Unified Communication Framework

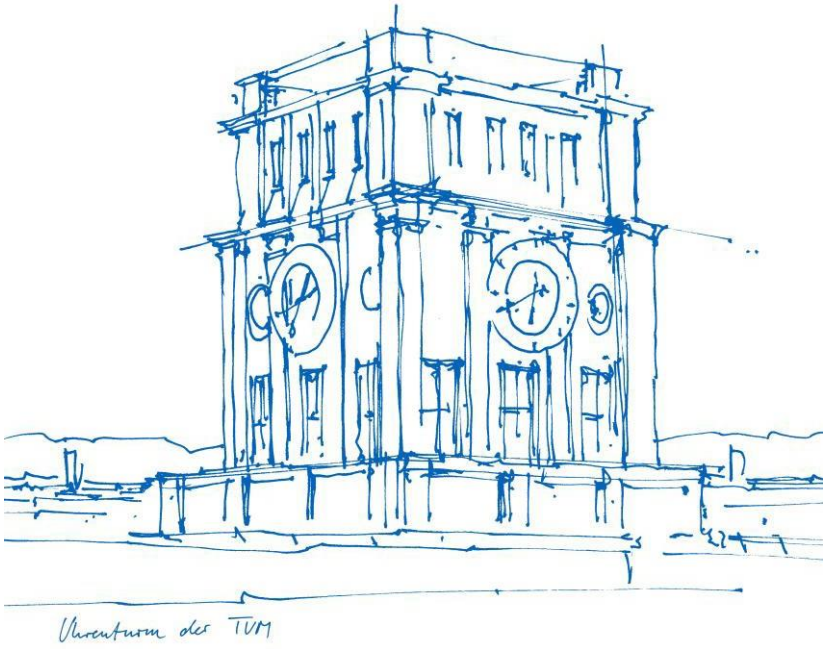
Stefan Huber, Dominic Gaisbauer

Technische Universität München

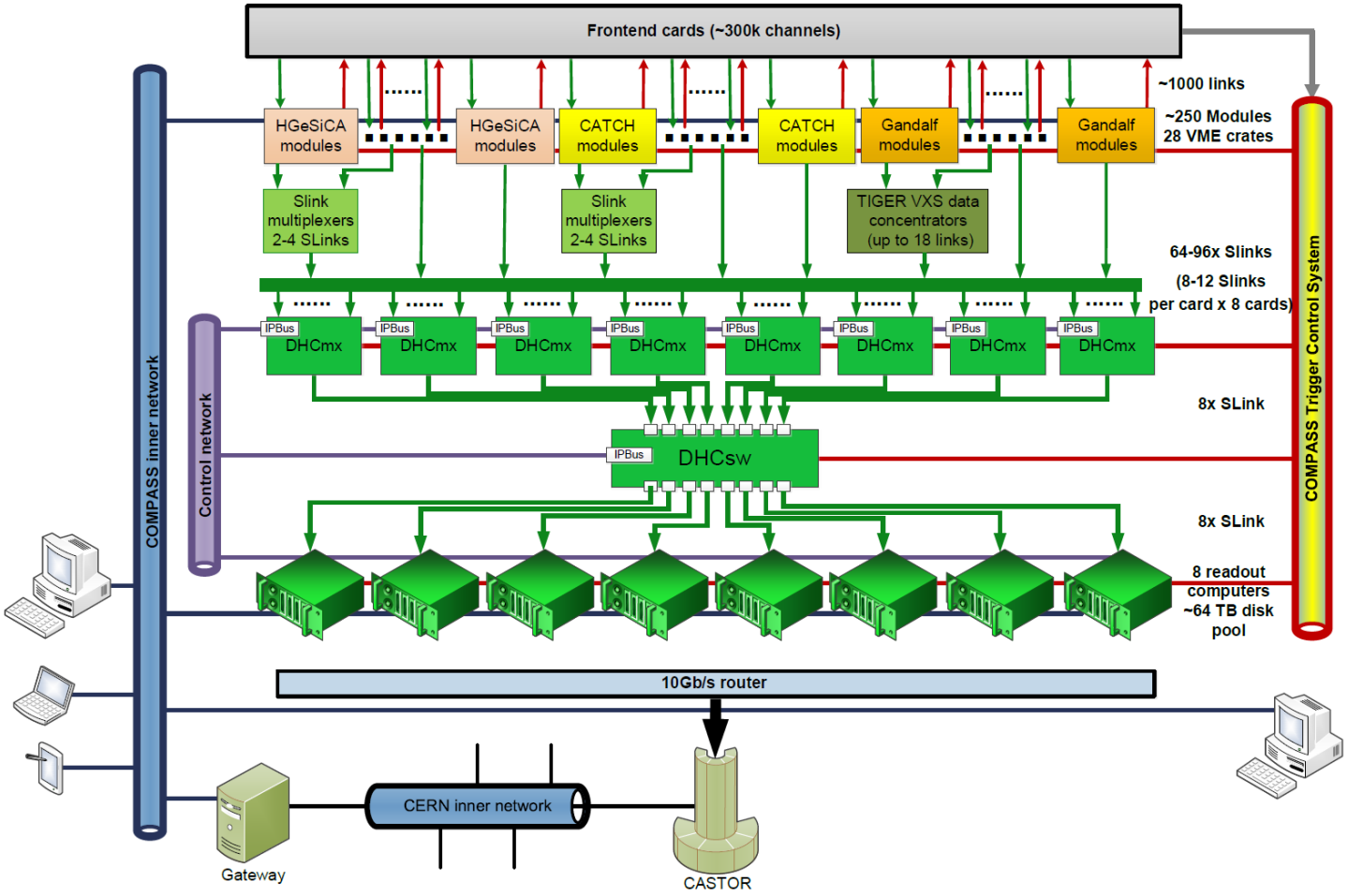
