



The ladybird guides

Book II

Day 5 - Protocols

Consider two messages from the spacecraft

- 1) Switch A is open
- 2) Switch A is closed

Remember the most bandwidth efficient way of passing the message between the receiver and transmitter is with a single bit

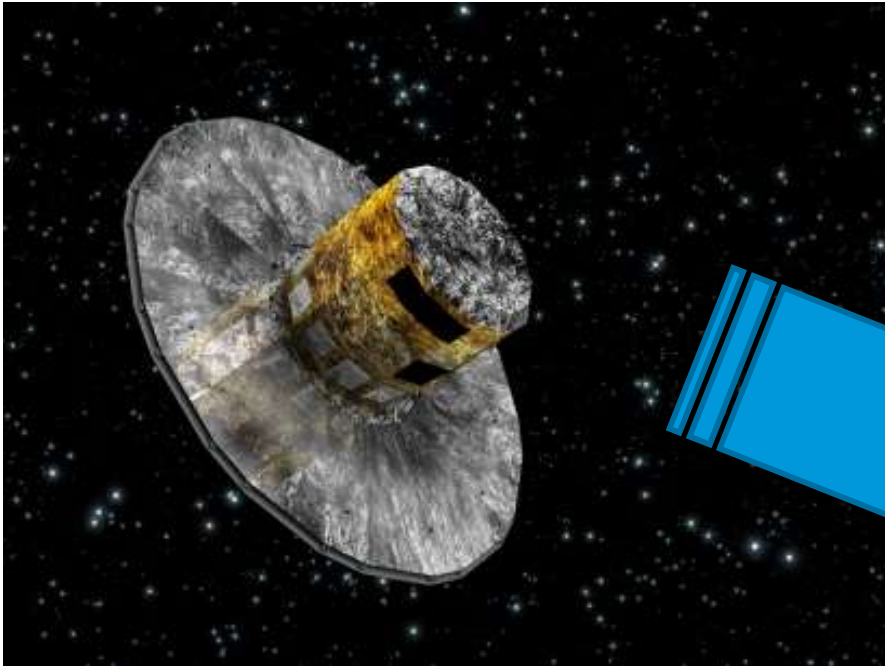
.....but only if both the receiver and the transmitter know which information that bit status is referring to

I.e. If Packet 301, bit 20 = 1 then Switch A is open
If Packet 302, bit 20 = 0 then Switch B is closed

This is the principle of the shared operational database and is the biggest contribution to improving the information rate we have!

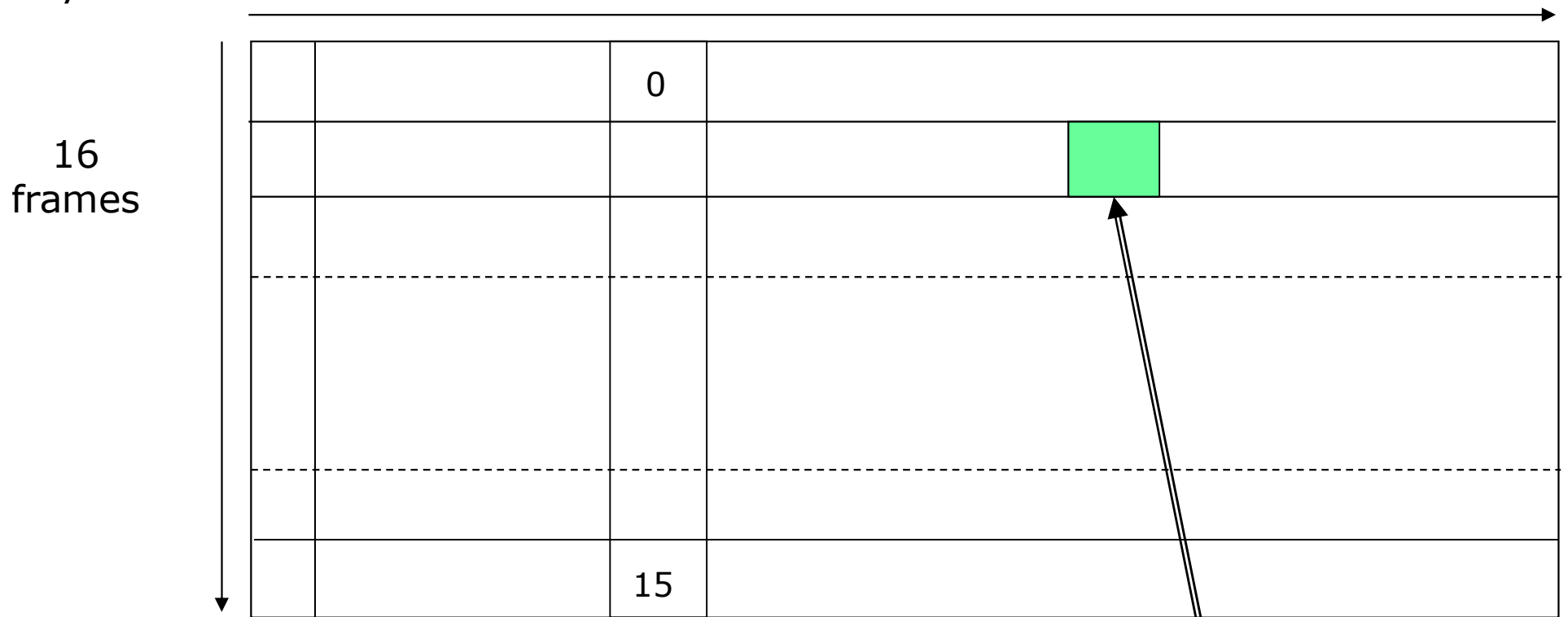
The process of extracting the value information from the telemetry and associating it to its relevant parameter is called DECOMMUTATION...

Telemetry Space Link Protocols



"Old" TM protocol

The on-board DHS collected information about itself and the other subsystems. It placed the VALUES for different parameters in a data structure, called a telemetry format.



sync word,
identifies start
of frame

frame
counter

temperature of IRES
always written here by
the on-board DHS

“Old” TM protocol



The format is made up frames where the start of each frame is marked by a synchronization word and at a set distance from this the number of the frame is written.

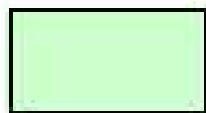
This structure is then transported to the ground station which slices up the constant telemetry stream where synchronization word (and hence start of frame) occurs. These individual frames are then sent to the mission server.

The server then looks for the frame number in the predefined position. It can use this as a key because it knows that all the other pieces of information are written in the same place relative to it. Hence the temperature of the IRES can always be found by looking at the value of word 15 in frame 3 etc.

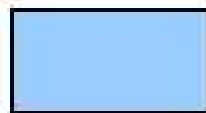
Therefore once the frame number is known everything else can be interpreted according to the format map.

"Old" TM protocol

FS	SFID	WD1	WD2	WD3	WD4	WD5	WD6	WD7	WD4
FS	SFID	WD1	WD8	WD3	WD4	WD5	WD6	WD7	WD4
FS	SFID	WD1	WD2	WD3	WD4	WD5	WD6	WD7	WD4
FS	SFID	WD1	WD8	WD3	WD4	WD5	WD6	WD7	WD4
FS	SFID	WD1	WD2	WD3	WD4	WD5	WD6	WD7	WD4
FS	SFID	WD1	WD8	WD3	WD4	WD5	WD6	WD7	WD4



Normal Commutation



Sub Commutation

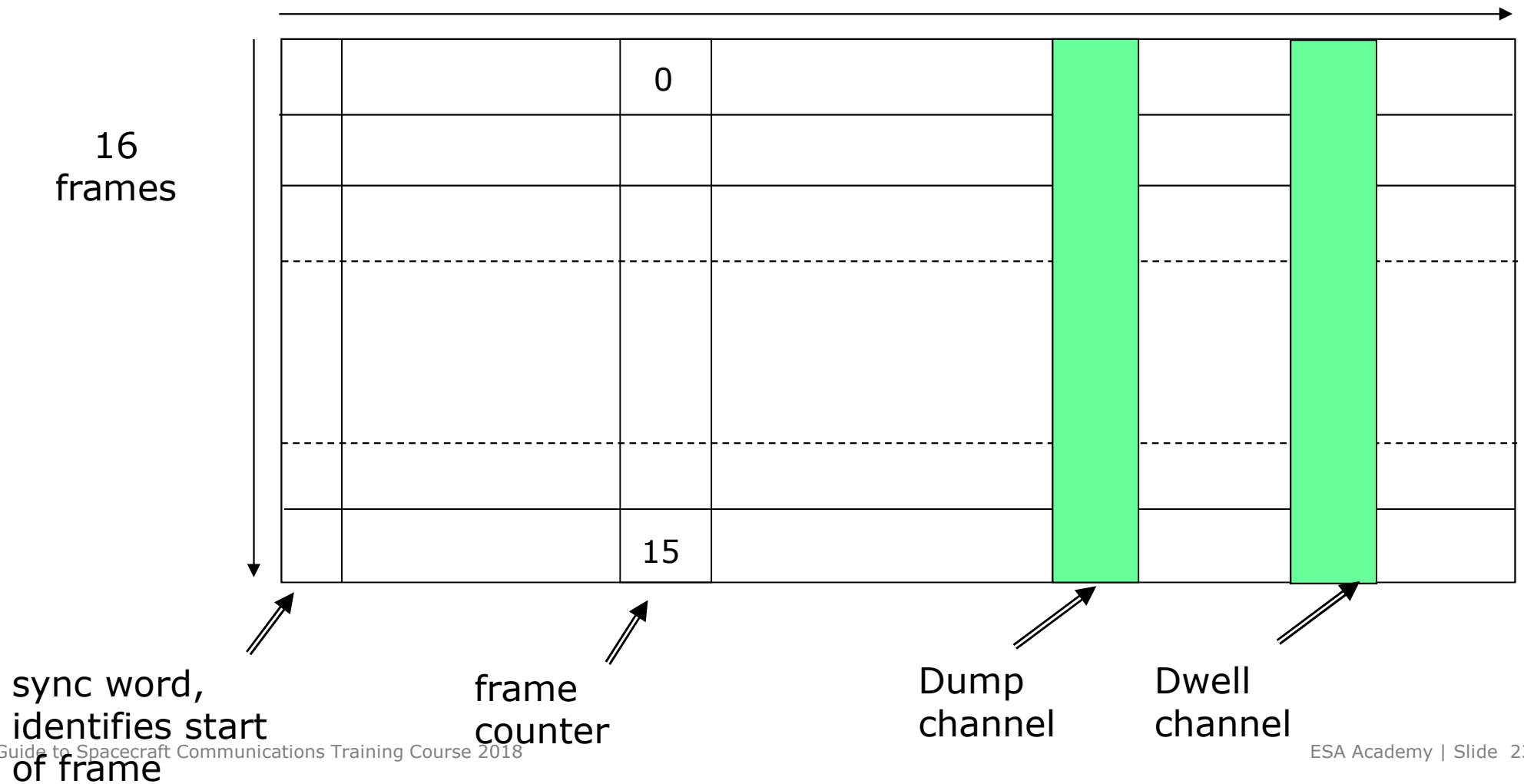


Super Commutation

"Old" TM protocol

But how can we do things like dump memory?

But how can we do get flexibility to look at things more often than usual?



Formats are very efficient but not very flexible.

In the late 80s's variable length Packet TM and TC CCSDS communications standards were written.

Individual packets could be sent at different downlink periods e.g. Packet 1 every 30 secs, packet 2 every 5 mins. This could be changed in flight.

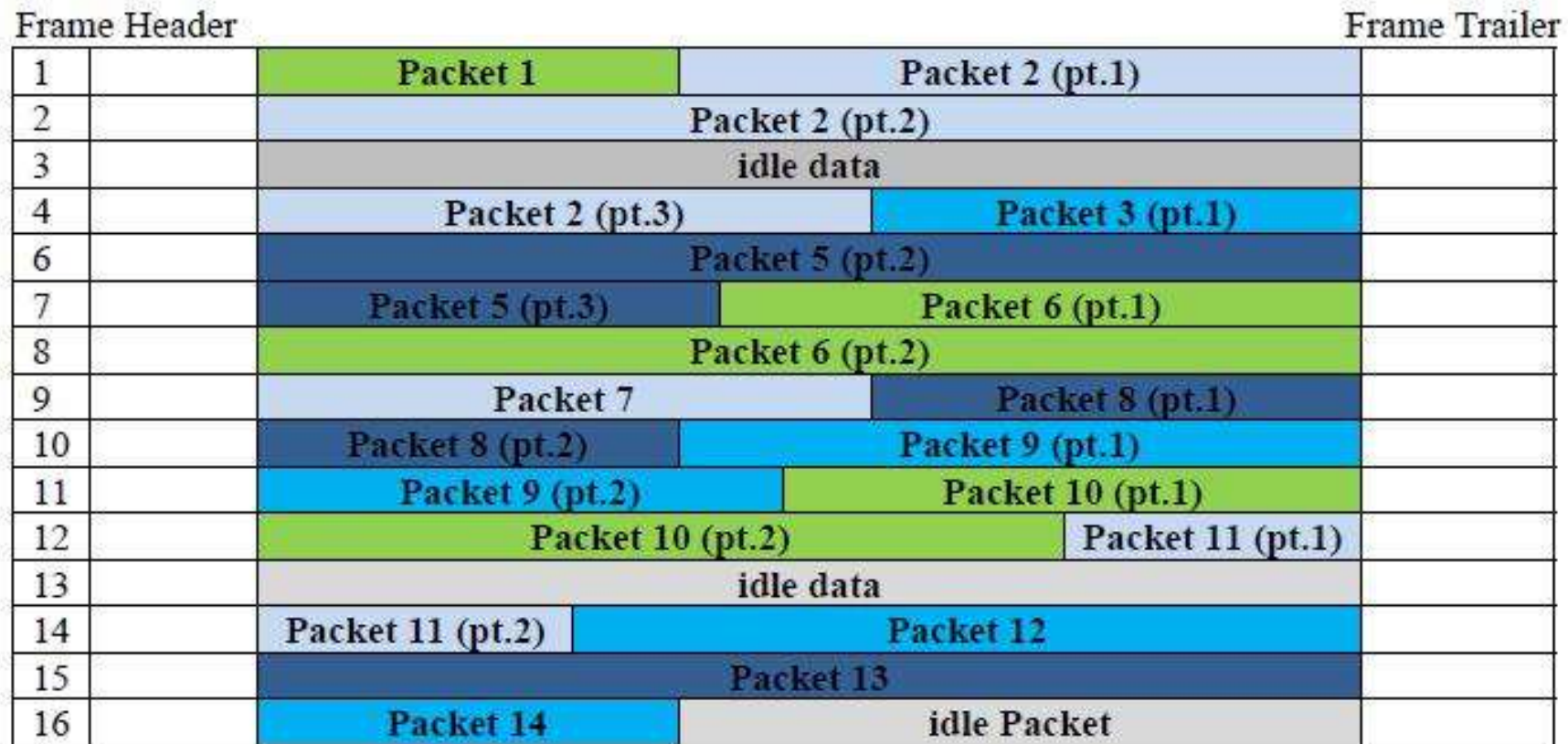
It was also possible to turn off and on the generation of different packets.

- Improve communications link flexibility (easy to change what you want)
- Improve link efficiency (no need to telemeter everything all the time)
- Maintain a minimum load on spacecraft resources

Frame Synchroniser

But hold on, if we use variable length packets how does the ground station know how to chop up the incoming bit stream?

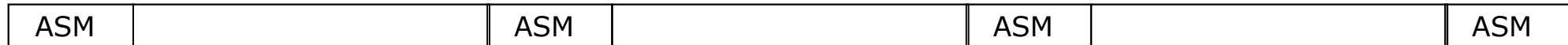
Answer: The packets still sit inside frames



↑
Frame Counter

A frame delimiter called the ATTACHED SYNCHRONISATION MARKER (ASM) must be added during the coding process.

The marker is simply this sequence of bits (1ACFFC1D) that is added to the start and end of each transfer frame.



There is a difference of **when** it is added according to the coding.

- 1) No coding, then just add it to the transfer frames and send to the modulator
- 2) R-S or turbo only coding, then add it after the transfer frame has been coded.
Then send to the modulator.
- 3) Concatenated coding, then add it after the transfer frame has been R-S coded.

NB: A different ASM must be used for turbo coding

Frame synchronization must take place before the Reed-Solomon decoding process.
Quiz: Why?

Answer: R-S is a block code so the decoder has to know where the parity bits are in the bit stream

However this is not true for the convolutional decoding process as this can work on a bit stream (no parity bit blocks)

Frame synchronization does **not have to** take place before the convolutional decoding process as this can work on a bit stream (no parity bit blocks).

However frame sync it is done at ESA before the convolutional decoding and at NASA afterwards.

How? Adding the ASM after the transfer frame has been R-S coded but before the convolutional coding causes us a problem in the decoding as we have to find a marker that has been convolutional coded!

There is a trick:

If we use standard CCSDS convolutional coding (i.e. $1/2$, $k=7$, no puncturing, no differential coding) only the first 12 bits of the coded ASM (1ACFFC1D) change.

The other 52 bits remain static and therefore recognisable as 8 1C 97 1A A7 3D 3E.

So ESA ground stations search for the corresponding bit pattern in the bit stream from the demodulator.

Note with suppressed carrier BPSK there is also a 180 phase ambiguity therefore the bits could be inverted.

QUIZ: Why are the first 12 bits always different when we convolutionally code an ASM with ($1/2$, $k=7$, no puncturing)?

Cast your minds back to the demodulation session and you will remember that there is a DTTL which is responsible for determining the timing at which the phase of the IF wave should be measured. If there are not enough transitions in the symbol stream then it will lose lock.



Remember the EURECA deployment story?

The EURECA low speed TM link to the shuttle was using non encoded data at 2 kbps, BPSK modulated.

Would the problem have occurred if the TM link was convolutional encoded?

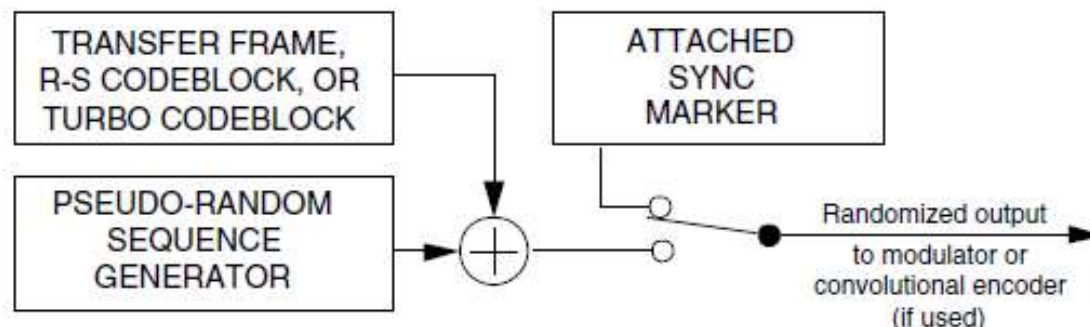
Would the problem have occurred if the TM link was convolutional encoded and QPSK modulated?

Pseudo Randomiser

A safe method for ensuring sufficient transitions occur in the symbol stream, no matter what the coding and modulation scheme is, is pseudo-randomisation. This is an exclusive-OR each bit of the incoming Transfer Frame with a standard pseudo-random sequence. This will remove any long sequences of zeros or ones.

