SAVOIR Mass Memory Day at ESTEC

HW/SW architecture implications in relation to CFDP and file system operation

Y. Charnet, R.Cissou, I.Herrera, F. Koebel, F.Peran, T.Pike, M.Staehle, E.Weih
Airbus Defence & Space Electronics
14th February 2014

Intro

For several years now, Airbus Defence and Space (ex Astrium) has been working on SSMM architectures supporting file systems and transfer protocols.

CFDP is the leading candidate for supporting reliable file transfers over unreliable communication links e.g. for deep space missions and EO Satellites using Ka-Band systems.

This presentation discusses the implications of using file systems and CFDP in SSMM architectures and the resulting consequences for HW/SW partitioning.

Related Heritage in Airbus Defence & Space

ExoMars PDHU HW/SW implementation:

- Based on PUS Service 13 LDT
- Based on a Flash memory array
- Supports memory partitioning
- Supports standard and private PUS Services for TM/TC
- Supports file management (open, close, delete, report catalogue, etc.)
- Supports simultaneous uplink and downlink file transfers
- Stores both packetized and non-packetized data
- Supports automatic re-transmission for science data files

Current and last Studies and Analysis

EUCLID

- CFDP analysis for EUCLID Mass Memory System

NGMMA (ESA Contract)

- CFDP for high speed systems (in work)

SpaceWire.Server (DLR Contract)

- CFDP for future Science Missions (in work)

CFDP context – File-based Operations (FbO)

ESA FbO Working Group main conclusions:

- ESA missions would benefit from FbO adoption, especially when involving data relays / long propagation delays[1]
- Packets and Files are complementary (not exclusive), both should be supported
- File utilization/transfer on top of existing space link protocols is possible
- **CFDP** is the preferred solution for point-to-point file transfers (i.e. class-1 and class-2 only)
- Existing standards to not cover all needs for FbO
- Investment to support FbO is primarily required at the Space Segment
- **Files+CFDP** solution is proposed to manage and downlink payload data

FbO/CFDP adoption status at NASA:

- NASA selected (more than a decade ago) CFDP as the standard on-orbit spacecraft file transfer mechanism
- Successfully used in several missions: MESSENGER, Deep Impact, LRO, etc.

FbO/CFDP adoption status at ESA:

- Several missions supported FbO with ad-hoc solutions, mainly based on PUS
- **Euclid** (a deep space mission) will be the 1st ESA mission to support CFDP

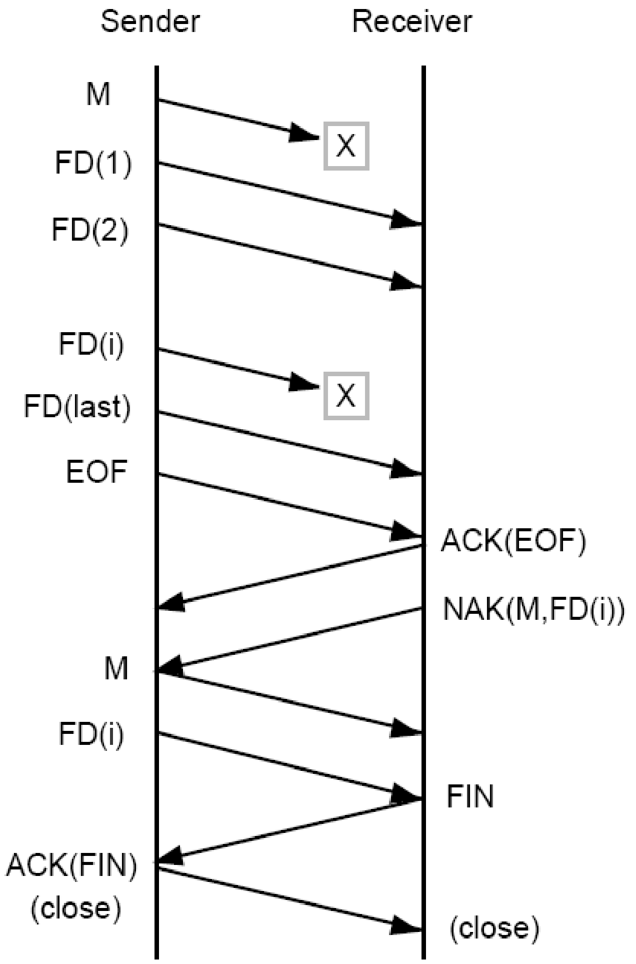
[1] Note: the benefit for EO missions with high-rate downlinks, unidirectional transfer and reliable links is less evident.

