LuxSpace CAN activities

LUXSPACE
An OHB Company

Klaus Schwarzenbarth

LuxSpace Sàrl

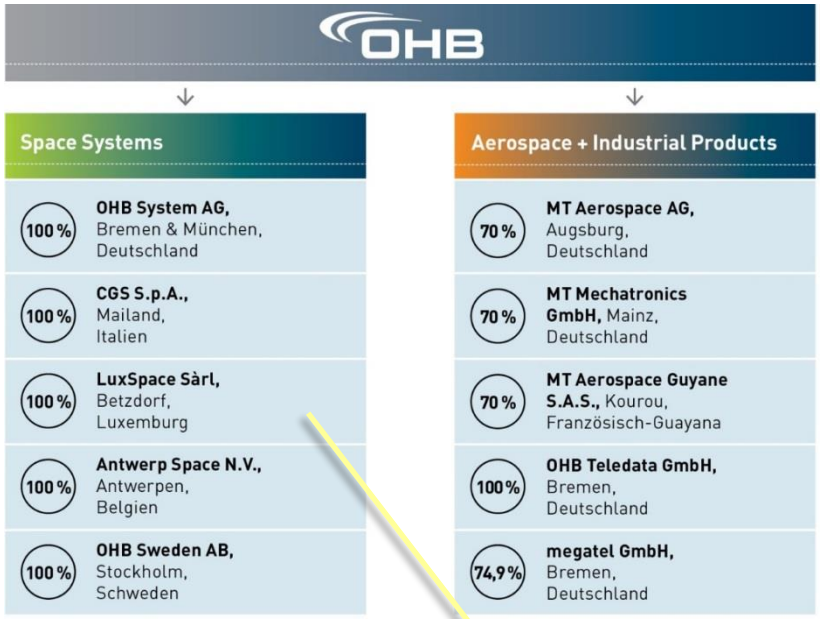
SBC - Rue Pierre Werner 9, 6832 Betzdorf, Grand-Duché de Luxembourg

Agenda

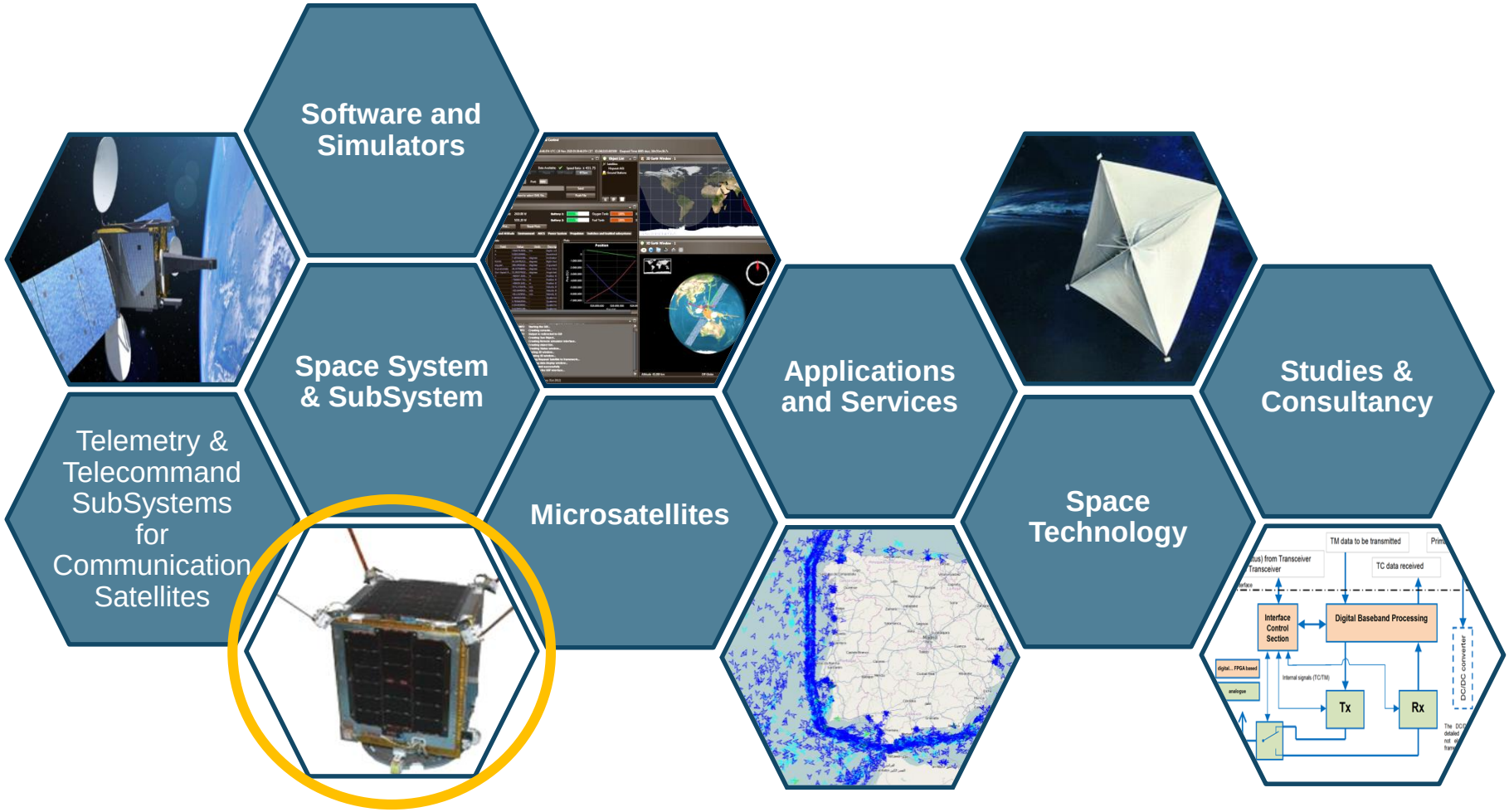
- LuxSpace company introduction
- Microsatellite history
- Introduction to ESAIL
- Why we use CAN bus
- Details on implementation of the specific ECSS CAN features
- Current results
- Conclusions