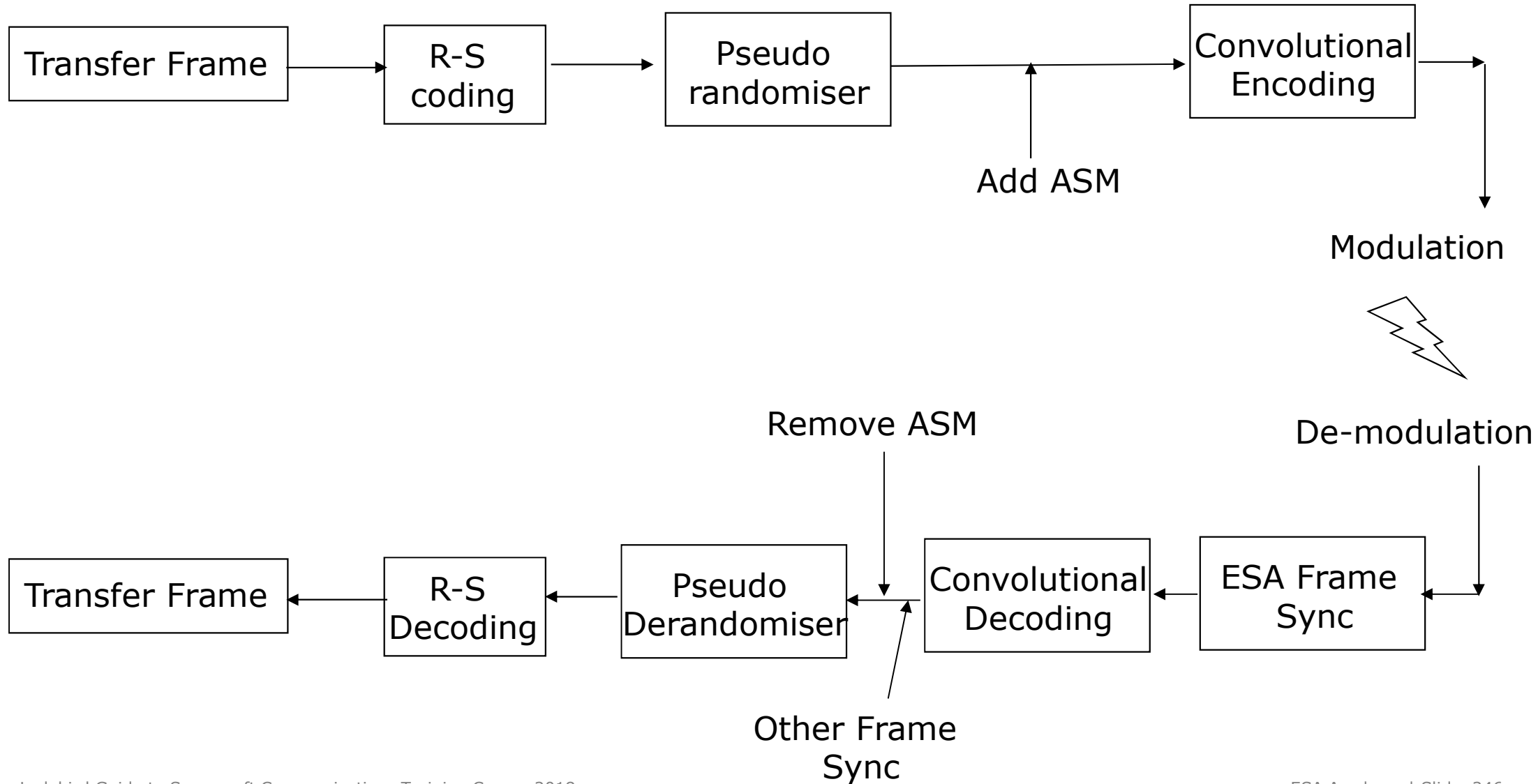
The following function is used to generate the sequence to OR the bit stream with where x starts counting at the start of the transfer frame after the ASM.

$$h(x) = x^8 + x^7 + x^5 + x^3 + 1$$

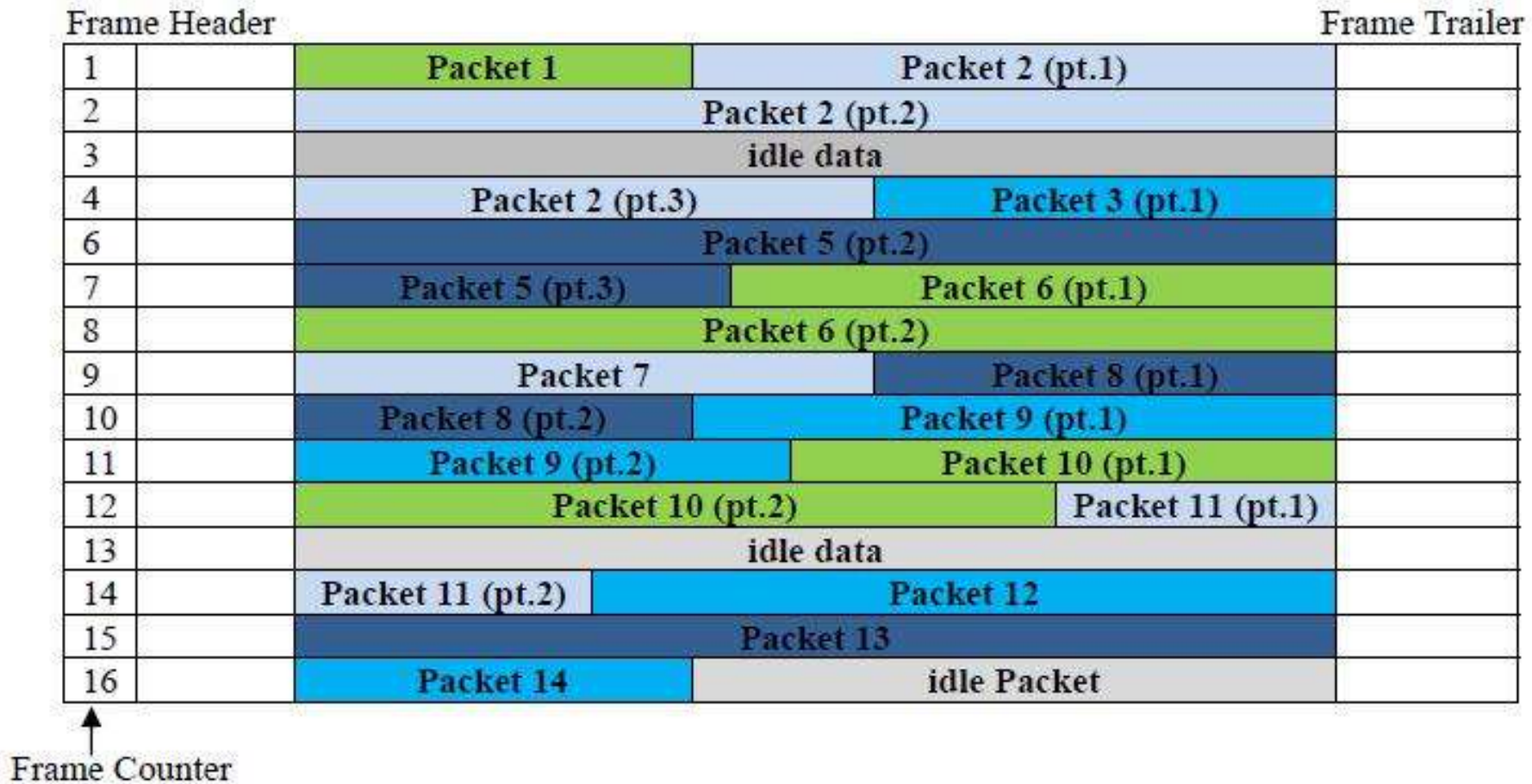


This is applied after the R-S (or turbo) encoding but before the convolutional coding. The reverse is applied immediately after the decoded ASM can be acquired in the decoding process.

Frame Sync/Pseudo Random - recap



Frame Synchroniser



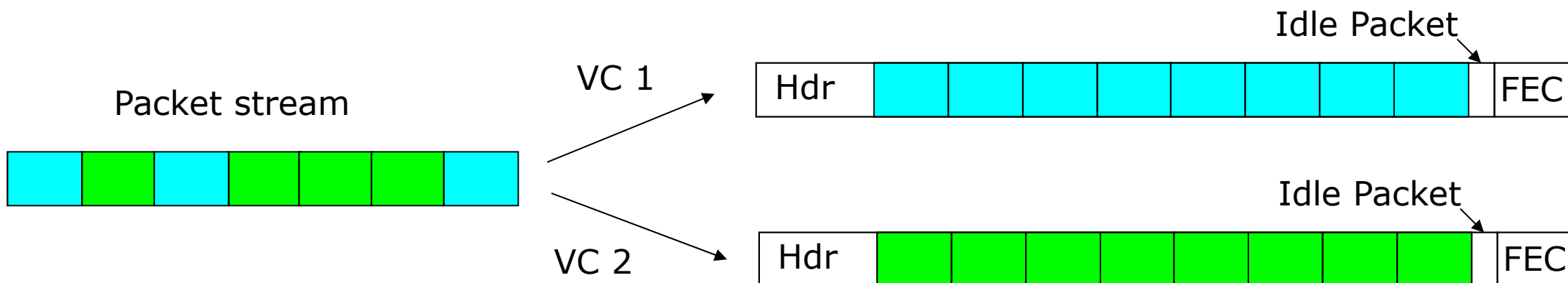
Source packets are grouped and placed in transport frames ready for transport.

Each transport frame is filled with packets that belong to one **virtual channel** only (e.g. HKTM real-time, science playback etc).

This is done to help processing the data on the ground.

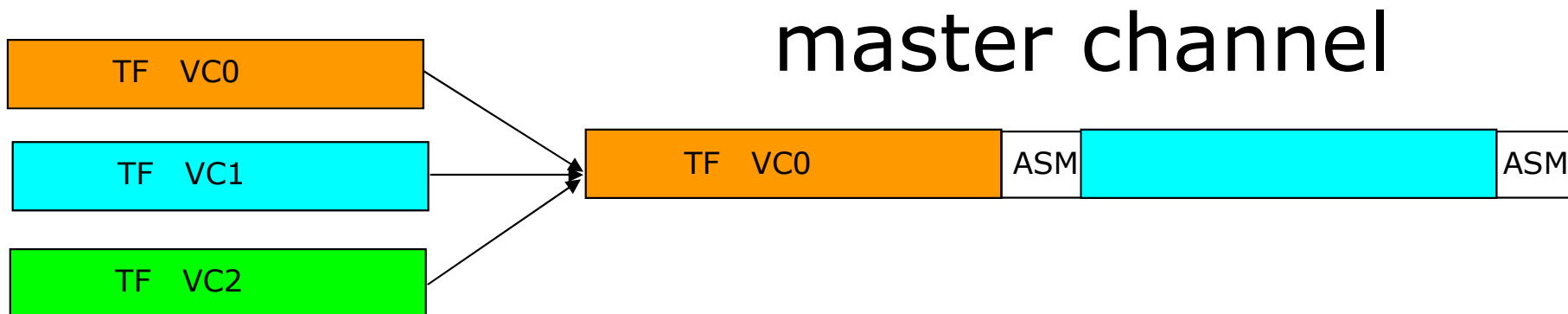
As packets are variable length and frames are fixed length it sometimes means that they will have to be filled artificially to get them to correlate. Idle packets are used for this.

TRANSFER FRAME BUILDING



All transport frames are then time-multiplexed into a single stream called the master channel.

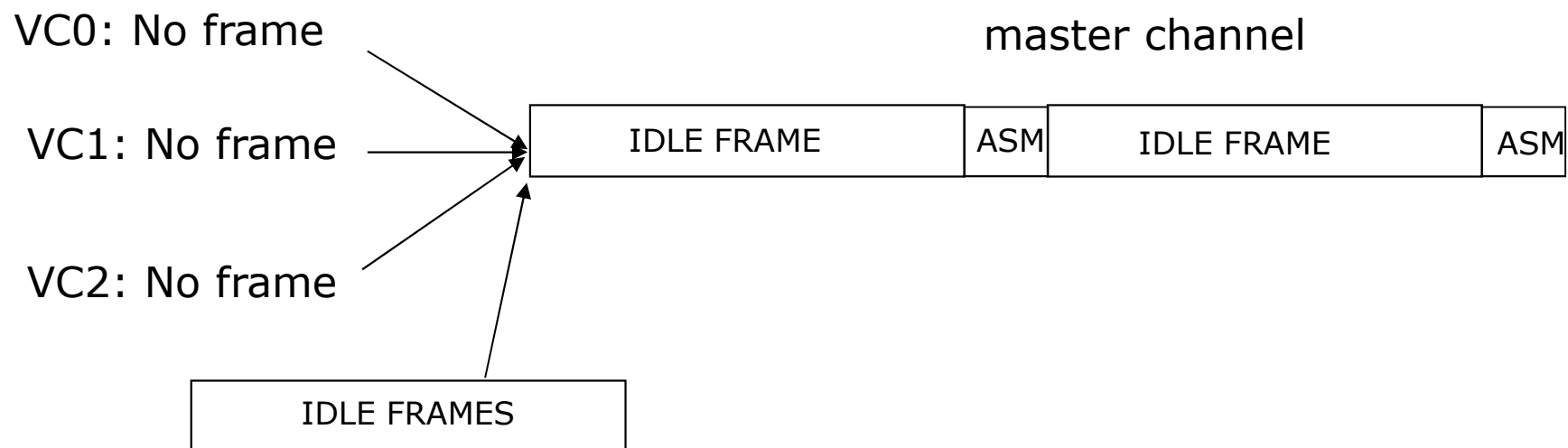
The start of each transport frame is signalled by an Attached Sync Marker (ASM). As the transport frames are always fixed length the ASM always appears at fixed places in the data stream. This is what the ground equipment uses to lock on to the downlink data stream and separate transfer frames.



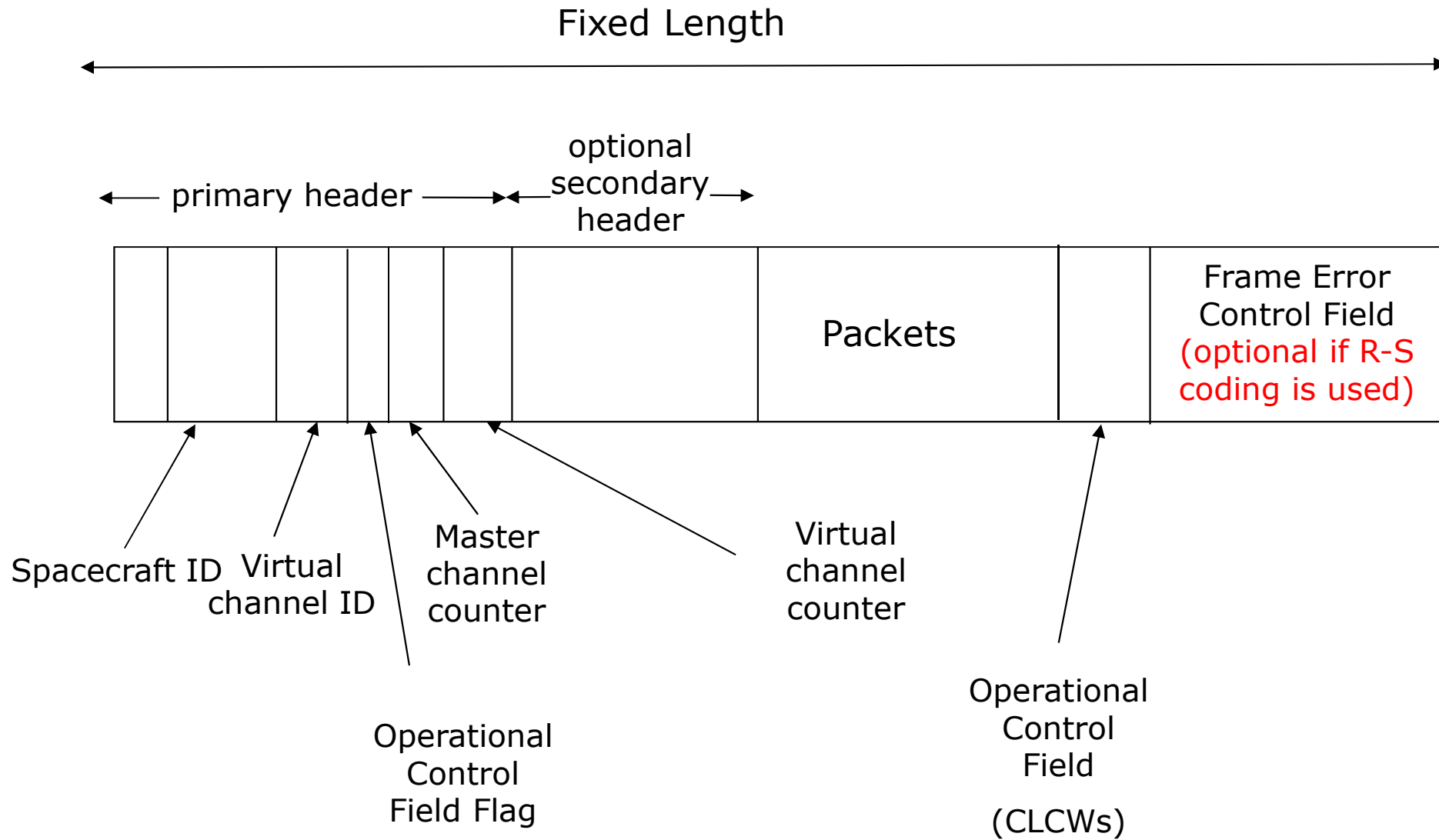
Synchronisation with the ground equipment is made simpler if the ASMs always appear at these regular intervals. This avoids having to re-synchronise again.

Therefore if the master channel has no data virtual channel transfer frames to send it until send an idle transfer frame. This is a transfer frame full of pseudo-random data. The correct name is Only Idle Data (OID) Frames.

Note OID frames may contain useful data such as CLCWs (see later).



