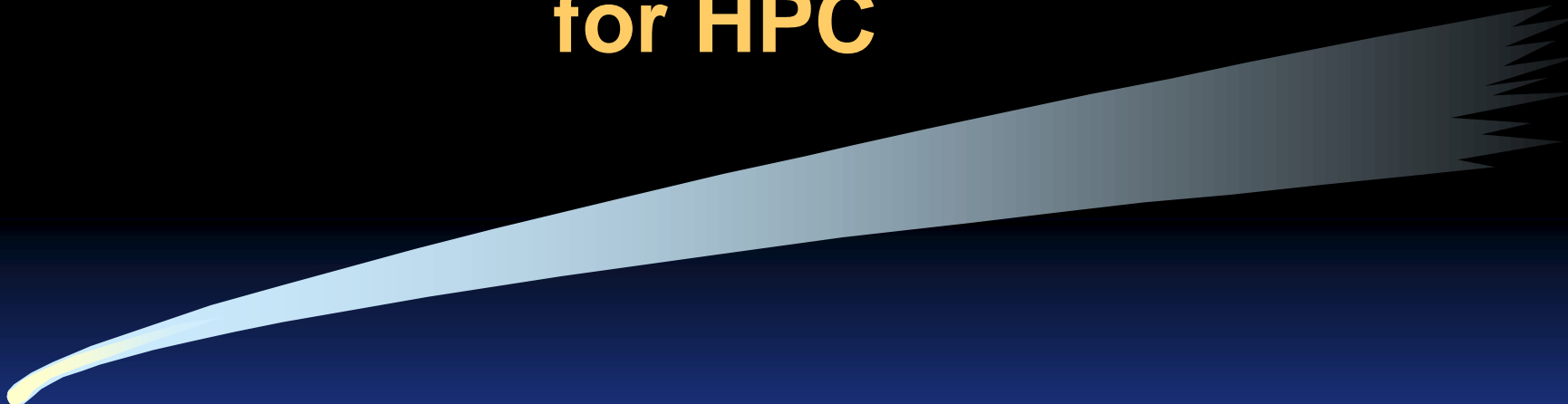


SPARC64™ XIfx: Fujitsu's Next Generation Processor for HPC



August 11, 2014

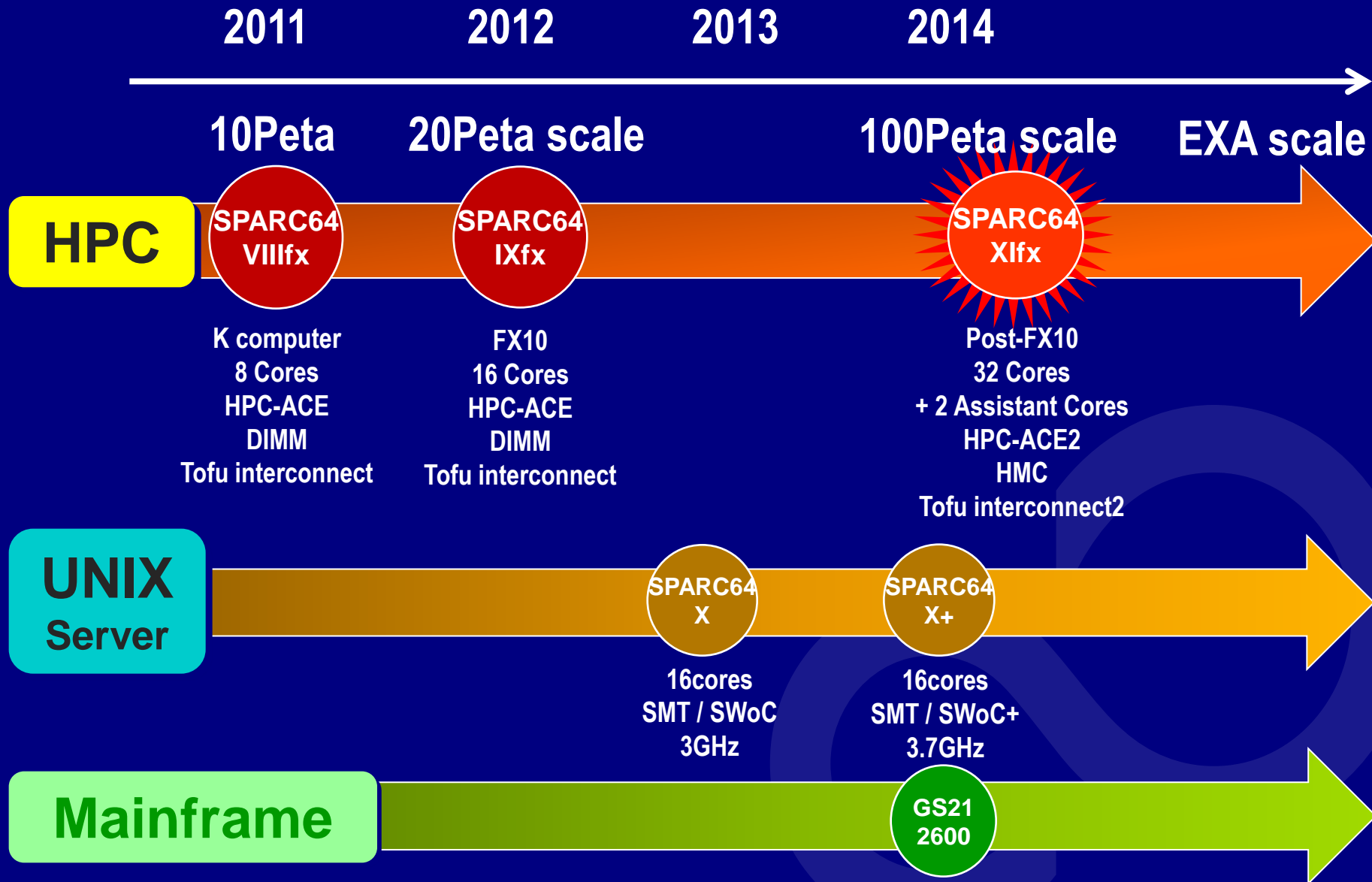
Toshio Yoshida

Next Generation Technical Computing Unit
Fujitsu Limited

Agenda

- ◆ Fujitsu Processor Development
- ◆ SPARC64™ Xlfx
 - Design Concept and Processor Overview
 - Node Architecture
 - HPC-ACE2: ISA enhancements
 - Microarchitecture
 - Enhanced VISIMPACT and Sector Cache
 - Assistant Core
 - Performance
 - RAS
- ◆ Summary

