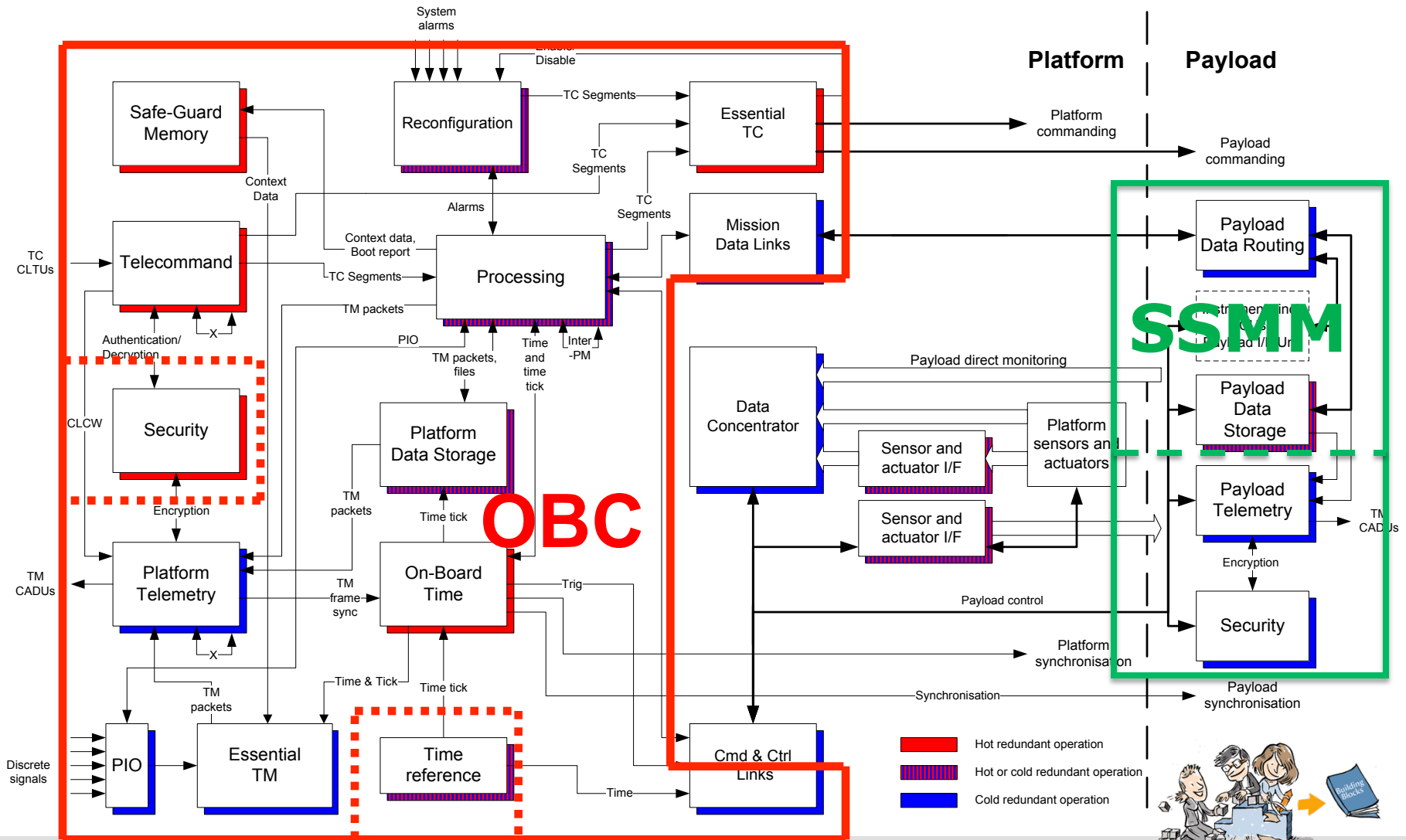
- OBC Spec TOC:
  - **Functional Requirements**
    - Telecommand reception, authentication, decoding and handling.
    - Telemetry Transfer Frame generation, coding and modulation.
    - Processing capability
    - Timing and Synchronisation management.
    - Platform Data Storage
    - Fault Detection Isolation and Recovery (FDIR): Reconfiguration Module (RM), Essential TC function , and a Safeguard Memory (SGM)
    - Security Function (optional)
    - Essential TM (optional)
  - **Interface Requirements**
  - Section with a list of suggested **CDMS/CDMU** ( and a list of non desiderata !!!) **requirements for System Requirement Docs**

Important: Modular Structure [parameters ranges , options are present !]