TM Transfer frames



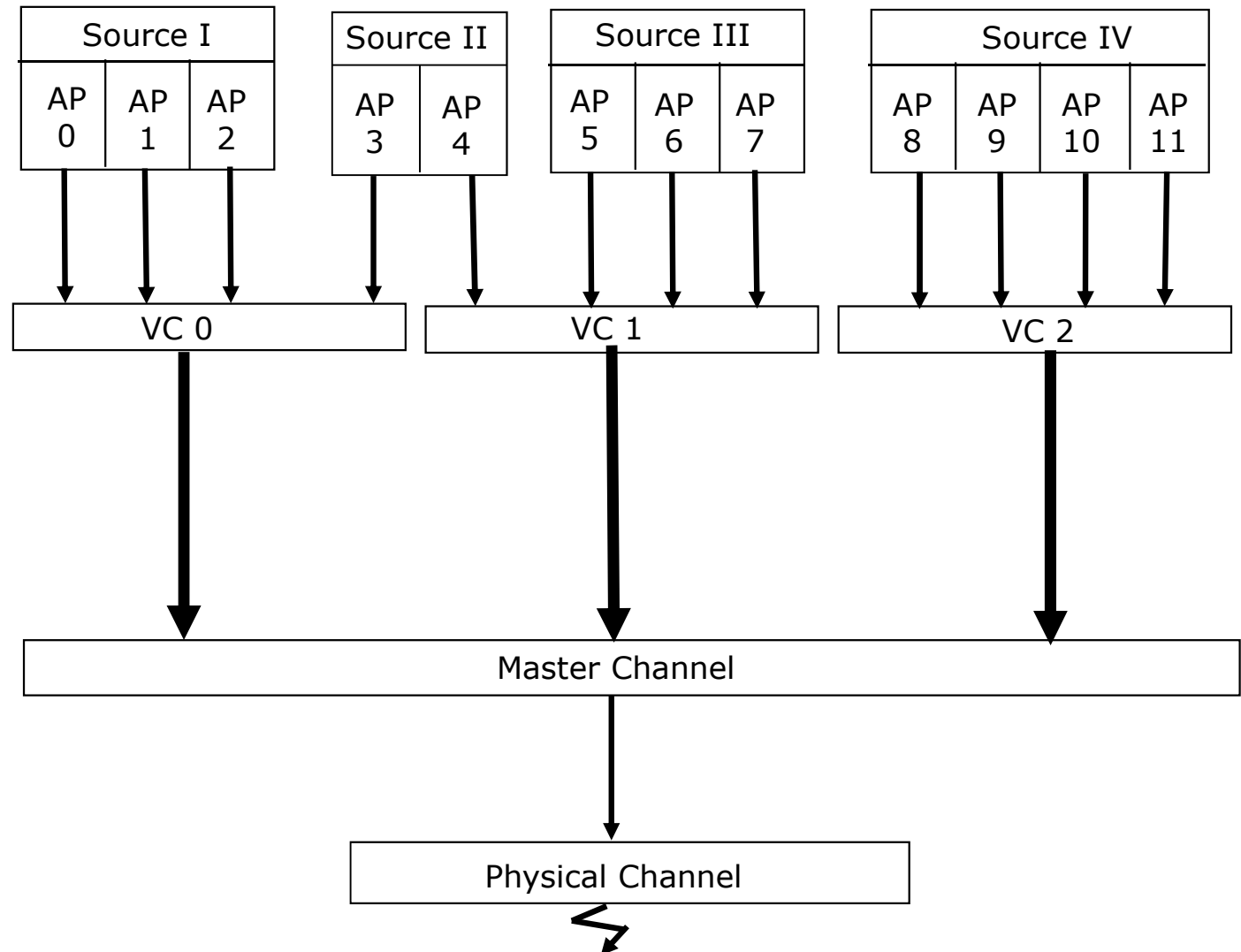
Telemetry data flow

Generate source packets

Multiplex source packets into Virtual Channels and Transfer Frames

Multiplex Transfer Frames into Master Channel

Apply coding and modulate RF



OSI Model

Application	Data moving between two applications
Presentation	
Session	
Transport	Protocols to move data between end systems
Network	End-to-End addressing, Routing, Forwarding
Data Link	Point-to-Point addressing, maps network to physical layers, media access (CDMA, TDMA etc)
Physical	Actual medium, RF, wire etc

OSI Model

Application
Presentation
Session
Transport
Network
Data Link
Physical

TCP/IP Model

Application
Transport
Network
Data Link
Physical

E-Mail, FTP,
Web Browsing, etc...

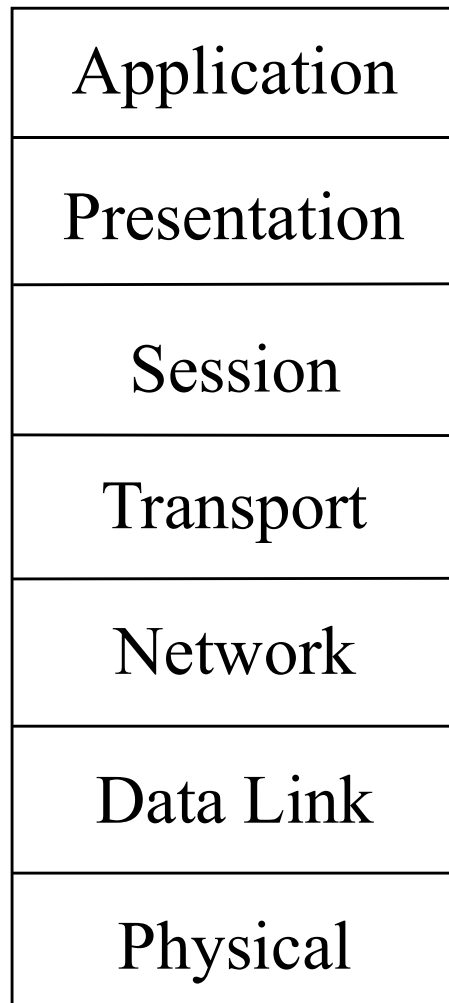
TCP, UDP, SCTP, etc...

IPv4, IPv6

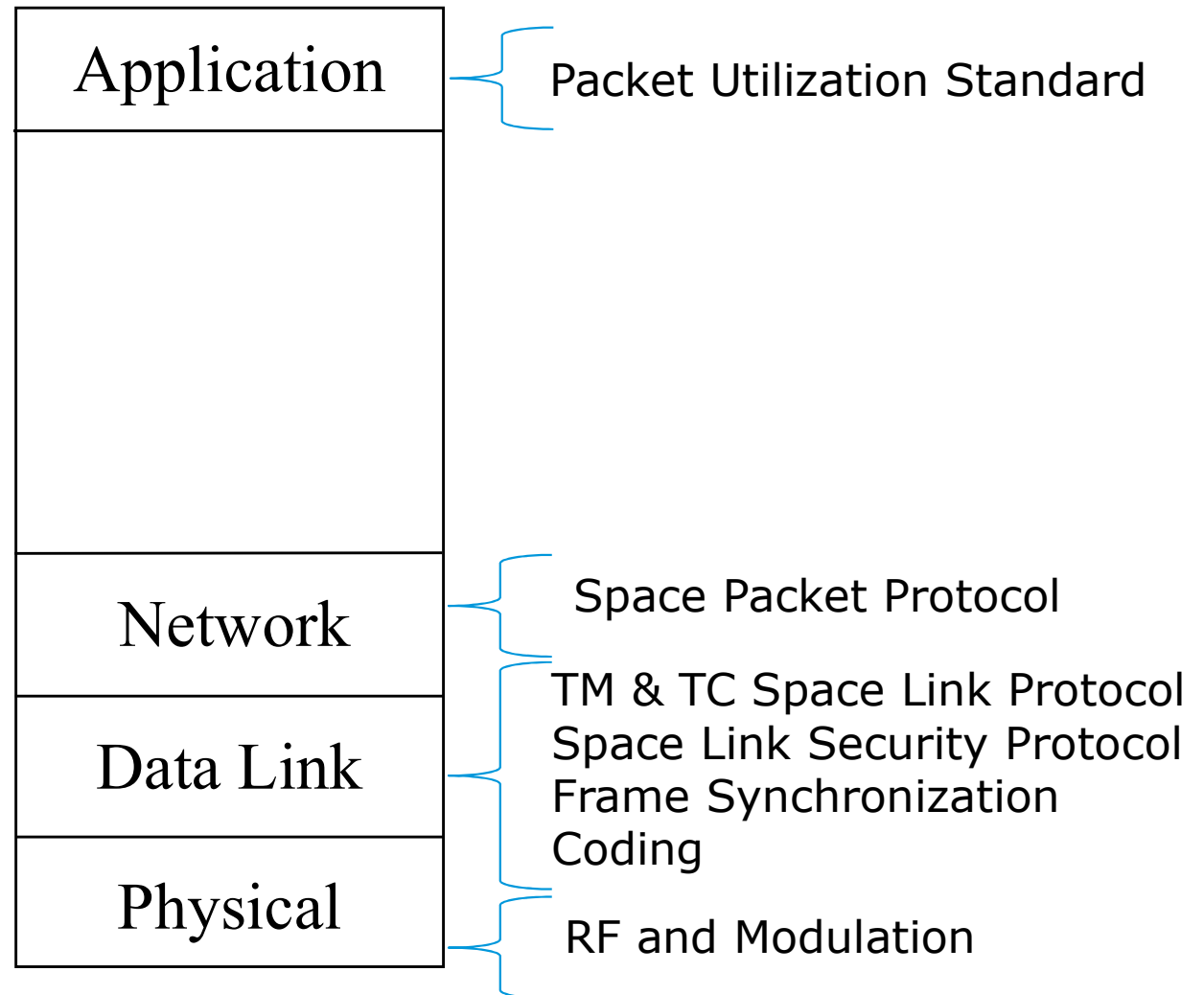
IEEE 802, SONET
Frame Relay (HDLC), etc...

Wire, Fiber, RF, etc...

OSI Model



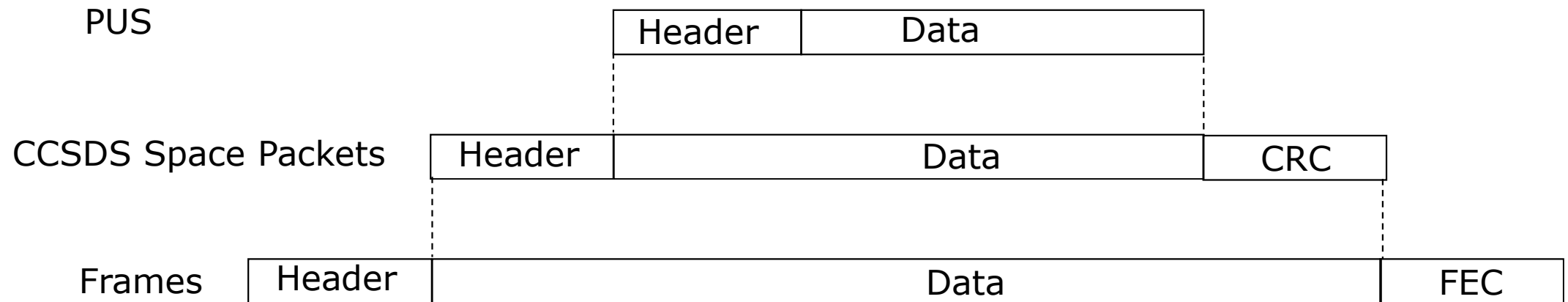
ESA Model



Not quite in line with the theory



THE THEORY



PUS packets are generated at source with associated CCSDS headers included.

The PUS part cannot stand alone without the CCSDS header information.

This is efficient but not ideal as the layers are mixed.

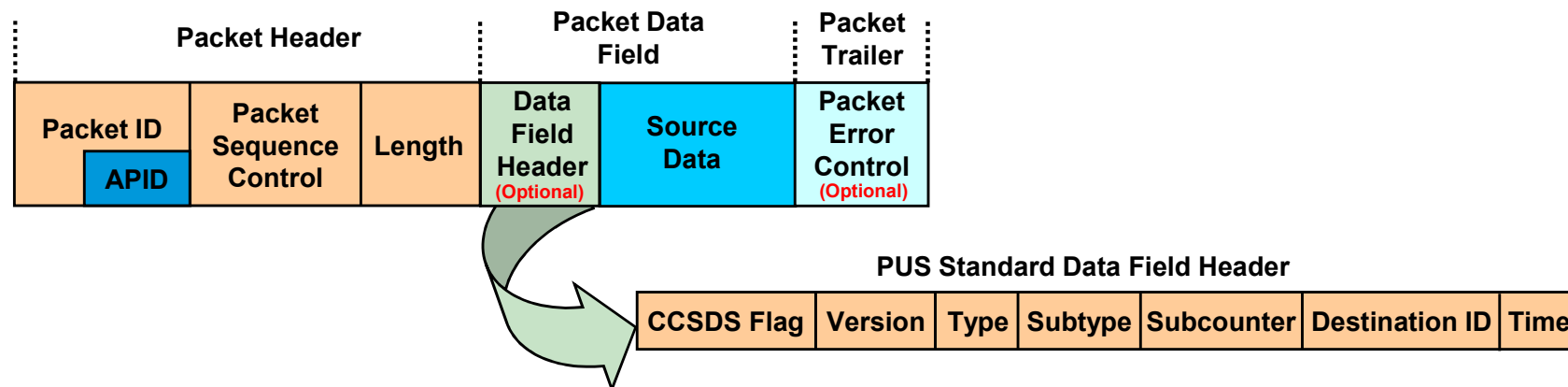
The CCSDS header contains a:

Number describing the source application, called the APID (application process ID). This could be a physical unit (e.g. a sun sensor) or a logical unit (e.g. thermal control).

Incremental counter (the source sequence count) counting packets from each APID.

Number describing the length of the packet data field as this is variable.

CCSDS TM PACKET



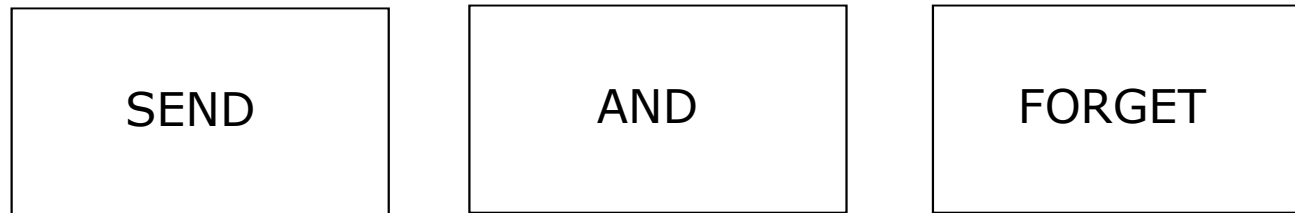
The PUS header will be described later.

Remember PUS TM is UDP like



It was decided that the TM standard **did not need** a mechanism to ensure complete delivery of all data generated by the spacecraft.

A simple transmission model without implicit hand-shaking dialogues for guaranteeing reliability, ordering, or data integrity is implemented.



If any extra services are required (e.g. completeness checks) they need to be done at application level.

However an optional error detection coding scheme is included.

Once the frames are available at the mission control system they are opened and the packets extracted.

By examining the packet headers the receiving application for each packet can be identified and the control system can route the packets that way.

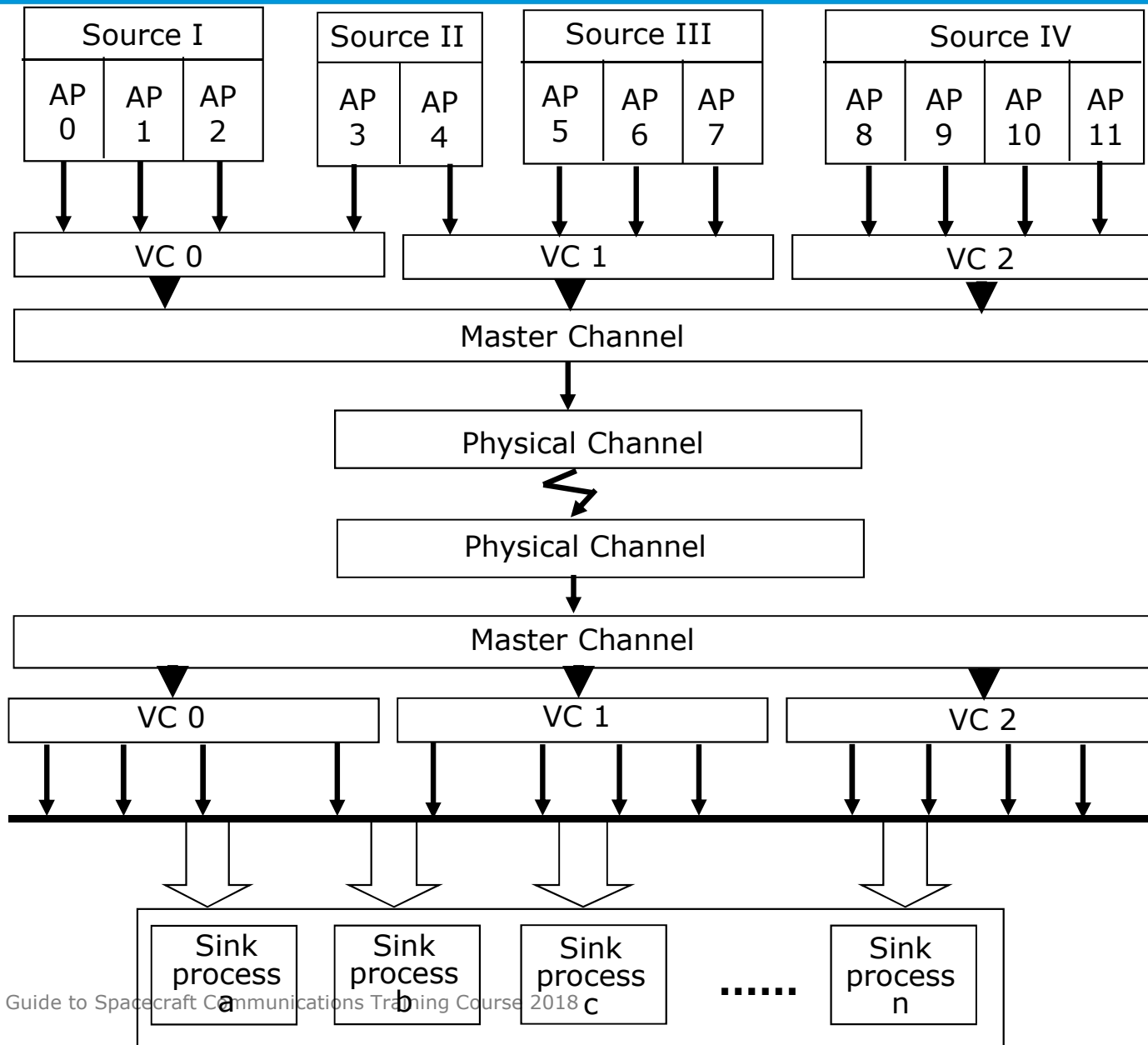
The data unit used between the spacecraft and the ground station is the transfer FRAME

At ESOC, the data unit used between the MCS and the ground station is the transfer FRAME

Hence you can only lose FRAMES

It makes no sense to talk about packet loss

Complete Telemetry data flow



Generate source packets

Multiplex packets into Transfer frames

Multiplex Frames into Master Channel

Code and modulate

Decode and demodulate

Demultiplex Master Channel into frames

Multiplex frames into virtual channels

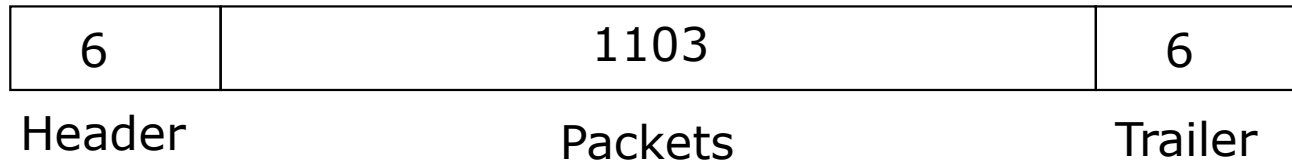
Transfer frames to MCS

Demultiplex packets and send to different applications

Typical Data Structures



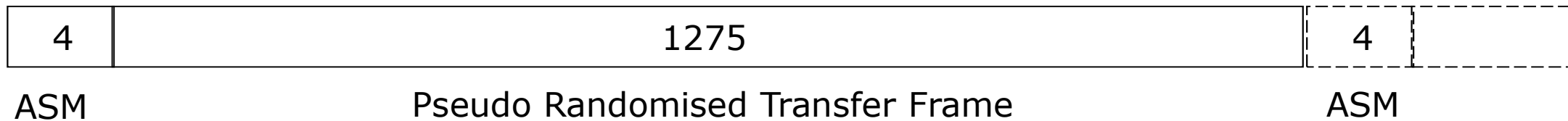
Typical Transfer Frame : 1115 bytes



Typical Transfer Frame after R-S encoding (255,223, I=5) : 1275 bytes



Typical Transfer Frame before convolutional encoding : 1279 bytes



Reed Solomon coding failed. Project confirmed that frame level checksums (CRCs), which could have been used to detect bad frames, could not be enabled onboard the spacecraft.

There was no choice but to disable the Reed Solomon checking at the ground station which could expose the MCS to bad telemetry. Nasty problems occurred when

1. the packet length got corrupted. SCOS would start allocating enormous amounts of memory to these mega packets and then run out!
2. The CLCW got corrupted, meaning commands could be resent in error!

The MCS software support team developed a mechanism to detect and file bad telemetry frames in a separate archive. An event message was also logged giving the reason. It was based on checking the values of fields in the frame/packet header that had known values.

