SpW-D Overview and Trade-offs

Ph. Armbruster
ESA/ESTEC
21/03/2011

European Space Agency

Introduction

- This presentation covers requirements, baseline considerations and a list of still open trade-offs for the definition of protocols running over SpaceWire networks, intended to provide a deterministic scheme for handling data transfers.
- The list of requirements is not meant to be exhaustive nor final as some trade-offs are still to be closed. The main objective of this presentation is to structure discussions during the SpW WG meeting#16.
- The features and protocols aiming at making SpaceWire networks Deterministic are referred to as **SpW-D**.
- A main evolution w.r.t previously mentioned options embedded in SpW-D is the removal of the "Multi-slotting" case as a mean to avoid segmentation. This has been done on the ground of the added complexity introduced by multi-slotting, in particular for fault containment in case of time slot interval violations.

2.1 Performance Requirements

- SpW-D is intended to provide the means to handle real time traffic on a SpaceWire network with a level of performance compatible with the following range of control loop frequencies:
  - 1Hz (Pulse Per Second (PPS) driven services)
  - 10Hz (AOCS, typically Mil1553 range)
  - 100Hz (high-end AOCS, pointing, max Mil1553 capability)
  - 1KHz (motor control, Robotics, µ-vibration compensation, fine pointing)
- Furthermore, the original characteristics of SpaceWire links in terms of high throughput have to be preserved. This extends as well to qualitative properties such as “simplicity” and low/medium implementation costs (e.g. gate count in an ASIC/FPGA).

- SpW-D shall allow the multiplexing of command & control traffic and High Throughput Data Transfers (HTDT) on the same links/network.

- SpW-D provides synchronisation properties to a natively asynchronous network.

2.2 Scheduling – Time distribution 1 / 2

- The main objective being to schedule data transfers on the network in a deterministic manner (by suppressing, by design, any contention at SpW Router level), every node in a scheduled SpaceWire network has to maintain a Local Clock that is synchronised to a common Master Clock.

- The node embedding the current Master Clock shall be the source of time codes broadcasted on the network by SpW Routers.

2.2 Scheduling – Time distribution (2/2)

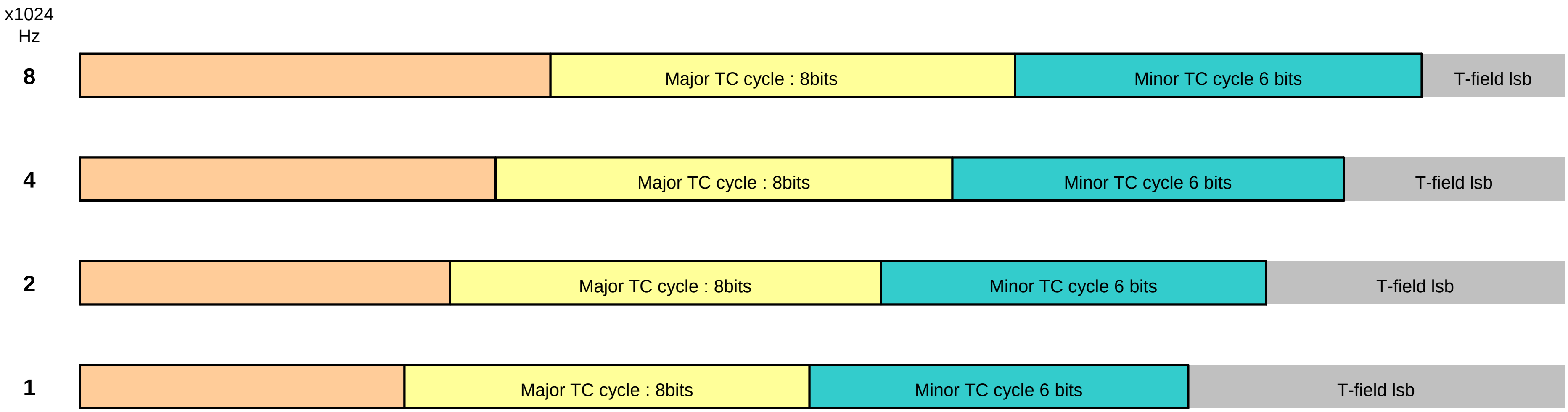
– Time distribution is therefore a core function of SpW-D, and includes the following features:

- CCSDS CUC format compliance.
- Time distribution and node synchronisation mechanism as described for instance in the document "SpaceWire – CCSDS Unsegmented Code Transfer Protocol – 1st of October 2010, Version 1.4, Aeroflex Gaisler). This has the consequence for valid Time Code Periods (TCP) to be a power-of-2 division of 1 s. (see following slide)
- Local Time is stored and managed by an Elapsed Time Counter in every node part of the scheduled network
- Time codes, coded on 6 bits, define a Minor Cycle
- Major Cycles extend the ambiguity period of Minor Cycles to more than a second (see following slide)

SpW-D Elapsed time counter format

SpW-D Elapsed time counter format

	CUC Time Code field																							
	coarse time									Fine time														
	TCI Approx. :								1s					32ms	16ms	8ms	4ms		1ms	500µs	250µs	125µs		60ns
TCF	2^{31}	**.....**	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	2^{-9}	2^{-10}	2^{-11}	2^{-12}	2^{-13}	**.....**	2^{-24}



2.3 Selection of Time code frequencies

The Frequency of Time code broadcasting has to be selected according to the following somehow antagonistic needs:

- The frequency has to be high enough to be compatible with the 1 KHz control frequency requirement while allowing multiple control loops to be handled.
- The frequency has to be low enough to maximise efficiency for HTDT while keeping segmentation optional in a range of practical cases.

Time Code Frequency	**Time Code Period**			**Minor Cycles per second**	**Minor Cycle period**	
	s	s	µs (rounded)		s	ms (rounded)
8 KHz	2^{-13}	1/8192	122.1	128	2^{-7}	7.8
4 KHz	2^{-12}	1/4096	244.1	64	2^{-6}	15.6
2 KHz	2^{-11}	1/2048	488.3	32	2^{-5}	31.2
1 KHz	2^{-10}	1/1024	976.6	16	2^{-4}	62.5

2.4 Links speed

- The speed of all links in a scheduled network shall be set uniformly to the same frequency.
- The following link frequencies are considered (as typical cases): 2, 10, 20, 50, 100 and 200 MHz.
- The following table provides an indication of the available bandwidth by deriving the number of 8-bit data characters (coded on 10 bits) that can be transferred in a Time Code Period.

Link Frequency MHz	**TC Frequency** (KHz)			
	8	**4**	**2**	**1**
2	24	48	96	192
10	122	244	488	976
20	244	488	976	1952
50	611	1222	2444	4888
100	1222	2444	4888	9776
200	2444	4888	9776	19952

2.5 Handling of multiple data transfers or transactions in the same time slot

- Some combinations between Link and TC Frequencies show that a high amount of data characters can be transferred in one time slot.

- In order to maximise throughput, multiple data transfers or transactions are allowed to be scheduled in a given time slot provided that:
  - Data transfers do not compete for conflicting network resources

