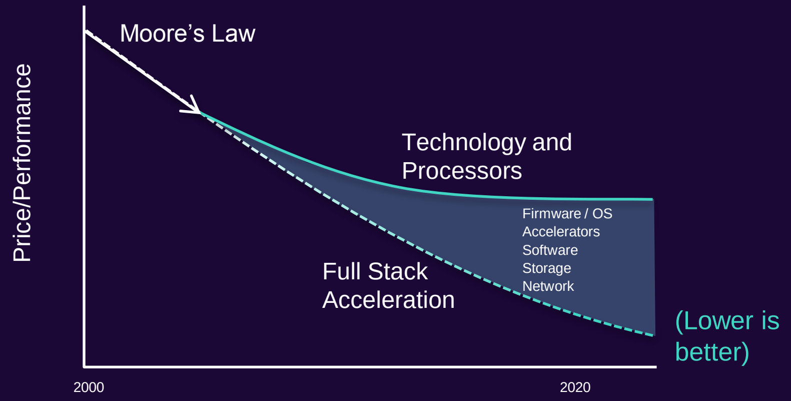
Changing the Game with OpenCAPI

Steve Fields
IBM Fellow
Chief Engineer of Power Systems

Fundamental forces are accelerating change in our industry

IT innovation can no longer come from just the processor



Full system stack innovation required

IT consumption models are expanding

Cognitive

Custom Hyperscale Data Centers

Hybrid Cloud

Open Solutions

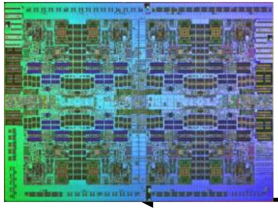
Not only is Moore's Law "coming to an end in practical term, in that chip speeds can be expected to stall, but it is actually likely to roll back in terms of performance …"– William Holt, Intel Executive Vice President and General Manager

Acceleration Can Have a Bigger Impact on Cost/Performance than Processors

Multiple Options to Design with POWER Technology Within OpenPOWER

Hardware

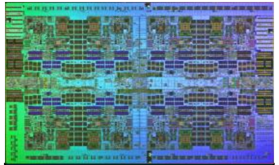
POWER8



CAPI over PCIe

“Standard POWER Products” – 2014

Framework to Integrate System IP on Chip



Customizable

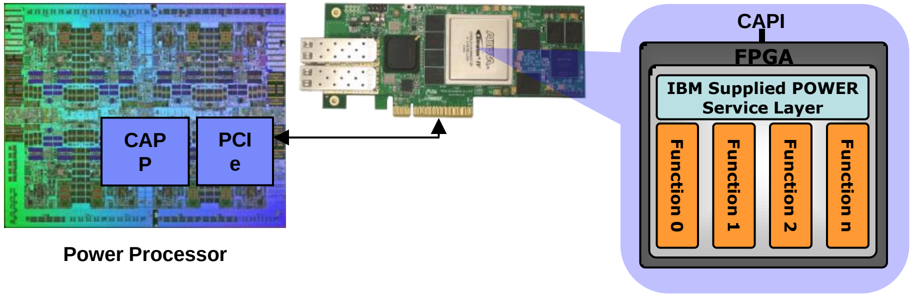
Industry IP License Model

“Custom POWER SoC” – Future

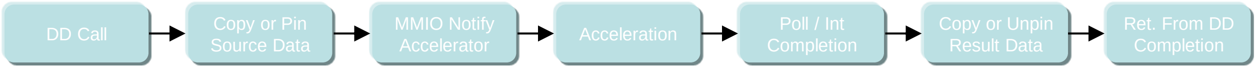
OpenCAPI & Nvlink

FPGA
GPU
Storage Class Memories
Highperf Networking
Other Stuff…

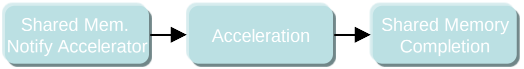
POWER8 CAPI Overview



Typical I/O Model Flow



Flow with a Coherent Model



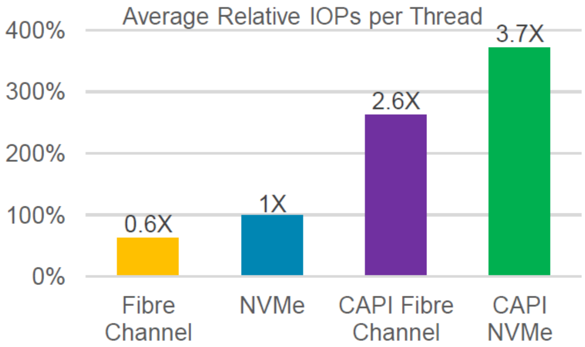
Advantages of Coherent Attachment Over I/O Attachment

