Space Systems Finland

EagleEye Evolution towards Time and Space Partitioning

Final Presentation, ESTEC, Dec 11, 2013

On the Grounds of Quality

Outline

- TSP
- EagleEye TSP project
  - Before: V4.0
  - After: V5.0
  - Design issues
- EagleEye TSP validation
- Recommendations for Applying TSP to OBSW
- Recommendations for Future Work

Space Systems Finland

Consortium

- Space Systems Finland
  - Panu Kauppinen, Niklas Holsti, Victor Bos
- Bright Ascension
  - Peter Mendham
- Universidad Politécnica de Madrid (UPM)
  - Juan Zamorano, Juan A. de la Puente
- fentISS
  - Miguel Masmano
  - Universidad Politécnica de Valencia (UPV)
    - Alfons Crespo

On the Grounds of Quality

The project

- Goal: port the EagleEye reference mission to a Time and Space Partitioned Platform
- Customer: ESA, technical officer: Felice Torelli
- Started in February 2012

Time and Space Partitioning

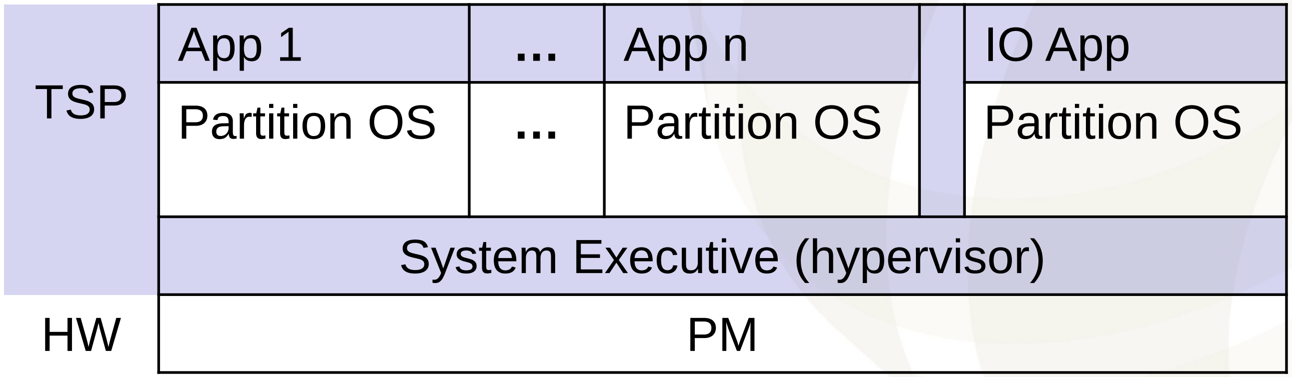
On the Grounds
of Quality

Space Systems Finland

Time and Space Partitioning

- Origin: Integrated Modular Avionics (‘90s)
- Why:
  - Save on mass, volume, and power
  - Mixed criticality systems on one OBC
  - Fault containment
- How: IMA-SP

IMA-SP


TSP
App 1
...
App n
IO App
Partition OS
...
Partition OS
Partition OS
System Executive (hypervisor)
HW
PM

Space Systems Finland

Evolution towards TSP

- From federated systems
  - combining CSW PM and Payload PM(s)
- From monolithic systems
  - Distributing CSW over partitions