It was accepted that such checks would be imperfect and bad data might still reach the main MCS telemetry processing chain.. but the new software performed very well indeed. At times significant periods of bad telemetry were received which the MCS detected, rejected and logged exactly as required.

Without Reed Solomon encoding it became important that downloads of SSMM stores were not scheduled at times when bad data may be expected e.g. at low elevation.

What is our "useful data rate"



Typical Transfer Frame : 1115 bytes

6	1103	6
Header	Packets	Trailer

Information rate is defined as the bit rate at the input of the first encoder.

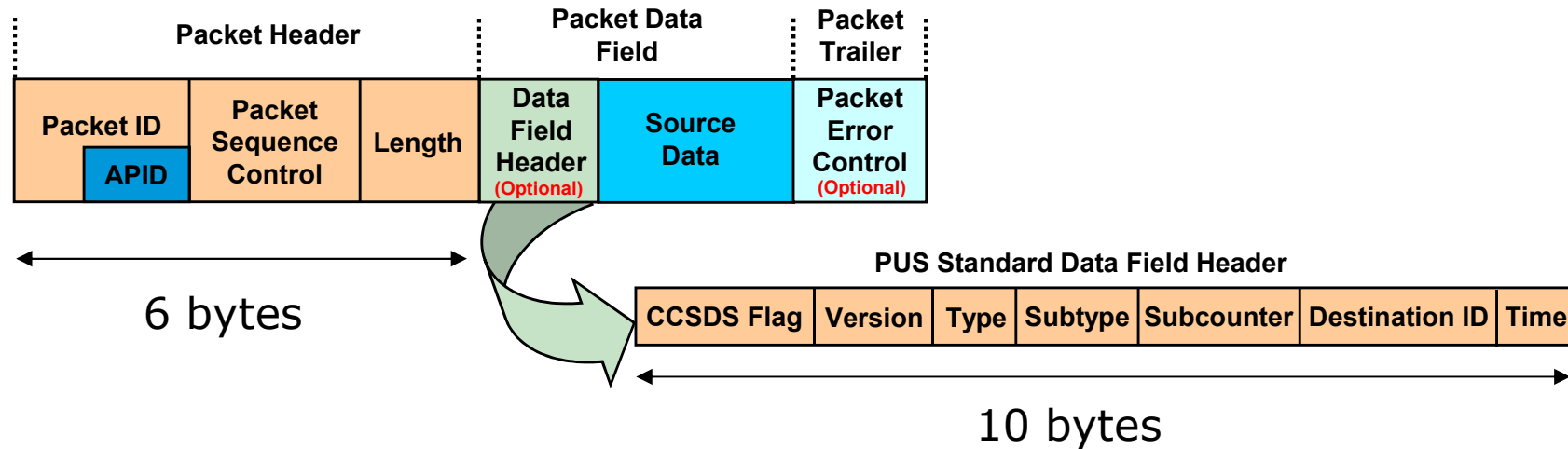
However it is not the useful data rate as this input is transfer frames.

There is still an overhead associated with frame headers and footers not covered. They have fixed headers and trailers of 6 bytes each.

The frame length depends on the R-S coding scheme. For a typical R-S coding scheme (255, 223, I=5, the transfer frames will be 1115 bytes long, i.e. a 1% overhead.

What is our "useful data rate"

CCSDS TM PACKET

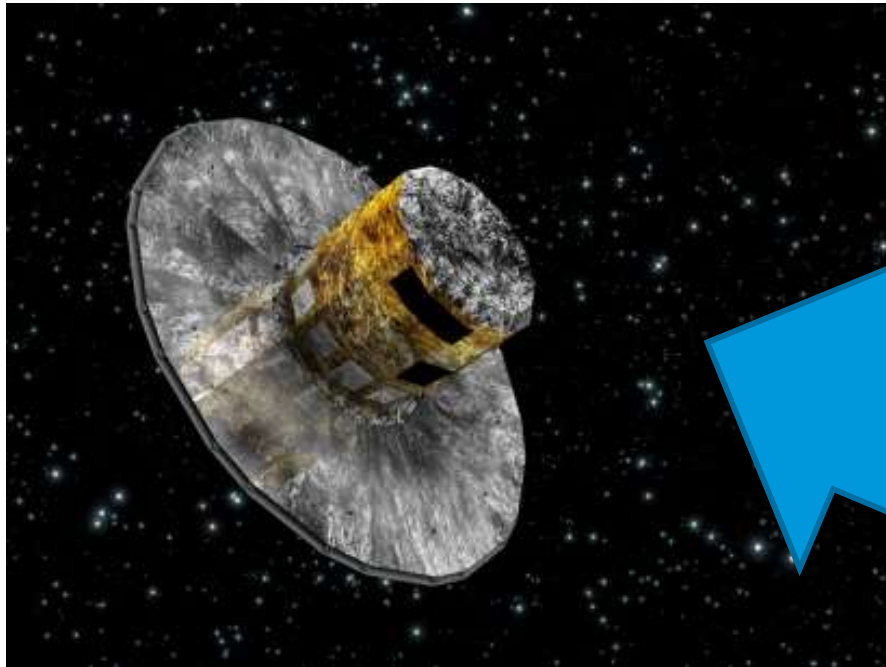


Still we are not at useful data rate because we must remember the frames are full of packets that also have headers and footers.

The overhead depends heavily on packet lengths (as these are variable and the headers and footers are fixed). However as a rule of thumb we can say for large packets that mostly fill the frame the overhead is about 1.3%. **Small packets are much, much worse.**

Remember also that at some point we will add a 4 byte ASM (which may or may not be convolutionally encoded i.e. doubled), again an overhead we should consider.

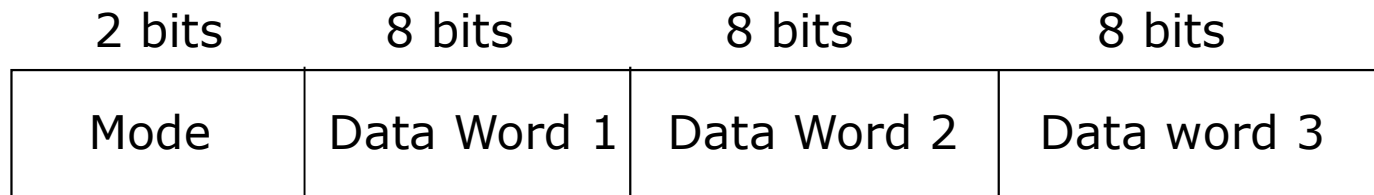
Telecommand Space Link Protocols



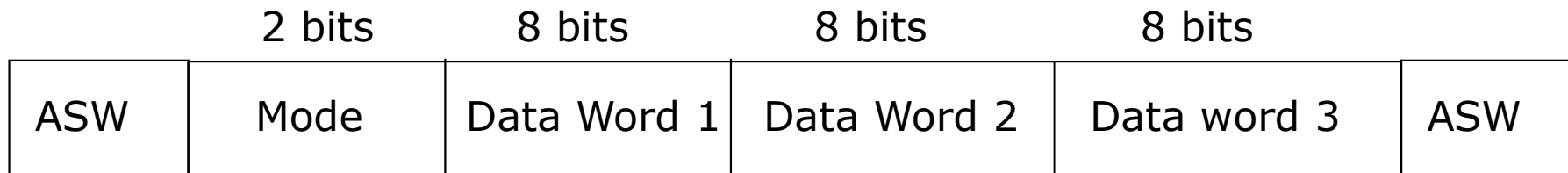
“Old” Telecommand Protocol



The ESA PCM telecommand standards used a basic telecommand format as follows;



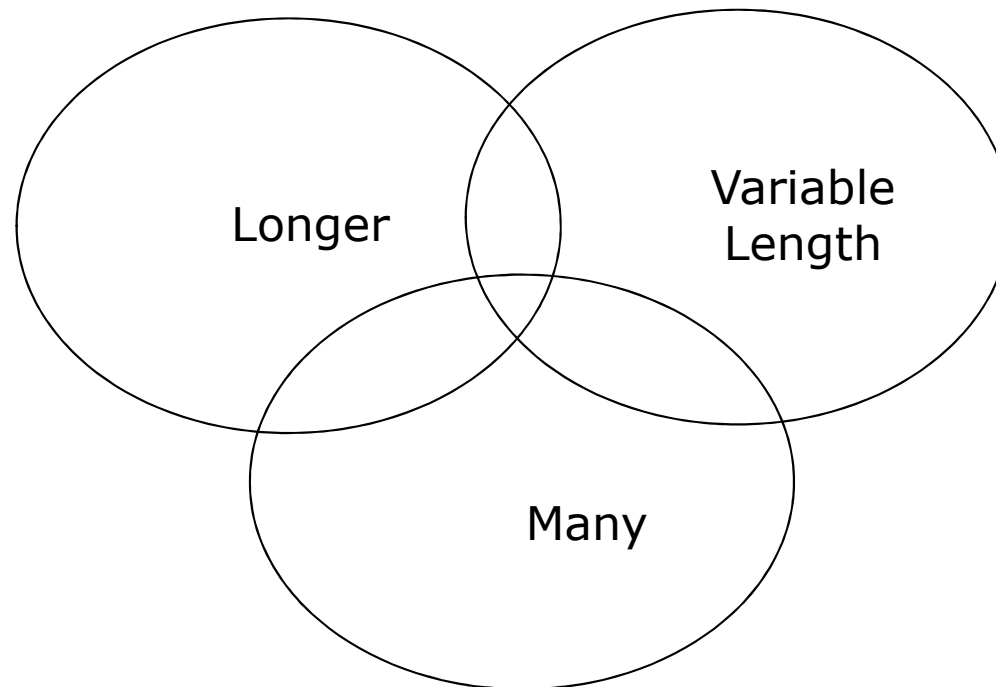
The ASW is then added from the user TC options and this all coded together to create the standard 96-bit ESA TC frame.



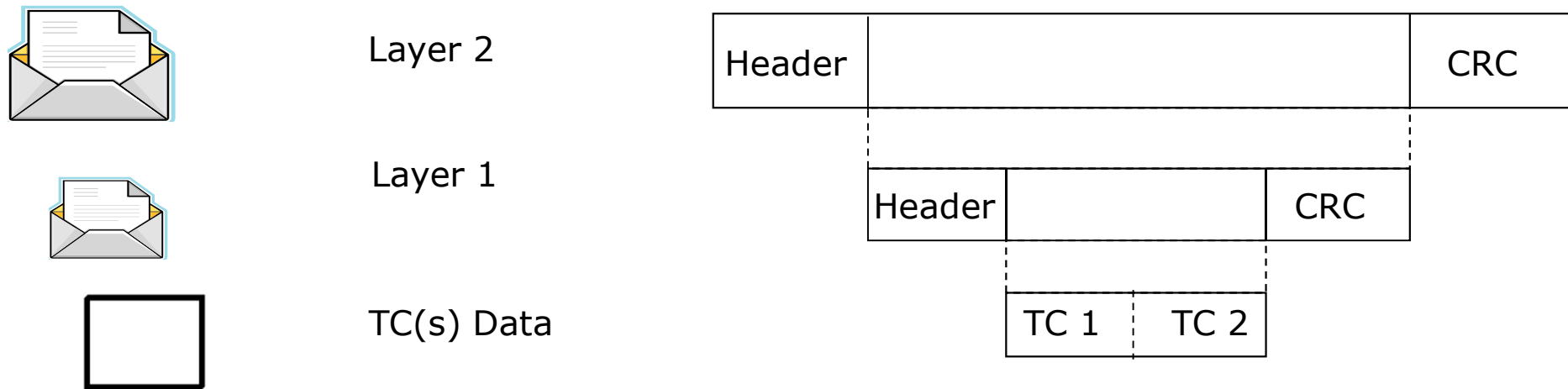
This system is simple because it always uses the same message format to transport the TCs but it is also inflexible for long commands as they need to be spread over several messages and re-constructed by the on-board DHS.

Then something changed. What?

The NATURE of control changed from being hardware based to software based. This meant much more **information** had to be transported rather than **switch commands**. The new telecommands had to transport commands that were:



Layers again



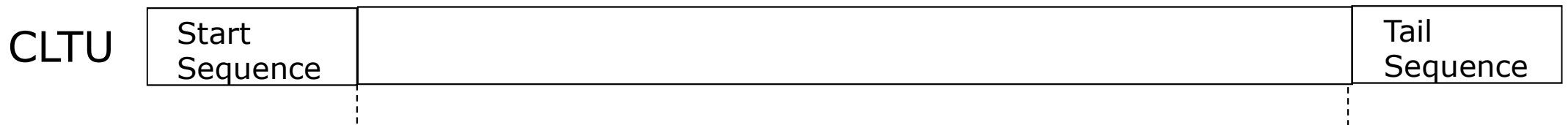
By using layers (view them as envelopes) we can address each bit of equipment in the transport system without changing the command data itself.

When an envelope arrives at the destination given in the header it is opened, checked and processed.

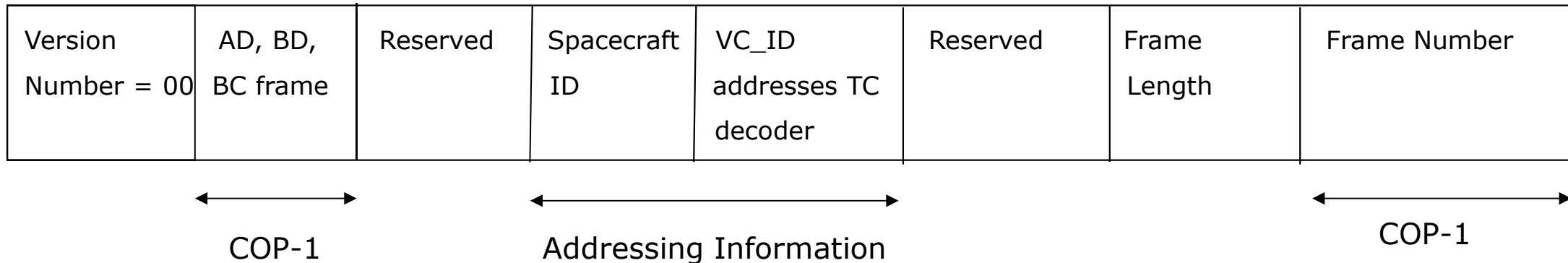
The outside layer is discarded and it is passed on to the destination given by the header of the inside envelope.

The last envelope (or layer) is opened to reveal the command inside.

The lowest layer is the Command Link Transfer Unit. It has a start and tail sequence so that the TC decoder can recognize a TC is being sent and has ended.



The next level is the TC transfer FRAME. This contains a header with spacecraft and virtual channel addressing information, frame length and COP-1 information (see later)



It also contains a trailer for forward error correction.

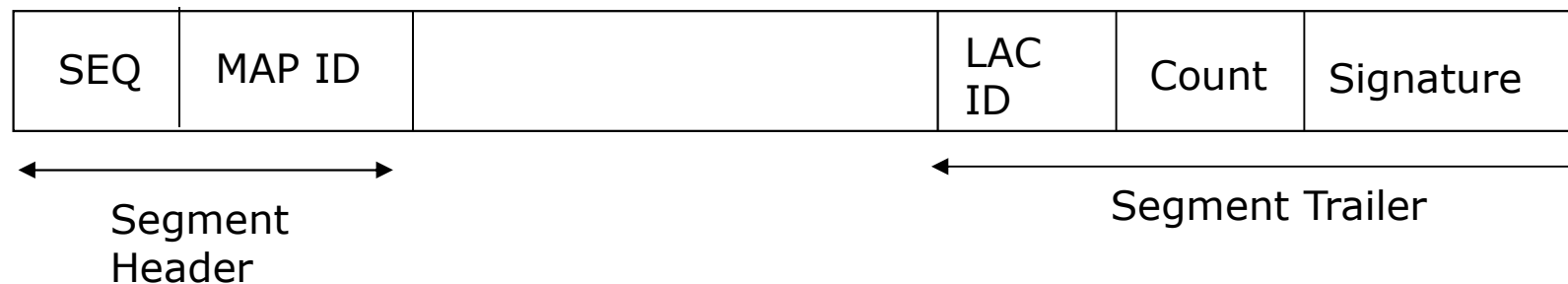
Note: we are running out of spacecraft IDs!! Version 1 only uses 10 bits for this field and version 2 only 8 bits. According to Gian Paolo Calozori in 2014 there were less than 27% for V1 and 13% for V2 of all possible values left.

The TC segment contains a header with

- Information on command routing inside the spacecraft (MAP ID), see Book 1
- 2 Sequence bit flags that tell the TC decoder if the frame contains a TC packet that has been split over different frames (01 = first, 00 = continuing, 10 = last, 11 = no split)