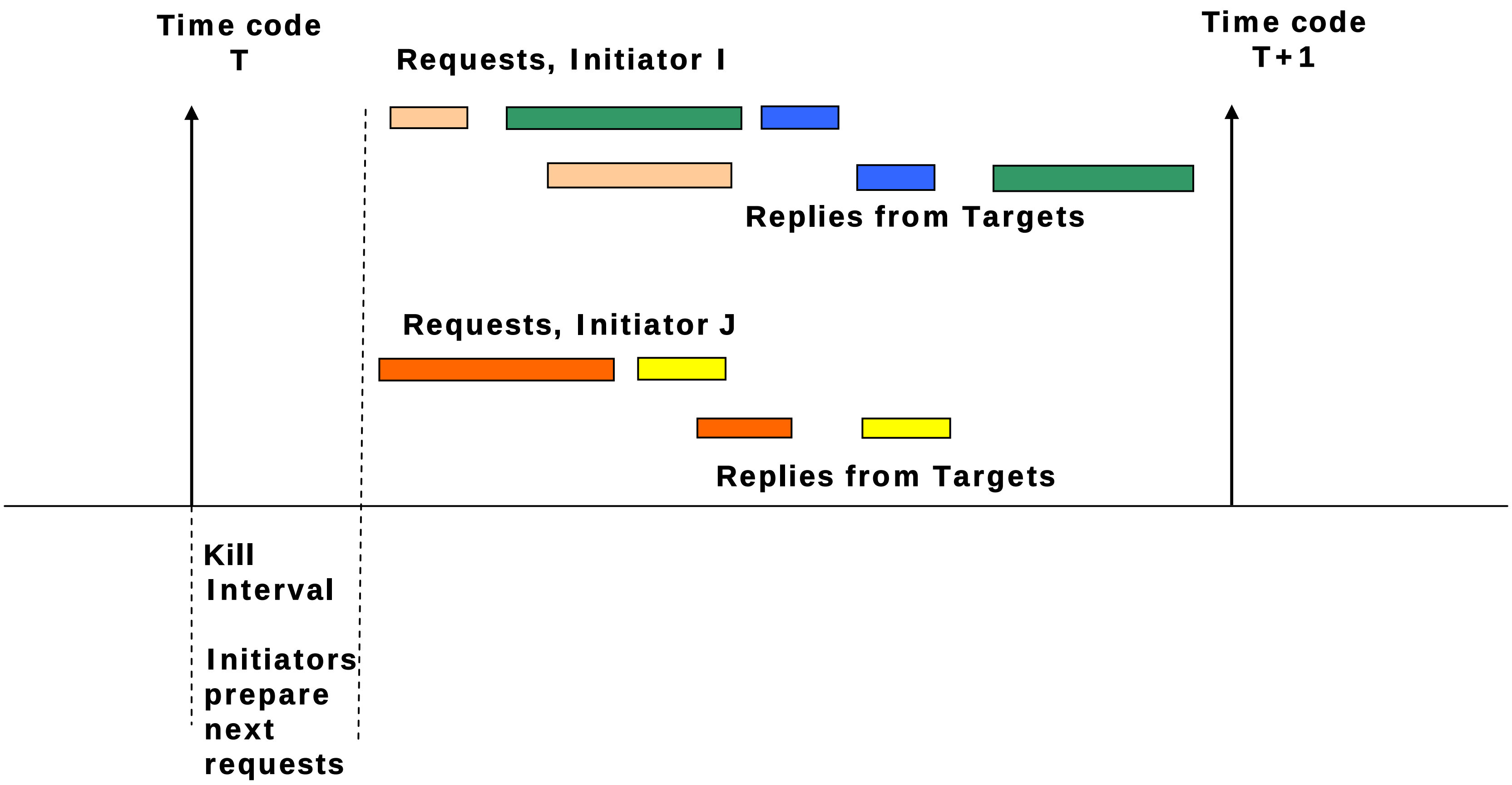
  - All Data transfers are completed before the next time code is being received (by Initiators, Routers and Targets).