About LuxSpace

- Founded in November 2004
- 100% owned by OHB AG/Bremen but acting independent
- Offices located close to the SES ASTRA Campus at Betzdorf, Luxembourg
- ISO 9001:2008 qualification since 2008
- 50 staff from 12 different countries
- Turnover approx. 10M€



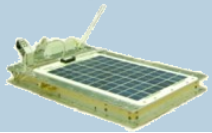
LuxSpace



LuxSpace microsatellites

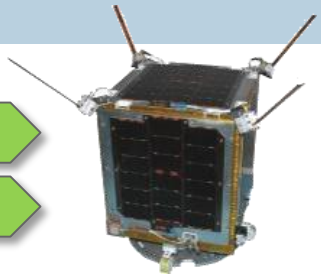
launched

Piggy Back Payload | Satellite



2009

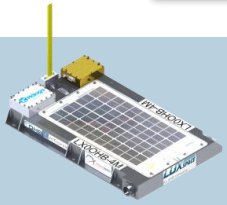
Pathfinder
- AIS receiver (ship detection) attached to third stage of the launcher



2011

2012

VesselSat-1 and -2:
- Platform 30 cm cube, 30 kg, 4W, payload 3 kg
- embarking AIS receiver



2014

Moon Mission 4M:
- Manfred Memorial Moon Mission
- Micro payload attached to third stage of chinese launcher