    E.g., **EagleEye CSW**

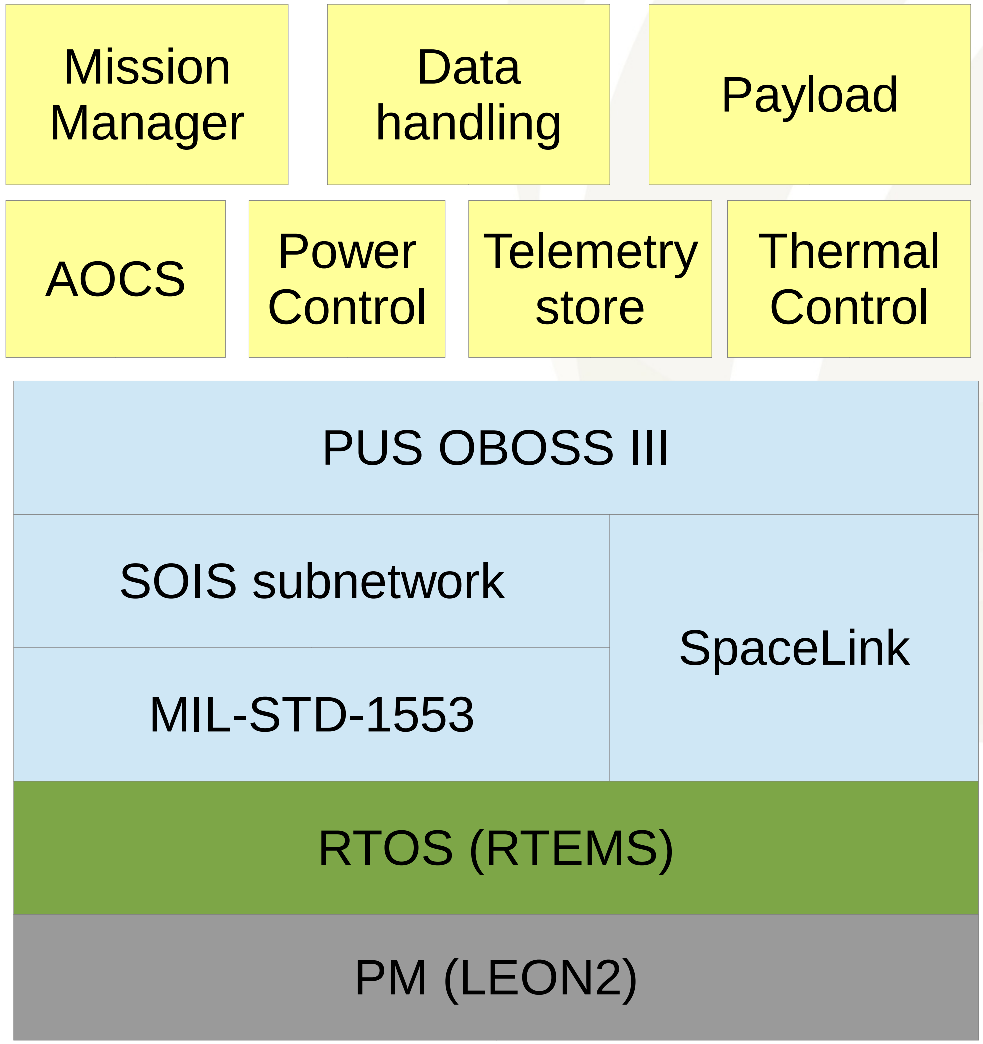
On the Grounds of Quality

Space Systems Finland

The EagleEye TSP Project

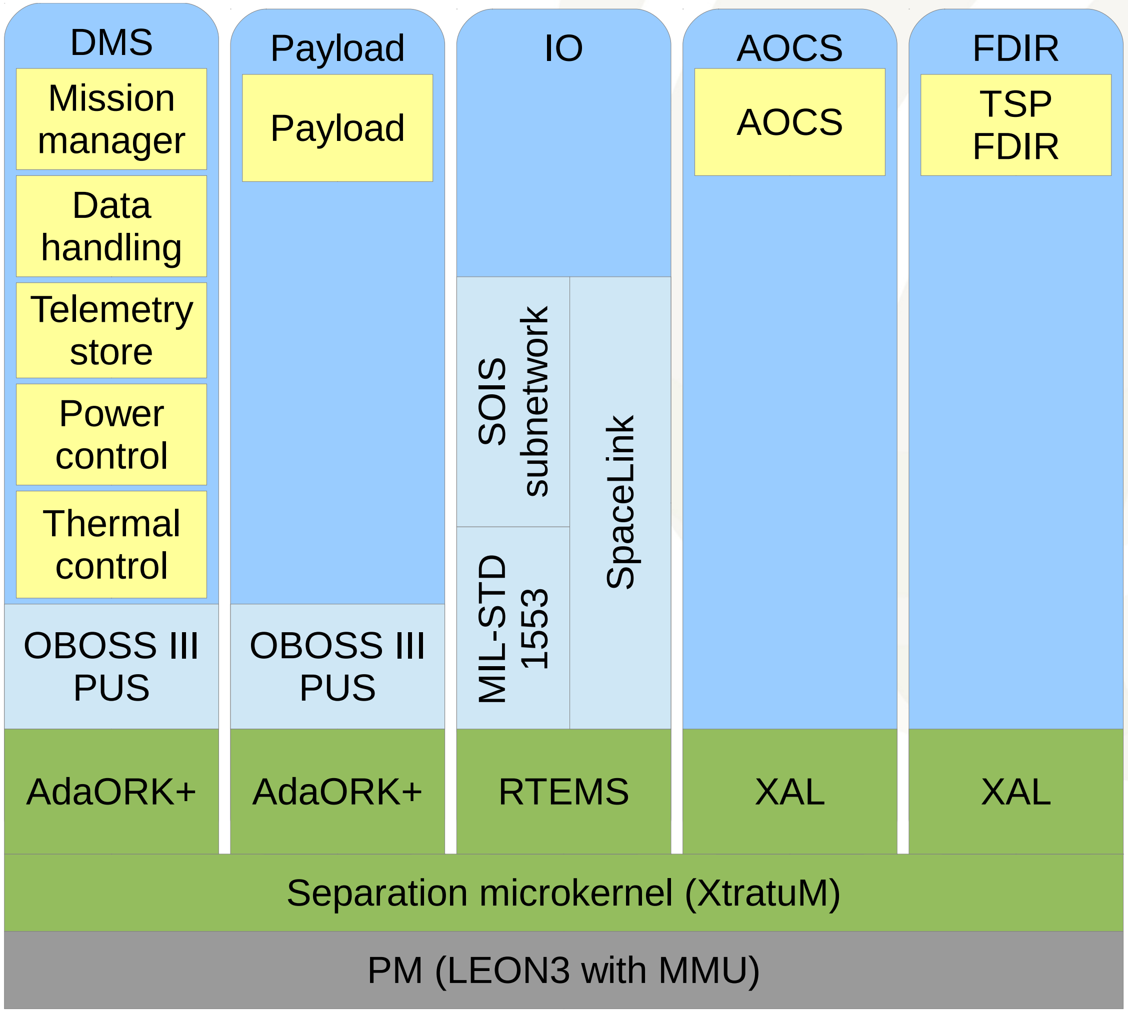
On the Grounds of Quality

Before: CSW V4



Space Systems Finland

After: CSW V5



On the Grounds of Quality

Rationale for this partitioning

- Obtain a single-language / run-time
- Anticipate mixed criticality system
  - I/O is critical
  - Therefore, qualified OS needed in I/O partition (RTEMS)
- Imitate typical business situation:
  - AOCS is developed by a third party
- Reuse CSW V4 as much as possible

Design issues

- SW components distribution
- Inter-partition communication (IPC)
- Partition programming language
- Partition OS
- PUS data handling
- I/O handling
- Scheduling
- FDIR strategy
- On-board SW maintenance

Space Systems Finland

Partition schedule

Time	Partition	Purpose
0	I/O	Sensor data acq, spacelink proc
25	DMS	DH, sensor data to AOCS
60	I/O	SSMM commanding, spacelink proc
75	AOCS	AOCS proc, actuator data to DMS
100	I/O	Thermal commanding, spacelink proc
125	DMS	AOCS data forwarding to I/O
150	I/O	Actuator commanding
175	Payload	Payload DH and commanding
200	I/O	PCS and spacelink proc
225	FDIR	FDIR monitoring, commanding, rep.

