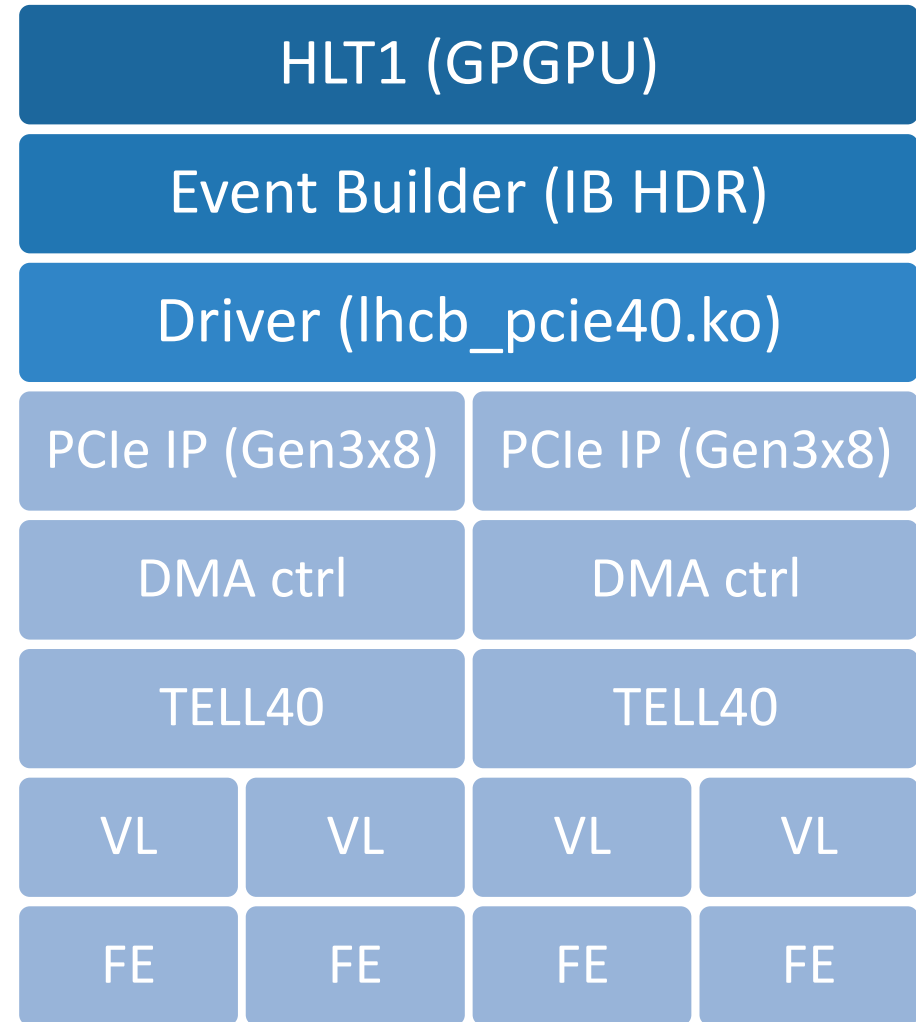# PCIe40→PCIe400 PCIe IP porting considerations