under development

2018

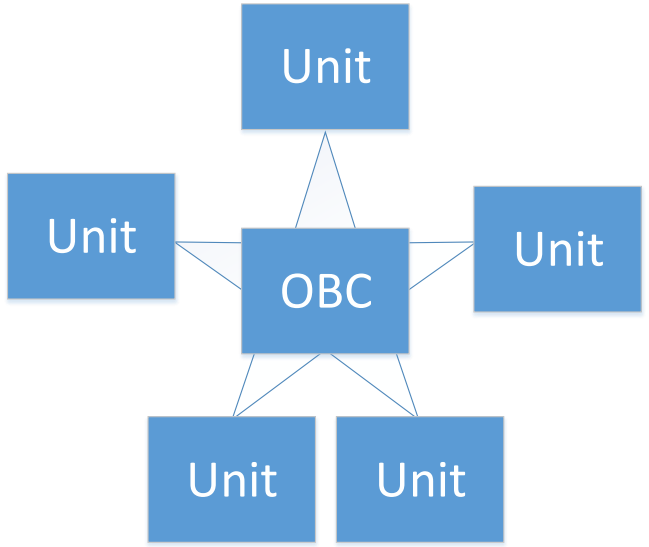
E-Sail Sat-A
- 70 cm cube, 100 kg, 10W, payload 15 kg, 3-axis stabil.
- ESA E-Sail program (AIS with high data rate link)

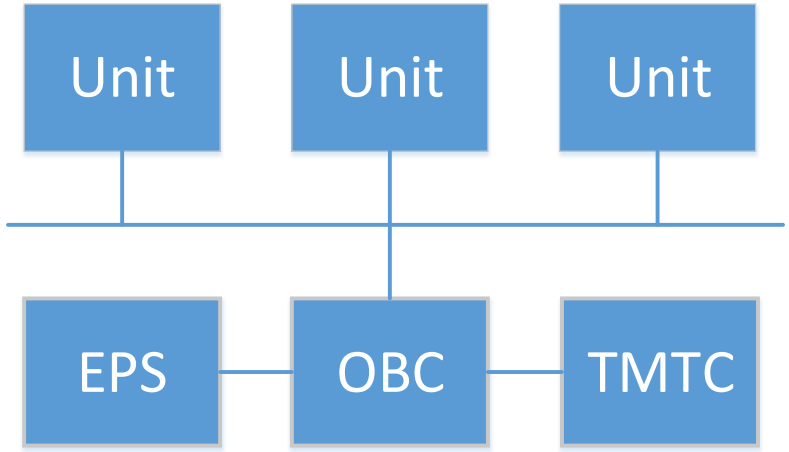
Introduction to ESAIL

- European SAT AIS Constellation
- ARTES 21
- LuxSpace is prime contractor
- Low cost ESA AIS mission
- 2 satellites
- 100 kg
- 3 axis stabilized platform
- High performance AIS payload
- High speed C-Band downlink
- Current status: System PDR closeout
- Launch planned in 2018

Why we use CAN bus I

- **Previous missions used star architecture**
- **Problematic for larger satellites**
  - Large number of interface required on OBC that take space and power
  - Low flexibility
  - Complicated Harness
- **Obvious solution to use Command and Control bus**
- **Highly critical interface (TMTC and EPS) still directly connected to the OBC**