Institute for Hadronic Structure and

Fundamental Symmetries

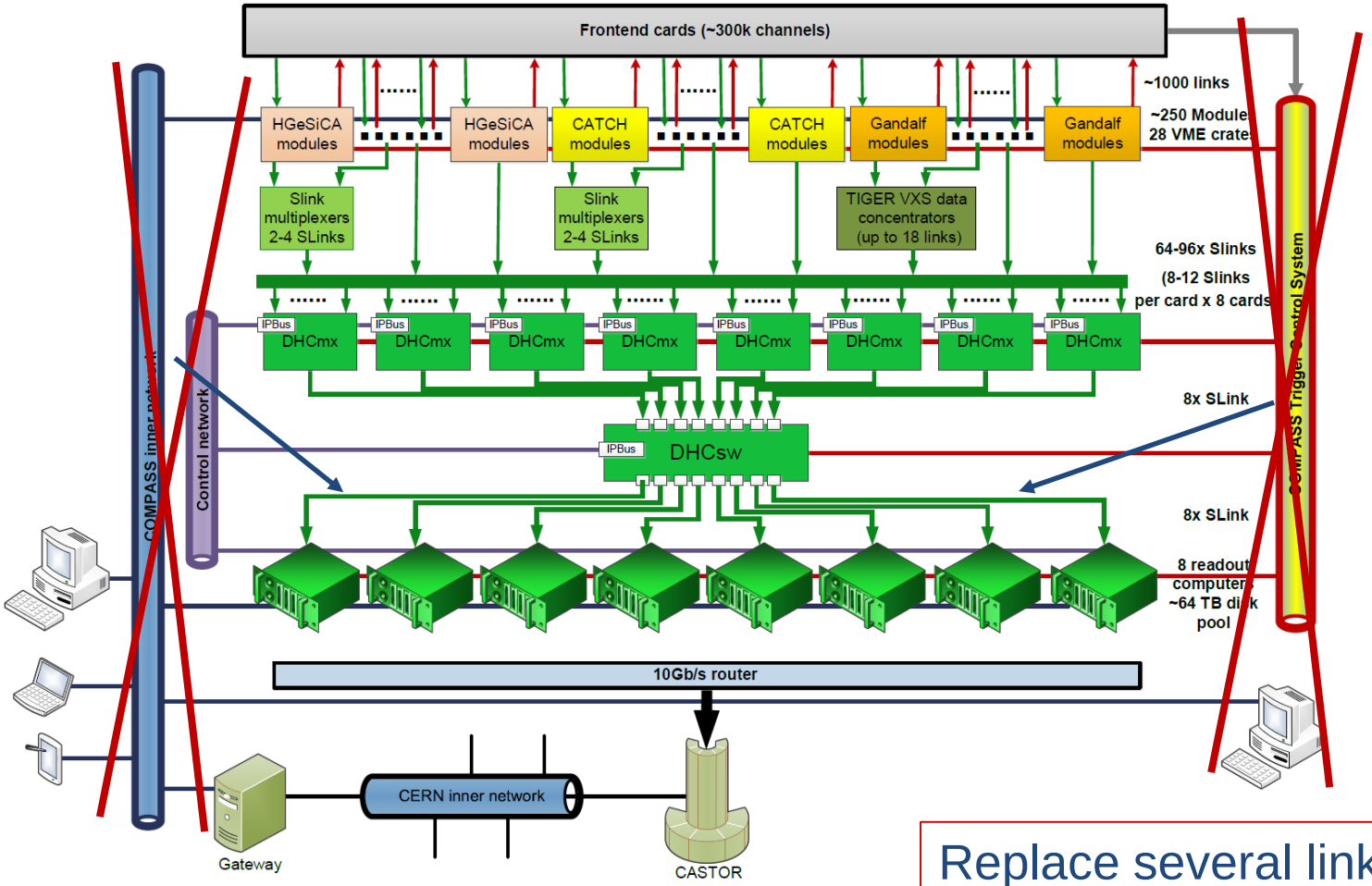
DAQFEET 2019



What can be gained in COMPASS



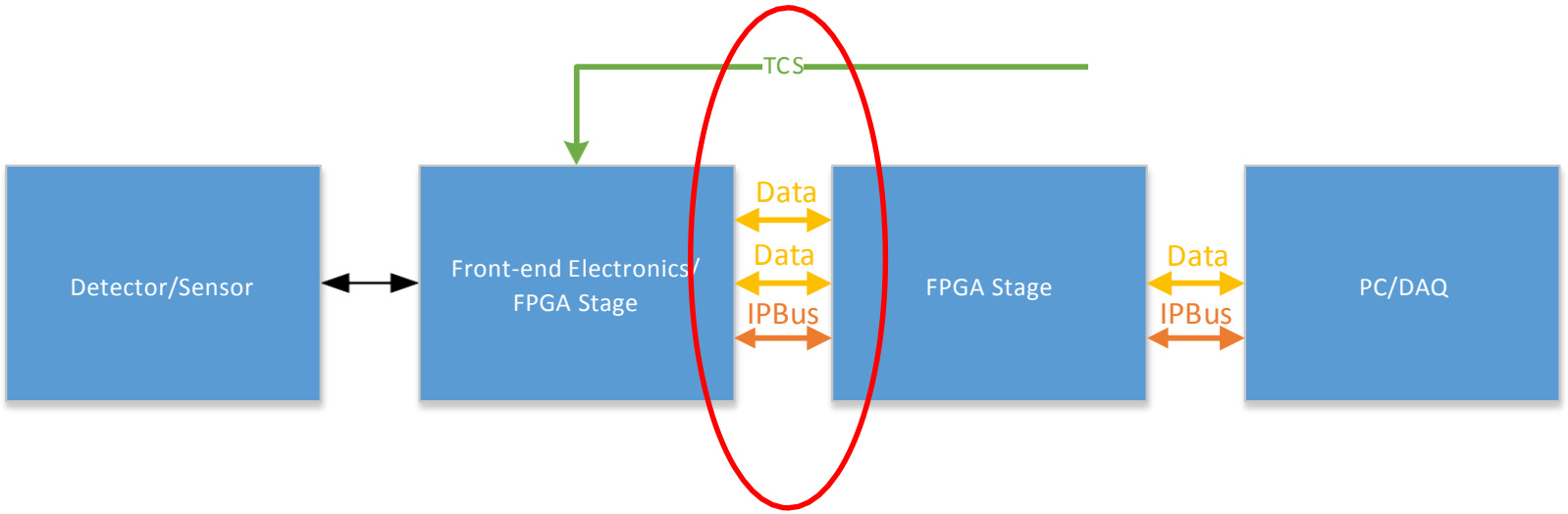
