13th ANNUAL WORKSHOP 2017

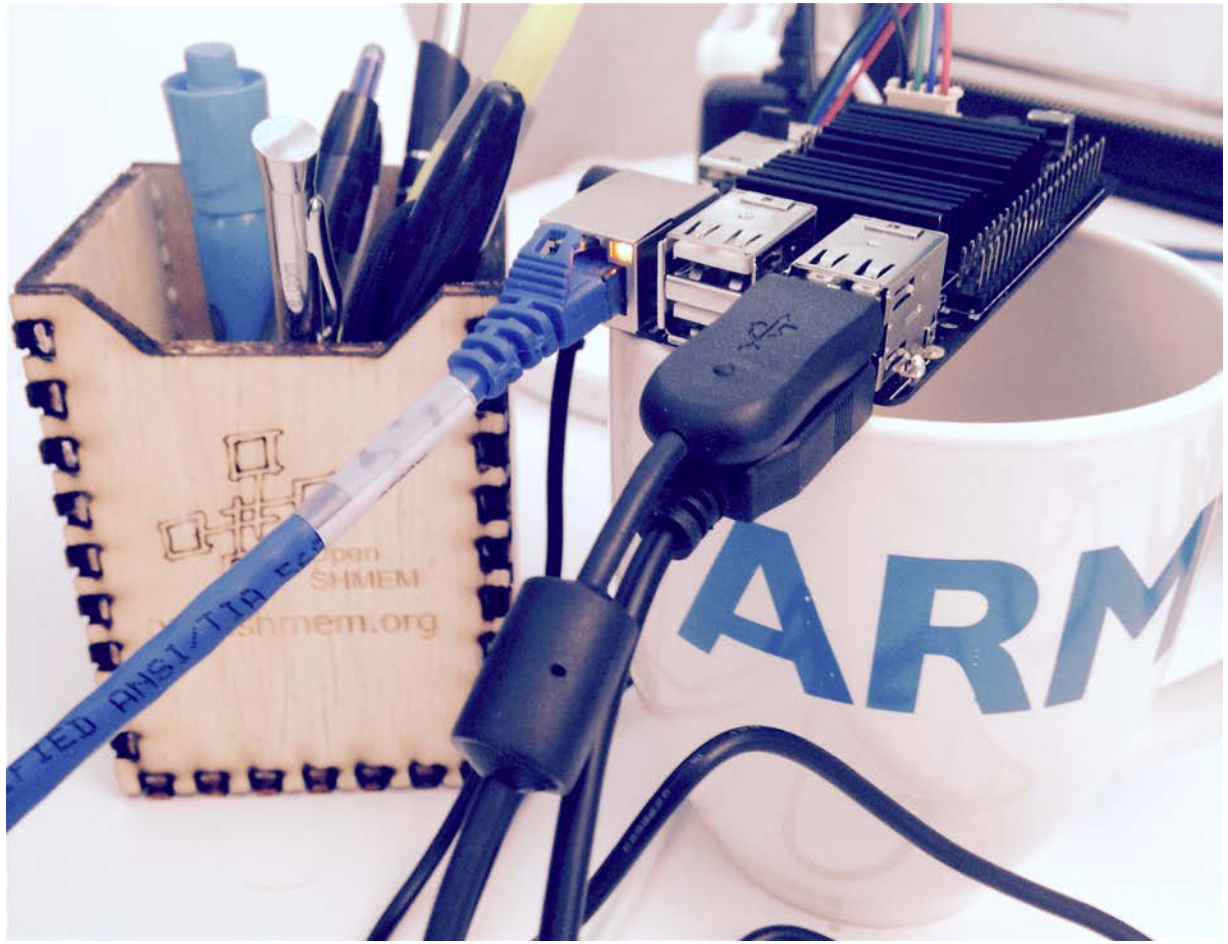
RDMA on ARM

Pavel Shamis (Pasha), Principal Research Engineer

ARM

March, 2017

RDMA on ARM ?



ARM Servers from multiple manufacturers

HP ProLiant
(Applied Micro, TI)

Softiron 64-0800
(AMD)

Gigabyte R120-P30
(Applied Micro)

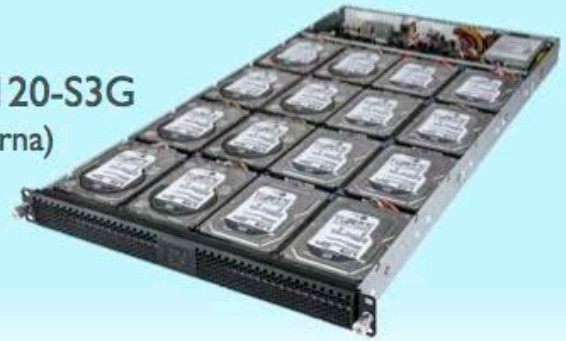
Wiwynn LN1148-10SL
(Marvell)

Gigabyte MT70-HD0
(Cavium)

Cirrascale RM1905D
(Applied Micro)

Mitac Datun
(Applied Micro)

Gigabyte D120-S3G
(Annapurna)



ARM Overview

An introduction to ARM

ARM is the world's leading semiconductor intellectual property supplier.

We license to over 350 partners, are present in 95% of smart phones, 80% of digital cameras, 35% of all electronic devices, and a total of 60 billion ARM cores have been shipped since 1990.

Our CPU business model:

License technology to partners, who use it to create their own system-on-chip (SoC) products.

- We may license an instruction set architecture (ISA) such as "ARMv8-A")
- or a specific implementation, such as "Cortex-A72".
- Partners who license an ISA can create their own implementation, as long as it passes the compliance tests.