CFDP class 2 (reliable transfer) with deferred NAK Mode

Principle CFDP Communication

- high speed downlink interfaces
  - Meta Data (M)
  - File Data Segment (FD)
  - End of File (EOF)
- TM/TC subsystem (PDU messages)
  - File Store Directives
  - Acknowledgement (ACK)
  - Negative Acknowledgements (NAK)
  - Finished (FIN)

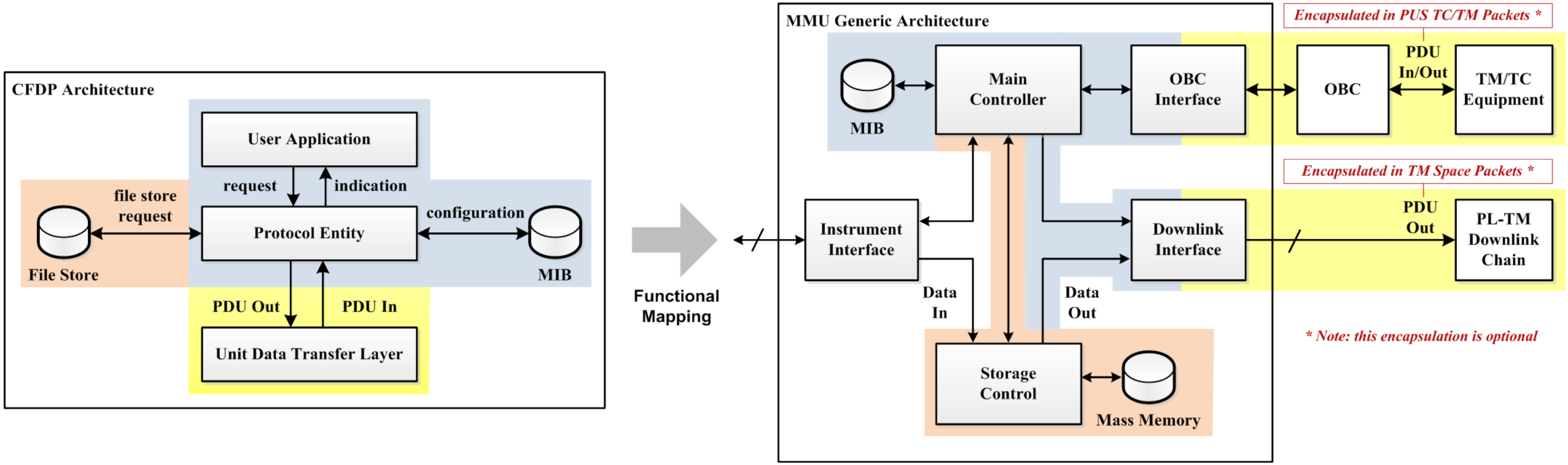
Protocol Data Units (PDU) exchanged with the OBC as is, or encapsulated in PUS packets.



CFDP functional mapping on a MMU generic architecture

Main points to be addressed w.r.t. MMU HW/SW architecture:

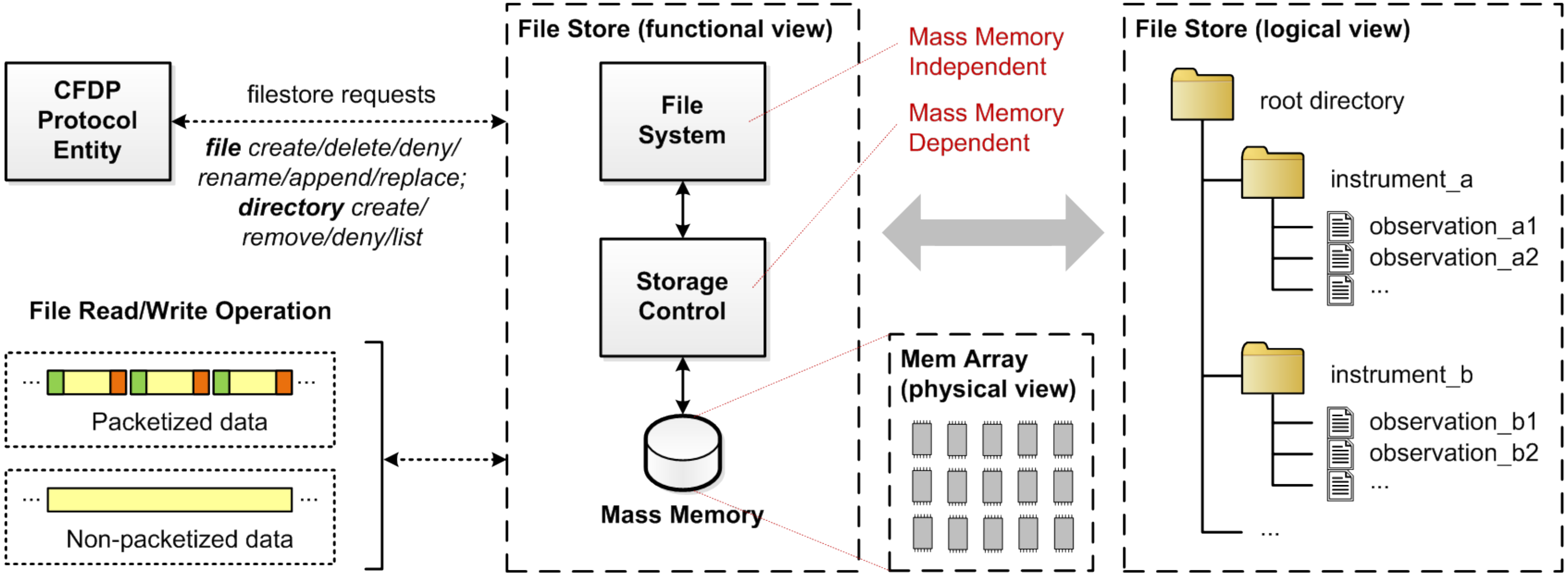
- For high-rate downlinks, part of the Protocol Entity must be implemented by HW at the Downlink Interface
- For closed-loop services, the Protocol Entity SW must receive File Directive PDUs from OBC (from uplink)
- Transmission of File Directive PDUs to OBC (for downlink) must be supported (optional use)
- Endpoints for CFDP requests/indications may be MMU SW or OBC SW



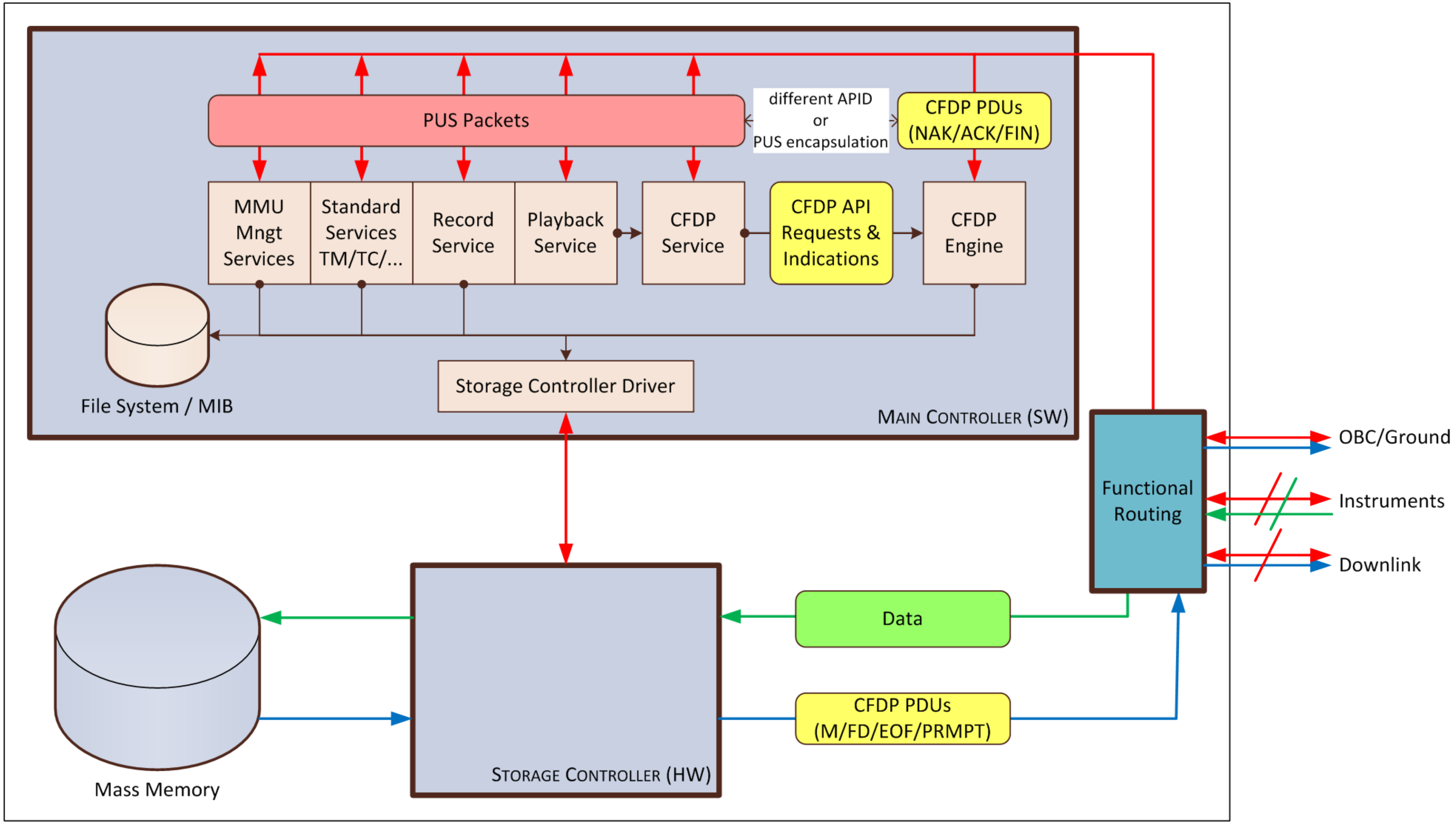
File operation on a MMU generic architecture