# SAVOIR Reference Functional Architecture

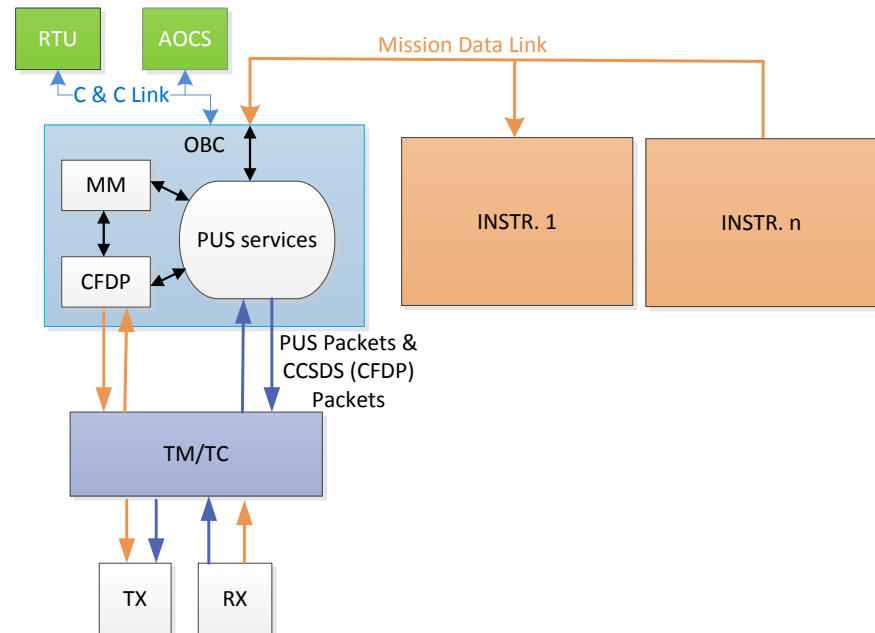


# Architecture #1

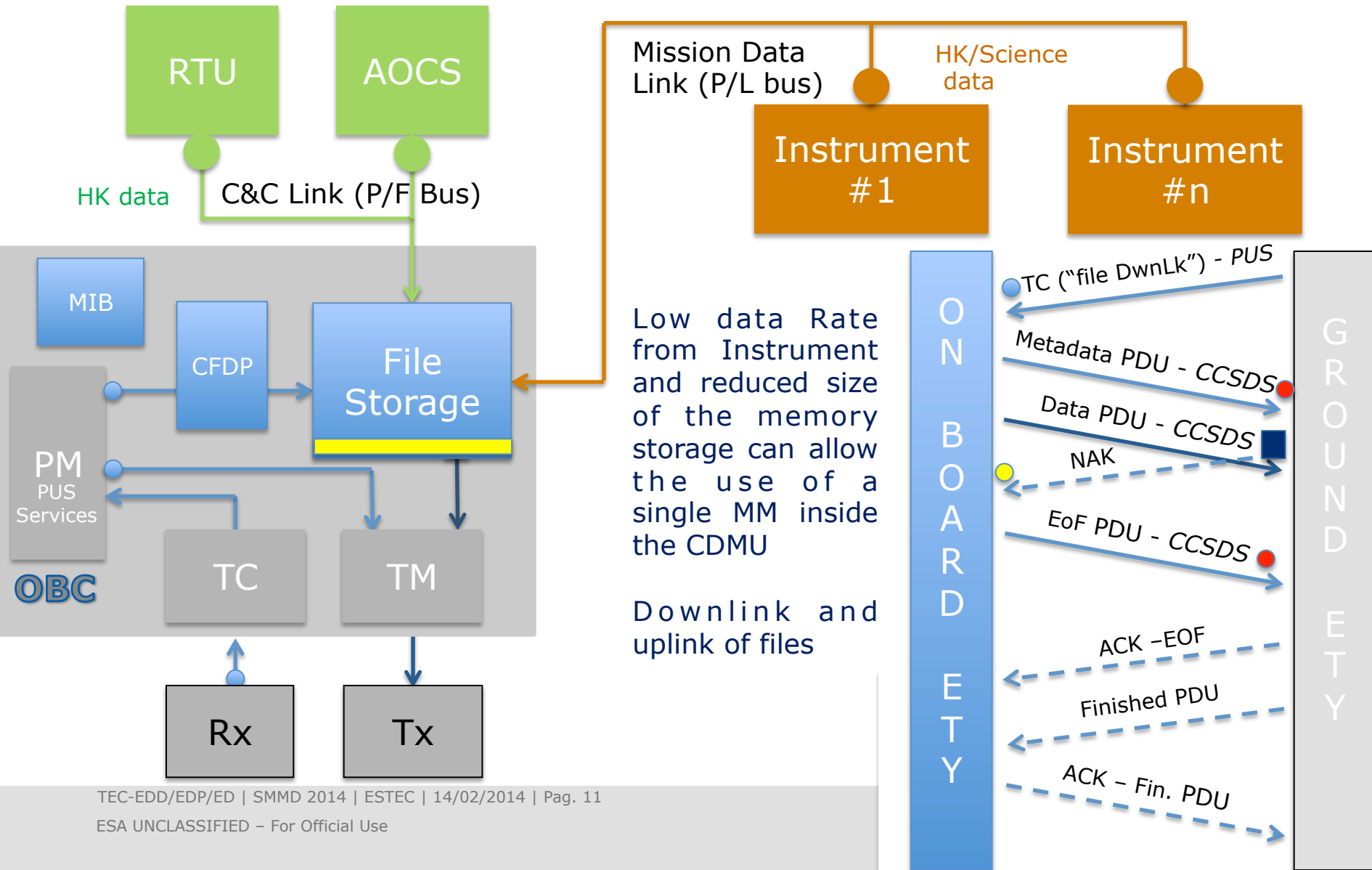


Different options for CFDP implementation on board are possible

- Low data Rate from Instruments and reduced size of the memory storage can allow the use of a MM inside the CDMU
- Downlink and uplink of files