- **Virtual Addressing & Data Caching**
  - Shared Memory
  - Lower latency for highly referenced data
- **Easier, More Natural Programming Model**
  - Traditional thread level programming
  - Long latency of I/O typically requires restructuring of application
- **Enables Applications Not Possible on I/O**
  - Pointer chasing, etc…

CAPI 2.0 NVMe Flash Accelerator (FlashGT+)

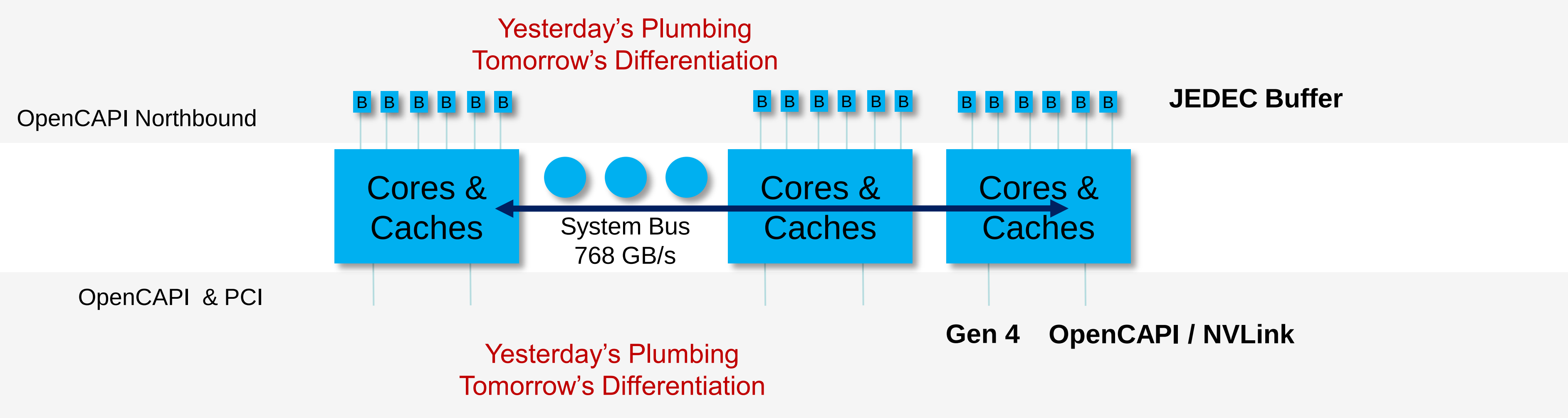
2016 Flash Adapter (CAPI 1.0)

- FPGA Controller
- 2x 960GB M.2 SSDs
- Supports User-Mode KVS and Block APIs + Linux CAPI filesystem
- ~4x reduction in CPU overhead compared to NVME