…and our IP extends beyond the CPU

ARM

A partnership business model

A business model that shares success

- Everyone in the value chain benefits
- Long term sustainability

Design once and reuse is fundamental

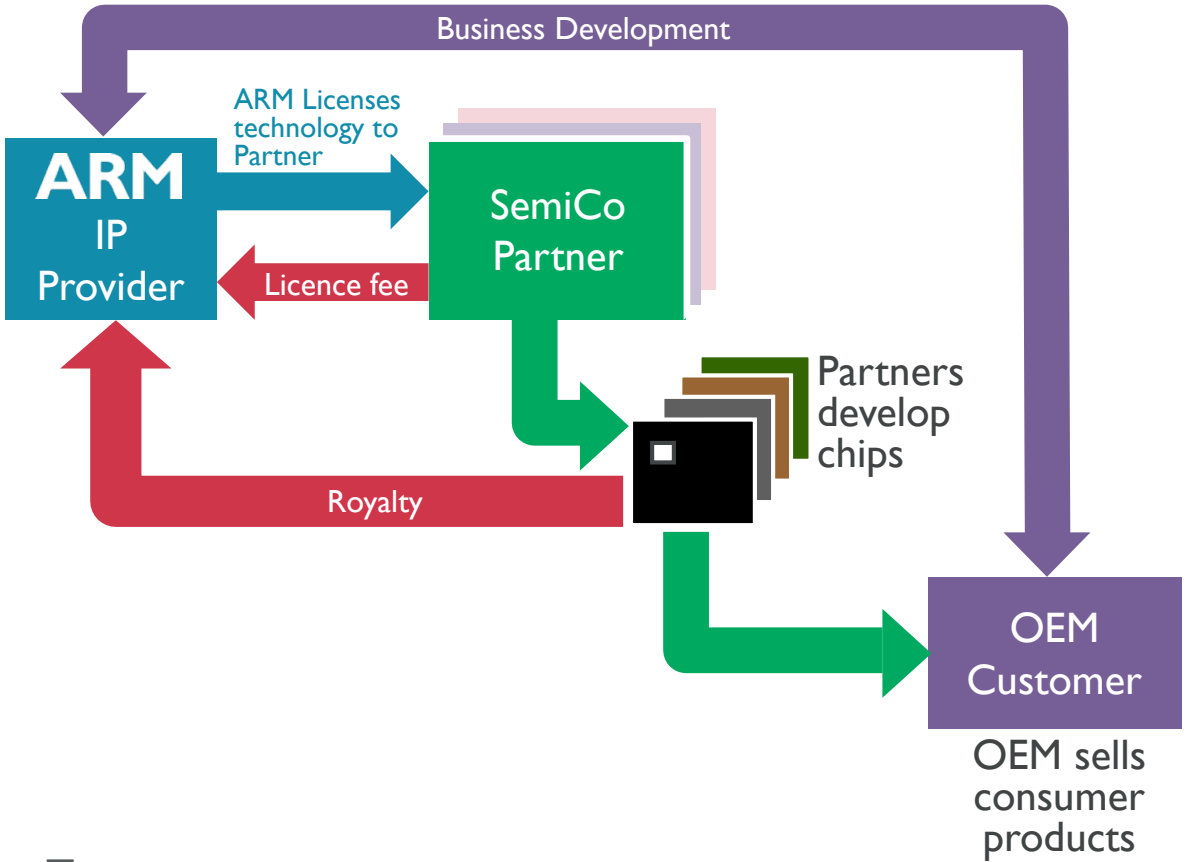
- Spread the cost amongst many partners
- Technology reused across multiple applications
- Creates market for ecosystem to target
  - Re-use is also fundamental to the ecosystem

Upfront license fee

- Covers the development cost

Ongoing royalties

- Typically based on a percentage of chip price
- Vested interest in success of customers



Approximately 1350 licenses
Grows by ~120 every year

More than 440 potential royalty payers

14.8bn+ ARM-powered chips in 2015

ARM

Partnership

Range of SoCs addressing infrastructure

XILINX ZYNQ UltraSCALE+

NXP QorIQ® Layerscape 2080A

SC2A11

QUALCOMM Centriq 2400

ALTERA Stratix 10 FPGA ▪ SoC

BlueField

applied micro X-Gene 3™

CAVIUM

One size does not fit all

ARM

Serious ARM HPC deployments starting in 2017

Two big announcements about ARM in HPC in Europe:

insideHPC

Bull Atos to Build HPC Prototype for Mont-Blanc Project using Cavium ThunderX2 Processor

January 16, 2017 by staff

Today the Mont-Blanc European project announced it has selected Cavium's ThunderX2 ARM server processor to power its new HPC prototype.

The new Mont-Blanc prototype will be built by Atos, the coordinator of phase 3 of Mont-Blanc, using its Bull expertise and products. The platform will leverage the infrastructure of the Bull sequana pre-exascale supercomputer range for network, management, cooling, and power. Atos and Cavium signed an agreement to collaborate to develop this new platform, thus making Mont-Blanc an Alpha-site for ThunderX2.

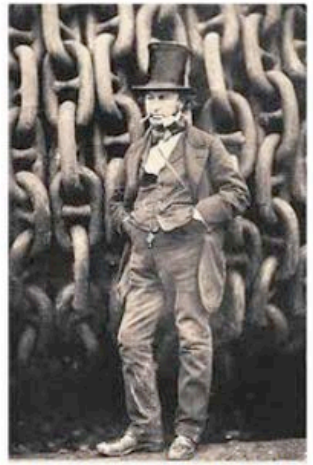
Bull atos technologies

University of BRISTOL GW4 January 17th 2017

Announcing the **GW4 Tier 2 HPC service, 'Isambard':** named after Isambard Kingdom Brunel

System specs:

- Cray CS-400 system
- **10,000+** ARMv8 cores
- HPC optimised software stack
- Technology comparison:
  - x86, KNL, Pascal
- To be installed March-Dec 2017
- £4.7m total project cost over 3 years



I.K.Brunel 1804-1859

Simon McIntosh Smith, simonm@cs.bris.ac.uk, @simonmcs 5 bristol.ac.uk

Japan

Post-K: Fujitsu HPC CPU to Support ARM v8

ARM FUJITSU

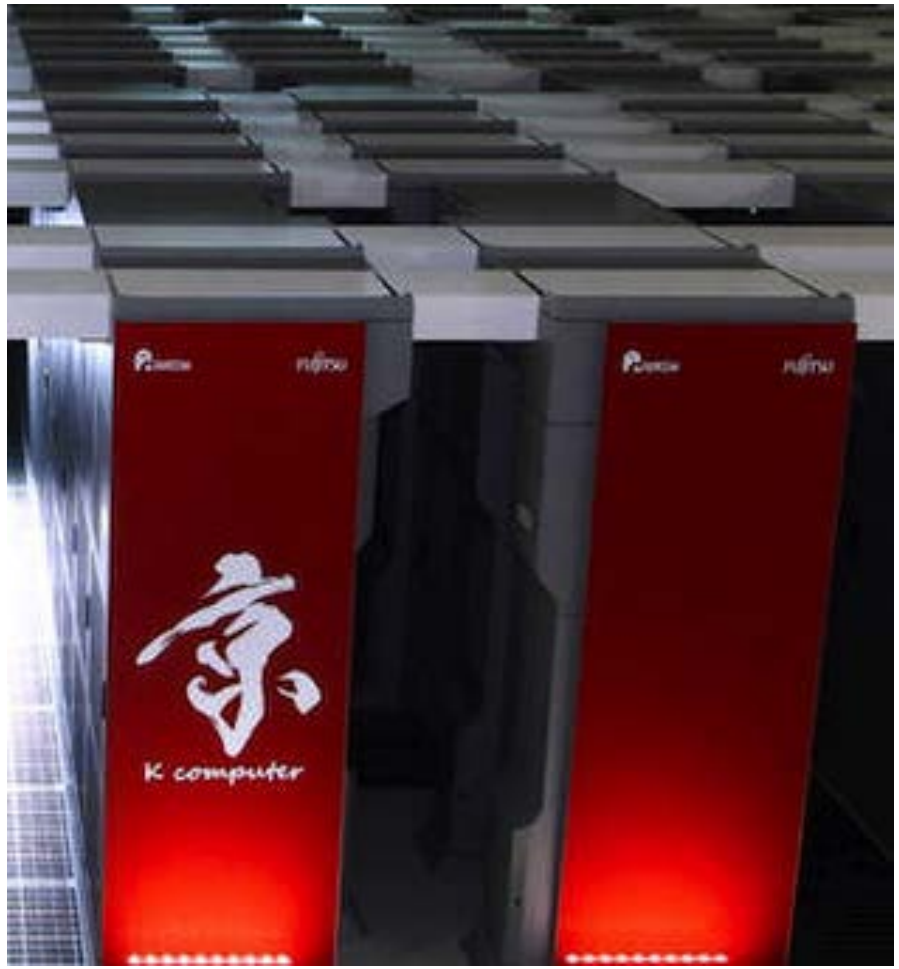
Post-K fully utilizes Fujitsu proven supercomputer microarchitecture

Fujitsu, as a lead partner of ARM HPC extension development, is working to realize ARM Powered® supercomputer w/ high application performance