# Architecture #1 File Downlink

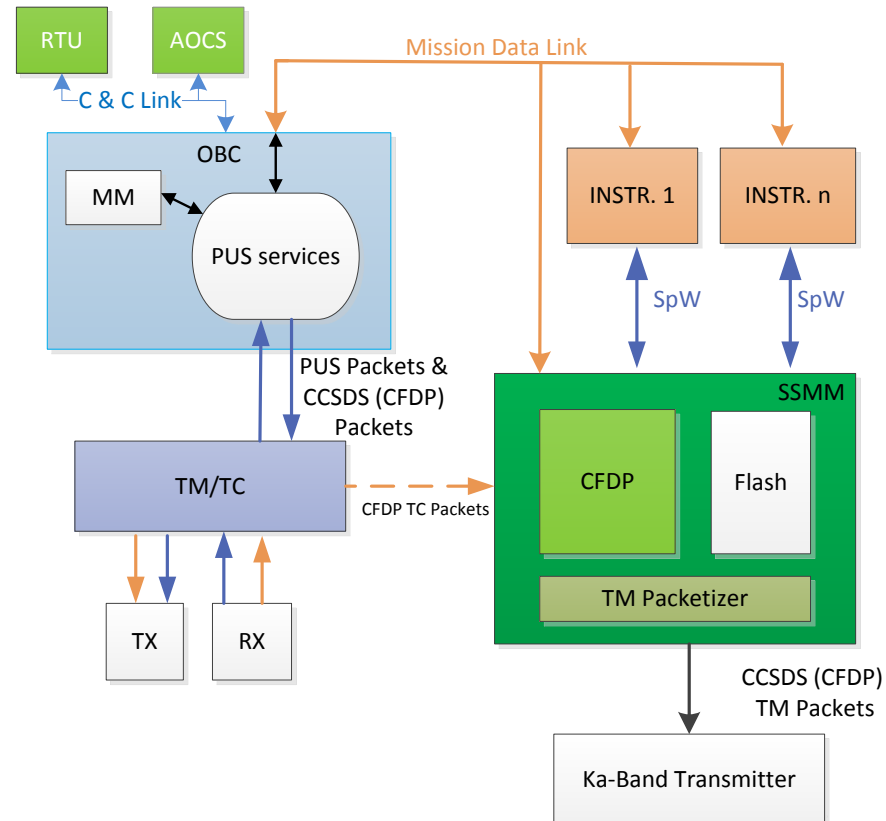


# Architecture #2



Different options for CFDP implementation on board are possible

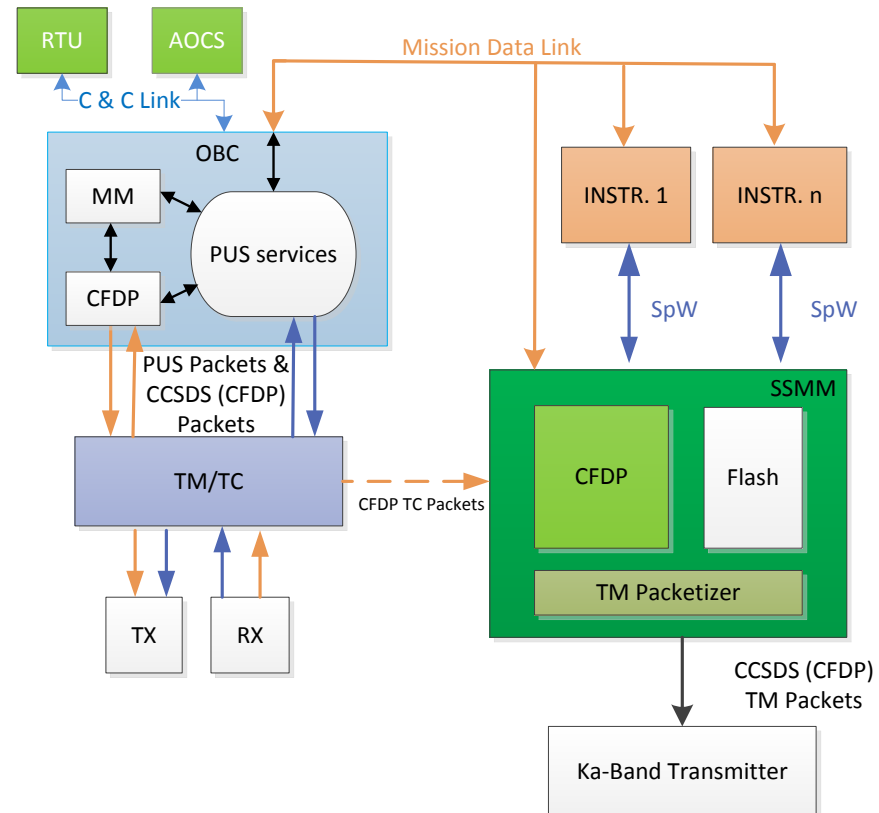
- CFDP entity located in the SSMM as independent unit from CDMU (The SSMM stores P/L data, but also SW images, OBCP, S/C telemetry...)
- Downlink and uplink of files
- Direct link from the SSMM to the Transponder (Ka band)
- CFDP TC (CCSDS packet) for CFDP transactions to SSMM could be routed directly to SSMM or passed via the OBC.
- Optional Mass Memory (no CFDP support) inside the OBC



# Architecture #3



- Two CFDP entities located in the CDMU (SW images, OBCP, S/C telemetry) and in the SSMM (P/L data)
- Downlink and uplink of files
- Direct link from the SSMM to the Transponder (Ka band)
- CFDP TC (CCSDS packet) for CFDP transactions to/from SSMM could be routed directly to SSMM (dedicated MAP-ID).