What can be gained in COMPASS



Replace several links to Fes by one optical fibre

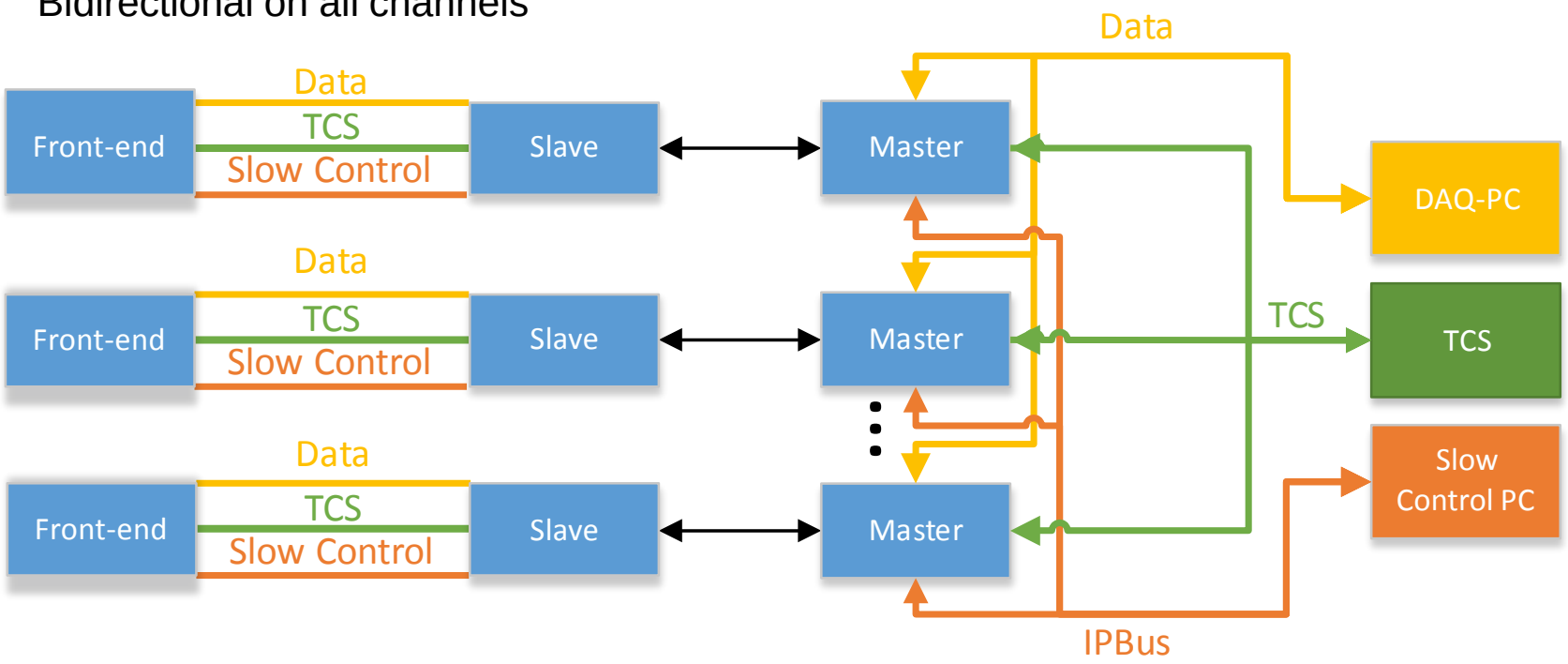
Unified Communication Framework (UCF)