ARM v8 brings out the real strength of Fujitsu's microarchitecture

HPC apps acceleration feature	Post-K	FX100	FX10	K computer
FMA: Floating Multiply and Add	✔	✔	✔	✔
Math. acceleration primitives*	✔Enhanced	✔	✔	✔
Inter core barrier	✔	✔	✔	✔
Sector cache	✔Enhanced	✔	✔	✔
Hardware prefetch assist	✔Enhanced	✔	✔	✔
Tofu interconnect	✔Integrated	✔Integrated	✔	✔

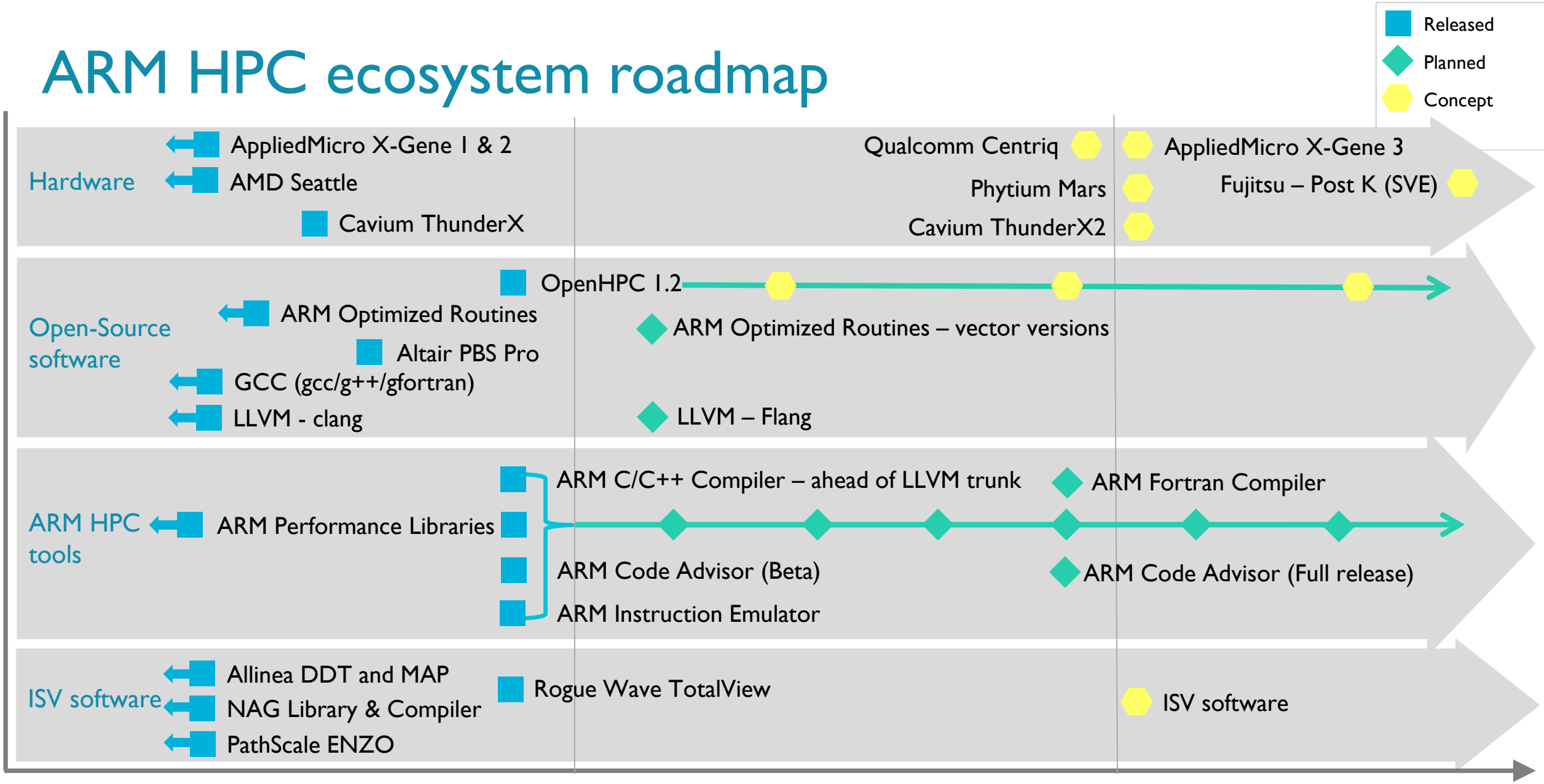
* Mathematical acceleration primitives include trigonometric functions, sine & cosines, and exponential...



slides from Fujitsu at ISC'16

Software Eco-system

ARM HPC ecosystem roadmap



Linux / FreeBSD w/ AARCH64 support

Debian 8 adds AARCH64 – April 2015

12.04LTS & 14.04LTS ← Also 14.10 & 15.04 releases released

Fedora 22 released – May 2015
Fedora 23 released – Nov 2015

Red Hat Enterprise Linux Server for ARM
7.2 BETA – Sept, 2015

CentOS Linux 7 for AArch64
GA – August 2015

OpenSUSE 13.2 – Nov 2014

SUSE Launches Partner Program to Bring SUSE Linux Enterprise 12 to 64-bit ARM
July 2015 @ ISC

FreeBSD ← Engaged with FreeBSD foundation / Semi-half & Cavium to get FreeBSD on ARMv8
FreeBSD Beta version demo'd by Semihalf – Nov. 2015

Open source and commercial compilers

- GCC
  - C, C++, Fortran
  - OpenMP 4.0

- PathScale
  - C, C++ Fortran
  - OpenACC
  - OpenMP 4.0

- LLVM
  - C, C++
  - OpenMP 3.1, (4.0 coming soon)
  - Fortran coming Q1 2017

- NAG
  - Fortran
  - OpenMP 3.1

- ARM C/C++ Compiler
  - LLVM based
  - Includes SVE

Celebrating 5 years of Open Source Engineering on ARM

"The ARM situation has just improved tremendously over the last several years. It used to be a major pain to me, it has gone to almost being entirely painless."

- Linus Torvalds
May 2015

Note:
Linus Torvalds Image from Linux Foundation. Icons made by Freepik. Logos & trademarks remain the property of their respective owners and represent a range of products and services supported by Linaro.