PAOLO DURANTE
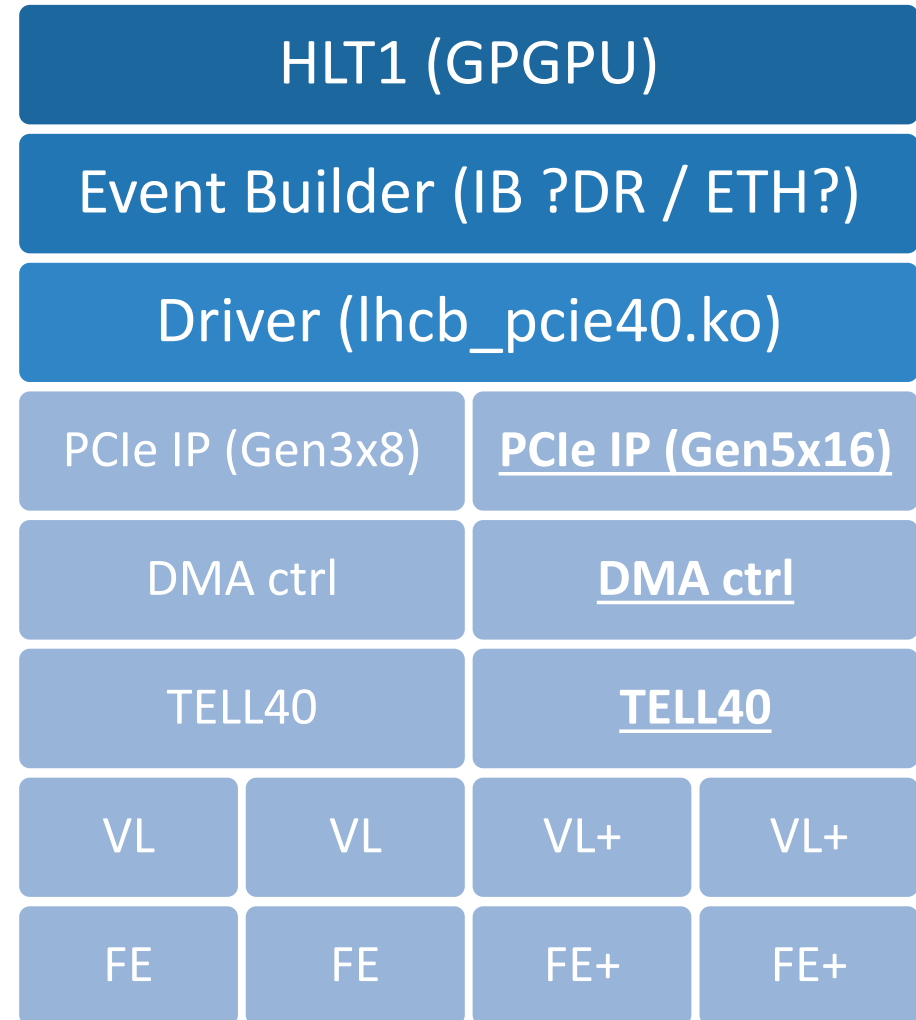
LHCb HW/SW stack (Run3)

- In use at LHCb since 2021

- Front-End with Versatile Link protocol
  - ~5Gbps

- 110Gbps DMA interface from TELL40
  - 55Gbps x 2

- Up to 40 Tbit/s aggregate EB bandwidth

- New HLT1 implementation on GPU

- Each layer of the stack is monitored and configured by the LHCb control system

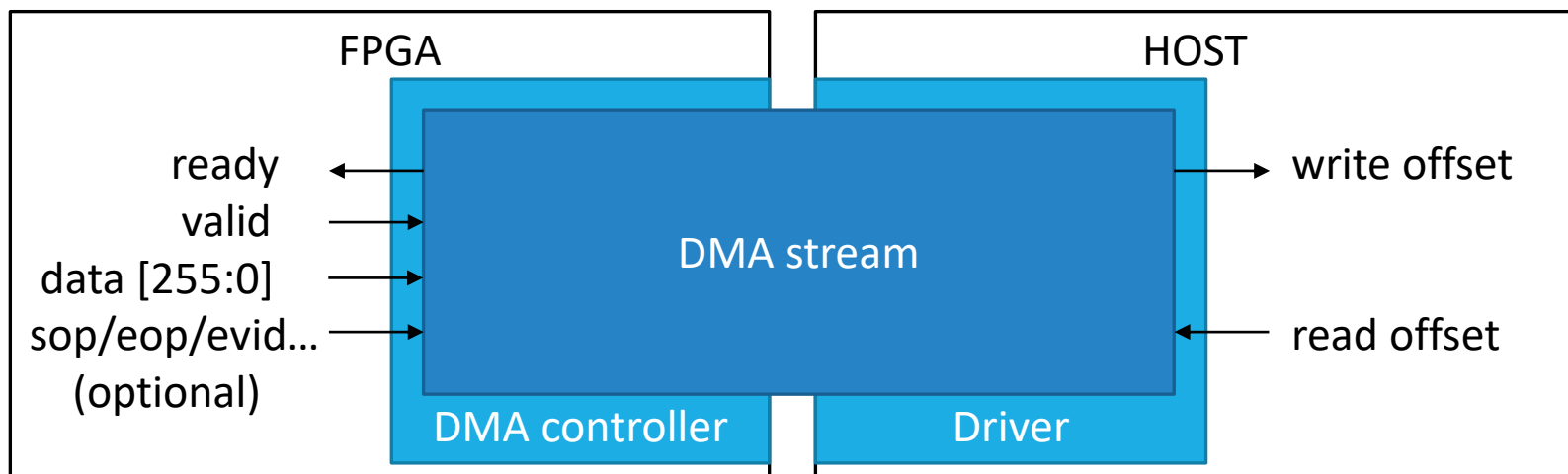| HLT1 (GPGPU) | | | |
Event Builder (IB HDR)			
Driver (lhcb_pcie40.ko)			
PCIe IP (Gen3x8)		PCIe IP (Gen3x8)	
DMA ctrl		DMA ctrl	
TELL40		TELL40	
VL	VL	VL	VL
FE	FE	FE	FE