Fujitsu Processor Development



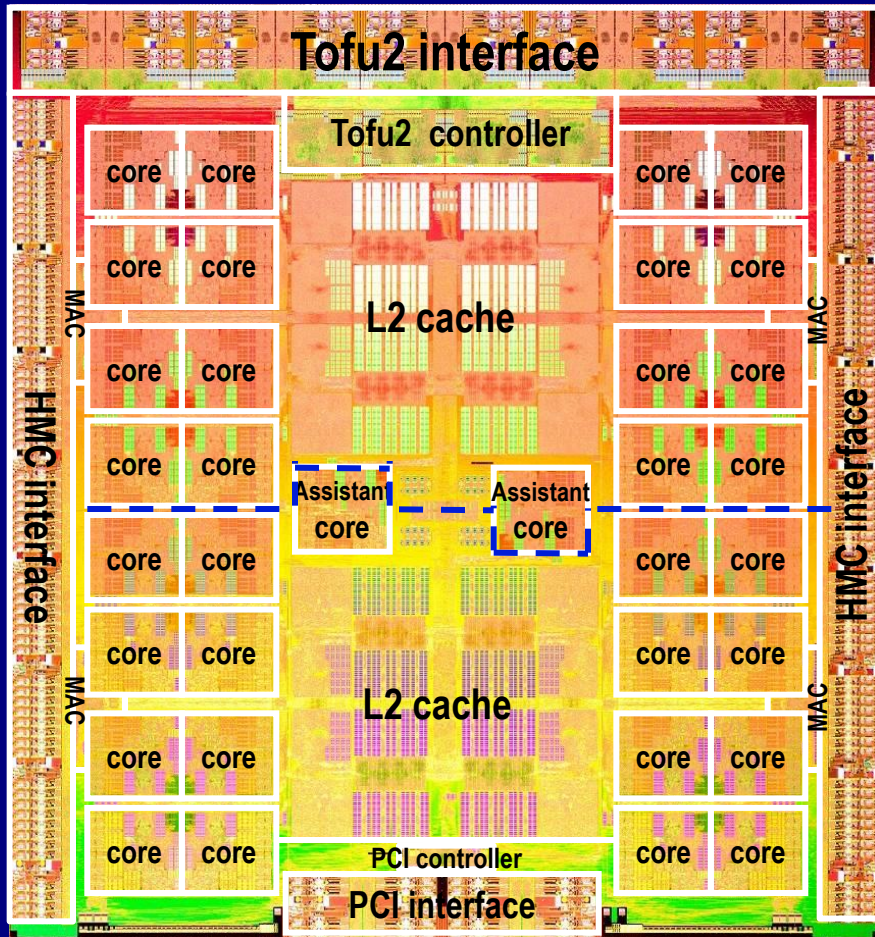
Agenda

- ◆ Fujitsu Processor Development
- ◆ SPARC64™ Xlfx
 - Design Concept and Processor Overview
 - Node Architecture
 - HPC-ACE2: ISA enhancements
 - Microarchitecture
 - Enhanced VISIMPACT and Sector Cache
 - Assistant Core
 - Performance
 - RAS
- ◆ Summary

Design Concept of SPARC64™ Xlfx

- Designed for massively parallel supercomputer systems
 - High performance for wide range of real applications
 - High scalability
 - Low power consumption
 - Groundwork for EXA scale computing
 - Enhance and inherit K computer features
 - Stand-alone scalar many-core architecture
 - Enhanced VISIMPACT and Sector cache
 - On-chip integrated Tofu interconnect 2
 - Introduce new technologies to EXA scale
 - Wider SIMD enhancements
 - Leading-edge memory technology
 - Cores dedicated for non-computation operation
- HPC-ACE2
HMC
Assistant cores

SPARC64™ Xlfx Chip Overview



● Architecture Features

- 32 computing cores + 2 assistant cores
- HPC-ACE2
- 24 MB L2 cache
- HMC, Tofu2 , PCI Gen3

● 20nm CMOS

- 3,750M transistors
- 1,001 signal pins
- 2.2GHz

● Performance (peak)

- 1.1TFlops
- HMC 240GB/s x 2(in/out)
- Tofu2 125GB/s x 2(in/out)

Agenda

- ◆ Fujitsu Processor Development
- ◆ SPARC64™ Xlfx
 - Design Concept and Processor Overview
 - Node Architecture
 - HPC-ACE2: ISA enhancements
 - Microarchitecture
 - Enhanced VISIMPACT and Sector Cache
 - Assistant Core
 - Performance
 - RAS
- ◆ Summary

Node Architecture

- Stand-alone scalar many-core with wider SIMD
 - No accelerator
- Non-hierarchical and high bandwidth memory
 - 8x HMCs (32GB, 240GB/s x2 (in/out))
- Isolation of non-computation operation for jitter reduction
 - 32 Computing cores
 - 2 Assistant cores
 - Daemon, IO, MPI asynchronous communication, etc.
 - Sector cache is used for assistant core to avoid cache pollution
 - Computing cores and Assistant cores keep cache coherency
- Single OS manages computing and assistant cores
 - Single OS minimizes memory management overhead