Main points to be addressed w.r.t. MMU HW/SW architecture:

- File System SW must support CFDP filestore requests
- File System SW must support the storage of packetized and non-packetized data into files
- File System SW must be independent from the Mass Memory technology
- Storage Control must include HW to support high-rate data read/write into physical Mass Memory
- Storage Control HW must be designed for specific Mass Memory technology



CFDP MMU – Focus on SW Partition (1/2)



CFDP MMU - Focus on SW Partition (2/2)

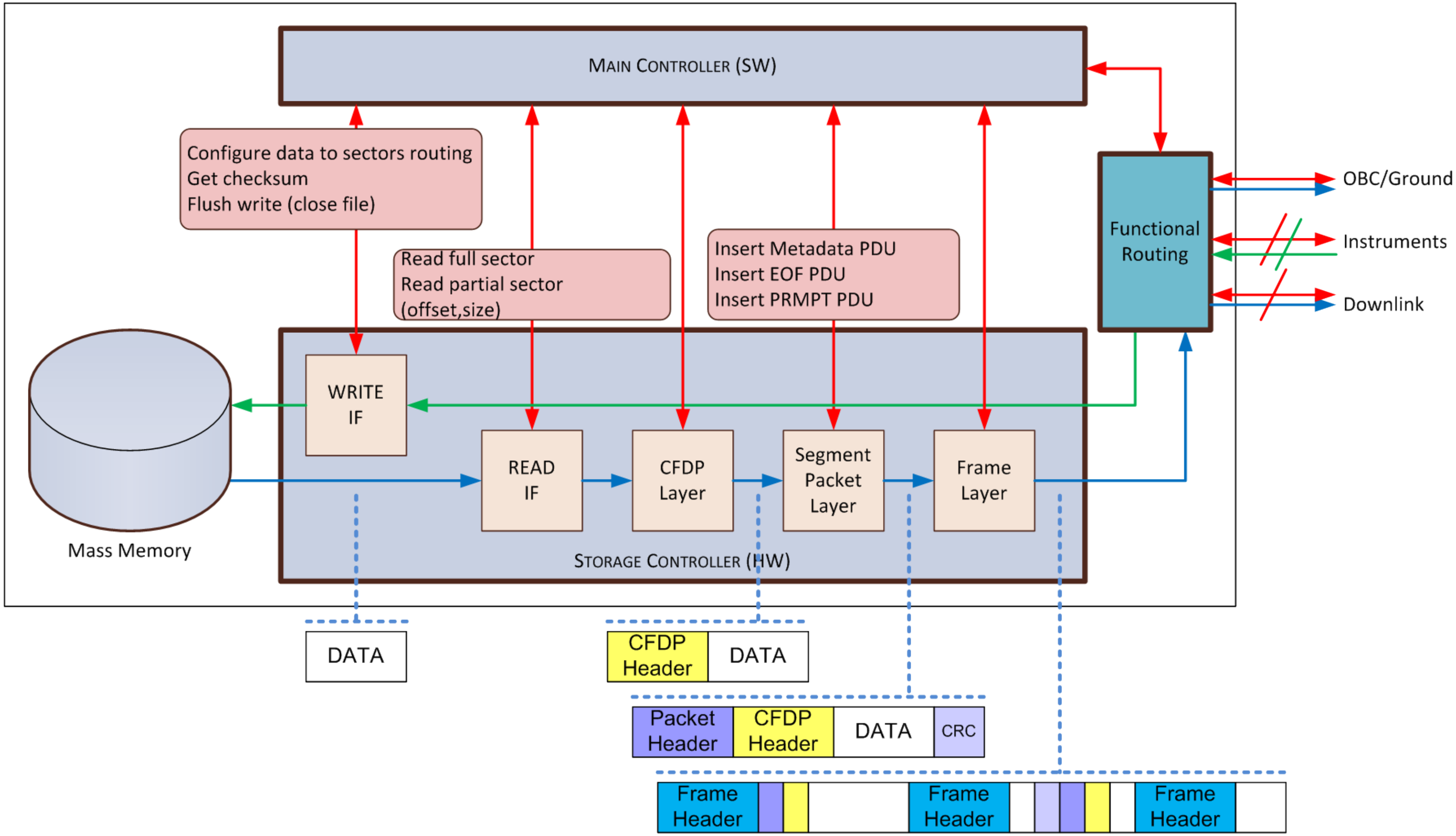
Services with PUS TM/TC interface

- Standard services (TC,TM,...)
- Record service for data recording