LHCb HW/SW stack (Run4)

- Expected to go into production in 2029

- Versatile Link+ Front-End uplinks
  - 10Gbps

- 500Gbps PCIe interface from PCIe400
  - Gen5 = 32T/s/lane/direction

- Must coexist with Run3 system
  - Same timing distribution
    - PON
  - Same event builder architecture?
    - To be finalized, depends on system size
  - Same driver?
    - Not required but nice to have
  - Same control system

HLT1 (GPGPU)			
Event Builder (IB ?DR / ETH?)			
Driver (lhcb_pcie40.ko)			
PCIe IP (Gen3x8)		**PCIe IP (Gen5x16)**	
DMA ctrl		**DMA ctrl**	
TELL40		**TELL40**	
VL	VL	VL+	VL+
FE	FE	FE+	FE+

Current PCIe implementation

Already described in previous presentation (June 8th)

- Implementation provides a very generic abstraction (just a stream of words)
  - Each word is 256bits

- Supports multiple DMA channels, Host-to-Device (Tx) direction
  - Device-to-Host (Rx) direction implementation exists but is not tested

- 55Gbps/stream (Gen3x8), zero-copy interface, InfiniBand RDMA compatible

- Up to 1TiB addressable memory per Tx stream

External descriptor controller

- Altera PCIe IP supports external controller interface
  - Avalon ST for descriptors
  - Avalon MM for buffer access

- Compatible between PCIe IPs from different device families

- Allows LHCb PCIe implementation to support multiple device families with minimal modifications (device wrappers)
  - Stratix IV
  - Stratix V
  - Arria V
  - Arria 10
  - What about Agilex?

Figure 5-2: Avalon-MM DMA Bridge with External Descriptor Controller

MCDMA PCIe IP

- Only supports H-Tile, P-Tile, F-Tile

- Up to Gen4x16 (P-Tile devkit)
  - 512bit bus
  - 256bit bus (single x8 only)

- Similar enough to "older" external controller interface
  - Should be possible to support without a lot of extra work
  - Still requires bus width adaptation (256→512)

- Supported by F-Series devkit at CERN
  - But not tested

- Supported by Intel DPDK driver
  - But not tested

MCDMA with External Controller Interface
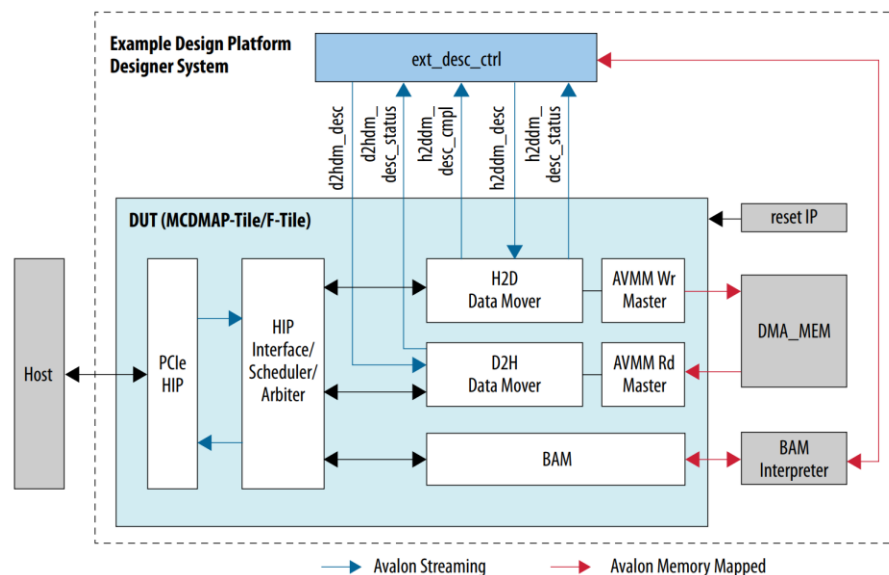
R-Tile PCIe IP

- Up to Gen5x16 (R-Tile)
  - 1024bit bus @ 500MHz
  - Segmented as 4x256 bits

- R-Tile IP still under development
  - Last UG document released on Oct 7th

- **No Avalon MM interface**
  - **Only Avalon ST**

- <u>**No external DMA controller interface**</u>
  - <u>**Might have to implement our own**</u>