24TB

data from June 2014 - May 2015
615,000 downloads from **>100 countries**

16 Connects
14 Cities on 3 continents

32 member companies
Six members at launch

4,410
gallons consumed at Linaro Connects

1,141,014
minutes of videos showing demos, talks and training sessions watched

More than **220 Engineers** from seed of twenty

#3
company contributor for
Linux Kernels 3.11 - 3.18

11,589
patches upstream since 2011

50,217
Wiki pages

>1 Million
website users

>50
Monthly Software Releases

openembedded · yocto PROJECT · redhat

~20
hardware platforms

www.linaro.org
www.96boards.org

RDMA

RDMA Support

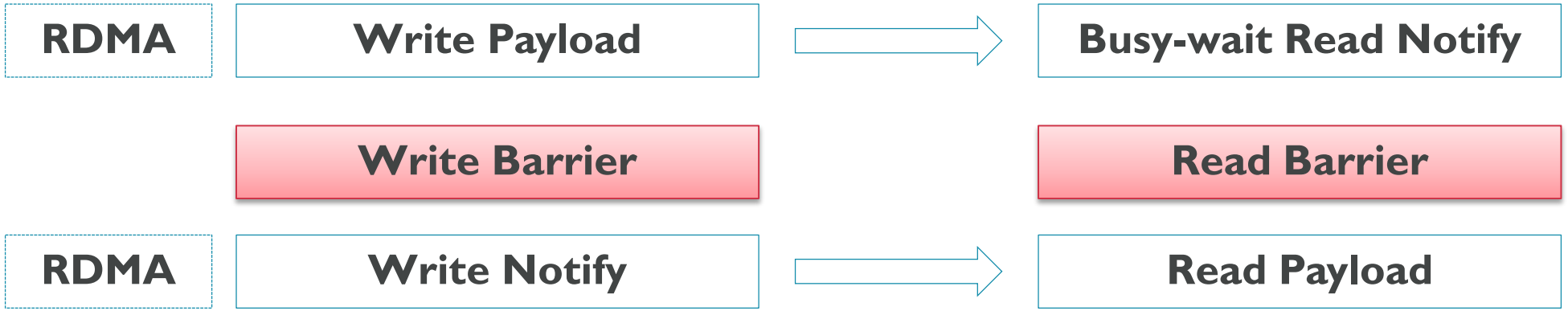
- Mellanox OFED 2.4 and above supports ARM
- Linux Kernel 4.5.0 and above (maybe even earlier)
- OFED – **No** support
- Linux Distribution – on going process

Lessons Learned

- Memory Barriers
  - Multithread environment
  - Software-hardware interaction
  - Examples https://github.com/openucx/ucx/blob/master/src/ucs/arch/aarch64/cpu.h#L25
  - You can "fish" for these bugs in MPI implementations around Eager-RDMA and shared memory protocols

```
#define ucs_memory_bus_fence()       asm volatile ("dsb sy" ::: "memory");
#define ucs_memory_bus_store_fence() asm volatile ("dsb st" ::: "memory");
#define ucs_memory_bus_load_fence()  asm volatile ("dsb ld" ::: "memory");

#define ucs_memory_cpu_fence()       asm volatile ("dmb sy" ::: "memory");
#define ucs_memory_cpu_store_fence() asm volatile ("dmb st" ::: "memory");
#define ucs_memory_cpu_load_fence()  asm volatile ("dmb ld" ::: "memory");
```



Maranget, Luc, Susmit Sarkar, and Peter Sewell. "A tutorial introduction to the ARM and POWER relaxed memory models." Draft available from http://www. cl. cam. ac. uk/~ pes20/ppc-supplemental/test7. pdf (2012).

Lessons Learned - continued

- Low-level timers
  - Typically found in benchmarks and MPI
  - Code examples https://github.com/openucx/ucx/blob/master/src/ucs/arch/aarch64/cpu.h#L35

```
static inline uint64_t ucs_arch_read_hres_clock(void)
{
    uint64_t ticks;
    asm volatile("isb" : : : "memory");
    asm volatile("mrs %0, cntvct_el0" : "=r" (ticks));
    return ticks;
}

static inline double ucs_arch_get_clocks_per_sec()
{
    uint32_t freq;
    asm volatile("mrs %0, cntfrq_el0" : "=r" (freq));
    return (double) freq;
}
```

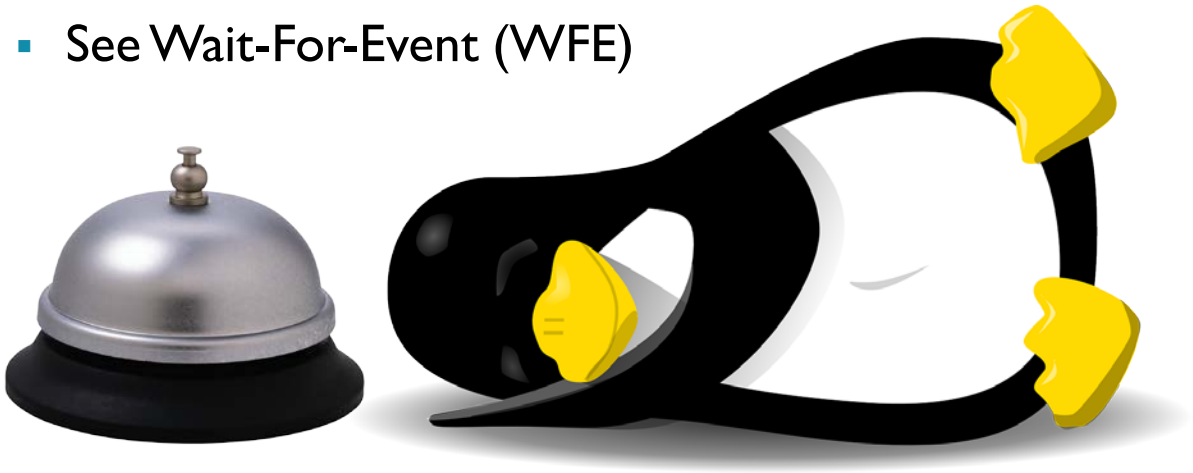
Lessons Learned – continued

- Not all cache-lines are 64Byte !
  - Implementation dependent
  - 128Byte and 64Byte

http://xeroxnostalgia.com/duplicators/xerox-9200/

Optimizations

- AVX => Neon
  - Mostly found around communication request initialization codes https://github.com/openucx/ucx/blob/891e20ef90257d1e2721da52461b0261220c82d8/src/uct/ib/mlx5/ib_mlx5.inl#L160
- Busy-wait loop
  - See Wait-For-Event (WFE)



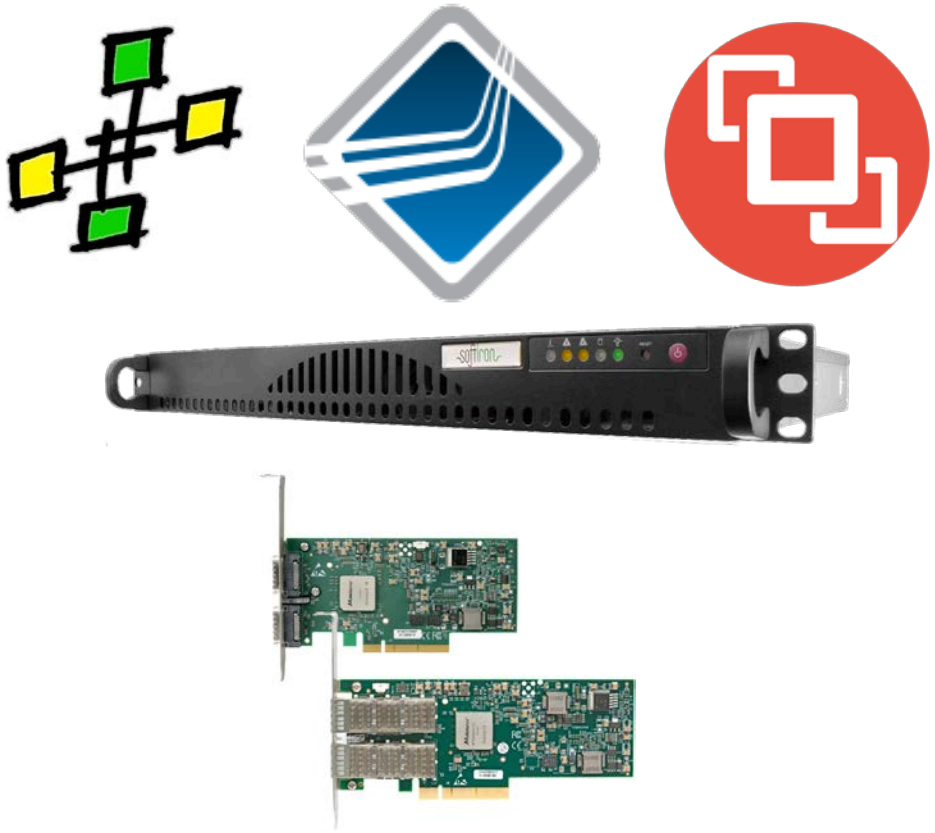
```
#if defined(__SSE4_2__)
    *(__m128i*)raddr = _mm_shuffle_epi8(
                _mm_set_epi64x(rdma_rkey, rdma_raddr),
                _mm_set_epi8(0, 0, 0, 0,            /* reserved */
                             8, 9, 10, 11,          /* rkey */
                             0, 1, 2, 3, 4, 5, 6, 7 /* rdma_raddr */
                             ));
#elif defined(__ARM_NEON)
    uint8x16_t table =  {7,  6,  5, 4, 3, 2, 1, 0, /* rdma_raddr */
                         11, 10, 9, 8,              /* rkey */
                         16,16,16,16};              /* reserved (set 0) */
    uint64x2_t data = {rdma_raddr, rdma_rkey};
    *(uint8x16_t *)raddr = vqtbl1q_u8((uint8x16_t)data, table);
#else
    raddr->raddr = htobe64(rdma_raddr);
    raddr->rkey  = htonl(rdma_rkey);
#endif
```