Agenda

- ◆ Fujitsu Processor Development
- ◆ SPARC64™ Xlfx
 - Design Concept and Processor Overview
 - Node Architecture
 - HPC-ACE2: ISA enhancements
 - Microarchitecture
 - Enhanced VISIMPACT and Sector Cache
 - Assistant Core
 - Performance
 - RAS
- ◆ Summary

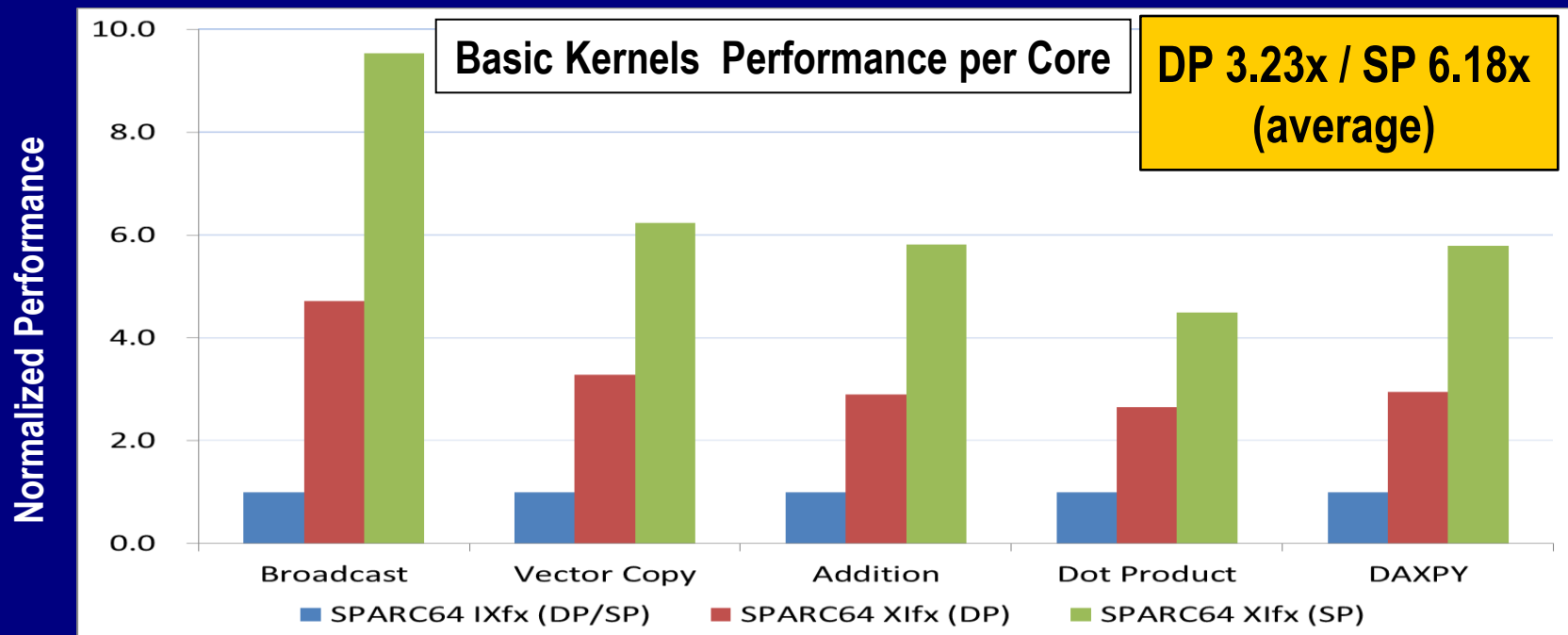
HPC-ACE2: ISA enhancements

- Wider SIMD enhancements from K computer / FX10
 - 256-bit wide SIMD (64-bit x 4 / 32-bit x 8)
 - More integer operations
 - Stride load/store
 - Indirect load/store
 - Compress
 - Round
 - Permutation