- Will MCDMA IP be ported to R-Tile?
 - Have to ask Intel

R-tile Top-Level Block Diagram in PCI Express Mode
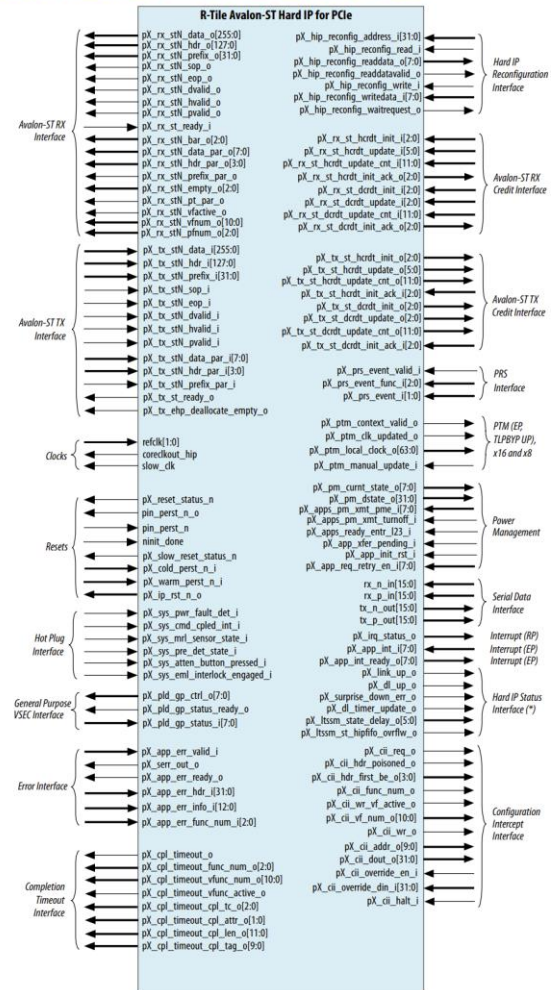
Next steps

- Understand MCDMA IP
  - Run simulation model
  - Program example design on devkit
  - Study DPDK driver
  - Study compatibility with old external controller interface

- Update LHCb DMA controller for configurable bus width
  - 256, 512, 1024

- Port MCDMA IP to R-Tile?
  - Understand TLP segmentation →
  - Feasible with a wrapper entity?

- Create our MCDMA from scratch?

- Ask Intel for timeline on DMA support for R-Tile IP

External Descriptor Controller Example Design
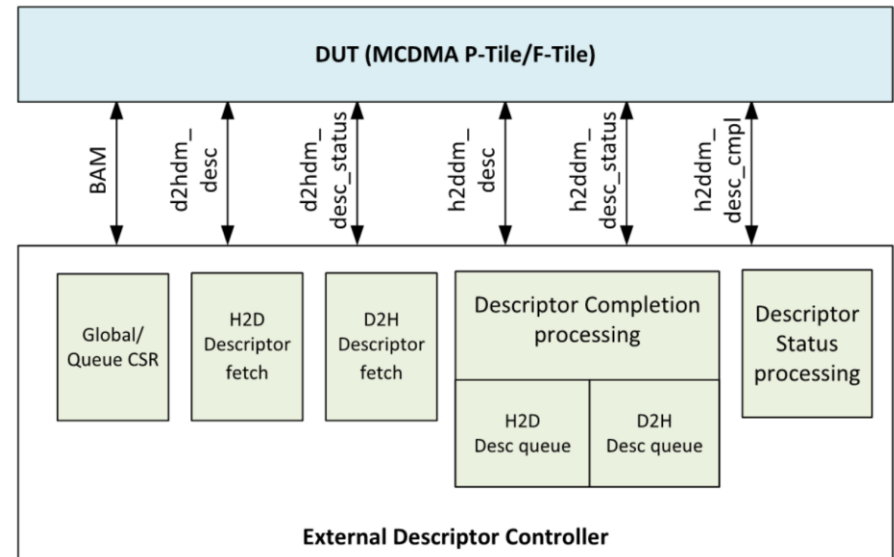
The Avalon-ST interface uses a split-bus architecture. In the x16 and x8 configurations, the 1024-bit Avalon-ST data bus consists of four segments of 256-bit data. This is done to improve the bandwidth efficiency of this interface. With this split-bus architecture, multiple TLP packets can be transmitted or received in a single clock cycle. For more details, refer to Avalon Streaming Interface.