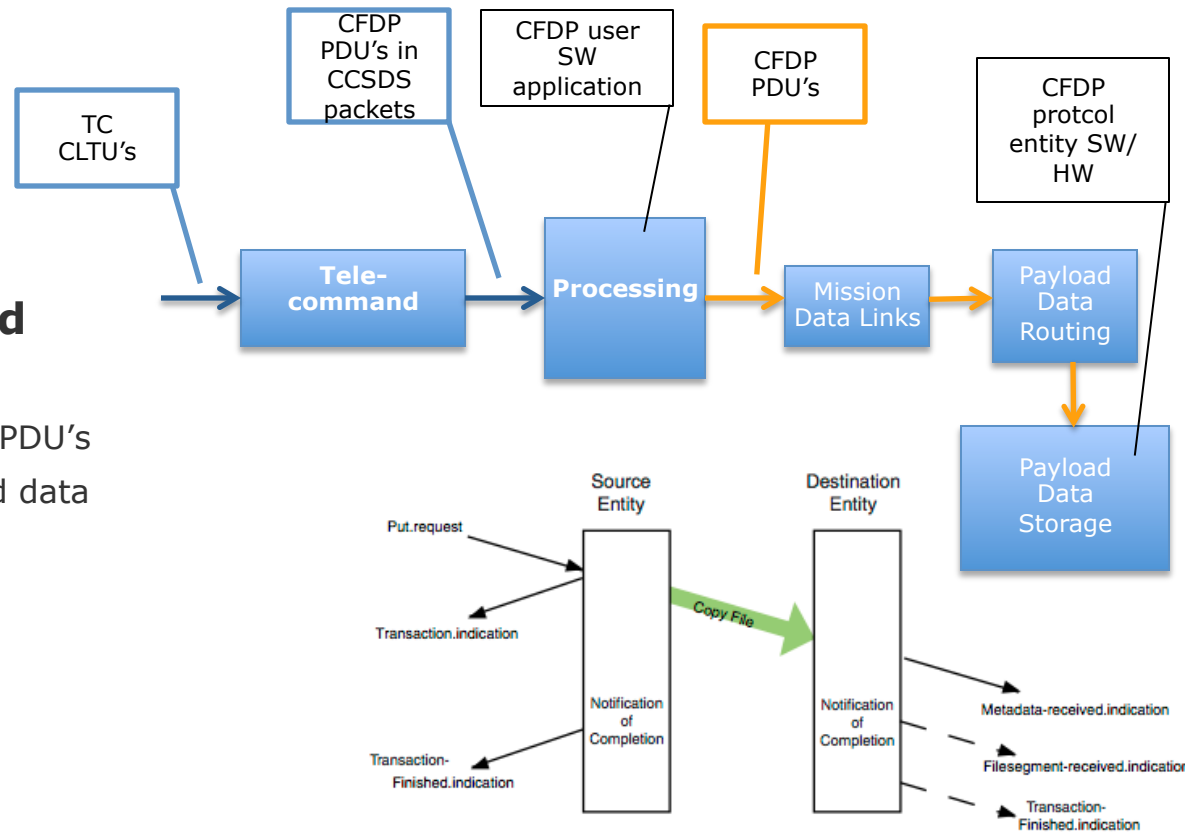


# Example - SAVOIR architecture – Files Uplink CFDP PDUs to SSMM sent through the OBC

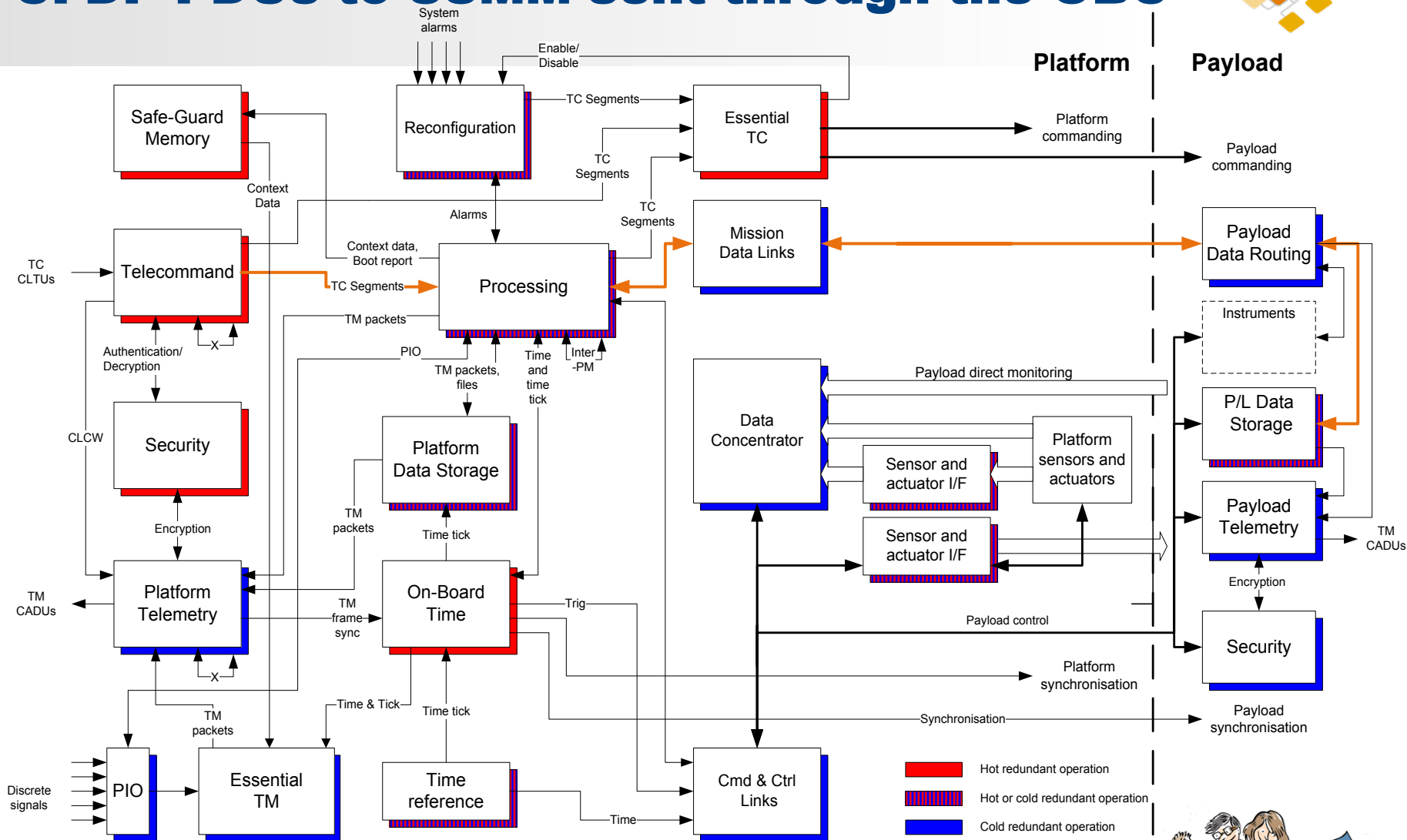


## Simplified SAVOIR functional architecture

- **CFDP user in OBC processing function**
  - Extracts CFDP PDU's
- **CFDP entity in payload data storage function**
  - Re-assembles CFDP PDU's
  - Stores file in payload data storage