- Single high-speed serial link for data, slow control, trigger, and timing information
- Up to 64 different communication channels (Trigger, Data, Ethernet)
- Fixed latency for one channel
- Priority handling for all channels
- Self recoverable after connection losses
- Independent from physical layer



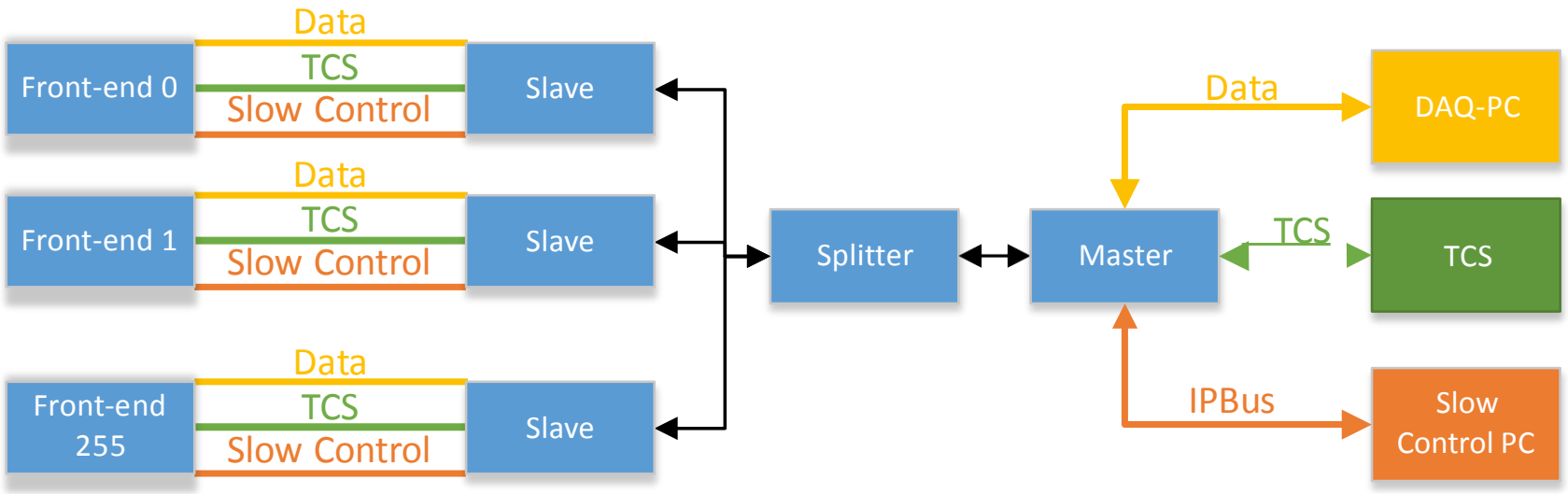
UCF – Example Topologies

- Point-to-Point topology:
  - Multiple or single 1:1 connections
  - Experiments with high data rates, …
  - Bidirectional on all channels