- CFDP Service to handle CFDP Requests & Indications embedded in PUS TM/TC => could be standardized
- Playback Service for automatized file transfer with rules
- Management Service

CFDP Class 2 Source entity with CFDP interface

- Internal API for CFDP Requests & Indications
- NAK/ACK/FIN PDU handling
- PDU could be received as native space packet or PUS packet.

CFDP MMU - Focus on HW Partition (1/2)



CFDP MMU - Focus on HW Partition (2/2)

SW

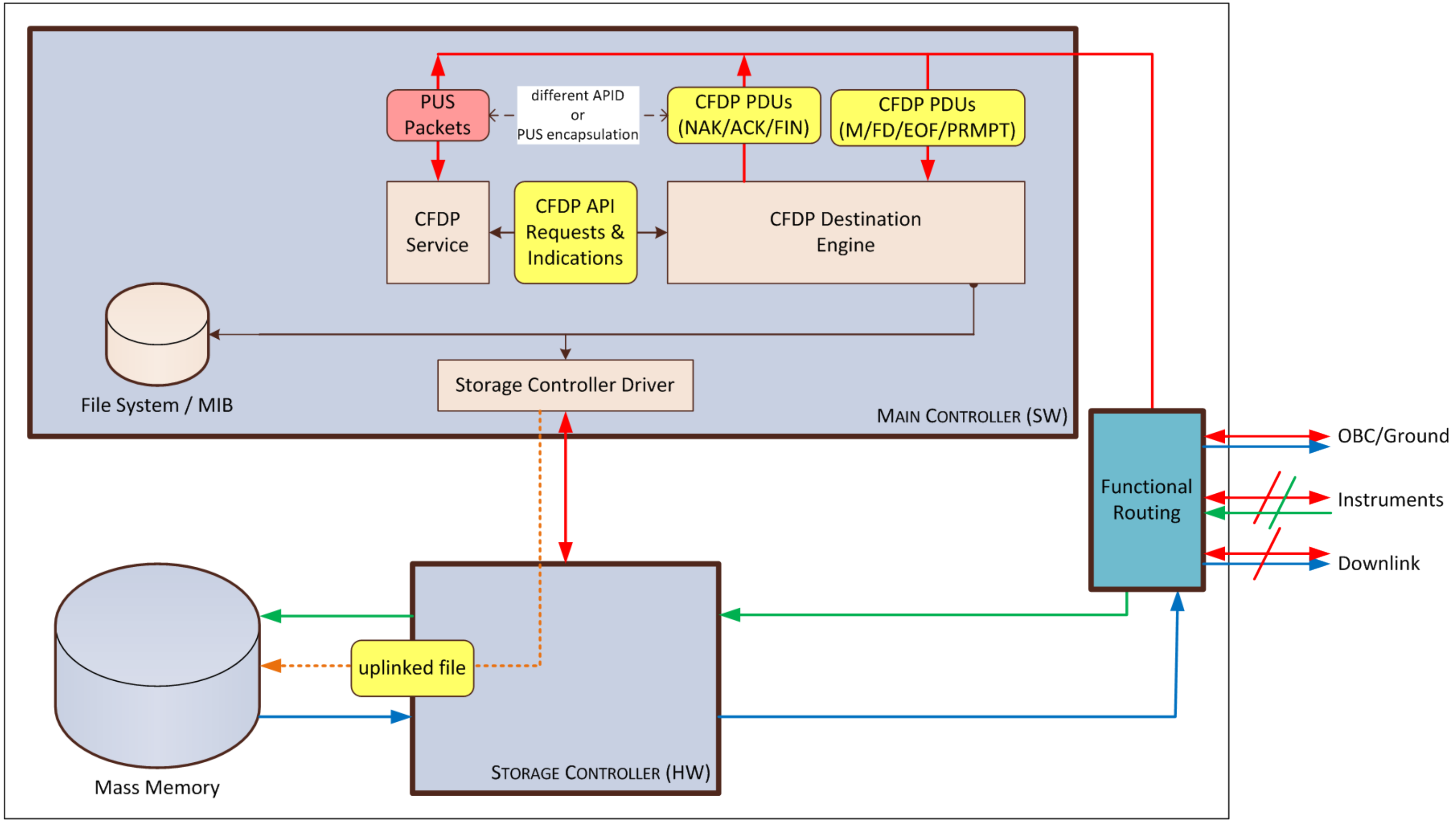
- handles files & directories (logical view)
- links files to sectors of the memory array (physical view)
- manages the memory array for FDIR
- operates the storage control hardware