# Example - SAVOIR architecture- Files Uplink CFDP PDUs to SSMM sent through the OBC

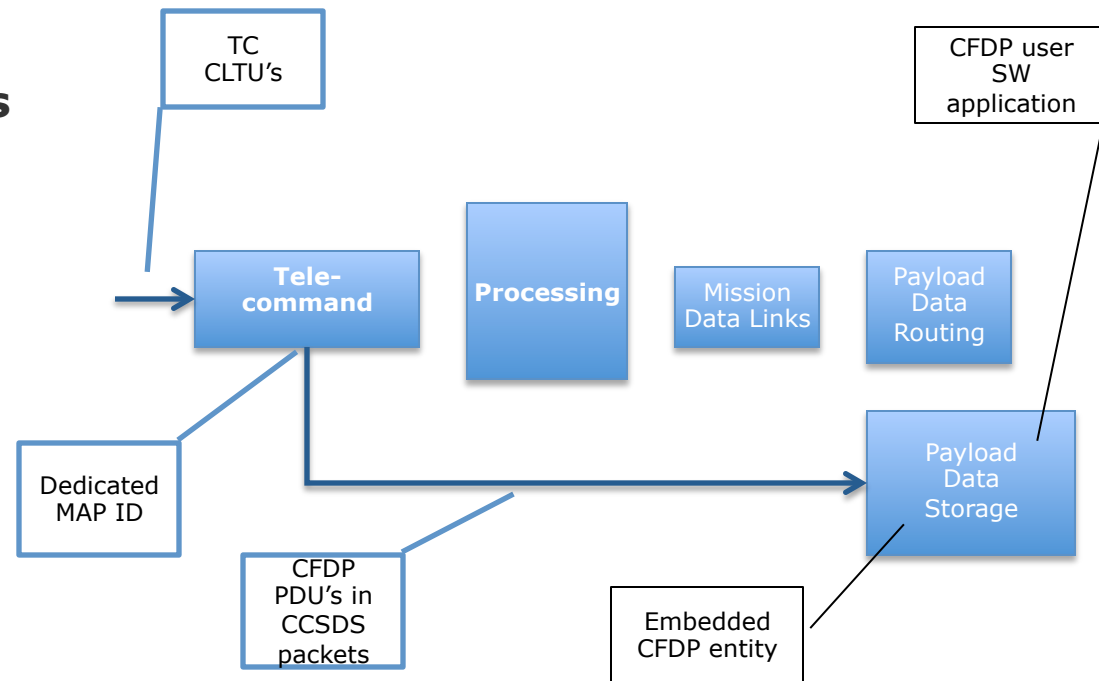


# Example - SAVOIR architecture- Files Uplink CFDP PDU sent directly to SSMM



- CFDP entity and user in payload data storage function
- Dedicated MAP-ID for the CFDP related PDU's

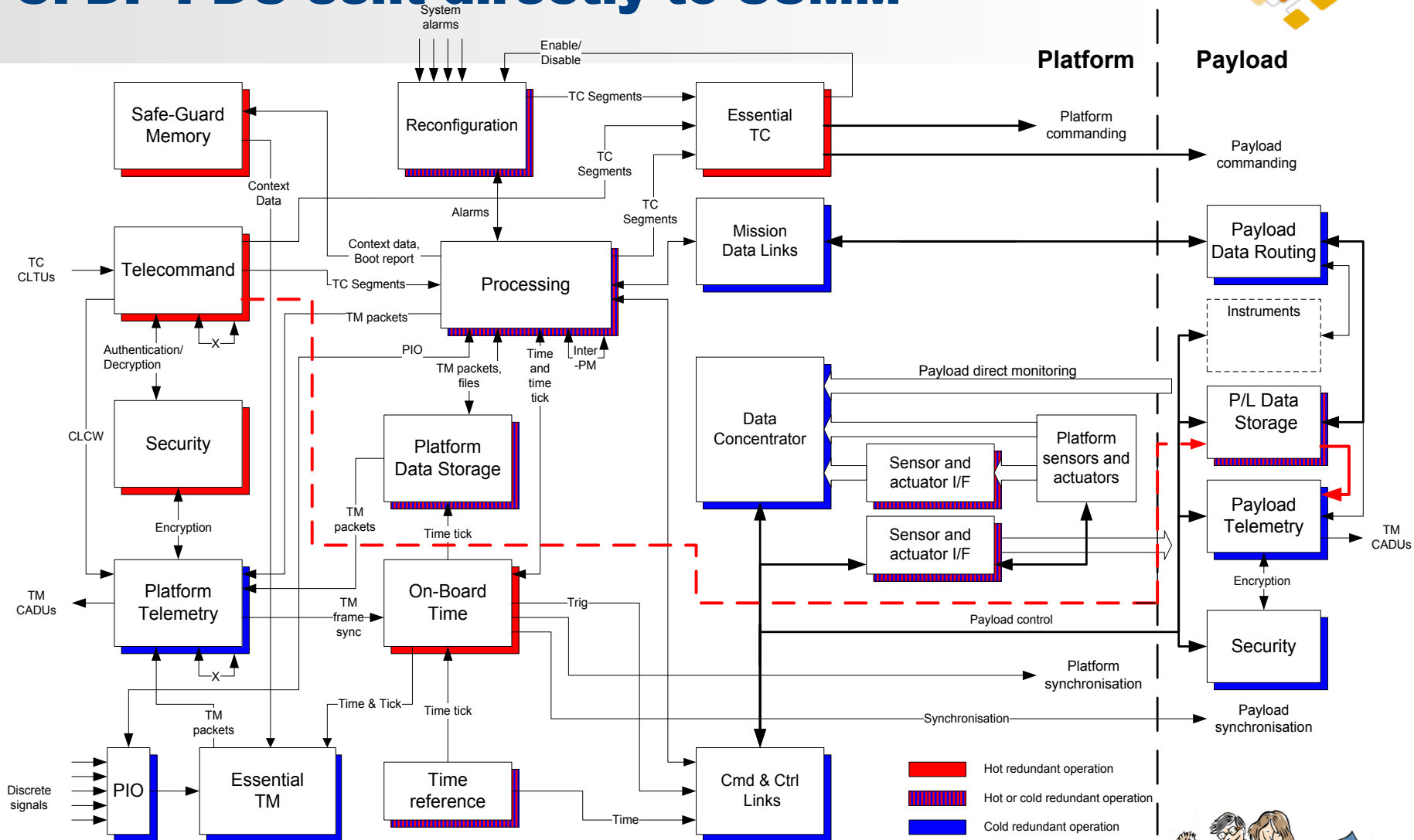
Simplified SAVOIR functional architecture