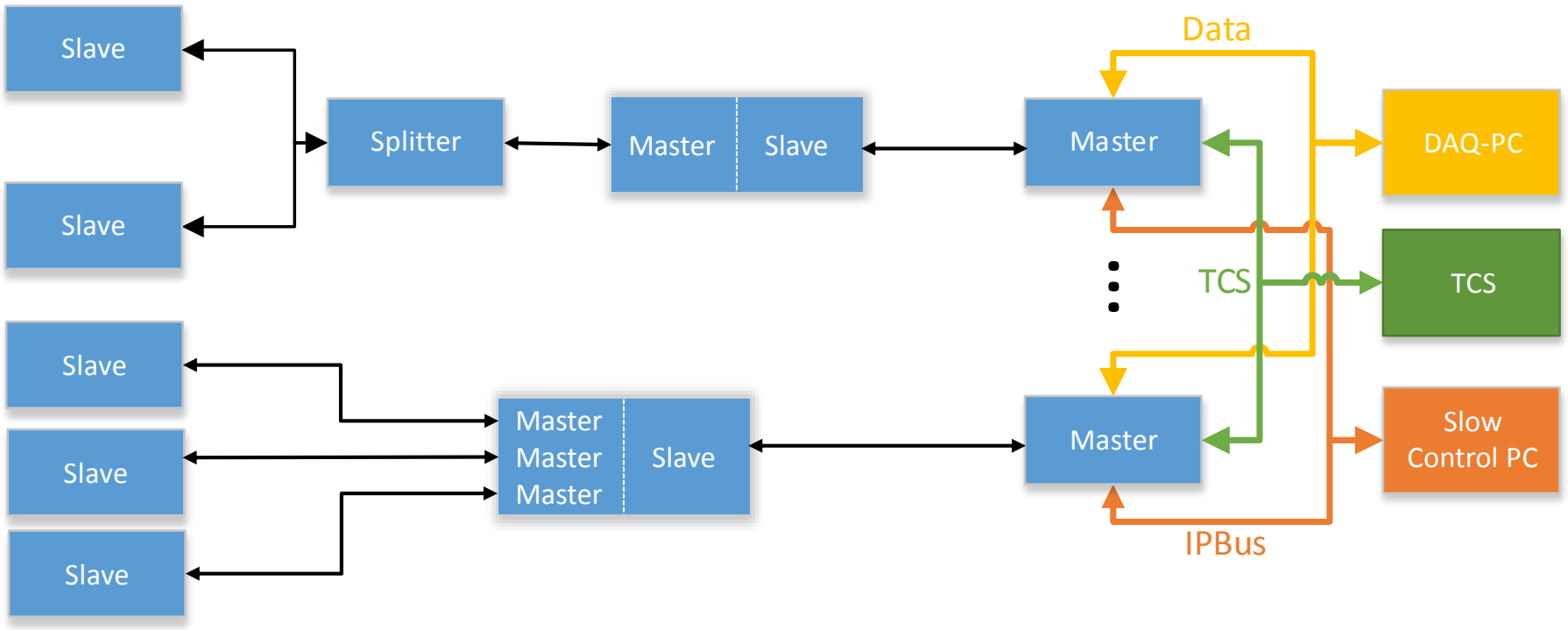
UCF – Example Topologies

- Star-like topology:
  - Single 1:n connections
  - Experiments with low data rates, time distribution systems …
  - Slaves share link in time division manner
  - Bidirectional on all channels



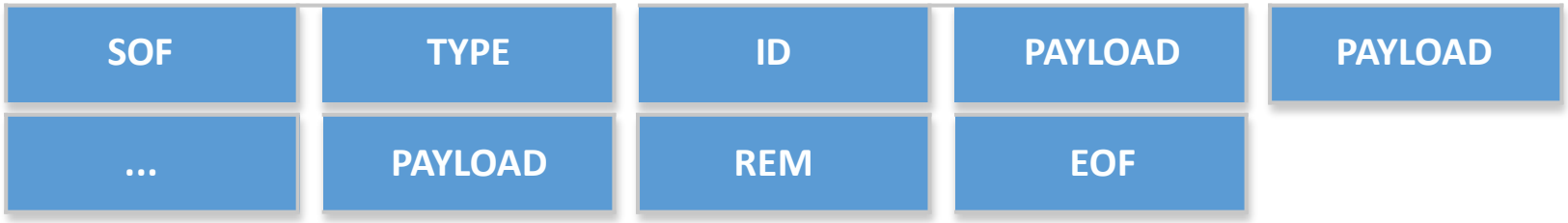
UCF – Example Topologies

- Hybrid topology:
  - Example for a full experiment
  - Several stages
  - Different topologies dependent on the FE requirements



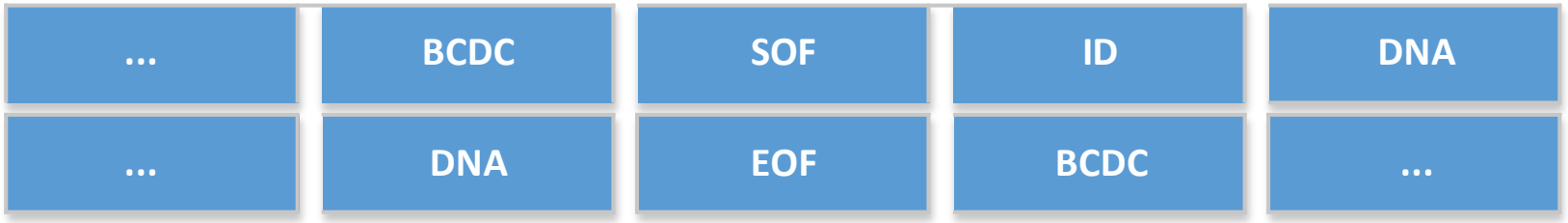
UCF – Low Layer Protocol

- Backbone of UCF
- Handles communication and initialization
- 8b/10b encoding scheme
- 10b K-characters for control and synchronization
- Protocol frames consist always of several character sequence:
  - Start of frame
  - Type of the message (either specific destination or broadcast)
  - Protocol identifier
  - Payload
  - Remainder defining the valid bytes in the last transmission
  - End of frame