2.6 Type of data transfers

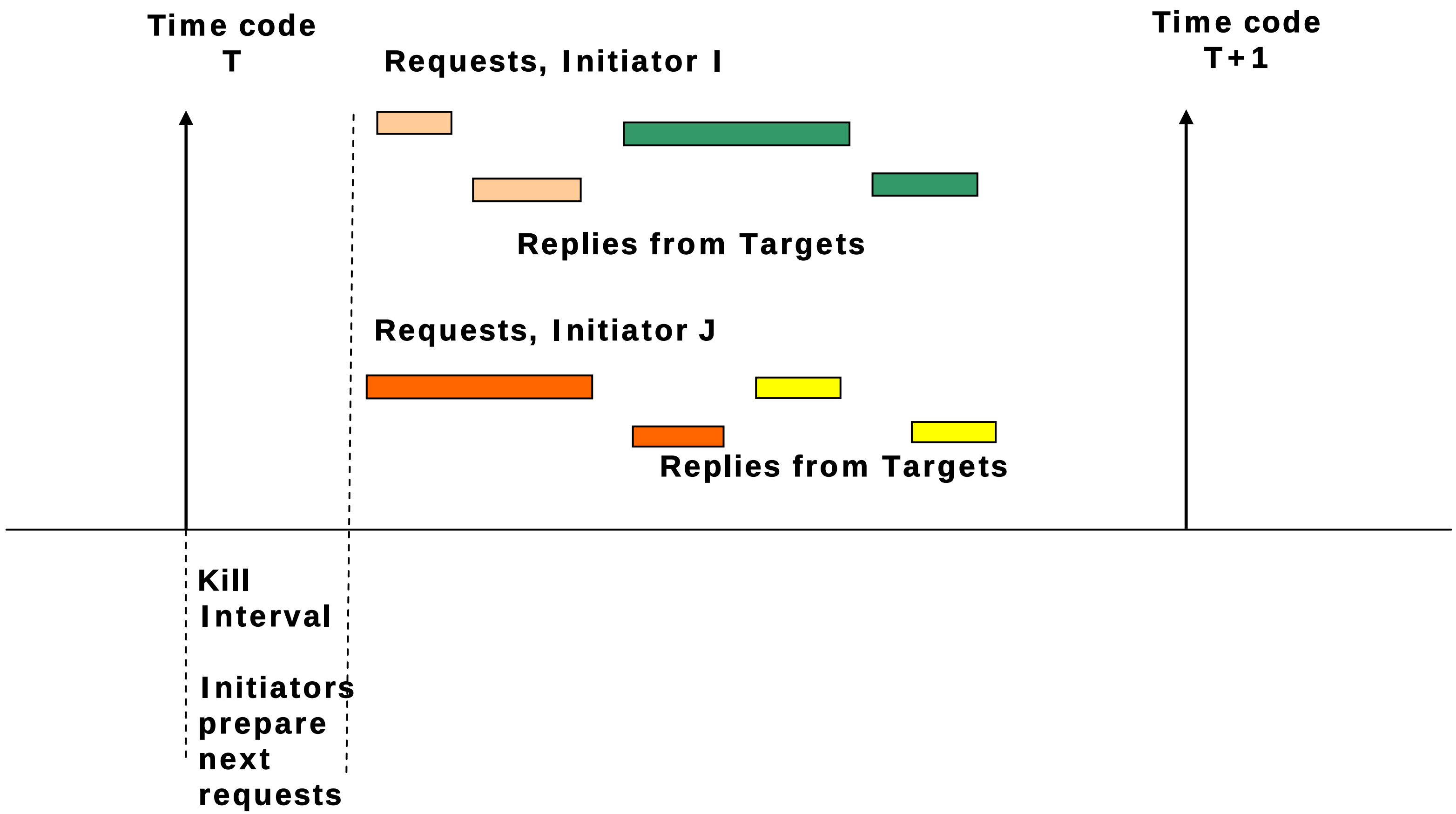
- Authorised data transfers include in the general case any (TBC) SpaceWire packet transmissions provided SpW-D scheduling constraints are respected.

- Data transfers include native SpW packet transmissions (TBC in particular regarding the addressing mode), transfers related to RMAP transactions or transfers of packets embedding specific protocols.