EagleEye TSP Validation

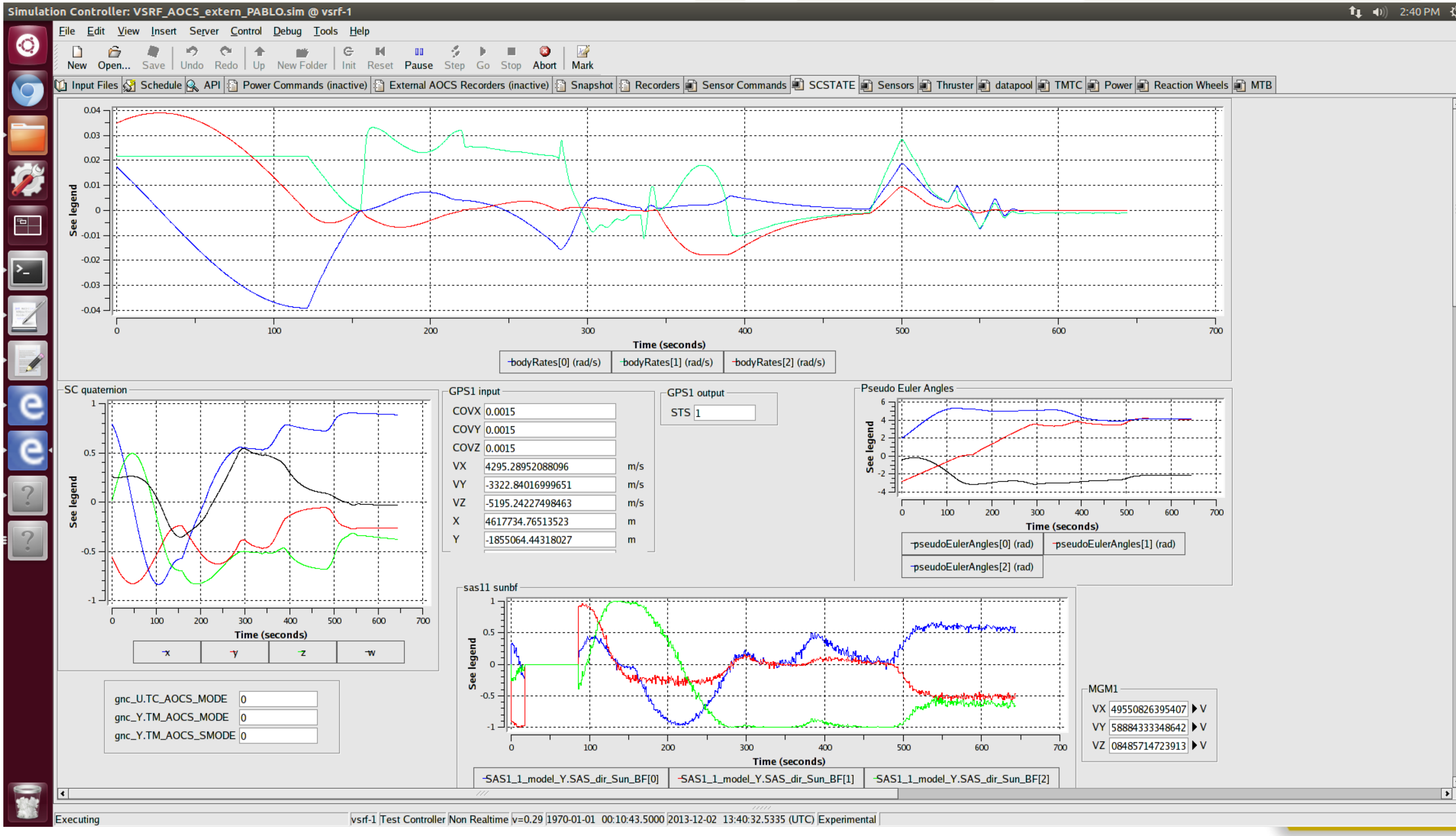
On the Grounds
of Quality

Test configurations for CSW V5

- Development SVF (ATB workstation)
- ATB SVF (open loop)
- ATB SVF connected to EuroSim (closed loop)