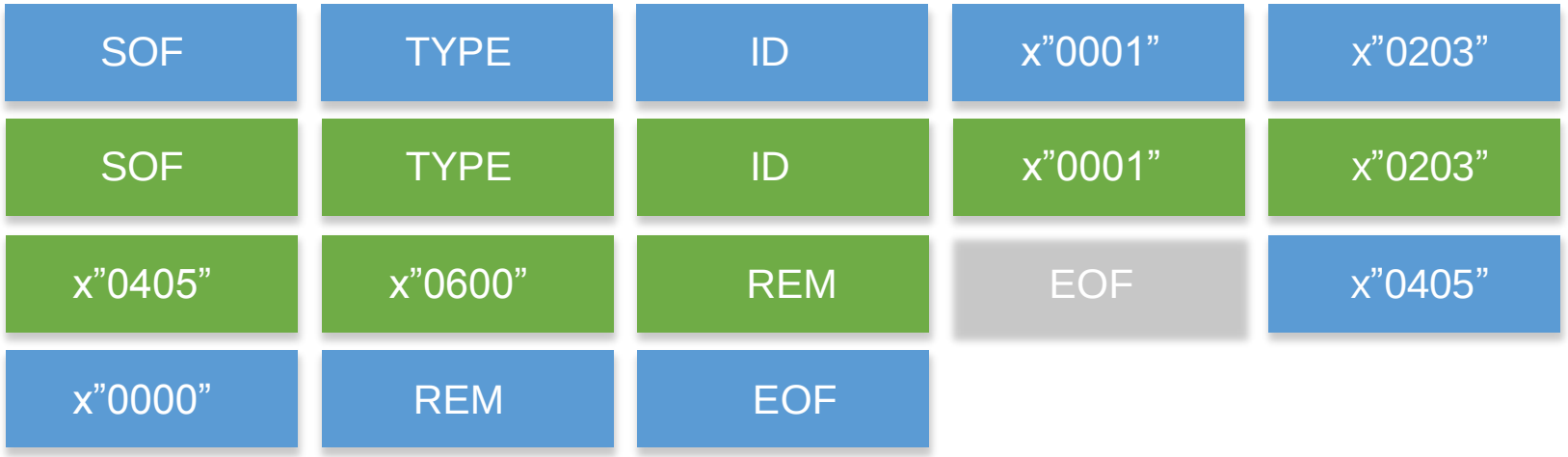
UCF – Initialization

- Fixed phase synchronization by sequence of two defined K-characters (x”BCDC”)
- Link polarity detection
- Synchronization character will be send for specific time to let the slaves synchronize
- Unique ID and IP assignment for all connection parties



UCF – Priority Handling

- All 64 communication channels have different priorities
- Protocol 0 has the highest and then it cascades down to the protocol 64 which has the lowest priority
- Frames with higher priority can always interrupt lower priority frames
- Maintains fixed latency for the timing channel



UCF – User Interface and Configuration

- UCF consists of a Logic block
- And a Hardware specific block
  to be adapted to new FPGA architectures
- Interface to user protocols via AXI-Stream
- Configuration parameters:
  - Link speed
  - Topology
  - Device type ….

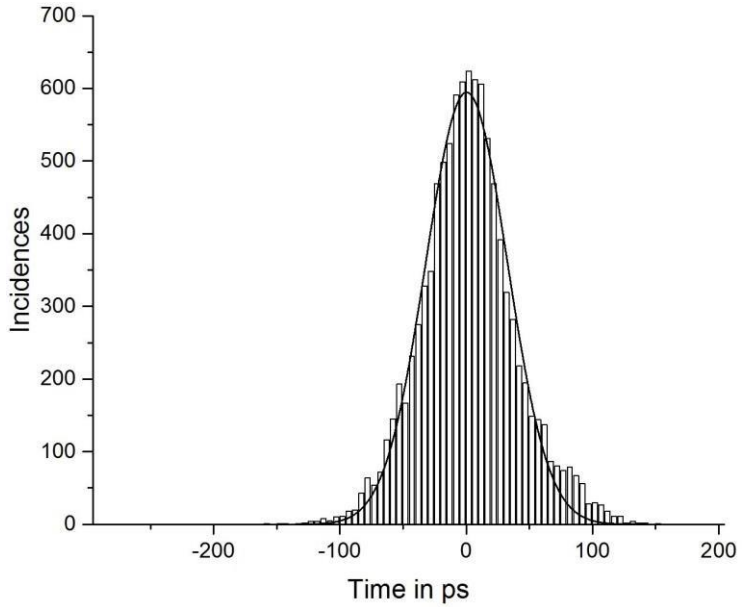
```
Inst_UCF_Wrapper : entity work.ucf_wrapper_ucf
generic map(
    recg.intInterfaces                  => intInterfaces,
    recg.intTransceivers                => intTransceivers,
    recg.intCyclesMin                   => intCyclesMin,
    recg.intSerials                     => intSerials,
    recg.intMaxTries                    => intMaxTries,
    recg.slP2PActivation                => slP2PActivation,
    recg.slCardPurpose                  => slCardPurpose,
    recg.slActivateSim                  => slActivateSim,
    recg.strDeviceType                  => strDeviceType,
    recg.PLL_DIVSEL_FB                  => PLL_DIVSEL_FB,
    recg.PLL_DIVSEL_OUT                 => PLL_DIVSEL_OUT,
    recg.PLL_DIVSEL_REF                 => PLL_DIVSEL_REF,
    recg.PLL_DIVSEL45_FB                => PLL_DIVSEL45_FB,
    recg.CLK25_DIVIDER                  => CLK25_DIVIDER,
    recg.RX_Polarity                    => RX_Polarity,
    recg.TX_Polarity                    => TX_Polarity
)port map(
  clk                                   => refclk125_bufg,
  reci                                  => reci(0),
  reco                                  => reco(0)
);
SFP_TX12_N                              <= reco(0).slTXN;
SFP_TX12_P                              <= reco(0).slTXP;
reci(0).slRXP                           <= SFP_RX12_P;
reci(0).slRXN                           <= SFP_RX12_N;
reci(0).slReferenceClk                  <= refclk127;
reci(0).slLinkReset                     <= '0';
reci(0).slLinkReinit                    <= '0';
reci(0).slvDNAList(0)                   <= x"00C0426EBA540890";
reci(0).slvDNA                          <= x"00C0426EBA540890";
reci(0).DIFFCTRL_IN                     <= DEF_DIFFCTRL;
reci(0).PRECTRL_IN                      <= DEF_PRECTRL;
reci(0).POSTCTRL_IN                     <= DEF_POSTCTRL;
```