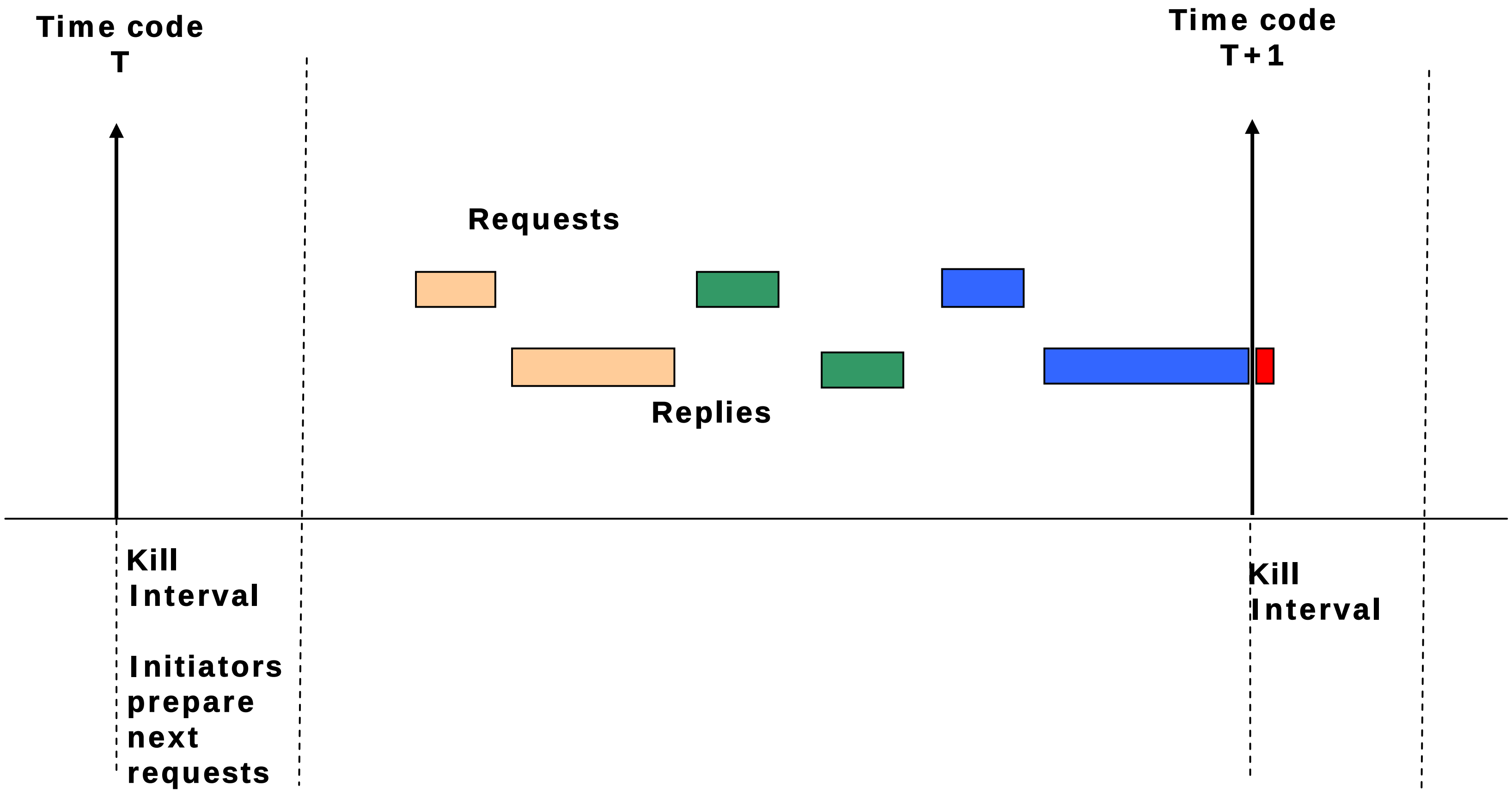
2.7 Structure of a time slot
Example A: Posted RMAP Transactions