# Example - SAVOIR architecture- Files Uplink

## CFDP PDU sent directly to SSMM



# SAVOIR – ARCHITECTURAL ELEMENTS of the FILE DELIVERY PROTOCOL



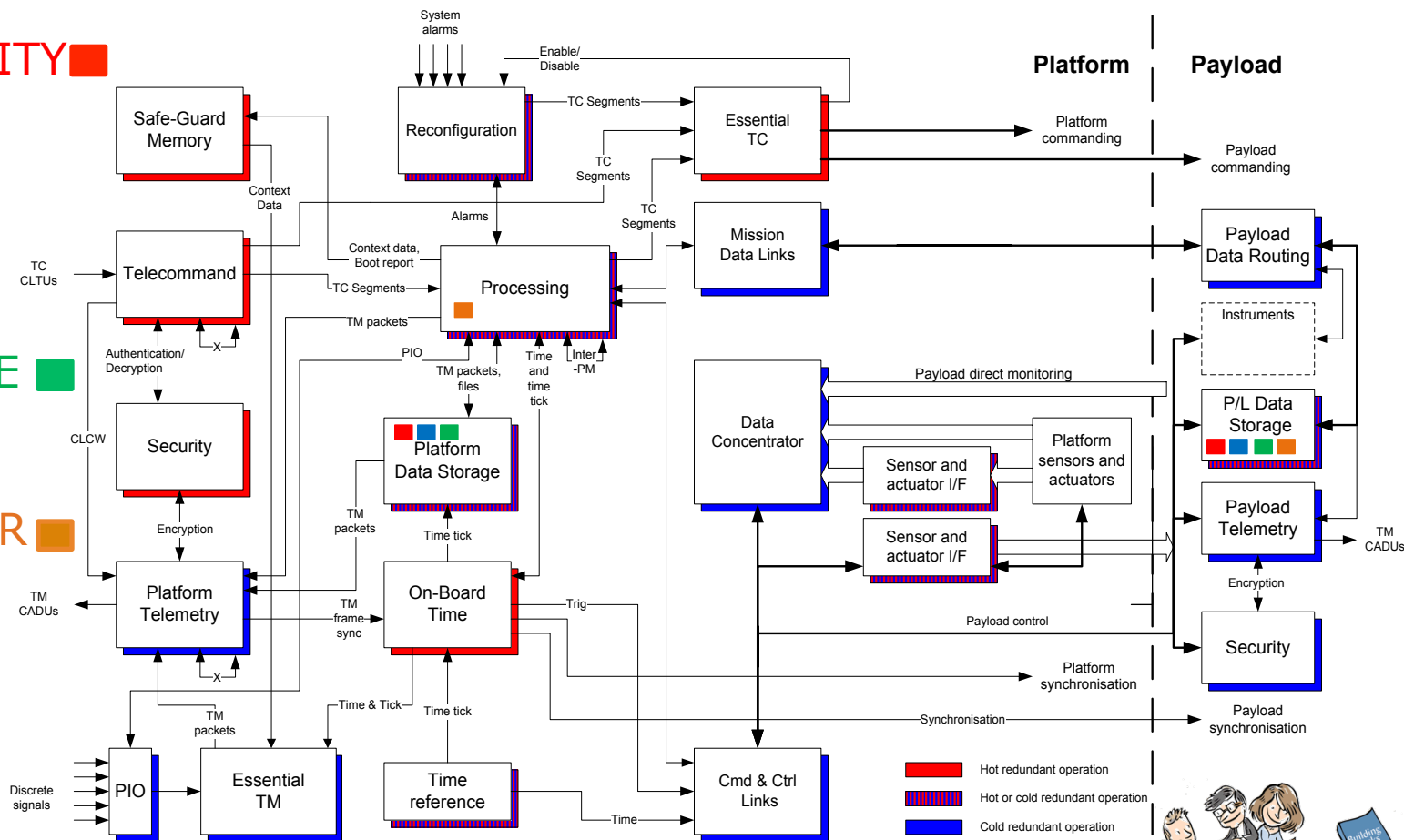
CFDP ENTITY

MIB

FILESTORE

CFDP USER

COMM' SYSTEM



# SAVOIR – proposed ADs and RDs for CFDP and file system in the SAVOIR OBC Spec



- *Applicable and Reference Docs proposed by ESA*
- ADx                      CCSDS 727.0-B-4 Standard CCSDS File Delivery Protocol
- ADx+1                  CCSDS 873.0-M-1 SOIS File and Packet Store Services
  
- RDy                      *File Based Operation Concept ESB-FbO-RP-1001 25/7/2012*
- Rdy+1                      *File Based Operation Study – Space Segment Trade-off Analysis EGOS-GEN-FIBOps-TM-1001 30/04/2013*



# SAVOIR – Mass Memory (1 of 3)



- *CCSDS 727.0-B-4 requires a filestore and a file system however: “ It is recognized, however, that the precise nature and capabilities of filestore management systems are operating-system dependent and, for that reason, the protocol assumes a virtual filestore and associated services that an implementation must map to the capabilities of the actual filestore used.”*
- CCSDS 727.0-B-4 defines a minimum set of functions/services that are required from the file system:
  - create file;
  - delete file;
  - rename file;
  - append file;
  - replace file;
  - create directory;
  - remove directory;
  - list directory.