HW operates at physical level (sector)

- A **Sector** represents the smallest entity which can be randomly accessed by the File System
- Sector size depends on required access granularity, data rate and processor performance (e.g. 1 Mbyte to 128 Mbytes)
- writes data to sectors
  - according to a route table, for example, separating packets according to APID or instrument
  - compute sector/file checksum
- reads data from sectors to generate two streams of CFDP File Delivery Units (FDUs)
  - one FDU process for nominal downlink data
  - one FDU process for retransmission of data

    => downlink bandwidth can be exploited for transmission of nominal data in real time and for retransmission of lost PDUs
- encapsulates FDUs and SW generated PDU (Metadata, EOF, Prompt) in CCSDS packets (optional) and frames for downlink

CFDP MMU – Uplink proposal



NAK management

NAK management requires access to small data portions, which may lead to reduced downlink speeds.

This can be easily handled by two parallel operated downlink data streams,

- one for nominal data to exploit the downlink bandwidth and
- one for retransmission of data not received on ground.

By this also different files can be downloaded at a time.

Implementation figures

CFDP Function needs hardware support

- Downlink Data Rate, typically several hundred of Mbit/s but for some cases up to 5 Gbit/s
- max. size of File Data PDU (FD): 64 Kbytes
- CFDP Handler Interaction Frequency: ~1024 PDUs/sec but for some cases up to ~10240 PDUs/sec
- Software Interaction Frequency on Sector Level: > 1.5 ms per Sector (@ sector size: 1 Mbyte)

Maximum File Sizes:

- max. size of CFDP-File: 4 Gbyte
- max. size of real file / packet store: up to completely available storage area (~Tbyte)
- => a file needs to be delivered by several chunks (here: transactions)

Conclusion

CFDP is seen as a major trend in supporting reliable file transfers over unreliable communication links.
This view seems to be shared by the space community in terms of ESA and industry.

A preliminary HW/SW partition of CFDP function has been proposed:

- SW managing the PUS TM/TC, the File System and the CFDP Protocol Engine (except Data PDU formatting for high-rate downlink)
- HW managing the file data read/write access to Mass Memory and output Data PDU formatting

The introduction of CFDP should be done with consideration of and in cooperation with well-established procedures:

- Instrument/uplink data will be processed and recorded into “files” directly (when non-packetized), or into “packet stores within files” (when packetized)
- “files” containing raw data or packets will be managed and transmitted/received by means of CFDP procedures
- a “packet store” might be separated into several files by means of a proper mechanism due to the limited size of CFDP controlled files (4 Gbyte)