2.7 Structure of a time slot
Example B: Sequential RMAP transactions



2.7 Structure of a time slot
Example C: RMAP transactions violating time code boundary



2.8 Behaviour in case of transfers or transactions exceeding time code interval boundaries

- If data traffic still exists when a time code is received. Taking RMAP transactions as an example, this would translate in:
  - Initiator and Target detecting the anomaly and stopping/aborting packet transmission
  - Routers detecting the anomaly and flushing/spilling packets

- The time slot period is headed by a Kill Interval (duration TBC) used:
  - to recover from a detected anomaly
  - to prepare for the next transfers or transactions

2.9 Determinism, integrity and reliability

- Determinism: All data transfers and transactions are:
  - Scheduled, synchronised with time codes
  - Time bound: are executed nominally within one time slot
  - Detected and Killed when exceeding the allocated time slot
- Strict Determinism :
  - A transaction exceeding the allocated time slot shall not have any impact in the next time slot(s)
- Integrity: Data integrity is provided by SpW, the protocol used for the transfer (when implemented) and/or the application. No specific data integrity scheme is provided by SpW-D per se.
- Reliability: is provided by SpW and the application. No additional reliability is provided by SpW-D (e.g. no retries)

2.10 Channels and other features (1/2)

Channels are being defined as the network resources needed to perform data transfers in a given time slot, but are not part of SpW-D itself. They can be seen as an abstraction used for the definition of SpW-D schedules, potentially used at system and application level.

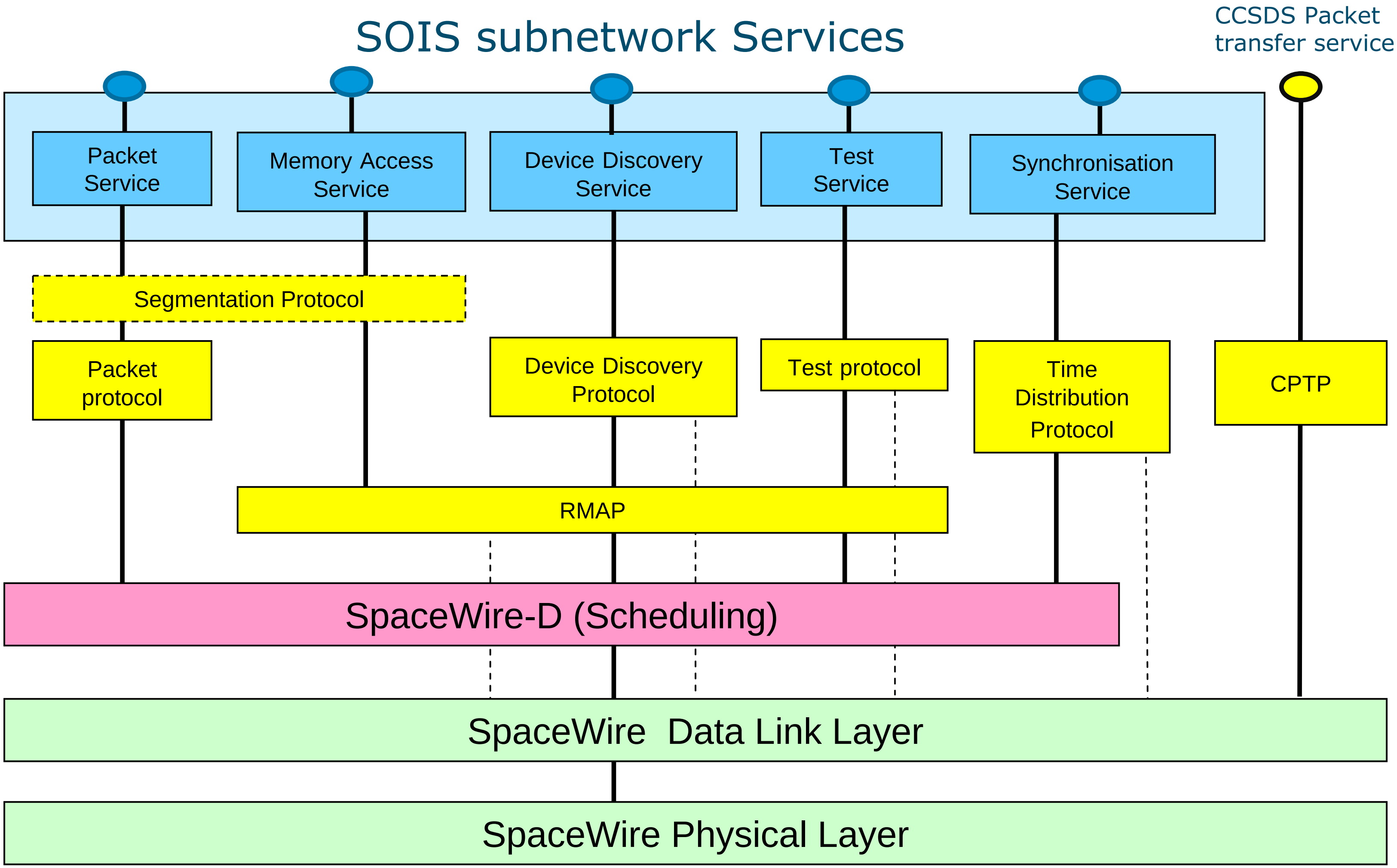
2.10 Channels and other features (2/2)