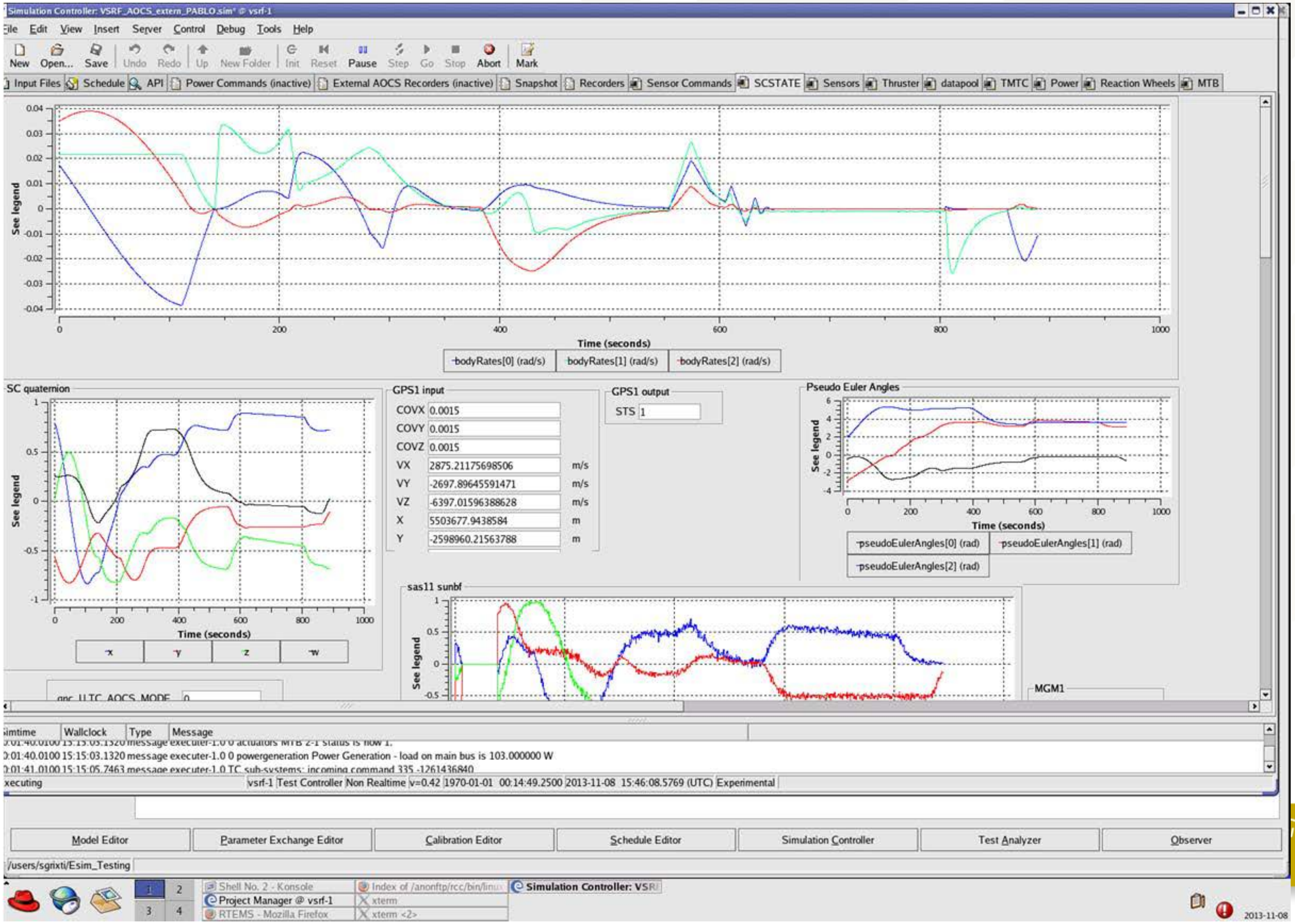
Space Systems Finland

CSW V4 AOCS: Full test



Space Systems Finland

CSW V5 AOCS: Full test



rounds of Quality

TSP Demonstrator

- TSP FDIR
  - Partition reboot / shutdown
  - TSP health monitoring
- Execution of partial system
  - During unit testing
  - During system integration
- Faulty applications (AOCS, Payload)
  - Application crash
  - WCET overrun