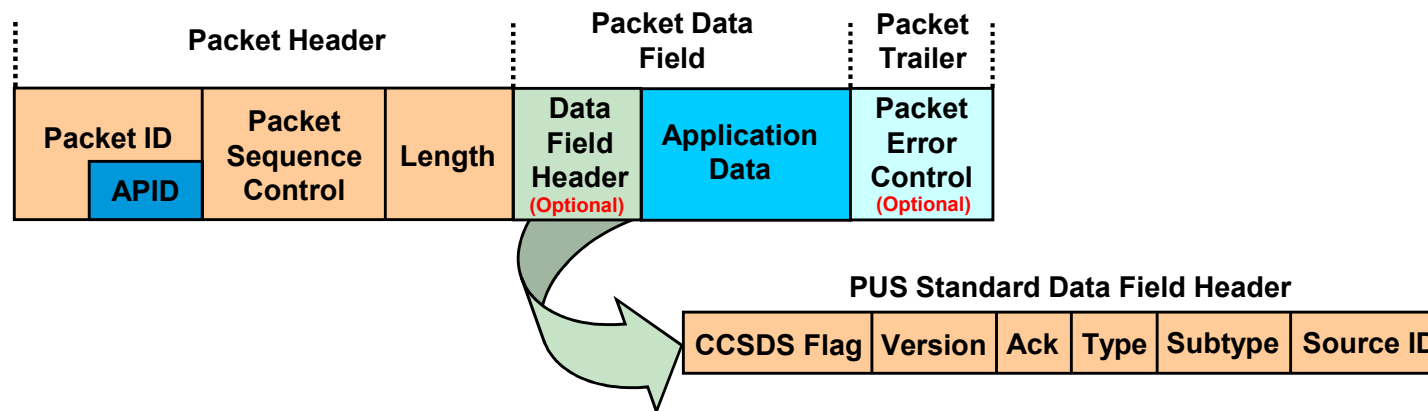
and a trailer with

- Password information to stop unauthorised users from commanding the spacecraft

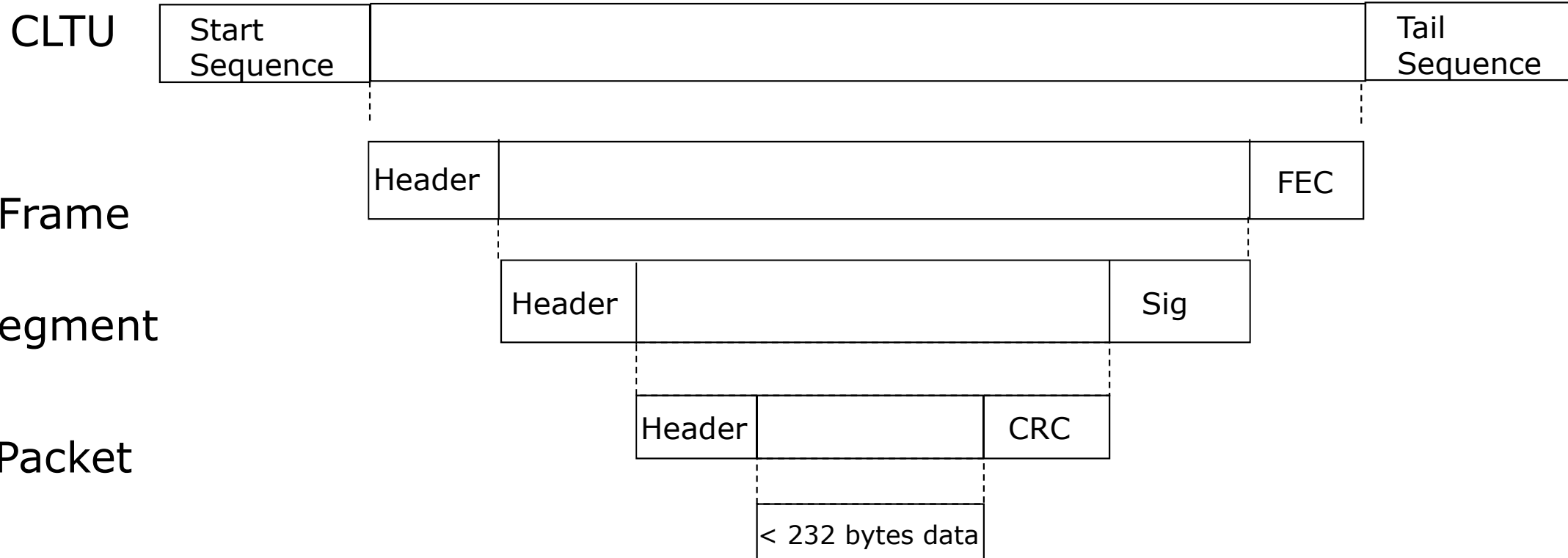


The final level is the TC packet itself

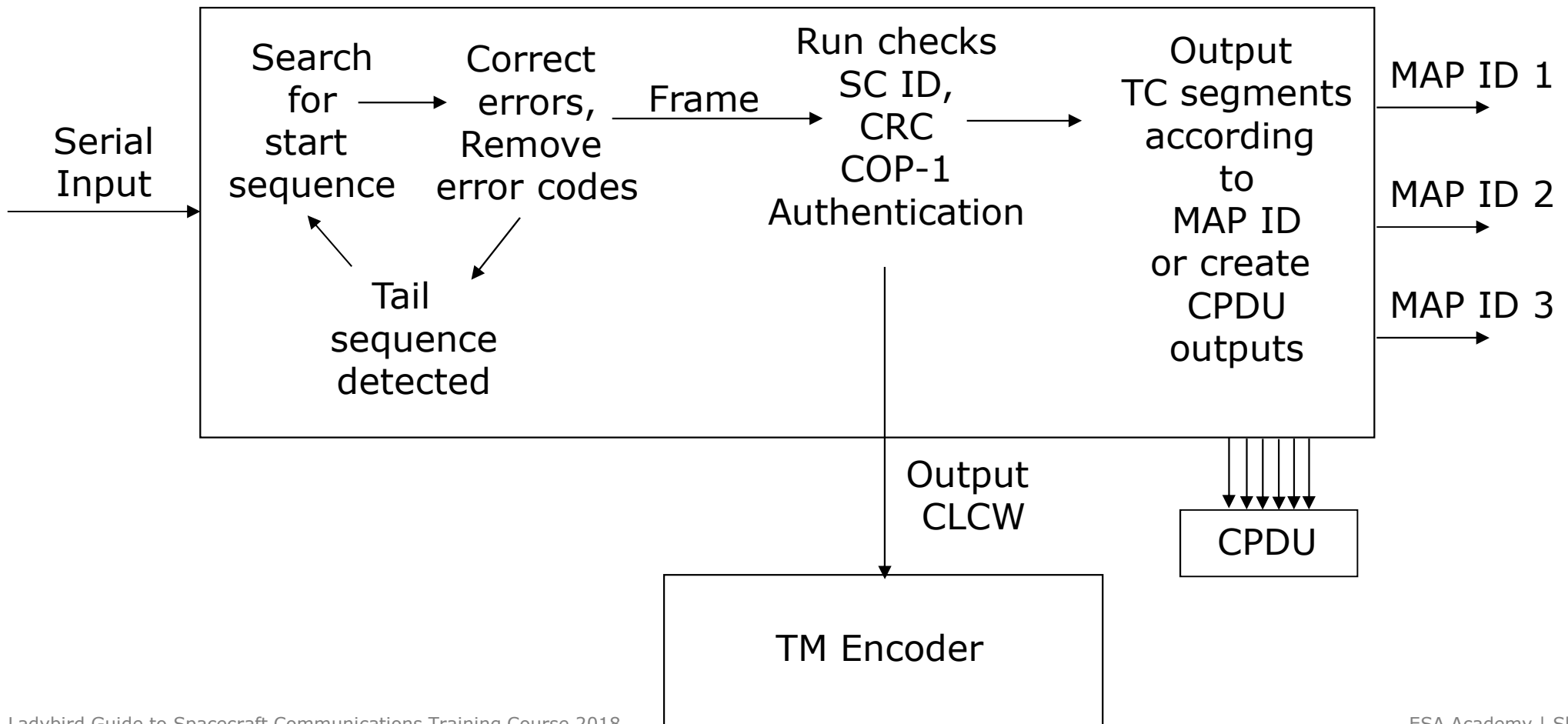
CCSDS TC PACKET



Telecommand Processing



ACQUISTION CHECKING DISTRIBUTION



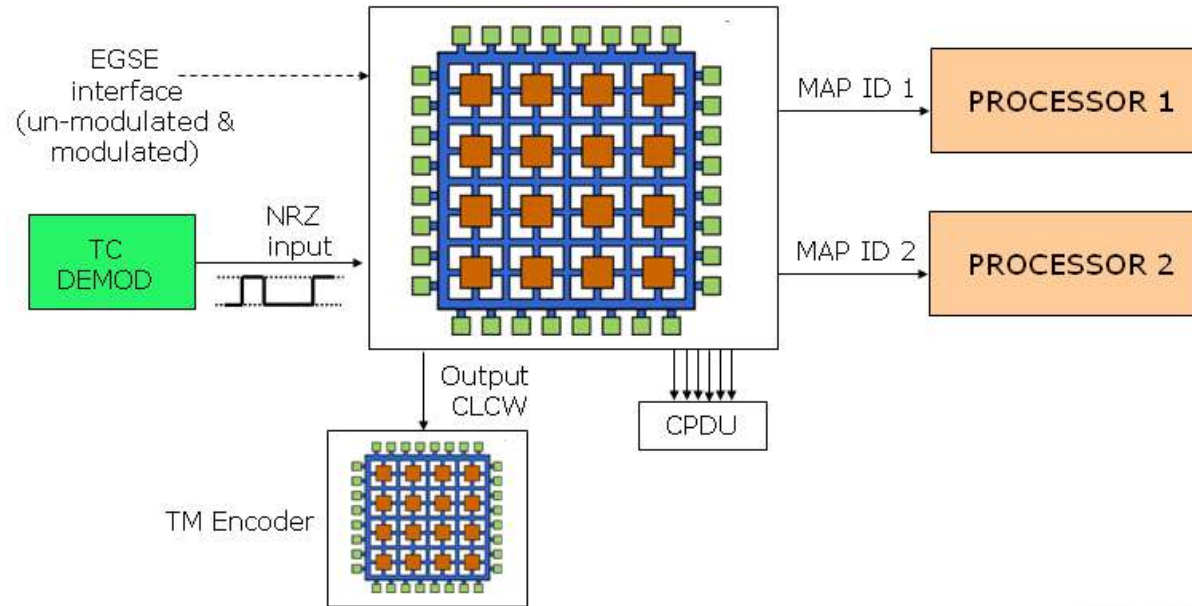
The telecommand packet standards have also meant that other services are available for improving the security of the ground station to spacecraft link:

- 1) Guarantee of command order and automatic requests for retransmission in the event of a TC loss, (called COP-1)
- 2) Automatic protection against unauthorized commanding by another party (called AUTHENTICATION)

Both features use a feedback loop between the commanding processes and the spacecraft telemetry.

The feedback loop involves the CLCW on the telemetry side and the TC frame header and segment header/trailer on the telecommand side.

The Command Link Control Word



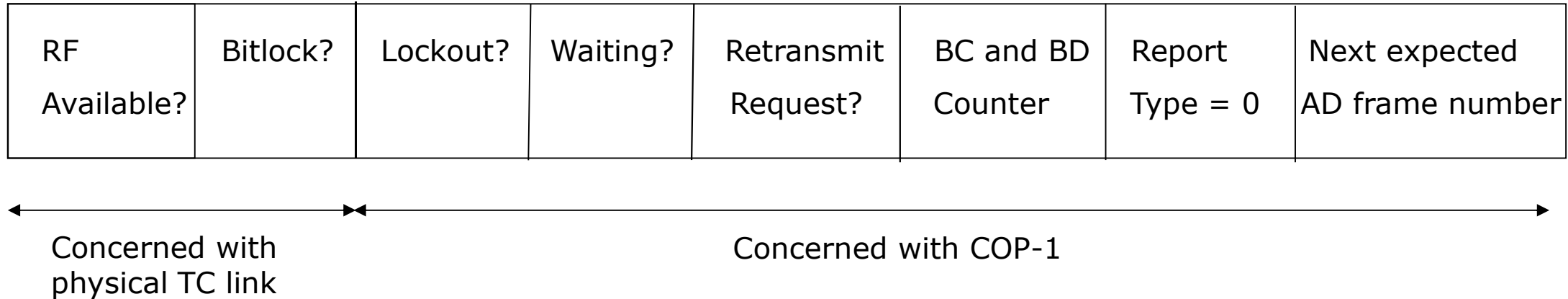
European Space Agency

The satellite telemeters the state of the on-board command reception process principally using the CLCW (Command Link Control Word).

This is written at the end of every spacecraft telemetry frame directly by the telecommand decoders in the operational control field (see earlier TM frame discussion)

Each virtual channel has its own independent CLCW which are written into telemetry.

CLCW contents



The first part telemeters whether the satellite is physically able to receive a command.

The ground will not command if the carrier is not locked or the bit synchronizer (DTTL) has not locked.

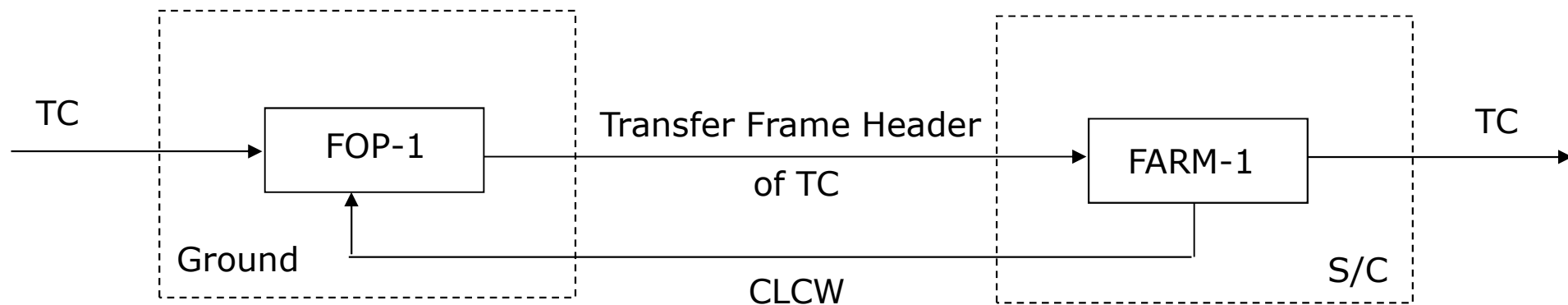
The second part telemeters whether the decoder has accepted the telecommand. It primarily deals with the on-board part of COP-1, see later.

Commanding Operations Procedure (COP-1)

QUIZ: Design a simple protocol that ensures commands arrive in the correct order.

COP-1 is split into a process on the ground called the Frame Operations Procedure (FOP-1) and a process running in the TC decoders called the Frame Acceptance and Reporting Mechanism (FARM-1).

These two processes talk to each other using the transfer header of the packet TC and the CLCW, essentially doing the same as our simple system.



Every TC frame is numbered sequentially by FOP-1 and this is written in the Transfer Frame Header.

The FARM process also has a counter which tells it the number of the last TC frame accepted and therefore the number of the next one expected.

This is telemetered to the ground in the CLCW.

If FARM-1 receives a frame number that is slightly more than expected it requests FOP-1 via the CLCW to retransmit everything again from the lost command onwards.

If it is slightly less than expected it simply rejects the frame.

If it is a long way off than it assumes FOP-1 is malfunctioning and enters the LOCKOUT state. It will not accept anymore commands until the ground sends a special UNLOCK command.

AD, BC and BD frames



When operating using COP-1 the transfer frames are said to be of type AD (“D” signifies that they contain TC data).

To control COP-1 type BC frames are used (“C” signifies that they are for control only).

Only two BC commands are available to control COP-1 on-board. They are the UNLOCK command and the command to set the expected frame number of FARM-1 to a given value.

Of course sometimes it may be necessary to command without telemetry. With no CLCW available, COP-1 does not work and must be bypassed.

To do this commands can be sent in frames of type BD. There is no guarantee that the commands will arrive in sequence if this option is used.

The Handshakes



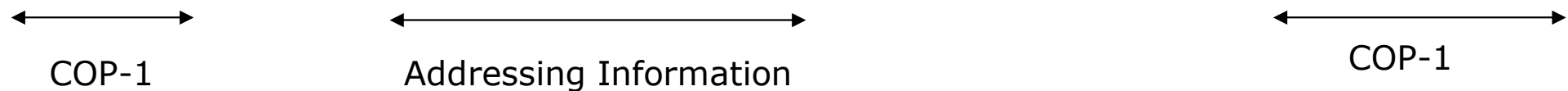
The CLCW

RF Available?	Bitlock?	Lockout?	Waiting?	Retransmit Request?	BC and BD Counter	Report Type = 0	Next expected AD frame number
---------------	----------	----------	----------	---------------------	-------------------	-----------------	-------------------------------



The TC Transfer Frame Header

Version Number = 00	AD, BD, BC frame	Reserved	Spacecraft ID	VC_ID addresses TC decoder	Reserved	Frame Length	Frame Number
---------------------	------------------	----------	---------------	----------------------------	----------	--------------	--------------



Note that FOP-1 can be implemented in the control system or the ground station software.

When a request for a command retransmission is received FOP-1 flushes the TC buffer at the ground station by remote command.

It then resends all commands from its own buffer from that number requested.

This means there are buffers of commands sitting around in various parts of the commanding chain.

I know of two independent systems in which a combination of events/bugs led to these buffers being transmitted to the spacecraft out of sequence or when not requested.

At Frame Level:

- Transfer Frame Version No ("00")
- Operational Control. Field Flag ("1")
- Transf. Frame Second. Header Flag ("0")
- Synch. Flag ("0")
- Packet Order Flag ("0")
- Segment Length ID ("11")
- Frame length: reject a frame larger/shorter than the standard SMART-1 frame size
- Control Word Type ("0")
- CLCW version ("00")
- Status Field ("000")
- COP in Effect ("01")
- Virtual Channel Identification ("000000" for VC0, "000001" for VC1)
- Spare ("00")

At Packet Level:

- Version number ("000")
- PUS version ("000")
- Spare field ("0000")

Authentication

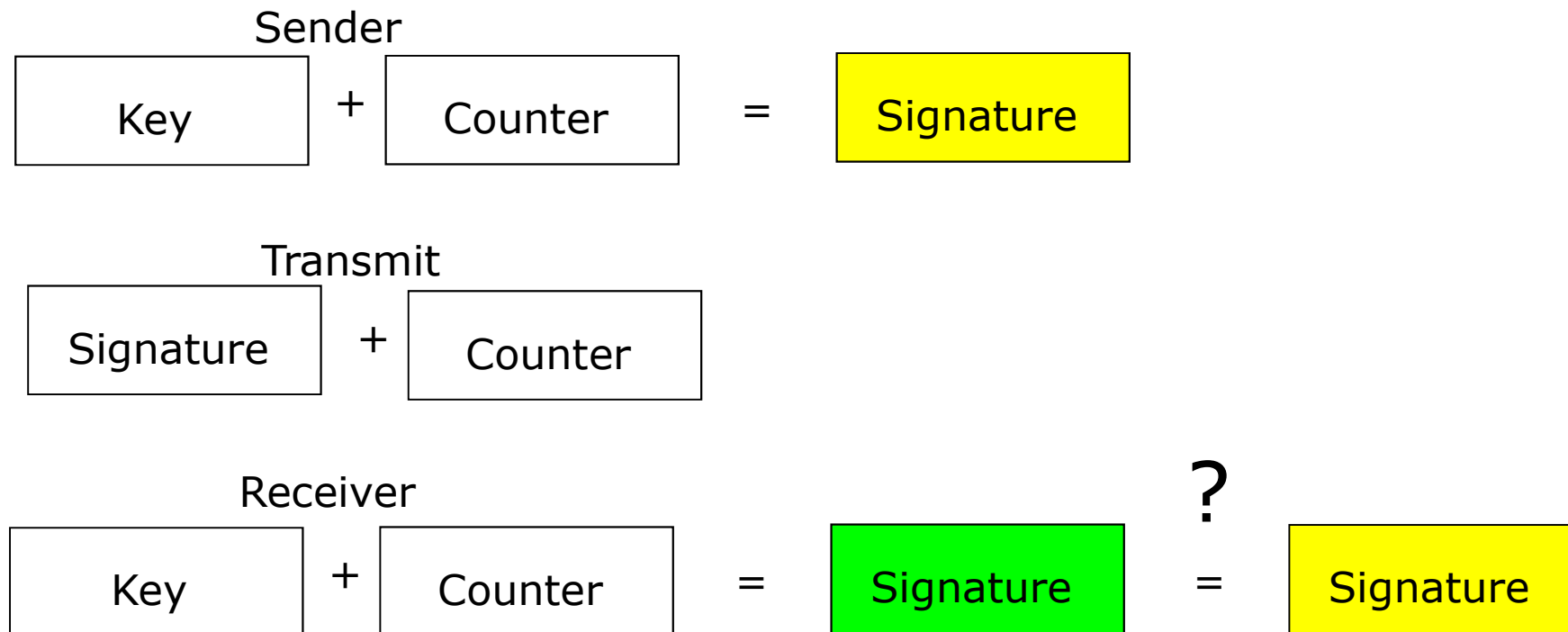
This system ensures that only telecommands from authorized users will be accepted.

The TC is not encrypted – but the option now exists in CCSDS.

A signature is calculated using the TC data, a “secret key” and a counter.

An on-board authentication unit will compare this signature to one it calculates on-board.

If they are identical it authenticates the TC, if not, it rejects the TC.



It is strange to consider that authentication is a recent service. For many years satellites worth hundreds of millions of euros were only protected with very weak, simple, unchangeable passwords.

The introduction of counters means someone recording a TC transmitted from the ground station cannot resend it successfully later.

As the counter increments after each TC is authenticated the signature on anything recorded will no longer be valid.

Note that the use of counters also means that the process should run after COP-1 has ensured a correct reception order, for this reason the signature is attached to the tail of the TC segment which is inside the Transfer Frame.