Power Systems

Future Evolution of System Architecture



Yesterday's Plumbing
Tomorrow's Differentiation

OpenCAPI Northbound

B B B B B B B B B B B B B B B B B B

JEDEC Buffer

Cores & Caches

Cores & Caches

Cores & Caches

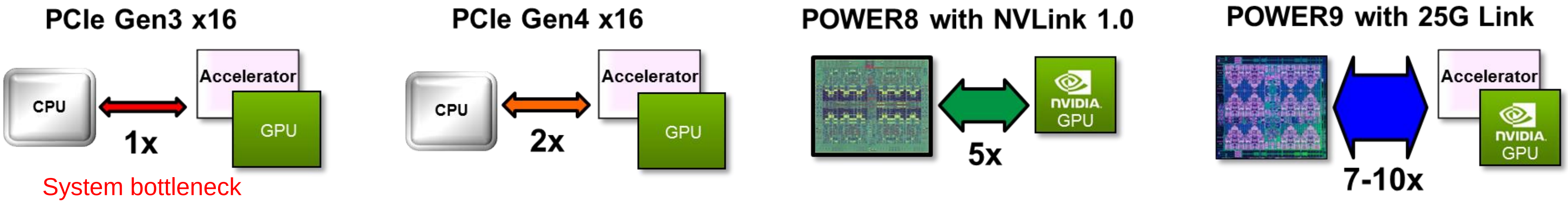
System Bus
768 GB/s

OpenCAPI & PCI

Gen 4 OpenCAPI / NVLink

Yesterday's Plumbing
Tomorrow's Differentiation

CPU/Accelerator Bandwidth



PCIe Gen3 x16

1x

System bottleneck

PCIe Gen4 x16

2x

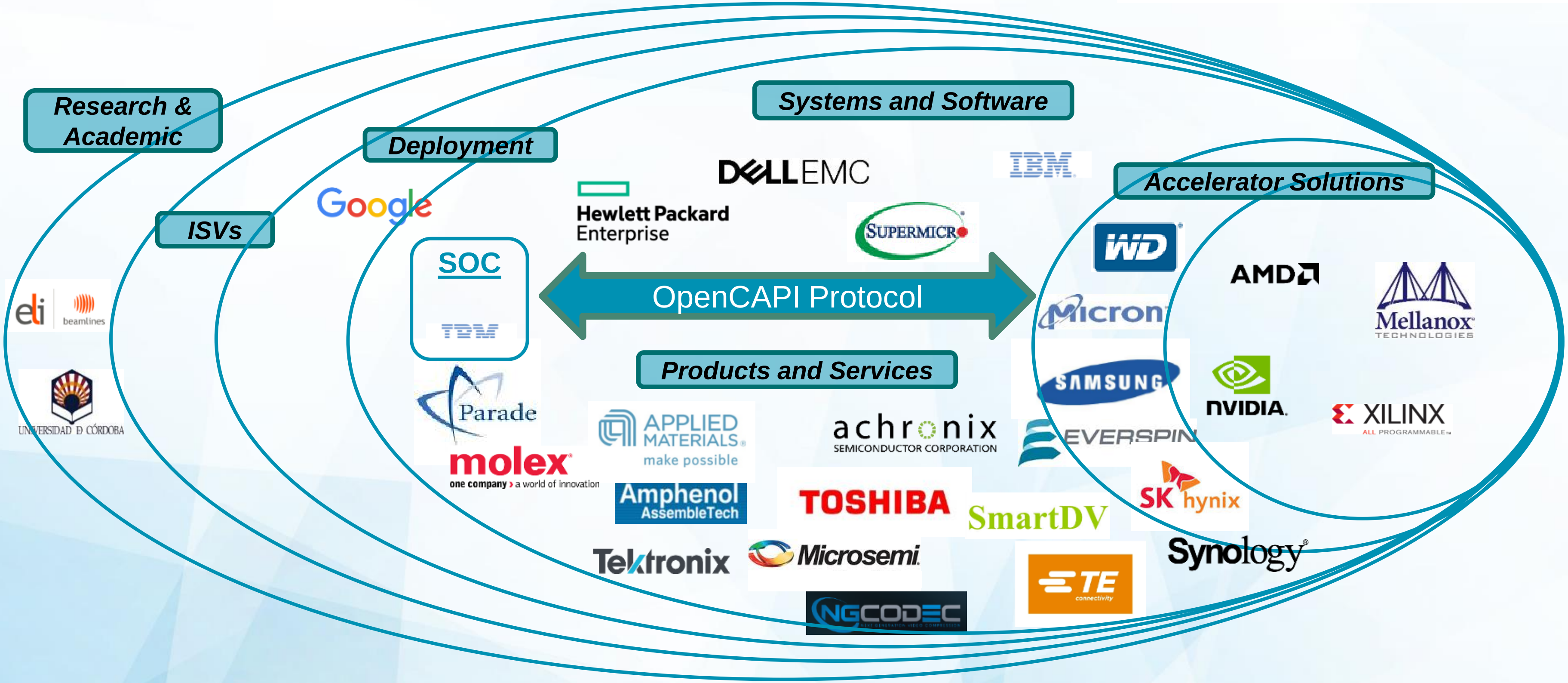
POWER8 with NVLink 1.0

5x

POWER9 with 25G Link

7-10x

Cross Industry Collaboration and Innovation



Research & Academic

ISVs

Deployment

Systems and Software

Accelerator Solutions

SOC

OpenCAPI Protocol

Products and Services

Welcoming new members in all areas of the ecosystem

Comparison of IBM CAPI Implementations

OpenCAPI

Feature	CAPI 1.0	CAPI 2.0	OpenCAPI 3.0	OpenCAPI 4.0
Processor Generation	POWER8	POWER9	POWER9	Future
CAPI Logic Placement	FPGA/ASIC	FPGA/ASIC	NA DL/TL on Host DLx/TLx on endpoint FPGA/ASIC	NA DL/TL on Host DLx/TLx on endpoint FPGA/ASIC
Interface Lanes per Instance Lane bit rate	PCIe Gen3 x8/x16 8 Gb/s	PCIe Gen4 2 x (Dual x8) 16 Gb/s	Direct 25G x8 25 Gb/s	Direct 25G+ x4, x8, x16, x32 25+ Gb/s
Address Translation on CPU	No – HPT	Yes – HPT/Radix	Yes – HPT/Radix	Yes – HPT/Radix
Native DMA from Endpoint Accelerator	No	Yes	Yes	Yes