Why we use CAN bus II

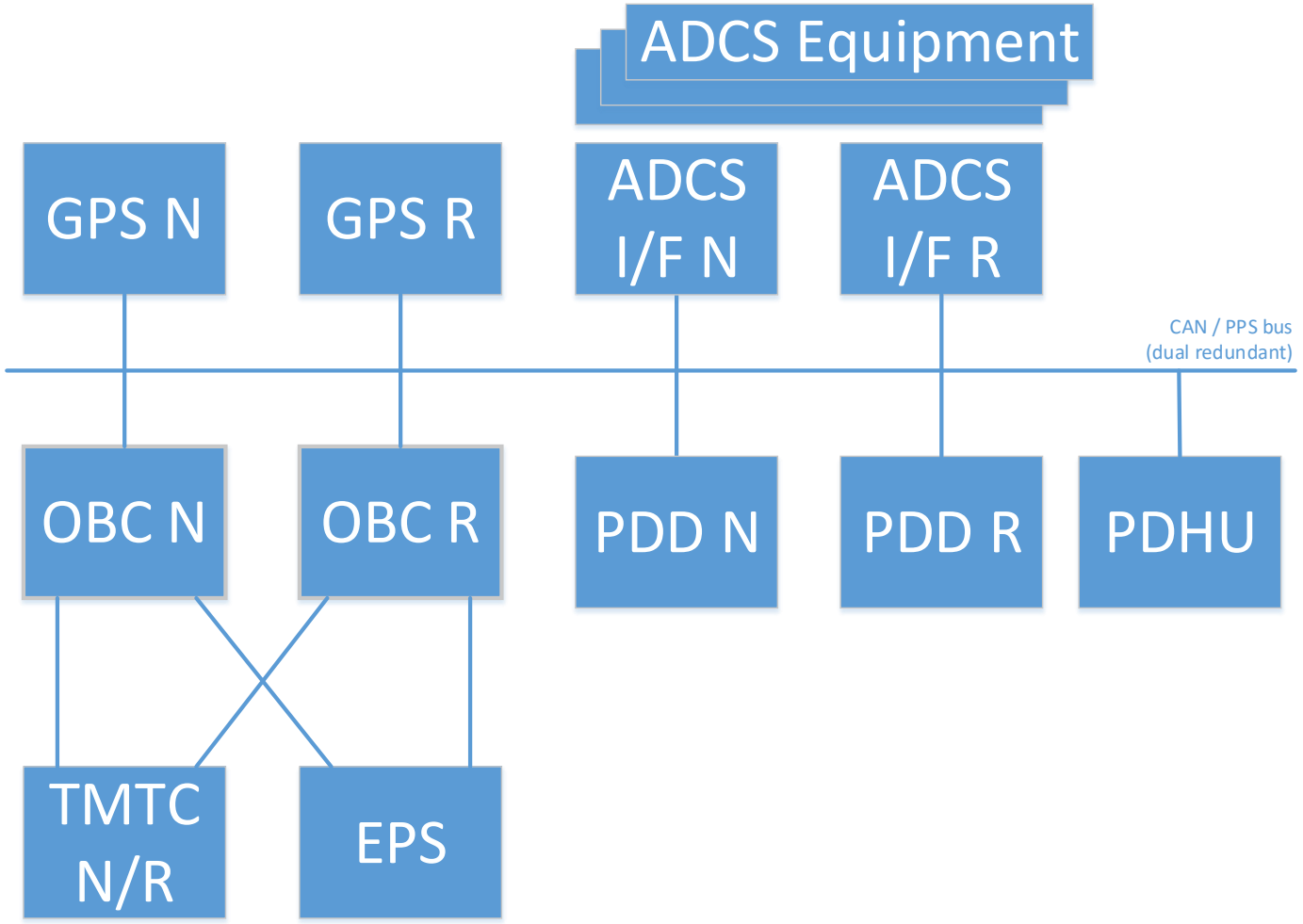
CAN

- Well known industry standard
- Robust, designed for harsh automotive environment
- Availabe on COTS parts
- Low power
- Standard test equipment available

CAN ECSS

- Just in public review when doing bus trade-off
- Based on widely known CANopen
- Later reusability
- Test equipment available

Block diagram



Physical layer

Transceivers

- **ISO11898 CAN transceivers (space grade) are used**
  - CAN over RS485 not recommended for new designs
  - Compatibility with standard test equipment

Connectors

- **Usage of standard connector and pinout left open to the suppliers**
- **Not always feasible due to limited space**

Harness

- **Daisy chain topology**
- **External termination resistors**
- **Grounding / shield adapted to mission specific needs**

Bit timing

- **1 Mbit/s**
- **No other bit rates are supported**

Network Management / Bus Selection

- **Heart beat message**
  - Implemented as per ECSS
- **Boot message**
  - Implemented as per ECSS
- **Emergency messages**
  - Optional, currently not used
- **Node State Machine**
  - Implemented as per ECSS
- **Bus selection**
  - Cold redundant bus architecture
  - Bus selection implemented as per ECSS

COB-ID and PDOs

- **Different priorities allow efficient exchange of data**
  - No need of predefined timeslots to ensure timely delivery
  - Priority not primarily based on unit, but on type of message (COB-ID).
- **Predefined connection set has been applied**
- **Remote Frame Request**
  - Not implemented (as recommended)
  - Functionality implemented by a pair of RX / TX PDO as described in “minimum implementation” in the ECSS
- **Limitation of size (8 bytes)**