# SAVOIR – Mass Memory (2 of 3)



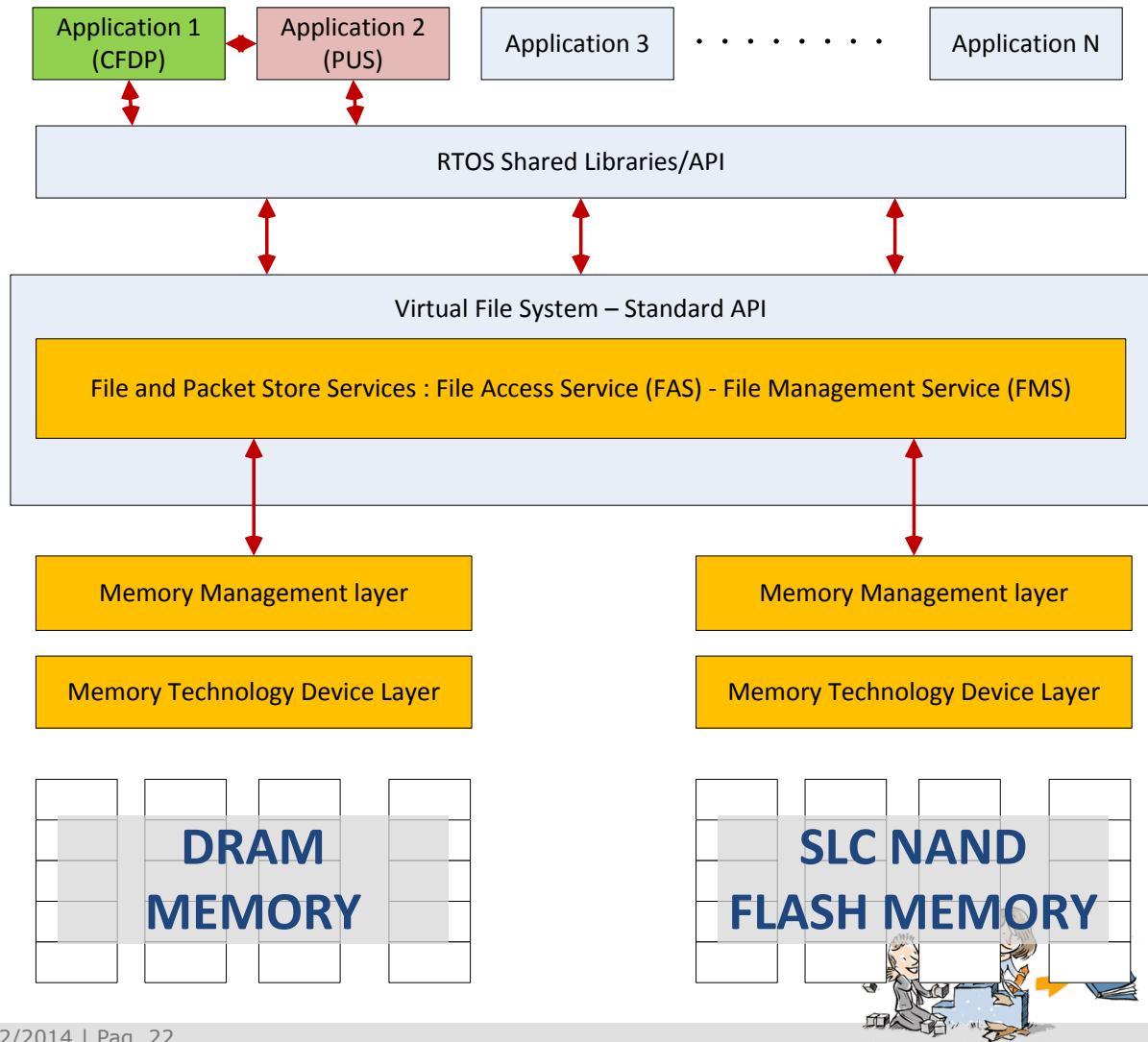
- CCSDS 873.0-M-1 SOIS File and Packet Store Service *defines the Spacecraft Onboard Interface Services (SOIS) File and Packet Store Services (FPSS).*
- The SOIS File and Packet Store Services are for use by onboard software to:
  - Access, and manage files residing in a file store.
  - Access and manage packets residing in a packet store.
    - Both files and packets can be present in the same filestore !
- The SOIS File and Packet Store Services provide a standard interface to allow onboard software to request:
  - access to files resident in a file store;
  - modifications to files in a file store;
  - management of files within a file store (e.g., create, delete, rename);
  - storing, retrieving, and deleting packets in a packet store;
  - management of packet stores.
- Different types of file systems (and file stores) may be considered:
  - Flat file systems
  - Hierarchical file systems
- The basic concept underlying the service is that the **onboard software should be able to access files in a file store or packets in a packet store independently of the precise physical location or address of the store.** The service is also **memory type independent** (Flash, SDRAM,...)



# SAVOIR – Mass Memory (3 of 3)



- Example of MMU based on Flash type memories:
  - the services are **memory type independent**,
  - Functions as wear levelling, garbage collector, bad blocks Mngt, etc are managed by the specific file system for Flash memories



# SAVOIR – Requirements for SRD Examples



- *Requirement Number* : SAVOIR.SRD.DMS.10  
Mass Memory shall store data in separate files
- *Requirement Number* : SAVOIR.SRD.DMS.20  
The Mass Memory shall allow the contemporary storage of data from all its input, downlink of files, execution of CDMU telecommands and provision of HK telemetry
- *Requirement Number* : SAVOIR.SRD.DMS.40  
The files in Mass Memory shall be addressed using file names
- *Requirement Number* : SAVOIR.SRD.DMS.50  
Files shall be transmitted to/from ground using the CFDP protocol
- *Requirement Number* : SAVOIR. OBC.CFDP.70  
The Mass Memory shall host a CFDP entity.