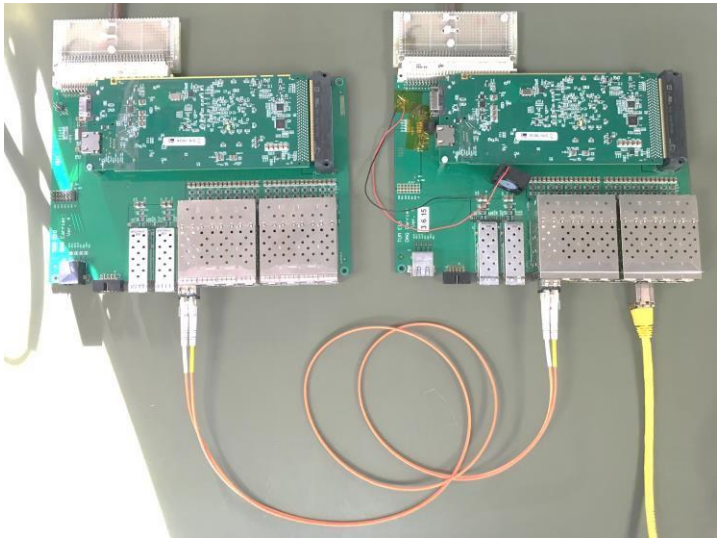
UCF Versions

- Version 1
  - 2Byte AXI-Stream interface
  - Version currently used in Belle II
  - Implemented for Virtex6, Artix7
- Version 2
  - 4Byte AXI-Stream interface
  - Supports backpressure
  - Link polarity detection
  - Transmission of 32bit parameters during startup
  - Implemented for Virtex6, Artix7, KintexUltraScale

UCF – Tests and Measurements

- Point-to-Point topology with 1 slave and 1 master
- 2.5 Gbit/s link speed
- Virtex 6 as slave and master
- Recovered clock jitter (σ) of 23 ps
- Requires 2 % slice register and 4 % slice LUT utilization on a Virtex 6 LX130T