Pavel Shamis, M. Graham Lopez, and Gilad Shainer. "Enabling One-sided Communication Semantics on ARM"

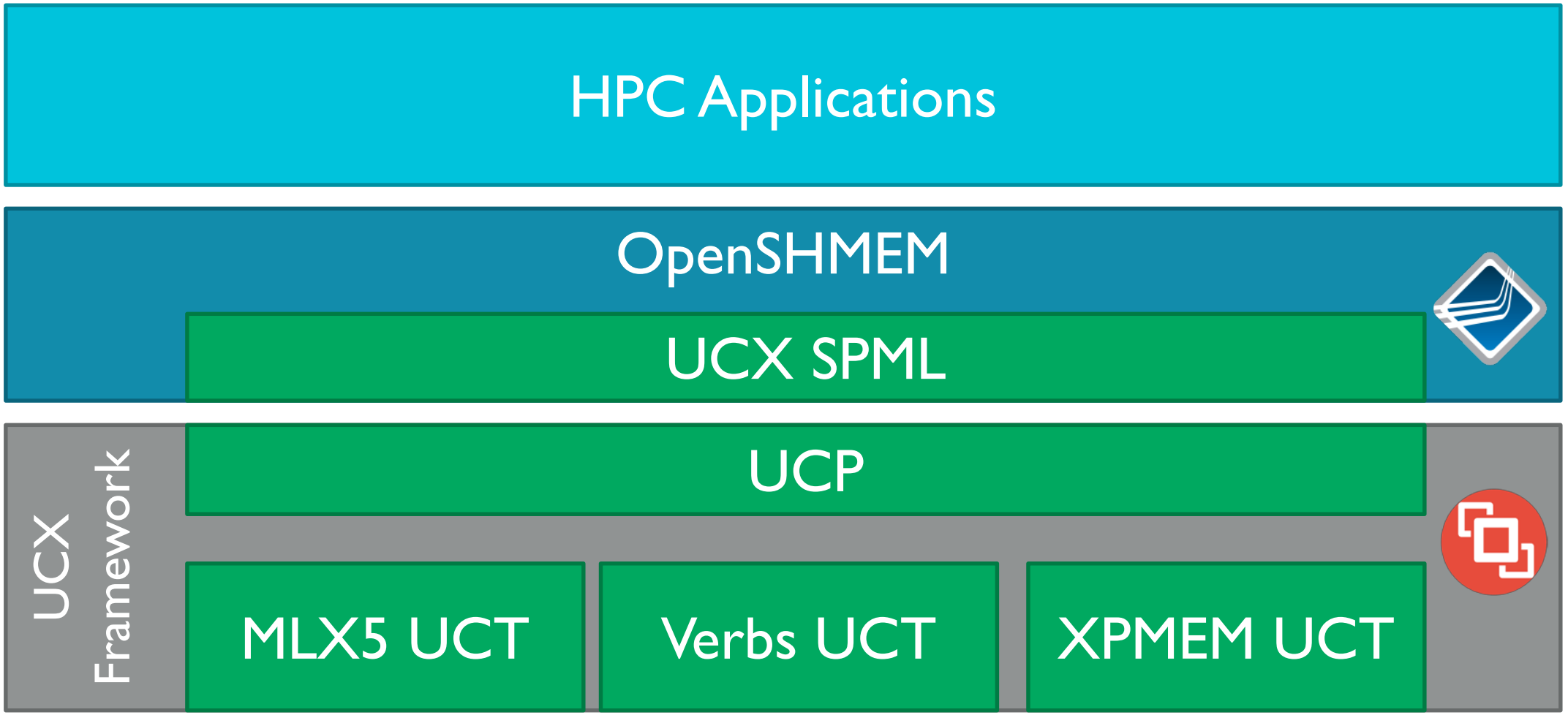
Preliminary Results

Testbed

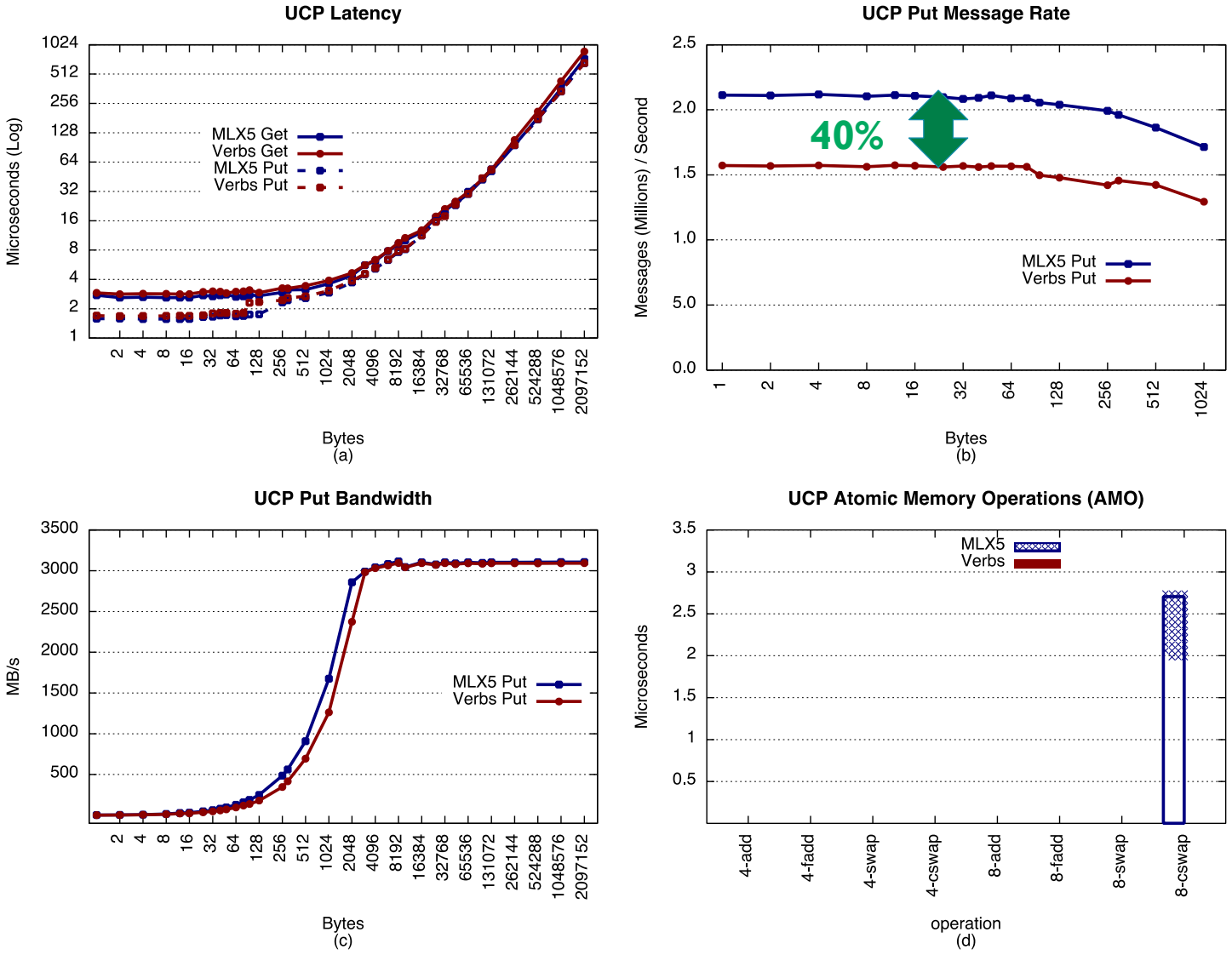
- 2 x Softiron Overdrive 3000 servers with AMD Opteron A1100 / 2GHz
- ConnectX-4 IB/VPI EDR (PCIe g2 x8)