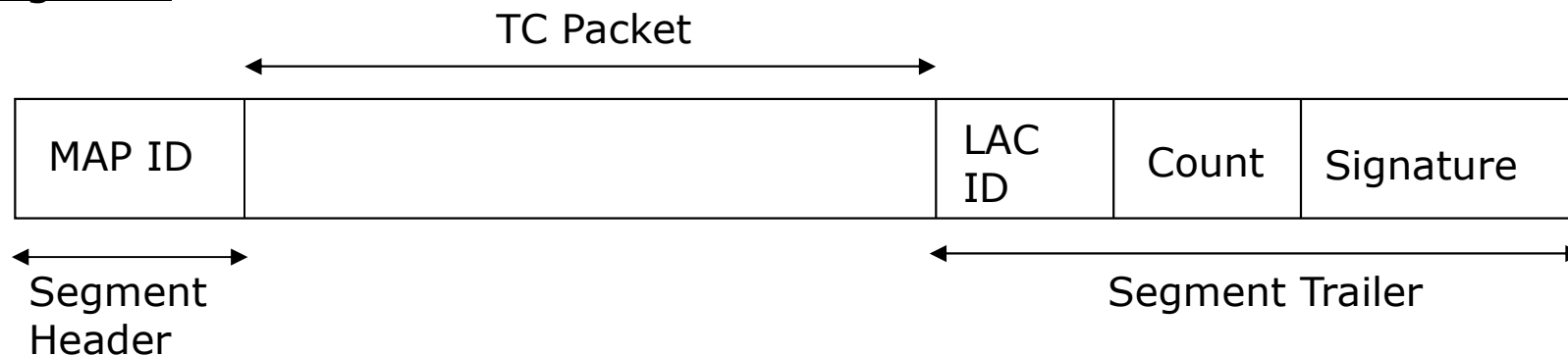
It is possible to deactivate the authentication process but only by using an authenticated command. This is why there are lots of counters and keys!

There are three counters which can be used allowing the possibility of several parties commanding the satellite independently without needing to synchronize their counters.

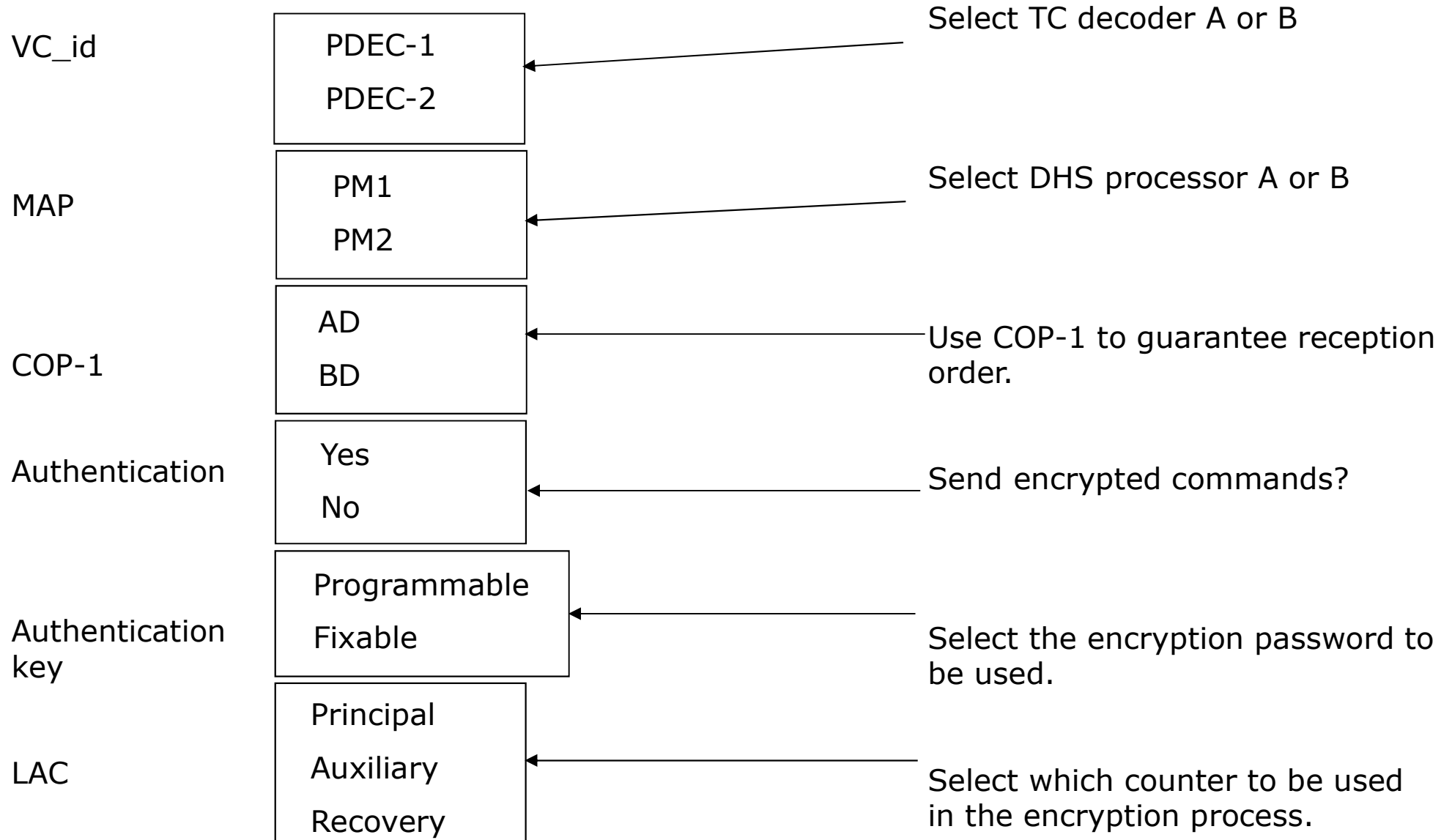
The values of each *LAC counter* (*Principal, Auxiliary and Recovery*) are written in telemetry every format. It is possible to generate signatures using any counter.

There is also two secret keys. One is programmable by the ground and one, which can be used for recovery, is fixed in the decoder PROM. The key used by the on-board authentication unit is selectable by ground command. These must be sent using the key to be selected.

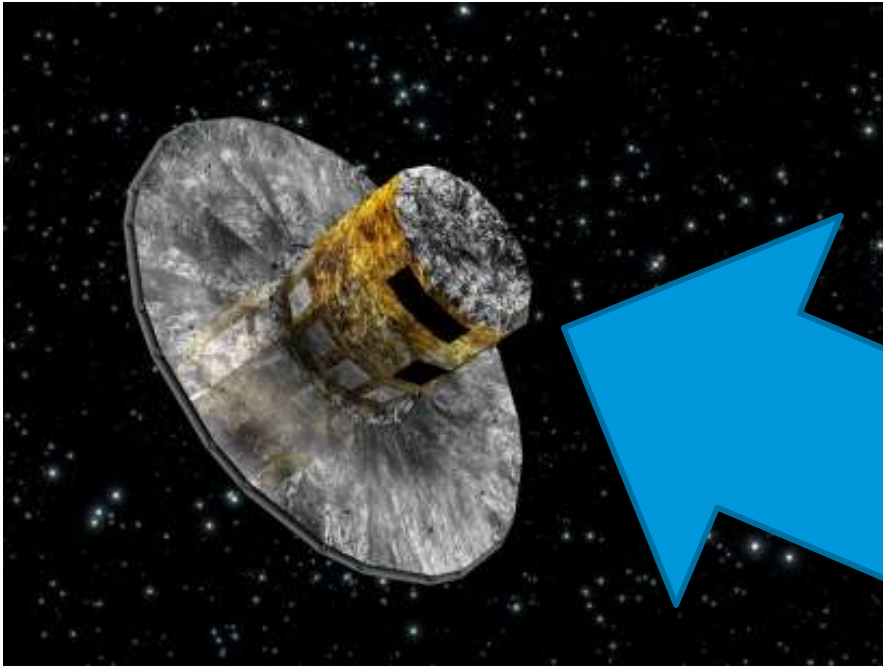
TC segment



Typical Packet TC protocol options



Application Level Protocols



An ESA standard not an CCSDS one. Careful when talking to the Americans!

There are many generic things that the ground will want a spacecraft to do.

PUS standardises the messages that must be exchanged between the ground and the OBSW when requesting these things.

Not all of the operations concepts described in the PUS are necessarily applicable to all missions, but are common to several missions.

The advantage of PUS is a standardised interface between the ground and OBSW. This should in theory lead to well tested libraries of OBSW and standardised ground software becoming available.

Standard PUS Services



No	Name
1	Telecommand Verification
2	Device Command Distribution
3	Housekeeping and Diagnostic Data Reporting
4	Parameter Statistics Reporting
5	Event Reporting
6	Memory Management
8	Function Management
9	Time Management

No	Name
11	On-board Operations Scheduling
12	On-board Monitoring
13	Large Data Transfer
14	Packet Forwarding Control
15	On-board Storage and Retrieval
17	Test
18	On-board Operations Procedures
19	Event-Action

Messages shall be produced by the spacecraft which indicate the **positive or negative acceptance and execution** of each command.

- These messages will contain enough information to identify the command, source and the reason for any failure.
- They will be generated with ten seconds of the corresponding event.

High Priority Commands (i.e. **direct**) will be provided to ensure commanding capability cannot be lost due to the failure or lock up of a piece of data handling equipment.

- They must work without the involvement of on-board software (usually directly from the TC decoder).
- Two independent routes for each command shall be provided.
- High priority command **functions** will also be commandable via the on-board processor.

A system of periodic reporting for **telemetry parameter values** or **memory address contents** will be provided.

- The information will be grouped and organised into different messages.
- The ground will be able to control this by sending telecommands to change the message contents, generation rates and to create new messages.
- There shall be a default set of messages and generation rates stored on-board.

All **events of operational significance** (progress of long duration process, on-board autonomous actions, errors etc) shall be accompanied by the generation of an unambiguous, **uniquely identifiable message**.

- The ground will be able to control this by sending telecommands to disable/enable the automatic generation of specific message types.

It shall be possible to **load** any changeable area of memory. It shall be possible to **dump** any area of memory.

- The system shall be simple, constraint free and based on static physical addressing.
- It shall be possible to request an automatic check of the checksum of a defined area of memory by telecommand and produce a report message.

Any on-board software application functions **not covered by the other services** should interact with the ground using a standard message set.

- This should include messages to start, stop, suspend, resume or pass parameters to the application.
- It should be possible to request a message containing these function parameters at any time.

You can effectively do anything with this!

Any on-board units that generate **messages** should use a time source synchronised with the master clock to **time stamp** them.

- It shall be possible to set the onboard time or add/subtract a delta time from it.
- The current on-board time should be regularly downlinked.

An on-board mission schedule shall be provided in which commands can be tagged to execute at a **predefined time or orbital position**. All commands except high priority ones should be able to be inserted.

- Ground shall be able to logically group commands in the schedule.
- Ground shall be able to enable/disable individual commands, logical groups or the whole schedule.
- Ground shall be able to insert, delete and append commands without stopping the schedule.
- Ground shall be able to add or subtract a delta time to all the commands in a specified part of the schedule.
- Ground shall be able to request summary messages detailing either the telecommands identifiers plus time/orbital positions for all commands or simply all the information for a particular command in the schedule.

Commands that are either disabled at execution time or have expired will not be executed, removed from the queue and a report message generated.

Different message types can be configured for routing **to on-board storage and/or real-time transmission.**

- Ground should be able to change this configuration at any time.
- A default configuration should be provided.

On-board storage organisation will allow logical grouping of the information.

- A priority system will be provided to ensure that the on-board storage is shared, efficiently managed and that each logical group is guaranteed a minimum quota of storage space.
- Different rules for data overwrite will be implemented.
- The ground will be able to configure the logical grouping; minimum quotas; priority and data overwrite rules.
- Ground shall be able to request report messages detailing the configuration.
- **The information will be stored as efficiently as possible.**

Ground will be able to **select messages for dumping to the ground in a flexible way** (as a minimum based on logical groups, generation time, age, and not yet dumped criteria).

- According to a configurable priority scheme information from several logical groups can be dumped simultaneously.
- Dumping can be suspended and resumed for any individual logical group or globally.
- Ground can request the deletion of specified information in storage.

Dummy commands shall be available to test the end-to-end connection between the ground and the on-board processor and/or any on-board intelligent user.

Standard PUS Services



No	Name
1	Telecommand Verification
2	Device Command Distribution
3	Housekeeping and Diagnostic Data Reporting
4	Parameter Statistics Reporting
5	Event Reporting
6	Memory Management
8	Function Management
9	Time Management

No	Name
11	On-board Operations Scheduling
12	On-board Monitoring
13	Large Data Transfer
14	Packet Forwarding Control
15	On-board Storage and Retrieval
17	Test
18	On-board Operations Procedures
19	Event-Action

Service 4 describes a service whereby the ground can ask for a set of statistical information on selected parameter values e.g.

- Min
- Max
- Average

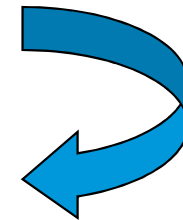
This can be very useful. In fact it is a form of information compression.

However it is not easy to implement on-board considering the whole PUS/CCSDS concept of packet being generated at source rather than parameter values.

Service 12 and 19 need to be considered in parallel to obtain the full benefit. Together they provide a form of simple scripting.

Monitoring (If parameter $X >$ threshold THEN generate Event Y)

Event – Action (If Event Y occurs THEN proceed with Action Z)



Events can also be generated by other OBSW modules such as FDIR

This is a more advanced type of on-board scripting. E.g.

- IF THEN ELSE
- LOOPS
- Variables
- Constants...

The PUS service only deals with controlling the procedure on-board not editing it.

An example of a PUS service: Memory Management Service



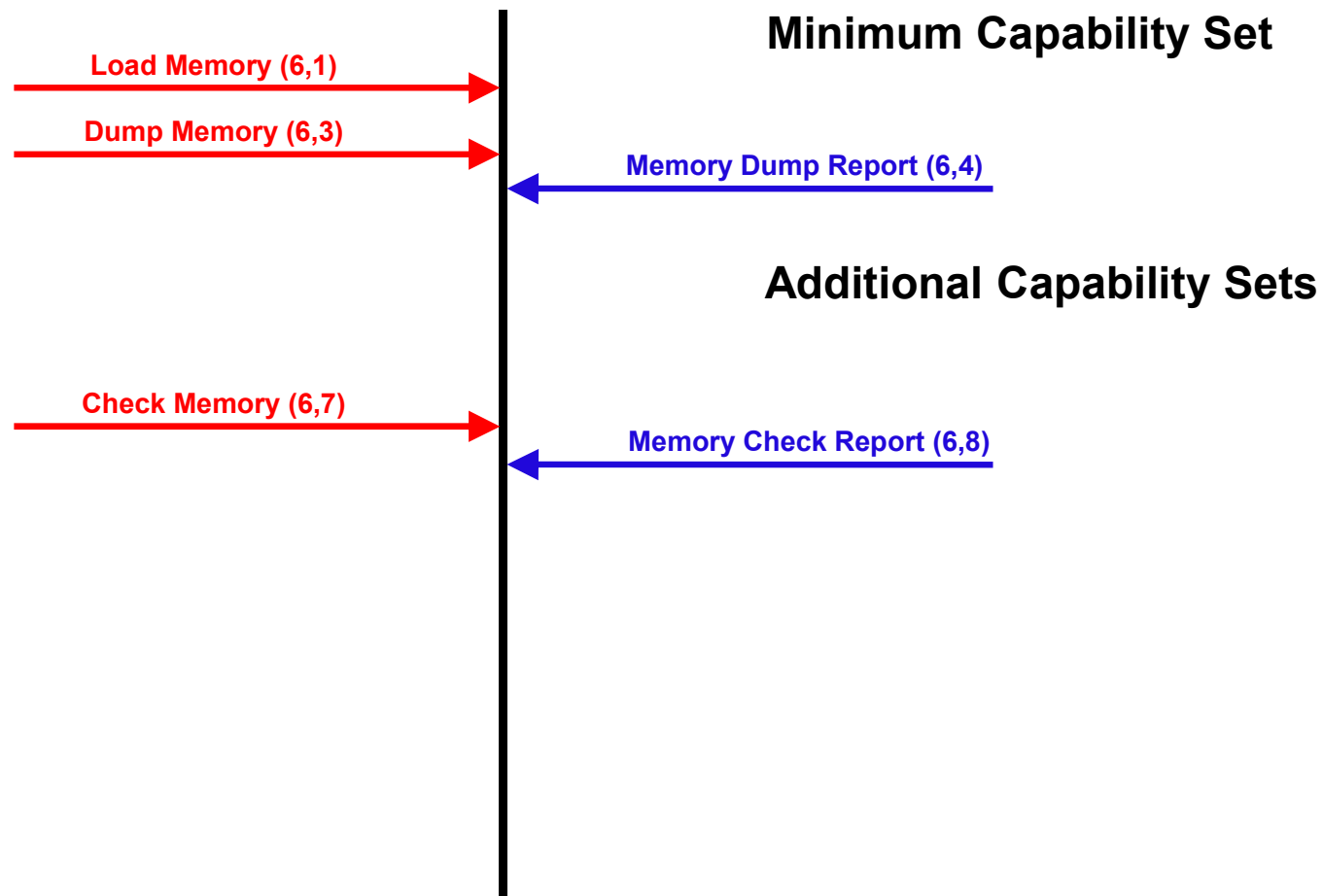
Service Model

- Capability to Load, Dump and Check the contents of on-board memories managed by the Application Process hosting the service
- These operations may be performed on a contiguous memory area or several non-contiguous areas (scatter-load, scatter-dump, scatter-check)
- An optional checksum may be appended to a memory load. A checksum is calculated and compared before transferring the data to memory
- Two distinct memory addressing techniques can be used:
 - Base reference plus an offset
 - Absolute address

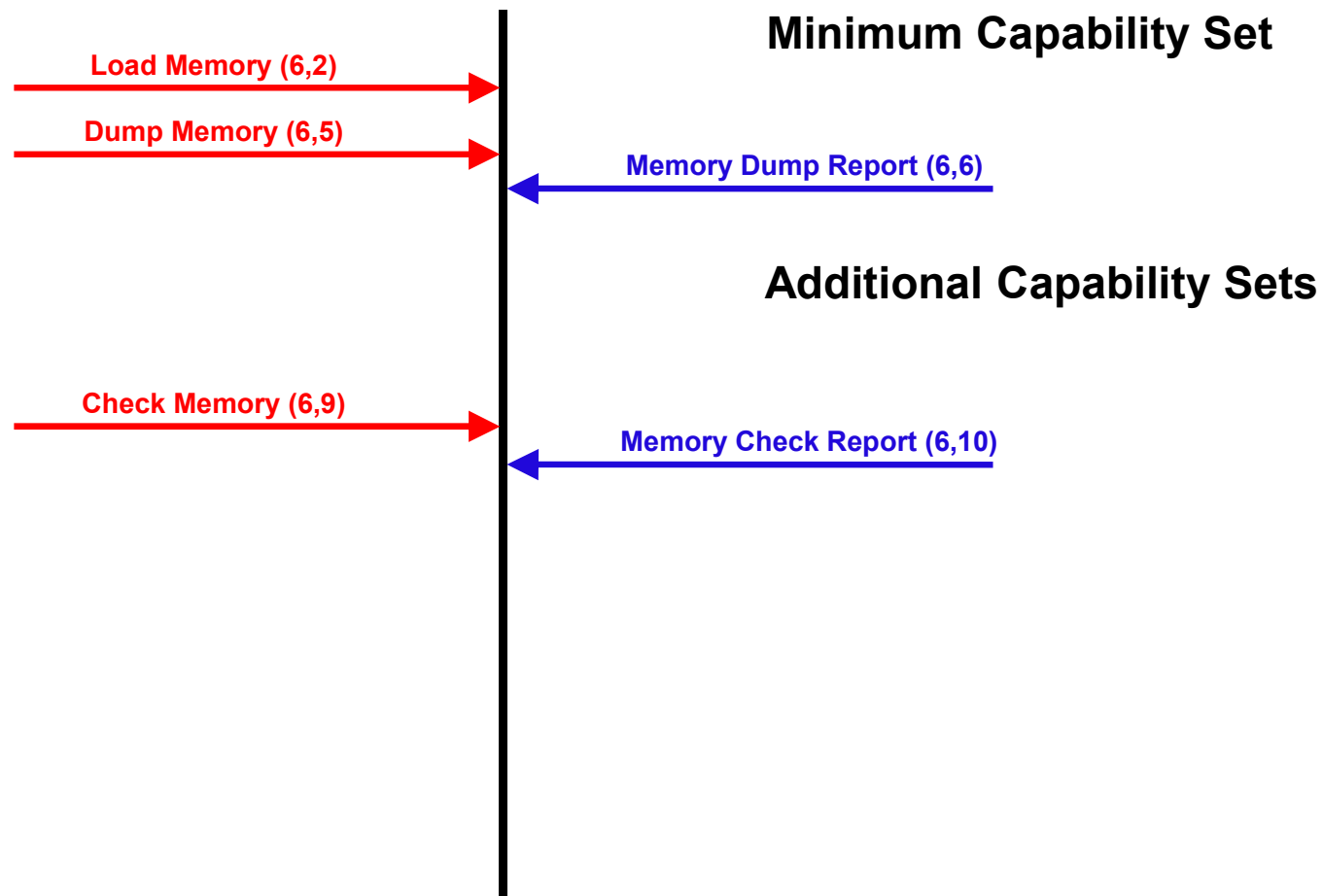
Memory Management Service: Base + Offset Addressing