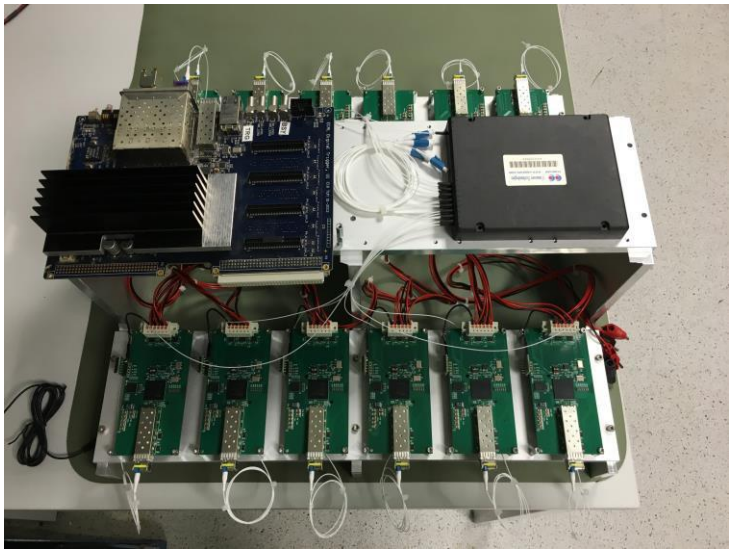




UCF – Tests and Measurements

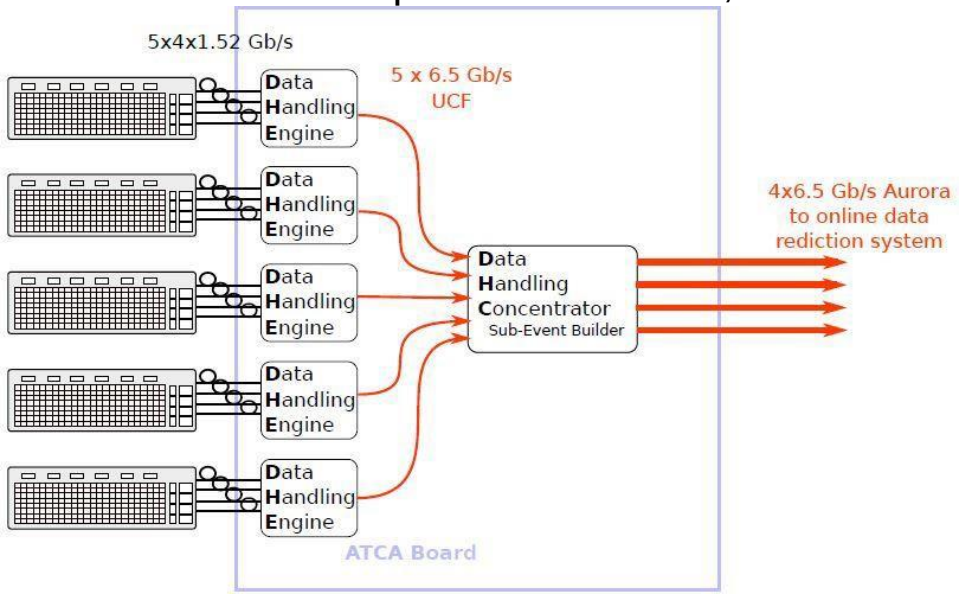
- Star-like topology with 12 slaves and 1 master
- 1.25 Gbit/s link speed
- Spartan 6 FPGA as slave and Virtex 6 as master
- Switching time of 16 µs (includes character transmission and synchronization)
- Long term stability test with 99 % link utilization over two weeks
- IPBus over UCF

Transmission Time [µs]	Efficiency [%]
25000	99,93
10000	99,84
1000	98,42
500	96,90
100	86,20



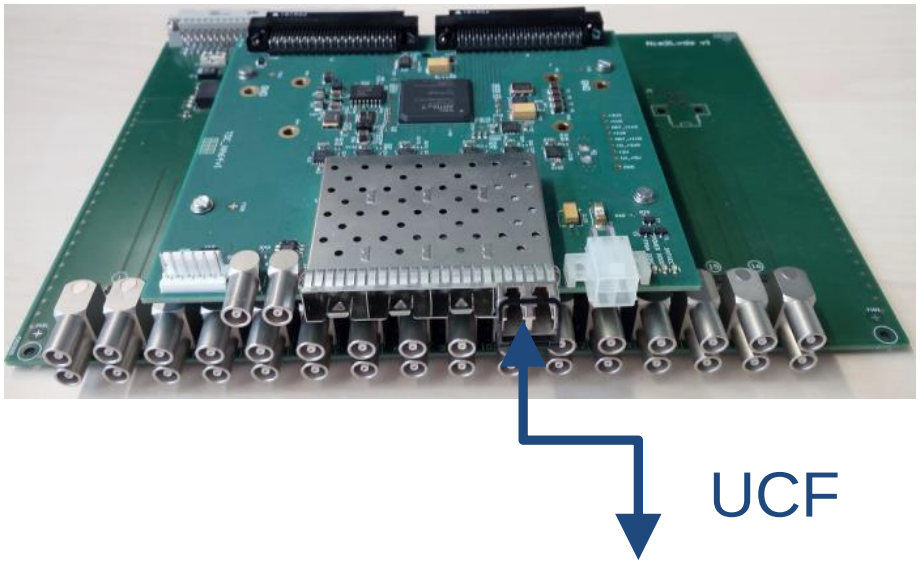
UCF – Belle II Pixel Detector

- Point-to-Point topology with 1 DHC and 5 DHE
- 22 GB/s data rate of the detector
- 127.21 MHz reference clock
- ~30ps clock jitter of the recovered clock
- 6.5 Gbit/s link speed, currently 2.5 Gbit/s
- IPBus, data and trigger distribution
- Communication via copper lines on ATCA carrier
- Four modules operational at KEK, UCF reliable for about a year



UCF – COMPASS iFTDC

- In 2018
  - UCF with 2Byte interface
  - 155.52MHz reference clock
  - 2.5Gbit/s to TDC multiplexer
  - 3 Protocols
    - TCS
    - IPBus
    - Data
- Future
  - Two UCF links
    - TDC multiplexer
    - Trigger processor



UCF – Conclusion

- UCF can be used as an interface between electronics modules in COMPASS
- Fixed latency for one channel
- Standardized ACI-Stream interface for user data
- Multiple 1:n and 1:1 connections possible
- IPBus over UCF
- Tested setups
    - Belle II
    - iFTDC
    - Starlike => LabTests

Plans

- Switch to 4 Byte version
- Deploy it as standard interface for all Fes
- Get feedback from other institutes about missing features and possible improvements

Thank you for your attention