- Ubuntu 16.04
- MOFED 3.3-1.5.0.0
- UCX [0558b41]
- XPMEM [bdfcc52]
- OSHMEM/OPEN-MPI [fed4849]



OpenUCX and OpenSHMEM on ARM



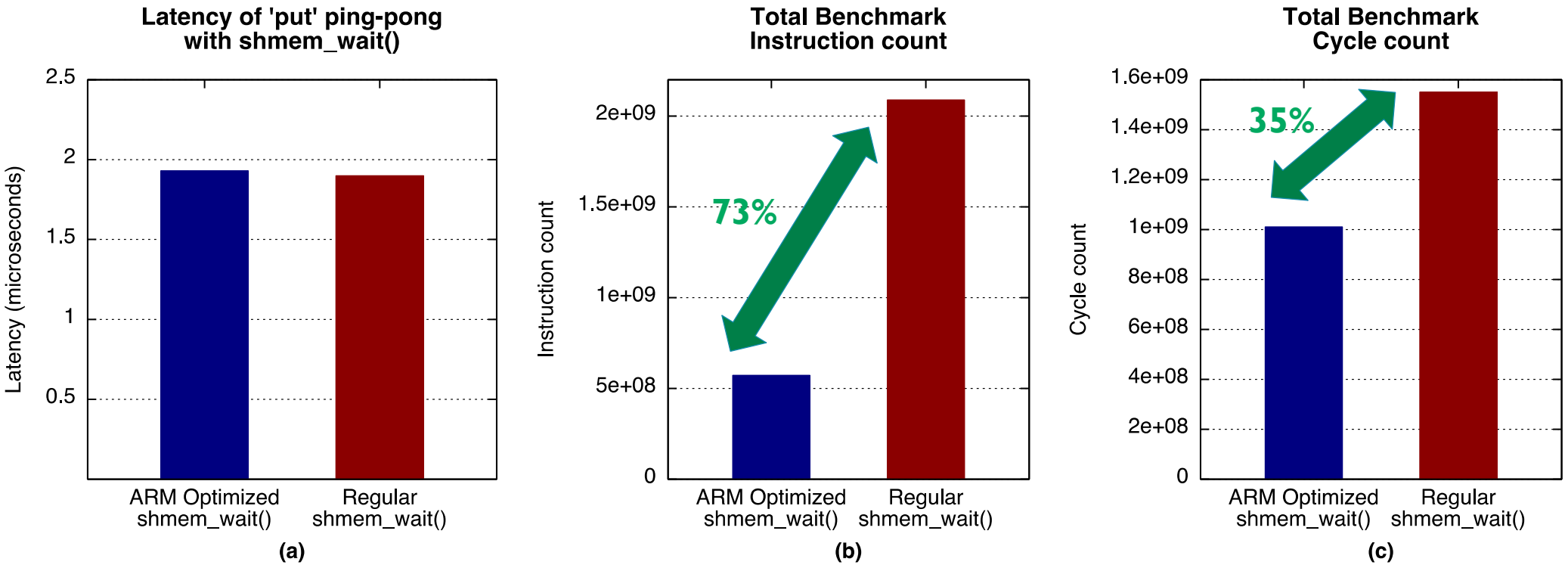
OpenUCX with InfiniBand



Pavel Shamis, M. Graham Lopez, and Gilad Shainer. "Enabling One-sided Communication Semantics on ARM"