Standard Messages \xrightarrow{TC} \xleftarrow{TM} and Implementation levels



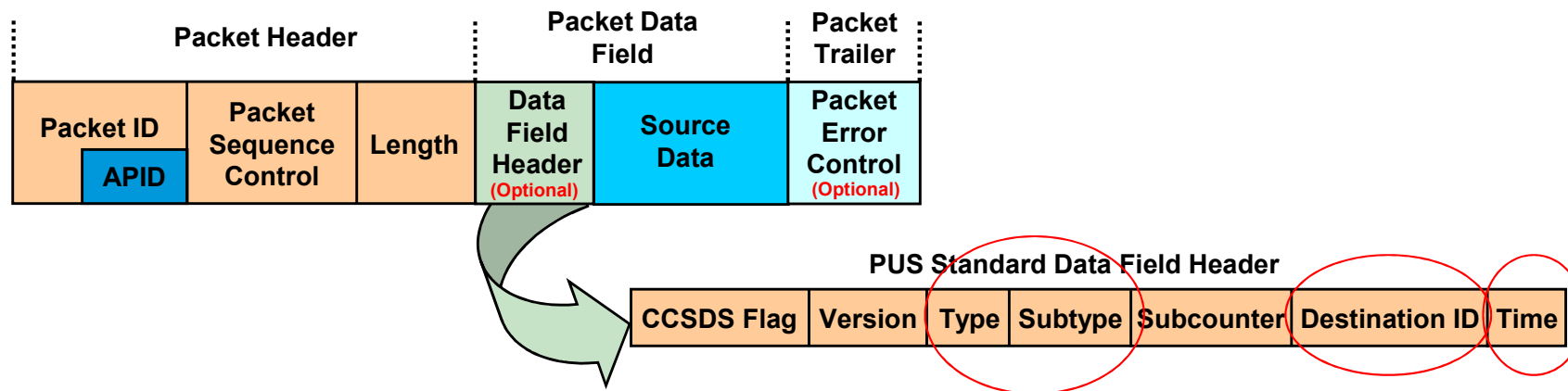
Standard Messages \xrightarrow{TC} \xleftarrow{TM} and Implementation levels



PUS – TM and TC packets

One can now see how this concept is implemented the PUS header. The Type and Subtype fields are used to describe the PUS service and subservice.

CCSDS TM PACKET



The destination ID is the application on ground/on-board that should receive the packet.

The time field is the generation time of the packet. This is applied to all the parameter values in the packet.

If there are multiple readings of the same parameter value at different delta times (i.e. super commutation) then the mission control centre applies this delta time to that parameter when interpreting the packet.

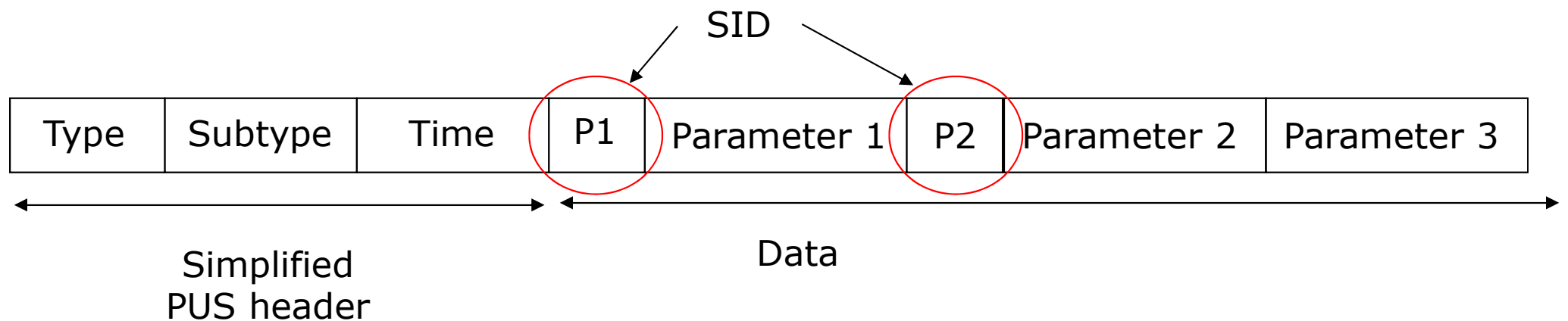
PUS gets messy

However, only using types and subtypes has limitations.

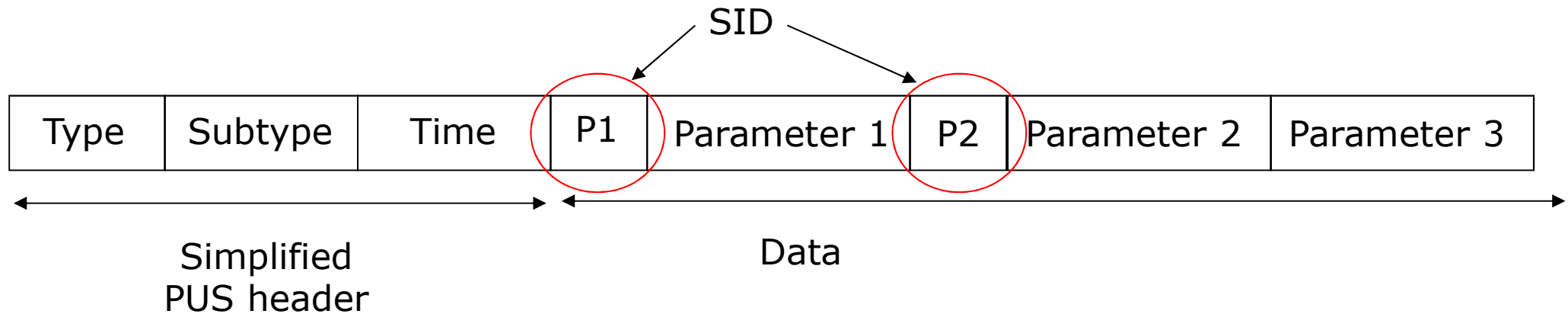
In order to provide this extra information it uses two fields in the data field (called P1 & P2).

These two fields are combined to produce a Structure ID (SID).

P1 and P2 can be inserted in different places in the packet and can have different lengths for different packets.



PUS gets messy



Consider the housekeeping service (3) the mission control centre needs to know more information than “I am a housekeeping packet from this application source” in order to decommutate the packet.

In this case P1 and P2 are combined into a SID that identifies a table on the ground listing the TM parameters in the packet and their start positions (offsets) in the packet.

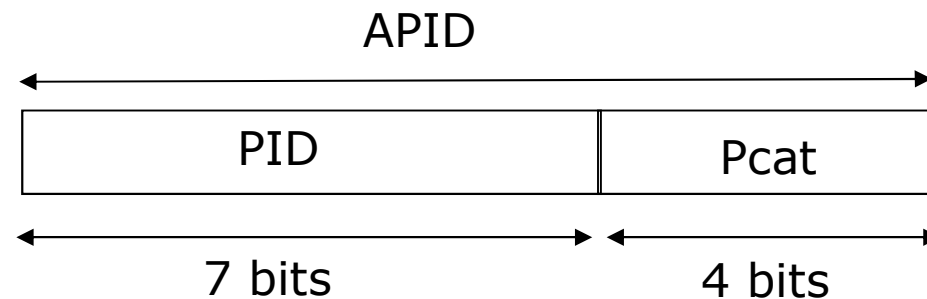
Consider the memory management service (6). In this case P1 might be used to define the start address of the dump and P2 the length of the dump.

PUS gets messy

As the SID is in the data field it makes routing of packets both on-board (e.g. service 14 & 15) and on the ground impossible using header information alone.

Also the source sequence count is only done at CCSDS header level. Therefore it will be difficult to identify which packets are missing if there is a gap.

The ESA solution:



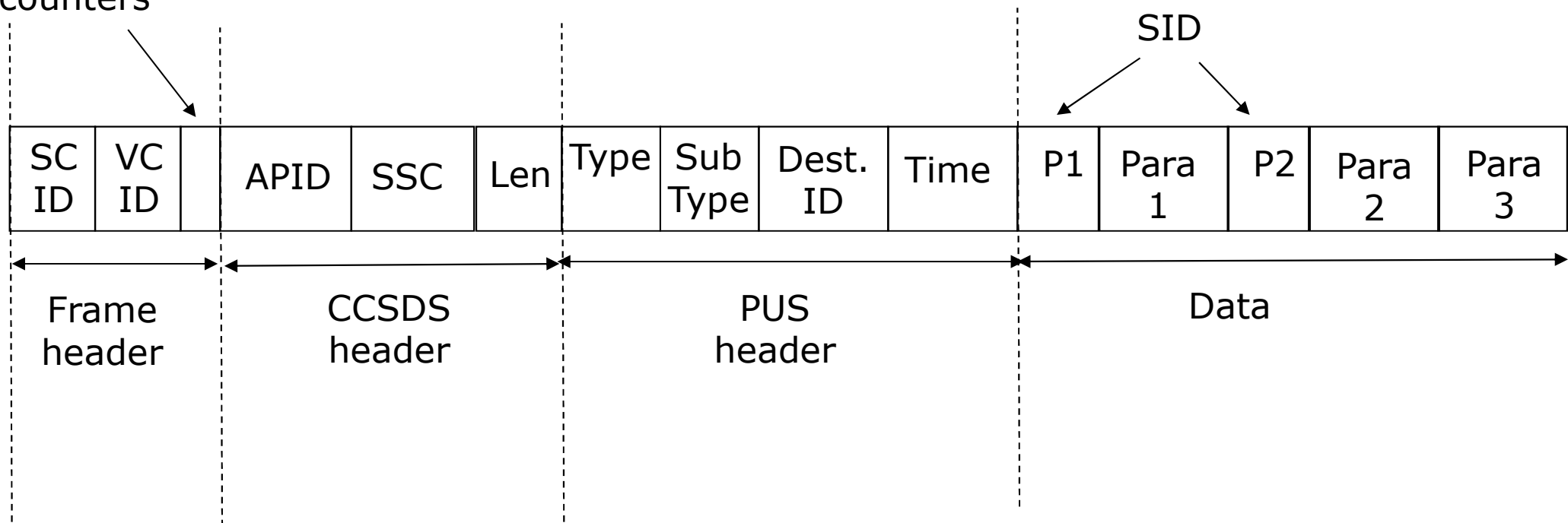
The PID does the same job as the APID.

The packet category identifier (Pcat) groups certain types of packets (coarsely mapping service types) so that they can be routed correctly using headers only.

One disadvantage is that the APID is effectively smaller. The original APID allowed for 2056 applications while the smaller PID is limited to 128.

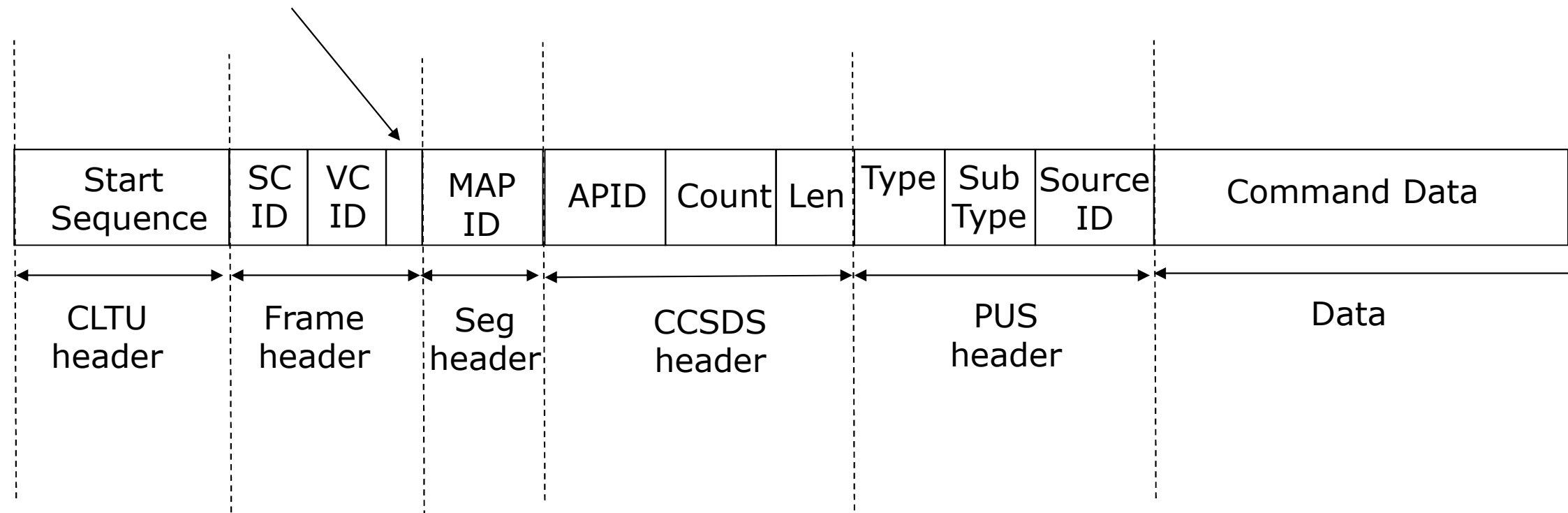
CCSDS/PUS summary – a simplified TM header

Frame counters



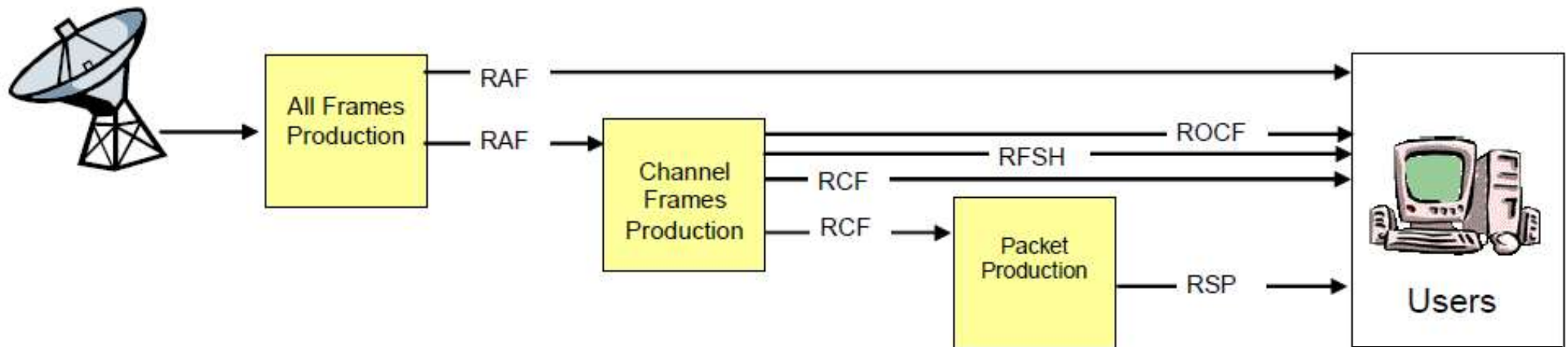
CCSDS/PUS summary – a simplified TC header

Frame counter &
COP-1 fields



- The Space Link Extension (SLE) services are a way of controlling the data flow between a ground station and a mission control centre.
- It only works with CCSDS TM, TC or AOS data structures.
- The SLE service provides a proxy at the ground station with a corresponding peer entity at e.g. the control centre.
- The added value of SLE is two fold:
 - It defines a protocol to make a request for a transfer as well as to actually manage that transfer
 - It is used by ESA and most other agencies. Therefore if you implement SLE and are CCSDS compliant in theory you can get support from a wide array of resources.

Example:



RAF: Return All Frames

RCF: Return Channel Frames which provides Master Channel or specific Virtual Channels

RFSH: Return Frame Secondary Header

ROCF: Return Operational Control Field

RSP: Return Space Packet, which enables single users to receive packets with selected Application Process Identifiers (APIDs) from one spacecraft VC

OSI Model

Application
Presentation
Session
Transport
Network
Data Link
Physical

TCP/IP Model

Application
Transport
Network
Data Link
Physical

E-Mail, FTP,
Web Browsing, etc...

TCP, UDP, SCTP, etc...

IPv4, IPv6

IEEE 802, SONET
Frame Relay (HDLC), etc...

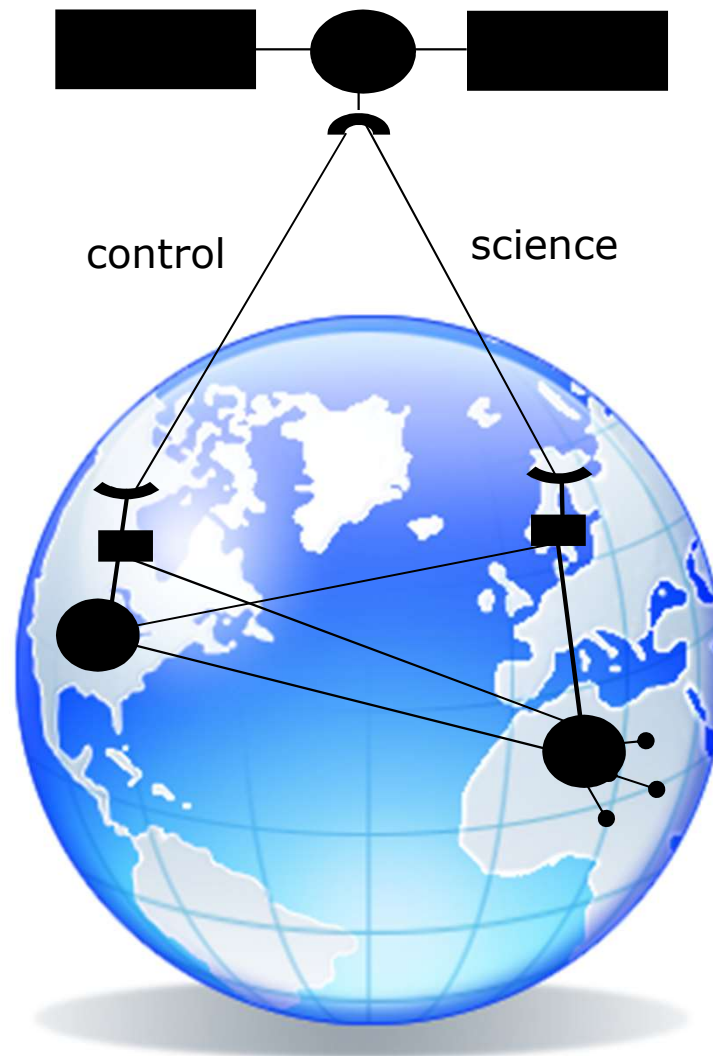
Wire, Fiber, RF, etc...