Home Agent Memory on OpenCAPI Endpoint with Load/Store Access	No	No	Yes	Yes
Native Atomic Ops to Host Processor Memory from Accelerator	No	Yes	Yes	Yes
Accelerator -> HW Thread Wake-up	No	Yes	Yes	Yes
Low-latency small message push 128B Writes to Accelerator	MMIO 4/8B only	MMIO 4/8B only	MMIO 4/8B only	Yes
Host Memory Caching Function on Accelerator	Real Address Cache in PSL	Real Address Cache in PSL	No	Effective Address Cache in Accelerator

Virtual Addressing

- **An OpenCAPI device operates in the virtual address spaces of the applications that it supports**
  - Eliminates kernel and device driver software overhead
  - Improves accelerator performance
  - Allows device to operate directly on application memory without kernel-level data copies or pinned pages
  - Simplifies programming effort to integrate accelerators into applications

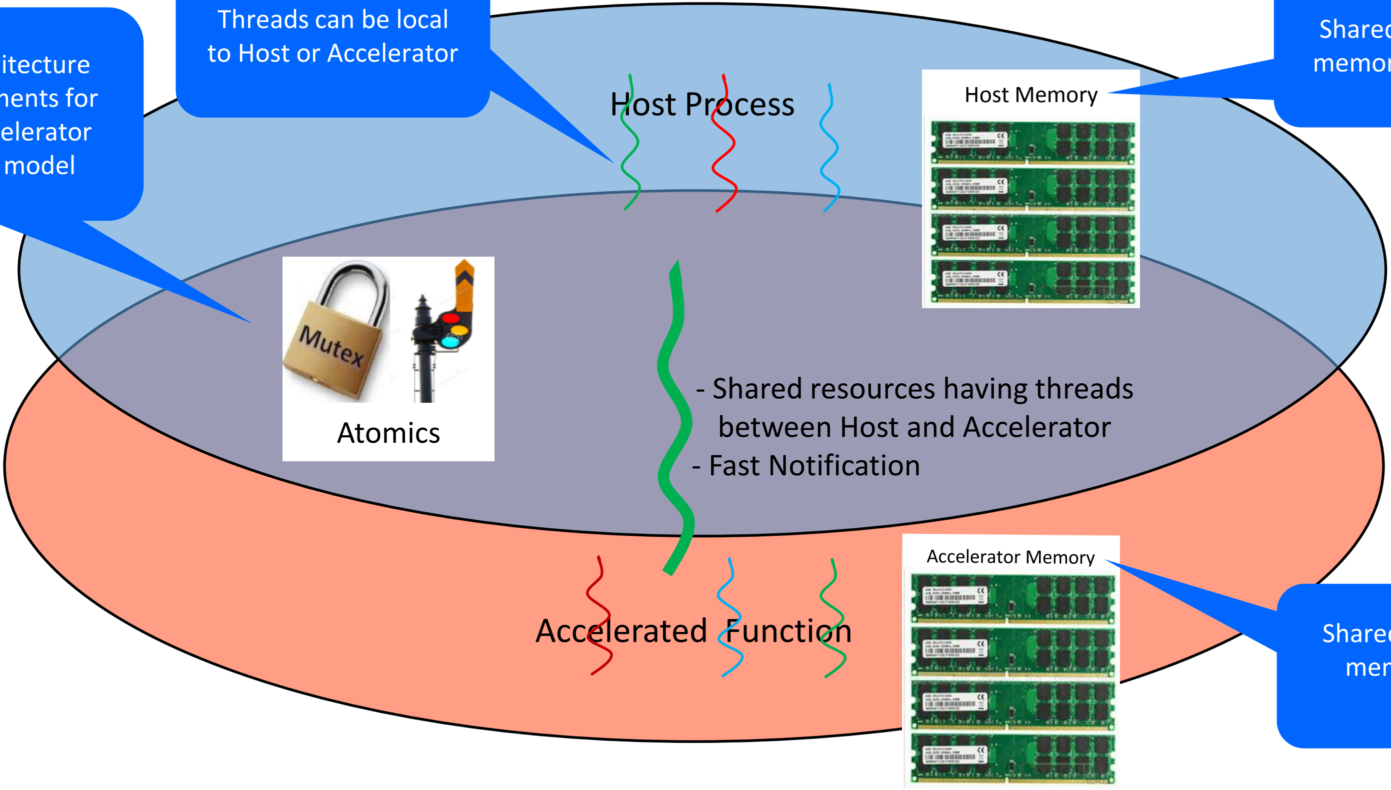
- **The Virtual-to-Physical Address Translation occurs in the host CPU**
  - Reduces design complexity of OpenCAPI-attached devices
  - Makes it easier to ensure interoperability between an OpenCAPI device and multiple CPU architectures
  - Since the OpenCAPI device never has access to a physical address, this eliminates the possibility of a defective or malicious device accessing memory locations belonging to the kernel or other applications that it is not authorized to access