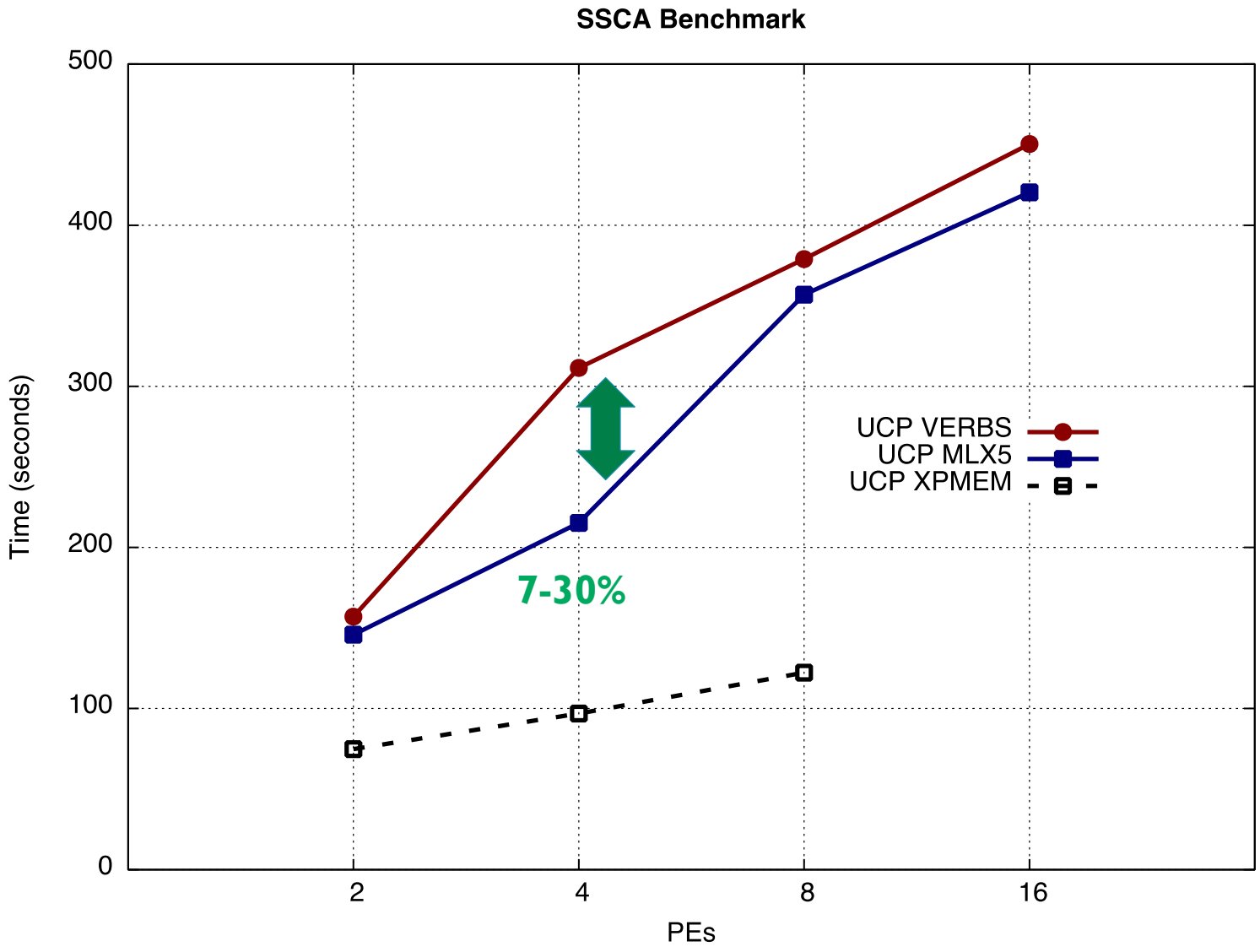
ARM

SHMEM_WAIT()



Pavel Shamis, M. Graham Lopez, and Gilad Shainer. "Enabling One-sided Communication Semantics on ARM"

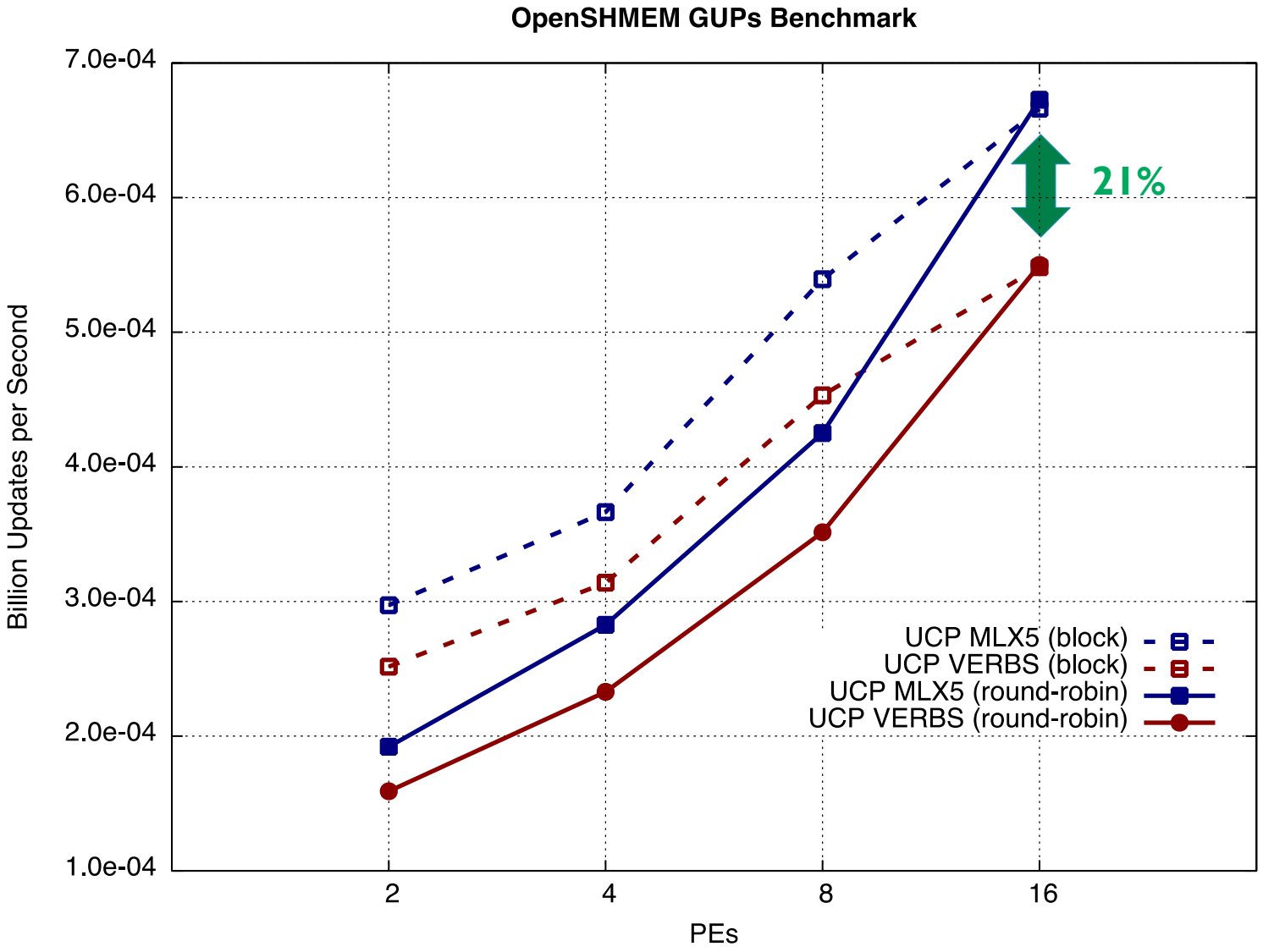
OpenSHMEM SSCA



ARM

Pavel Shamis, M. Graham Lopez, and Gilad Shainer. "Enabling One-sided Communication Semantics on ARM"