# SAVOIR – OBC Requirements (1 of 2)

## Examples



### *Functional Requirements*

- *Requirement Number* : SAVOIR.OBC.CFDP.10

The exchange of files between the Mass Memory and ground shall follow the CFDP protocol specified in [AD1]

- *Requirement Number* : SAVOIR.OBC.CFDP.20

The source and destination of files on board will be a location within the Mass Memory

- *Requirement Number* : SAVOIR. OBC.CFDP.30

The Mass Memory shall allow the on-board storage of :

- telemetry data (both instruments and CDMS)
- Scientific data
- SW images
- SW patches
- OBCPs

To be completed

- *Requirement Number* : SAVOIR. OBC.CFDP.40

The Mass Memory shall store in each open file the continuous stream of data coming from the associated input port starting after the open command for the file and until the close command for the file is received, or until the max file size limit is reached.





# SAVOIR – OBC Requirements (2 of 2)

## Examples



### *Functional Requirements*

- *Requirement Number* : SAVOIR.OBC.CFDP.50

Mass Memory content shall be maintained across platform re-configuration

- *Requirement Number* : SAVOIR. OBC.CFDP.70

The Mass Memory shall host a CFDP protocol entity.

- *Requirement Number* : SAVOIR. OBC.CFDP.40

The Mass Memory shall provide full observability of the status value of the file system

- *Requirement Number* : SAVOIR. OBC.CFDP.40

The Mass Memory shall store in each open file the continuous stream of data coming from the associated input port starting after the open command for the file and until the close command for the file is received, or until the max file size limit is reached.

- *Requirement Number* : SAVOIR. OBC.CFDP.40

The Mass Memory shall discard data received over any of its input for which a file is not open.



# SAVOIR – Conclusions and way forward



- There are different ways to implement CDFP and file store on the Space segment depending on
  - Mission requirements
    - Instruments scientific data volume and rate
    - Observability
    - Data Transfer latency
  - Availability of building blocks
  - Operation strategies
  
- SAVOIR will define a set of reqs in the SAVOIR docs also for implementation of CDFP and file system in CDMS (not present in the current version of OBC spec under finalization):
  - A ToR for a dedicated WG will be produced (April 14)
  - Ops aspects will be considered (participation of ESOC to WG/WG meetings)
  - At different levels: SRD and Equipment spec
  - The set of proposed requirements/docs will be reviewed by the **European space community** (as per conclusion of SAVOIR SAG mtg #23)

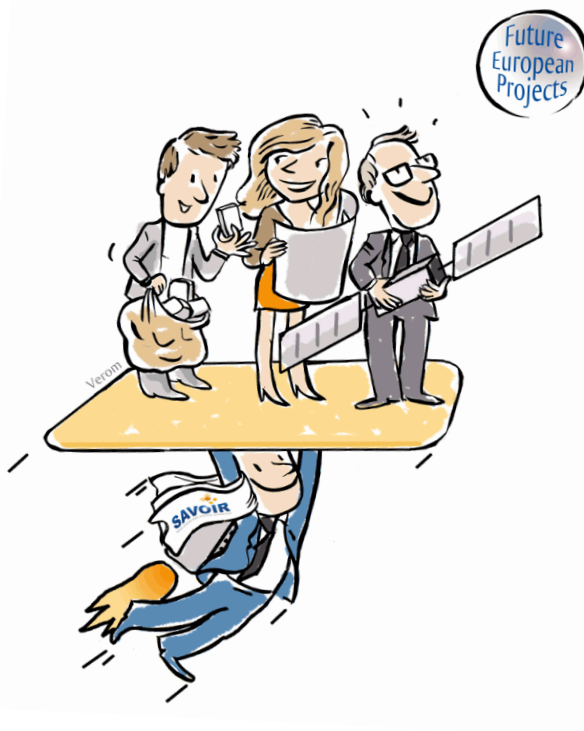


# Contact



Feedback: [savoir@esa.int](mailto:savoir@esa.int)

## Questions ?



# SAVOIR – ARCHITECTURAL ELEMENTS of the FILE DELIVERY PROTOCOL

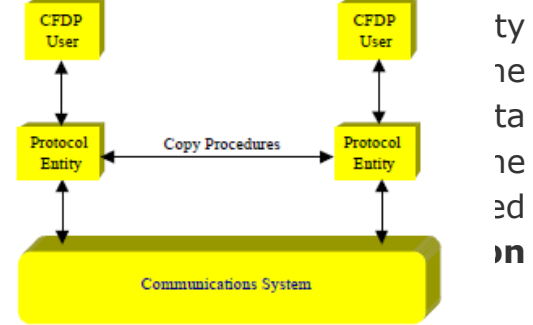
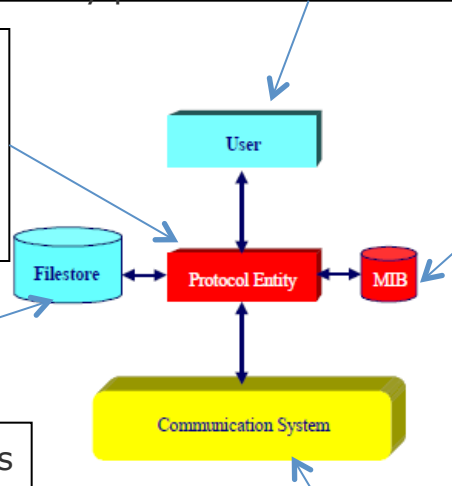
## (Optional- back-up slide !)



The protocol operates at the request of the CFDP user. A **CFDP User** is always a SW task. A CFDP user could be located inside the OBC processor or Mass Memory processor.

The **CFDP Protocol Entity** consists of implementations of the Core delivery procedures, which allow immediate file delivery and manipulation over a communication channel

To perform a file delivery, a significant amount of information must be passed by the and t remot is sta CFDP to as **Base**



The protocol operates by copying files from storage medium to storage medium, and it is therefore assumed that all CFDP entities have access to a local storage capability (**Filestore**) with the relevant SW services

The protocol assumes the availability of a single conceptual underlying **Communication System**, referred to as the 'Unitdata Transfer (UT) layer', to which all CFDP entities in a given CFDP addressing domain have access.