OpenCAPI Coherence Programming Model

OpenCAPI Architecture offers advancements for the Host<->Accelerator programming model

Threads can be local to Host or Accelerator

Shared Host-Accelerator memory attached to Host



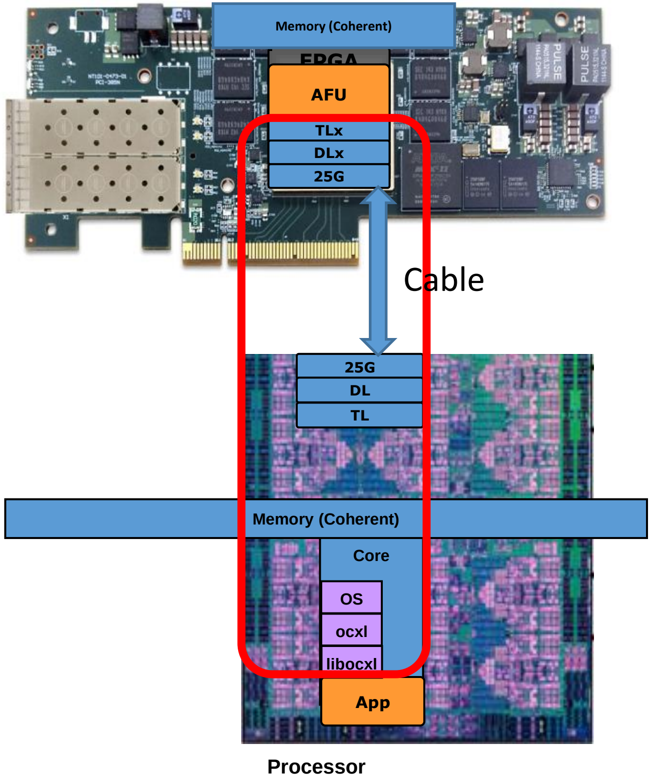
- Shared resources having threads between Host and Accelerator
- Fast Notification

Shared Host-Accelerator memory attached to Accelerator

Where is Processor Service Layer (PSL)?

- No PSL module on OpenCAPI device
- The Virtual-to-Physical Address Translation occurs in the host CPU
  - Reduces design complexity of OpenCAPI-attached devices
  - Makes it easier to ensure interoperability between an OpenCAPI device and multiple CPU architectures
  - Since the OpenCAPI device never has access to a physical address, this eliminates the possibility of a defective or malicious device accessing memory locations belonging to the kernel or other applications that it is not authorized to access

OpenCapi Simulation Environment (OCSE)
AFU/Application Development Enablement



- Hardware and reference designs to enable coherent acceleration
- Operating system enablement
  - Little Endian Linux
  - Reference Kernel Driver (ocxl)
  - Reference User Library (libocxl)
- Customer application and accelerator

- OCSE models the red outlined area
- OCSE enables AFU and Application co-simulation **when the reference libocxl and reference TLx/DLx are used.**
- OCSE dependencies
  - Fixed reference TLx/AFU interface
  - Fixed reference libocxl user API
- Will be contributed to the OpenCAPI consortium