SYNC

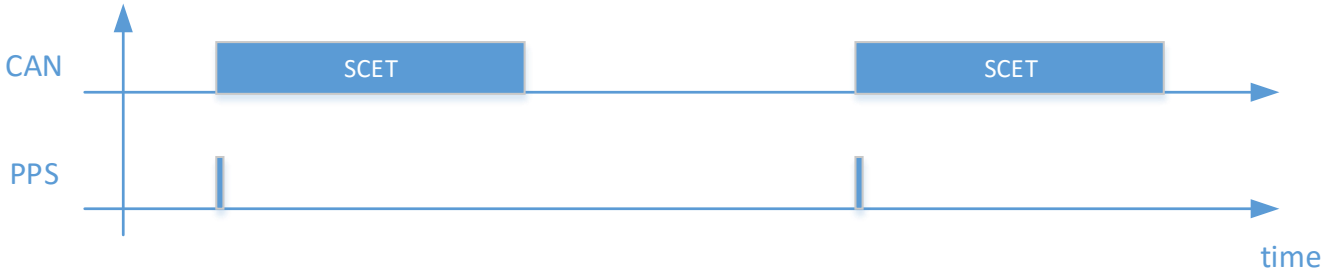
- **SYNC message is generated every 100 ms**
- **Synchronized to OBC time**
- **SYNC counter is reset every second (1..10)**

- **Main usage:**
  - Synchronize ADCS sensor readings
  - Synchronize ADCS actuator control
- **Usage is optional for other units**
  - E.g. to trigger PDO transfers

Time Synchronization

- Use SCET format from ECSS
- Time is represented in GPS time reference (Number of seconds from 05/01/1980 24:00:00)

- Additional PPS signal routed in parallel to CAN bus for high precision time synchronization
- Time value relates to previous PPS pulse that has been received on the PPS bus



- In case of PPS failure, time synchronization still possible as described in ECSS

Object Dictionary / EDS / DCF

Object Dictionary

- Standard way of access data in a unit
- Simple units don‘t implement OD

EDS / DCF

- Standard way of describing the behaviour of the unit
- EDS is provided for all units, even if no OD is implemented, to describe unit behaviour
- Saves time for SW development because one common format is used

PDO definitions

- All PDO definitions (mapping/parameters) are hardcoded
- Flexible PDO definition as per CANopen would highly increase complexity on unit and OBC side

SDO

- **Standard way to configure units**
- **Mandatory for CANopen, but not for ECSS CAN**
- **Simple units:**
  - No SDO implemented
  - Parameters can be set with PDO (command/response protocol)
  - Configuration on ground by configuration file or hardcoded
- **Complex units:**
  - Change configuration with SDO (but no dynamic PDO reconfiguration)
  - Firmware upload

- **SDO is only standard way to transmit bigger blocks of data**
- **No different priorities for different data types**

Current results

- **As the standard has been released in 2015, currently not much units with ECSS CAN are available on the market**
- **Adaption of existing units**
  - Use separate FPGA to convert from heritage protocol to ECSS CAN. Not really efficient in terms of power and cost
  - All ADCS heritage units are connected via ADCS I/F board (RTU)
  - On long term ECSS CAN must be implemented directly in the unit FPGA / processor
- **New developments**
  - Directly implement ECSS CAN in software or FPGA
  - Complex software implementation to handle different protocols (PDO, PDO command/response, SDO expedited, SDO segmented, SDO block)
  - Interest to have a standard way to transmit data >7 bytes in a prioritized way
- **High interest on CAN enabled microcontrollers for space**
- **CCIPC core**
  - Used by at least 2 suppliers
  - Additional R-CCIPC without SDO, but with more PDO would be very interesting

Conclusions

- LuxSpace has adopted ECSS CAN within the ESAIL project
- Phase C about to be started, first launch scheduled 2018
- Implementation based on “minimum implementation” from ECSS
- No major deviations from the ECSS CAN

- No major problems identified

- Need for transmitting larger data blocks with priority
- Big interest in space CAN enabled microcontroller
- Big interest in CAN-ECSS FPGA IP core with multiple PDO, but no SDO

Thank you