- Major cycles management: introduced for cyclic monitoring/commanding with periods above Minor TC cycle durations

- SpW Port Host interfaces shall respect the following constraint:
  - Packets have to be accepted or discarded or the interface shall allow data to be overwritten (TBC, new behaviour).
  - Data transmission cannot be stopped or denied. When data is being overwritten, this shall be notified to the application

- Compatibility with existing devices shall be maximised while defining SpW-D but not at the expense of bending determinism requirements.

SpW-D Protocol Stack



SpW-D Protocol Stack

- The layered diagram indicates the mapping of the existing SpaceWire protocols along with the proposed SpaceWire-D. It is a work in progress that may need updates as functions are firmed up. The following is worthy of note:
  - A segmentation protocol is required if it is desired to send data greater then the SpW-D slot size
  - Many protocols may need direct access to underlying layers (indicated by a dotted line in the diagram)
  - The CPTP provides a service for transferring CCSDS packets but it does not support the SOIS packet service, as this will transfer any fixed length data structure
  - Many protocols do not yet exist: Segmentation, Packet transfer, Device discovery, Test, Time distribution
  - A decision must be made on the Time distribution protocol if the full synchronisation service is to be support or not (signalling , especially requested by Yuri)
  - Network management including FDIR is not yet addressed by the diagram

Summary

SpW-D includes:

- A SpW Network scheduling concept
- The specification of discrete time code periods
- A CUC Time distribution and Synchronisation scheme
- *FDIR mechanisms: slot usage monitoring, packet routing timeouts and kill actions when time slot boundaries are violated -> needs to be further elaborated*

SpW-D schedules deterministically any (TBC) data transfers on SpW networks and in particular:

- RMAP transactions
- Data transfers according to the CCSDS Packet Transfer Protocol
- SpW – CCSDS Unsegmented Code Transfer Protocol packets

SpW-D will allow to support:

- CCSDS SOIS Service implementation (e.g. Packet Transfer Service, Memory Access Service, Device Discovery Service, Test service, ….)

Open points and Trade-offs

- Allowing any SpW native packet to be transferred, irrespective to the addressing mode (path or logical) or limiting it to specific protocols (e.g. RMAP, CPTP, etc).
- Verifying the need to distinguish between Determinism and Strict Determinism.
- Allowing posted RMAP transactions or only Sequential RMAP transactions
- Defining Schedules in a standardised way, e.g by defining channels
- Defining an optional Segmentation layer
- Elaborating a unified SpW-D Stack representation
- Elaborating on Network monitor function support
- Establishing the compatibility level with existing devices
- Maximising interoperability: scrutinising the existence of options and keep variability to a minimum
- Cross checking the robustness of timing faults containment schemes.