OpenSHMEM GUPs

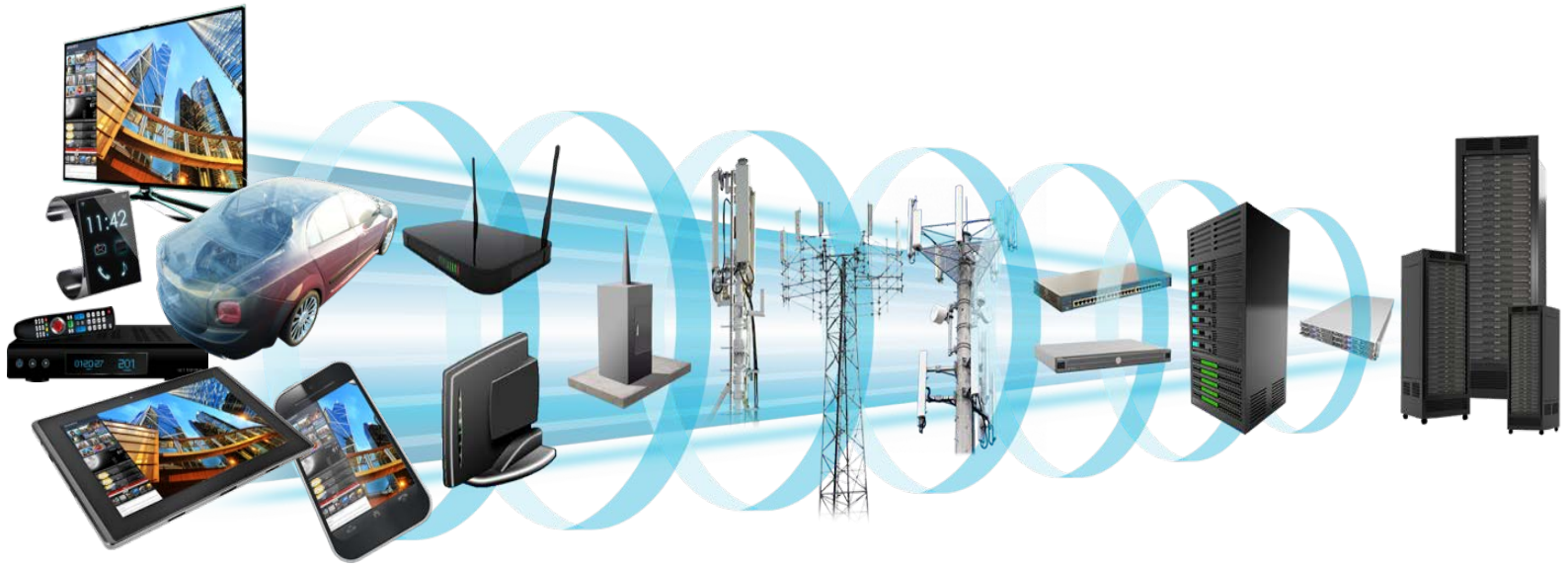


ARM

Pavel Shamis, M. Graham Lopez, and Gilad Shainer. "Enabling One-sided Communication Semantics on ARM"

Summary

- Linux RDMA community is doing great job !
- A lot of progress was made in ARM HPC/server software eco-system



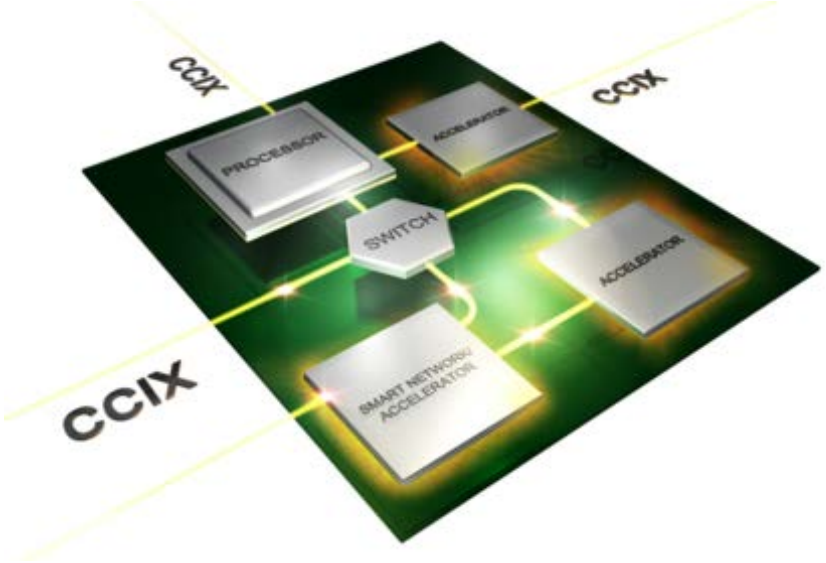
ARM

Backup

CCIX

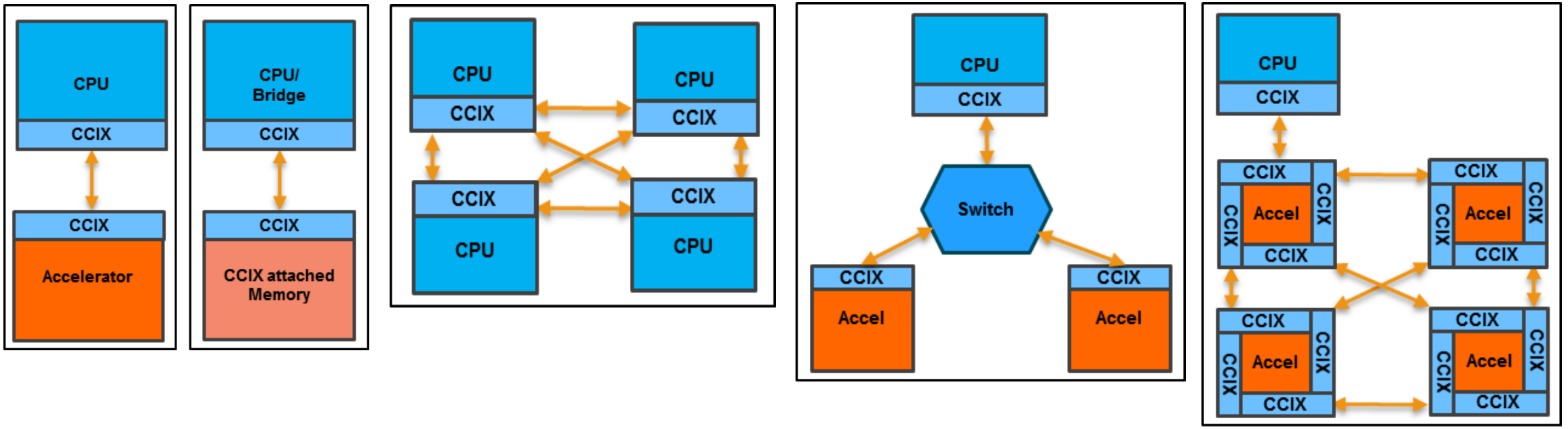
CCIX Cache Coherent Interconnect for Accelerators

- Accelerators and Network (NIC/HCA/etc.) as a first class "citizen" in the system
- Seamless process and accelerator hardware cache coherence support
- Low-latency and high-bandwidth
- Allow in-line acceleration
  - Bump in the wire processing (network packet processing, storage acceleration, etc.)
- Allows "off-line" acceleration (co-processor model)
- Driver-less / interrupt-less usage model
- http://www.ccixconsortium.com



AMD Amphenol ARM Arteris
avery design systems BROADCOM Bull atos technologies cadence
CAVIUM HUAWEI IBM IDT
KEYSIGHT TECHNOLOGIES Mellanox TECHNOLOGIES Micron NETSPEED SYSTEMS
QUALCOMM redhat SYNOPSYS tsmc
TELEDYNE LECROY XILINX

CCIX



CCIX Enables System Connectivity For All Elements (CPU, GPU, IO, FPGA)

ARM

Gen-Z

- **All data is accessed by some form of a Read or a Write**
- Example of reads: DDR Row + Column Read, PCI DMA Read, SCSI Write, Socket Read, File Read, RDMA Read
- Example of writes: DDR Row + Column Write, PCI DMA Write, SCSI Read, Socket Write, File Write, RDMA Write
- **The Goal:** Simplify world to memory semantic Reads & Writes

Gen-Z Overview

- An open, standards-based, scalable, system interconnect and protocol.
- Optimized to support memory semantic communications
- Breaks Processor-Memory Interlock
- Split controller model
  - Memory controller
    - Initiates high-level requests—Read, Write, Atomic, Put / Get, etc.
    - Enforces ordering, reliability, path selection, etc.
  - Media controller
    - Abstracts memory media
    - Supports volatile / non-volatile / mixed-media
    - Performs media-specific operations
    - Executes requests and returns responses
    - Enables data-centric computing (accelerator, compute, etc.)