Exerciser Examples

- Two exerciser samples will be provided to members of the OpenCAPI consortium
    - MemCopy
        - The Memcopy example is a DMA mover from source address -> destination address using Virtual Addressing and includes these features
            - Configuration and MMIO Register Space
            - acTag Table
                - Ranging in size from 1 to 64 entries
                - Used for Bus/Device/Function and Process ID identification
            - 512 processes/contexts and 32 engines supporting up to 2K transfers
    - Memory Home Agent
        - The Memory Home Agent example provides for DDR4 memory to be implemented off the endpoint OpenCAPI accelerator to act as a coherent extension to the host processor memory
        - The Memory Home Agent example includes these features
            - Configuration and MMIO Register Space
            - acTag Table
                - Ranging in size from 1 to x entries
                - Used for Bus/Device/Function and Process ID identification
            - Credit counters that keep track of command, data and responses
            - Read and Write engines for performance

Reference Card Design

- Definition of FPGA reference card is being driven as part of the 25G workgroup within the OpenPower consortium
- Definition of the cable(s) are also driven as part of the 25G workgroup within the OpenPower consortium
- Currently IBM and Xilinx are driving the initial definition of a PCIE based form factor card
  - Representative Diagram is articulated below

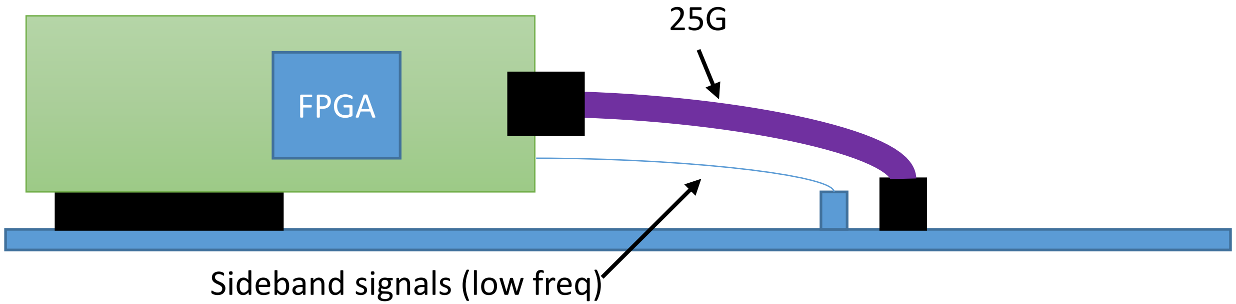


Table of Enablement Deliveries (UNDER CONSTRUCTION)

Expected delivery dates

Item	Date to be delivered to consortium	Availability date from consortium
TLx and DLx Reference Xilinx FPGA RTL and Specifications	February 2017	April 2017
AFU Interface Specification	February 2017	April 2017
Reference Card Design Specification	February 2017	April 2017
25G PHY publicly available	March 2017	March 2017
OpenCAPI Simulation Environment	May 2017	July 2017
Memcopy and Memory Home Agent Exerciser Examples	May 2017	July 2017
Reference Card Available	July 2017	Sept 2017
Reference Driver Available	TBD	TBD

Note: Consortium dates need confirmation

OpenCAPI Consortium Accomplishments

- **Open forum founded by AMD, Google, IBM, Mellanox, and Micron**
  - Manage the OpenCAPI specification
  - Establish enablement
  - Grow the ecosystem
- **Announced on October 14, 2016**
  - Press reviews very positive
  - Other 'open standards' forums announced the same week
- **Functioning Board with 8/9 BOD seats filled**
  - Founders: AMD, Google, IBM, Mellanox Technologies, and Micron
  - NVIDIA, WD, and Xilinx
- **Technical Steering Committee established**
  - Work Group Process defined
  - Initial Work Groups now being formed (TL Specification, DL Specification, PHY Signaling, PHY Mechanical, Enablement, Software, Compliance and more)
- **Closed Governing Documents (Bylaws, IPR Policy, Membership) with established Membership Levels**

OpenCAPI Consortium Accomplishments

- **Established website** www.opencapi.org
  - OpenCAPI Specification currently on web site and open to public (need to register first)
  - OpenCAPI overview document
  - Use Cases overview depicting where OpenCAPI can be used in a server
  - Miscellaneous including Members, Board of Directors, how to join, news, etc.
  - Governing documents (Bylaws, IPR Policy, Membership Agreement)
- **OpenCAPI Specification**
  - Current specification contributed to consortium and will be starting point for the Work Group
  - Updated OpenCAPI 3.0 specification replaced by OpenCAPI 3.1 posted February 6
  - Follow-on specification forthcoming called OpenCAPI 4.0 with added function
- **Target enablement schedule in place**
  - Including reference designs, documentation, SIM environment, etc.
- **Currently 28 members and receiving more enquiries**

Thank you!

Questions: sfields@us.ibm.com