The use of IP has two advantages:

1. When there is a network set-up (as opposed to point to point) using IP allows us to delegate many resource intensive tasks to the network layer.
2. We can use of lots of tools, libraries and expertise from the terrestrial world. The sheer size of the investment in technology in the IP terrestrial area and availability of cheap, mass produced equipment makes this very attractive.

Note that if you are only using point to point communication then using IP packets only introduces unnecessary overhead.

Networks on the ground are already here

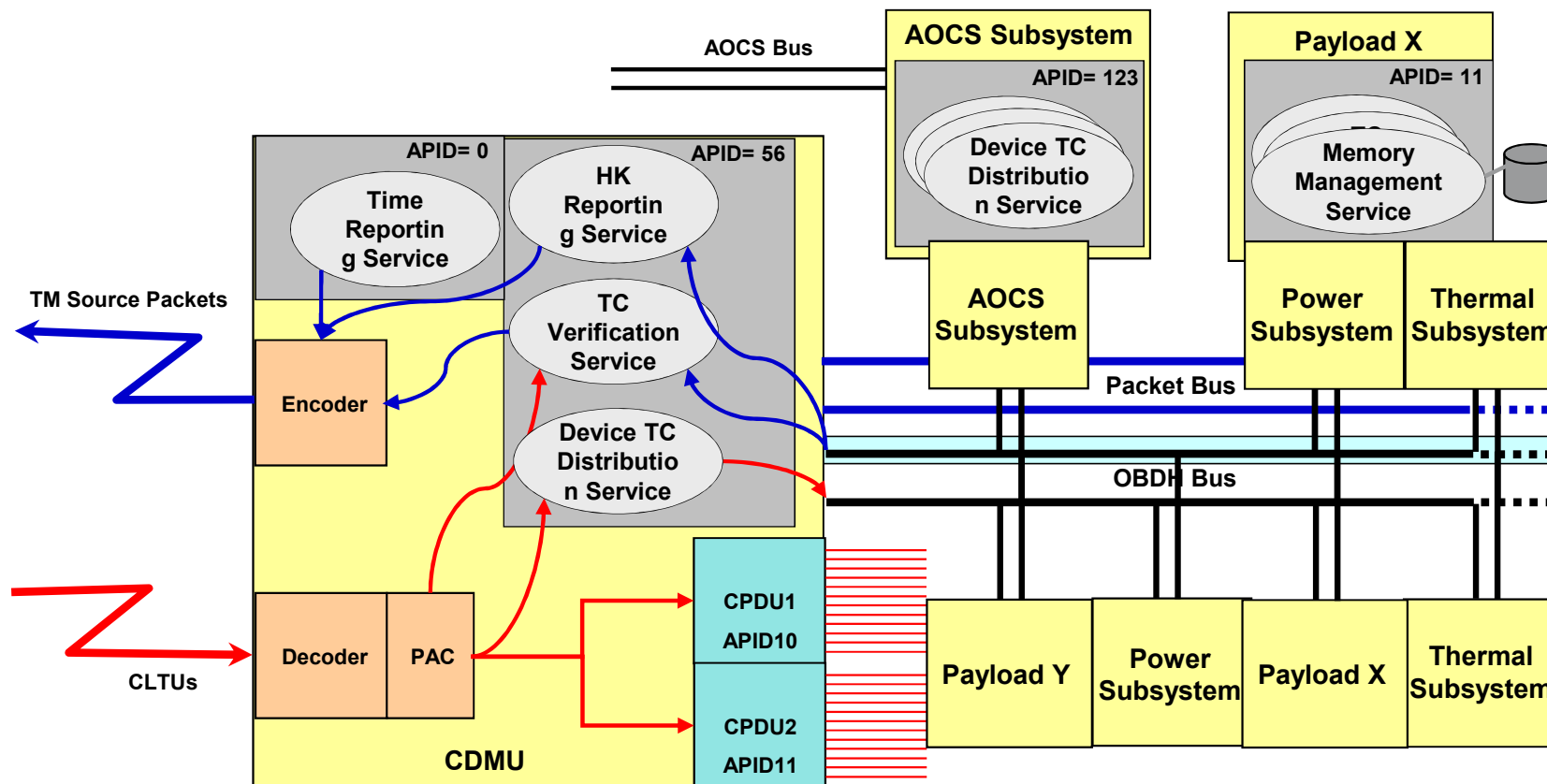


Nowadays, there is almost always a network being built (even in simple cases) consisting of

- Mission Control Centre
- Science Operations Centre
- Ground Stations
- Remote users

Networks on-board are coming

On-board the spacecraft this network could be extended with the payloads and processors being extra nodes. This gives the possibility of an institute remotely controlling **just** their payload. This is much easier using IP.



Mobile IP – a potential space use candidate

- IP mobile is a layer on top the IP stack that makes sure that the applications don't panic when there is a break in communications. It will smoothly establish a new link when one becomes available and signal to the applications to continue work.
- In this way the network handles the handover without extra work on the operators or applications. It is widely used in terrestrial systems (e.g. RIM blackberries)
- Using IP mobile we can consider using several ground stations as network resources. This would enable seamless switching between them to perform a task like data dumping (without complicated mission planning)
- The same network advantage can be leveraged if we increase the number of spacecraft in the network i.e. constellations

- CCSDS provide a recommendation for transporting IP packets over the normal data link layer.
- Basically each packet must be wrapped in a small header and trailer before it is sent to be framed as normal. This is called the encapsulation service.
- The overhead is relatively small but the main problem remains that space specialized ground equipment is required to deal with those CCSDS frames. Therefore some of the advantage of using IP is lost.

Pure TCP does not work well on space links. This is because of

1. the way it reacts when a packet is lost.
2. the long delays between the sender and receiver

When TCP detects that packet has been lost it resets the congestion window to a minimum. This then builds up again as packets are transported correctly. The congestion window refers to the number of packets that are allowed to be unaccounted for at any one time. Another way of thinking about this is “packets in transit”.

If the link is not so good then on average the congestion window will be small. A long delay exaggerates this effect i.e. the link will be relatively empty. Because of this various methods of modifying TCP have been tried for space links.

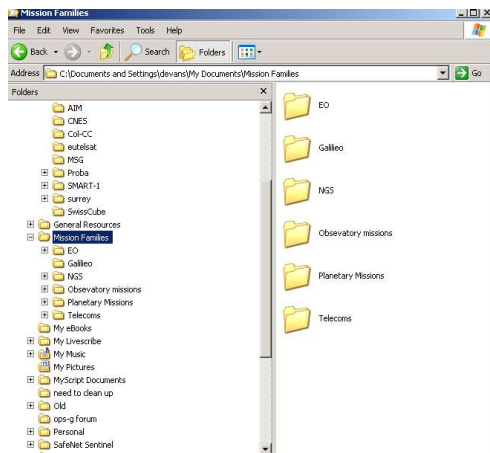
Also application level protocols working on negative acknowledgement (i.e. only sends a request if didn't get something) have been successfully implemented.

Operational Experience: SSTL control centre

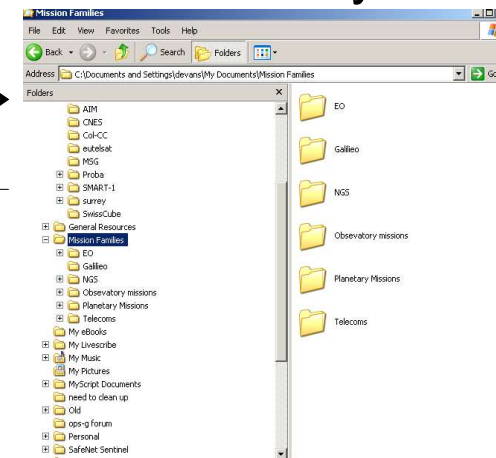


- SSTL have successfully used terrestrial based protocols all the way through the chain e.g. IP/TCP running over HDLC.
- The advantage can be seen in the following operations concept.

On-Board Directory



Ground Directory



Directory listing

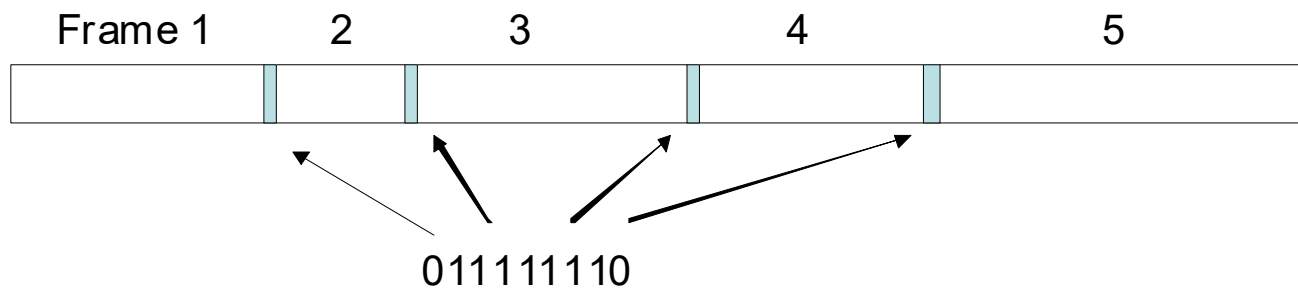
File downlink requests

File Transfer

Automatic Retransmit Requests

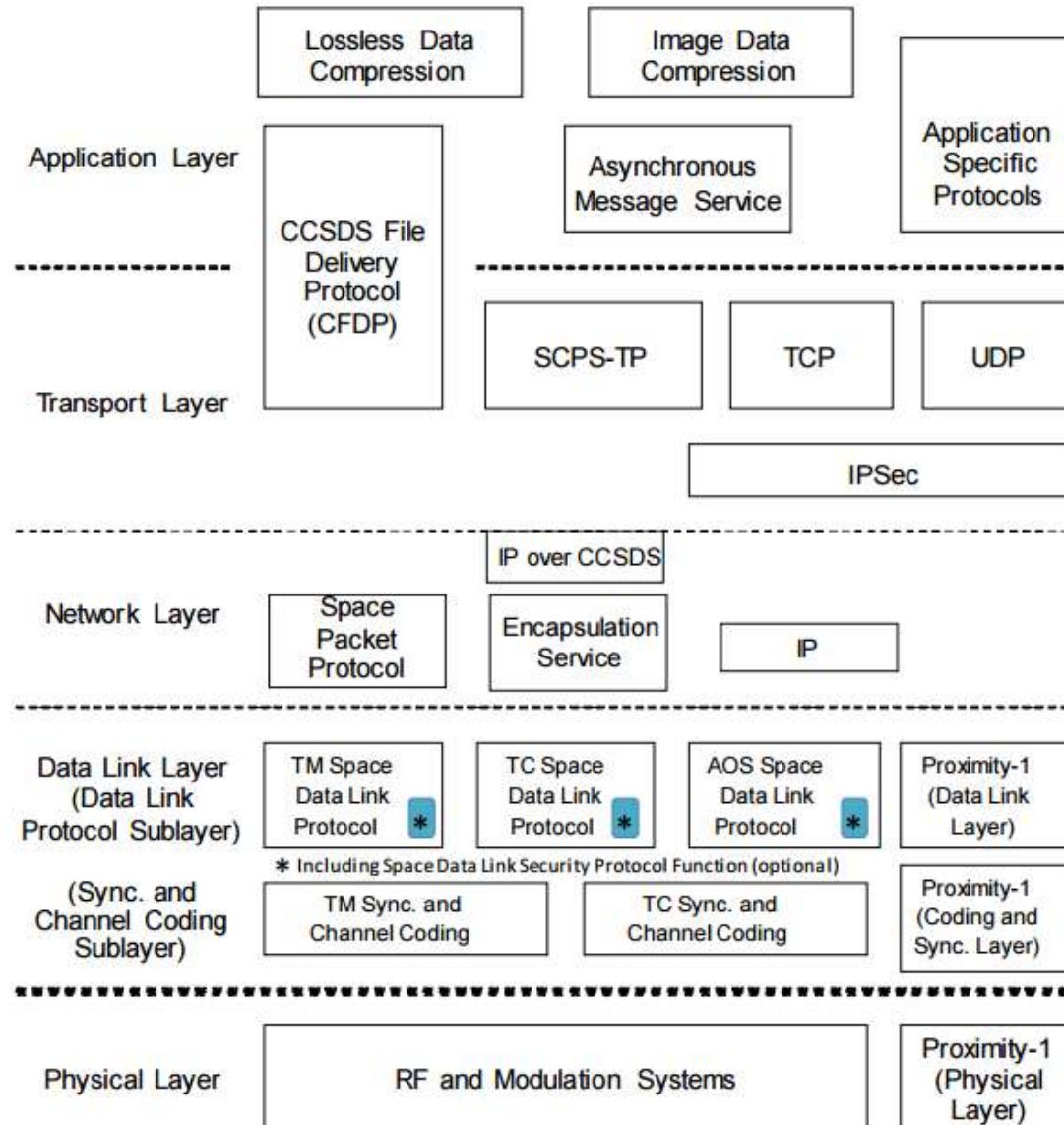
Operational Experience: SSTL control centre

- HDLC (high level data link control) is a simple protocol that allows variable length frames to be transmitted over a continuous bit stream.
- It does this by inserting a flag to mark the start of the next frame ('01111110').



- This means that the data cannot be allowed to contain seven 1's in a row and anytime five 1's in a row are detected a zero is inserted.
- This 'bit stuffing' means that HDLC frames are variable in length depending on the data that they are carrying but CCSDS frames **must** be constant in length.
- So SSTL benefit by using cheap CISCO routers instead of expensive space baseband units but their spacecraft are not compatible with ESA/NASA ground stations...

Other Space Link Protocols



- The Advanced Orbiting Systems (AOS) protocol was defined specifically for the ISS/Columbus programme and subsequently found its way into the spin-off Envisat downlink.
- It addresses one particular major issue with the limited size of the packet TM sequence counters which would roll over when used in high rate links with long (data relay enabled) contact periods.
- Very similar to conventional CCSDS TM and TC.

- Proximity-1 is a link layer protocol specifically designed for communication at non-terrestrial bodies.
- Will be used in the MEX support for the NASA mission Mars Science Laboratory (MSL).
- It is designed for operation with severely power limited resources (landers/orbiters) with dynamically varying link margins.
- To this end it incorporates hailing elements which allow maximum link use at the beginning of potential contact periods.

An extension to existing Internet protocols to improve performance in space environments.

- SCPS-TP - A set of TCP options and sender-side modifications to improve TCP performance in stressed environments including long delays, high bit error rates, and significant asymmetries.

Very rarely used up to now.

The emergence of file based operations has resulted in several competing space standards to transport and manage files.

CFDP

DTN

PUS service 13

The CFDP protocol consisted of two components:

1. A reliable data transfer engine incorporating negative acknowledgment and suspend/resume mechanisms which allow operation in noisy, long delay and disrupted transmission environments
2. A set of file manipulation primitives which can be used to manipulate remote file stores by operations such as file and directory creation, deletion, naming and, most importantly, copying.

The basic idea is that they implement store and forward. This means that one node sends data to another but keeps a copy.

Only when the receiver sends a message back to the sender to say all the data arrived correctly will the sender delete its copy. This capability is part of the application (bundle) layer of DTN.

The routers at the ground stations will need bigger buffers than normal to handle this for space missions because of the delays involved.

CFDP has also been extended to include this functionality although not to the same level as DTN.

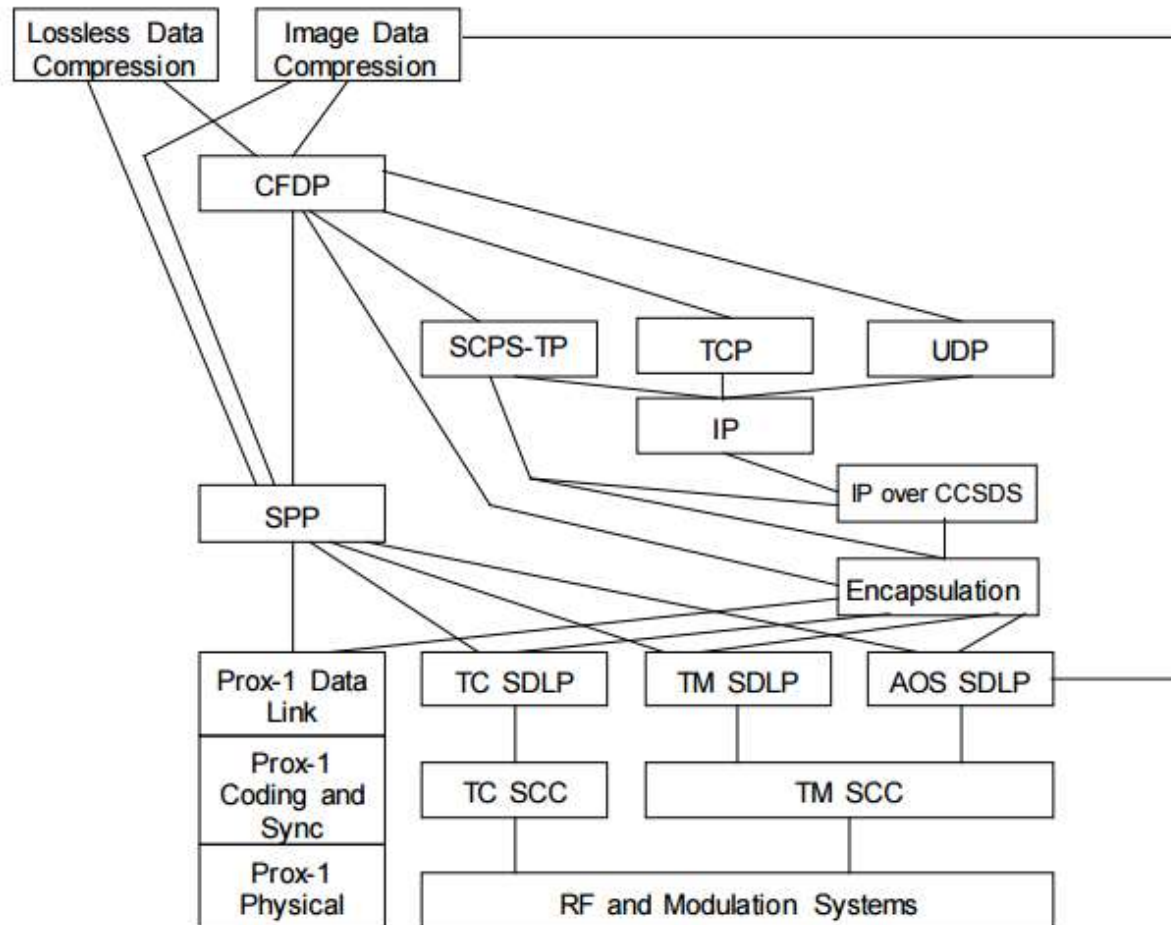
PUS Service 13 provides for the segmentation and acknowledgement of bulk data transfers.

It is simpler and not so fully featured as other protocols (e.g. DTN or CFDP) and not as optimised for long delay and disrupted links.

Limitations:

- retransmission capability not optimised for long delays, unlike CFDP
- transfers large service data units rather than files
- no file management capabilities
- no store and forward capability
- no proxy or remote operations

Possible Combinations of CCSDS Protocols



SCPS-SP and IPsec can be used between the Transport and Network layers in any combination of protocols.

SPP = Space Packet Protocol

SDLP = Space Data Link Protocol & Space Data Link Security (opt.)

SCC = Synchronization and Channel Coding