Recommendations for Applying TSP to OBSW

Space Systems Finland

On the Grounds of Quality

Space Systems Finland

Recommendations for TSP OBSW

- Many lessons learned on EETSP
- Can extract some general recommendations

- I/O handling
- Partition scheduling
- Centralised FDIR
- Location of PUS handling
- IPC patterns

On the Grounds of Quality

I/O Handling

- I/O is time intensive
  - Especially if I/O partition must be active during complete I/O operation
- Can lessen impact on partition schedule
  - DMA (e.g. MILBUS send lists)
  - Multi-core
- Impact on partitioning guarantees
  - Spatial partitioning impact of DMA – solved with IOMMU
  - DMA and multi-core have temporal impact

SpaceSystems Finland

Partition Scheduling

- Partition schedule is crucial system design issue
- Difficult if porting existing software
  - Sufficient dynamic execution information may not be available
  - Need WCET information for OBSW functions that are to be allocated to partitions
- When assessing risk of porting SW to TSP
  - Consider partition schedule
  - Analyse existing data pack for sufficient dynamic design/ WCET information

On the Grounds of Quality

Space Systems Finland

Partition-level FDIR

- Centralised partition-level FDIR in EETSP
  - Combination of hypervisor health monitoring and partition watchdogs
  - Worked well and is recommended
  - Could be a template for a “standardised” FDIR
- IPC health monitoring not robust enough
  - Partition failures may cause IPC queues to fill
  - Babbling idiot and failed receiver cases
  - Need ability to monitor health of queuing ports
  - Requires modification to hypervisor

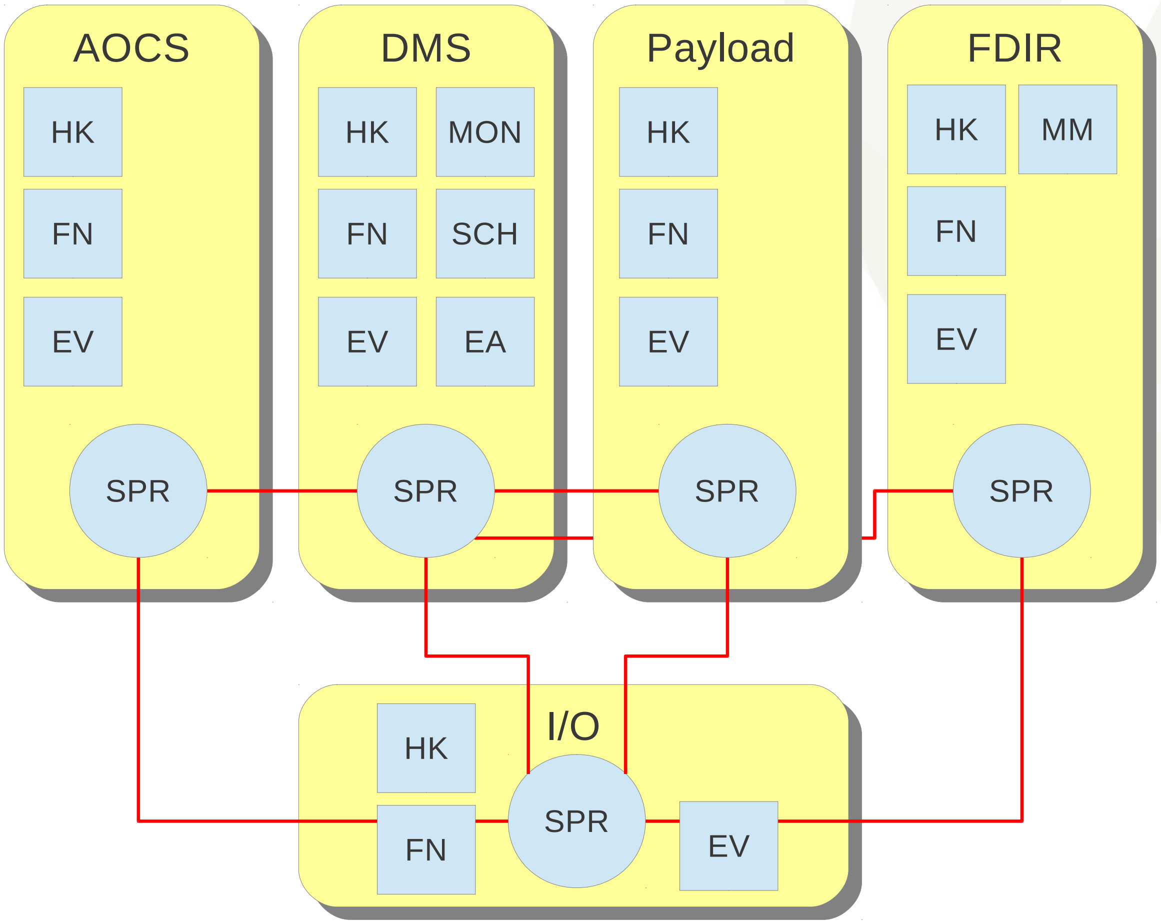
Space Systems Finland

Location of PUS Handling

- PUS Handling in CSW v4 all in Ada
- Requirement to partition on language
  - Either locate PUS handling only in Ada partition(s)
  - Or port some PUS handling to C
- A more distributed architecture for PUS is recommended
  - Similar to PUS split between OBC-Payload
- Not possible on EETSP due to effort (and risk) required to port OBOSS elements to C

On the Grounds of Quality

Distributed PUS



This is just an example!

On the Grounds of Quality

IPC Patterns

- Recommend IPC design patterns are used
- Propose three, based on experience

- Loosely-coupled messaging
- Loosely-coupled periodic update
- Client-server

Space Systems Finland

Loosely-Coupled Messaging

- Packet based
- No dependence on acknowledgements etc.
- Both source and destination are stateless
  - As far as communications are concerned
- For example
  - PUS packet forwarding
  - Some PUS service handling
- Uses queuing ports
- Need to be able to characterise flow rate

Space Systems Finland

Loosely-Coupled Periodic Update

- Periodic data, naturally becomes “stale”
- For example
  - Watchdogs
  - Data acquired from MILBUS
  - AOCS inputs and outputs
- Both source and destination are stateless
- Uses sampling ports
- Need to be able to characterise refresh frequency

On the Grounds of Quality

Space Systems Finland

Client-Server

- Request/response pattern
  - Simple, stand-alone transactions
- Stateless server
- Two-state client
  - Idle/waiting for response
- Need timeout conditions in client
- Need to
  - Match responses to requests
  - Characterise message flow and response times
- Suitable FDIR needed to protect IPC queues

On the Grounds of Quality

Space Systems Finland

Recommendations for Future Work

On the Grounds of Quality

Further Work

- EETSP has been a challenge but many useful lessons learned
  - ATB/EagleEye ready for more research in TSP
- More lessons could be learned
  - Using EagleEye including CSW
  - Using EagleEye but replacing CSW
  - In an alternative setting

Further Work on ATB/EagleEye (1)

- Improve the realism of SVF and RTB
  - MILBUS send lists on SVF
  - Hardware watchdog(s)
  - OBC redundancy and reconfiguration module simulations (SAVOIR OBC architecture)
  - Boot process (e.g. use of NV boot memory)
- Port TSP CSW to new SVF and RTB
  - Investigate I/O handling better, including DMA
- Investigate OBSW maintenance
  - Booting, patching etc. may require hypervisor work

On the Grounds of Quality

Space Systems Finland

Further Work on ATB/EagleEye (2)

- Investigate “standard” FDIR partition
  - Include IPC monitoring
  - Might require updates to hypervisor
- Investigate multi-core
  - Could use existing CSW as basis
  - Particularly interested in I/O handling
  - Could also look at an I/O co-processor
- Port or new implementation of EE mission using OSRA, including TSP

Recommended Tool Improvements

- XtratuM good hypervisor but better tooling needed
  - Needs centralised and coordinated build system
  - Better integration of configuration into code
  - Configurable unit and integration test framework
  - Better support for interactive debugging
- AdaORK+ good but would benefit from tool support for working on resource-constrained systems
  - Especially assistance with stack allocation