Wider SIMD Extensions

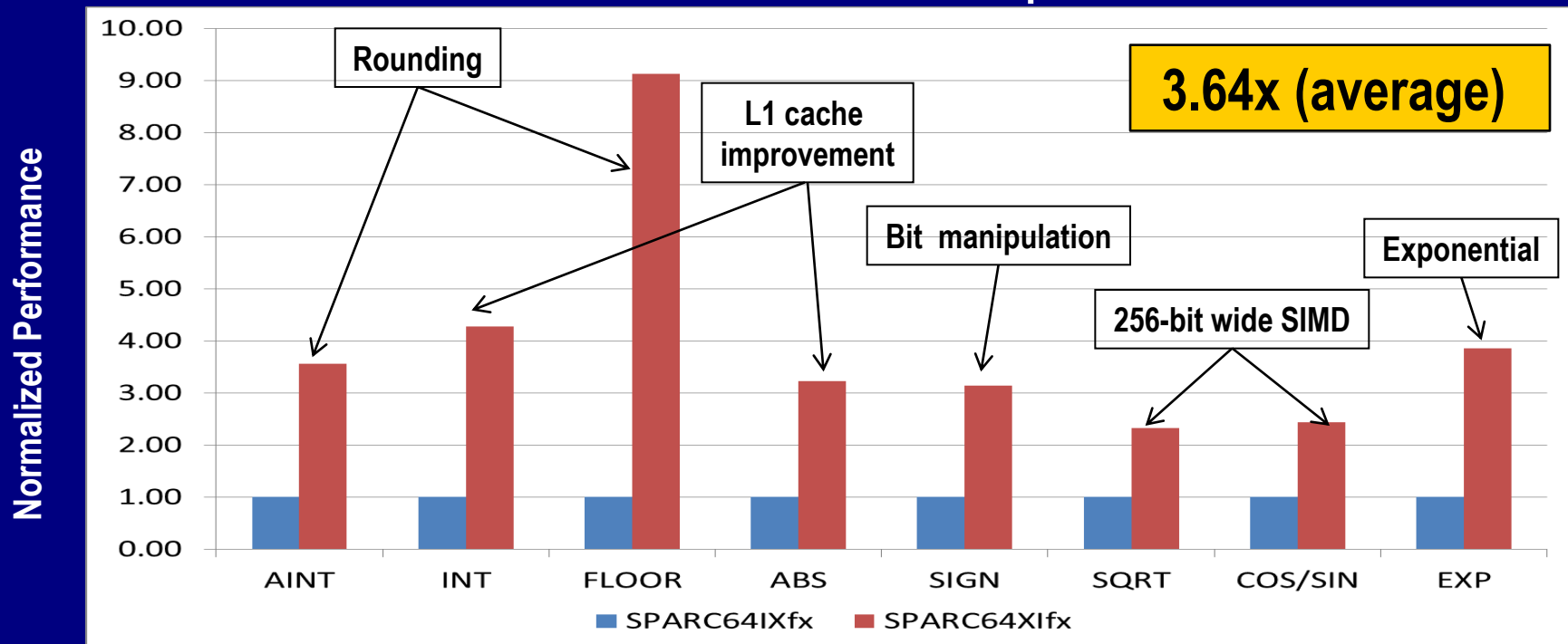
- 256-bit wide SIMD with 128 FPRs
 - 64-bit (DP: Double Precision) x 4 SIMD
 - 32-bit (SP: Single Precision) x 8 SIMD
- DP 3.2x, SP 6.1x faster than SPARC64™ IXfx in basic kernels
 - Improved L1 cache pipelines
 - Higher frequency 1.848GHz -> 2.2GHz



Built-in Functions

- Built-in functions accelerated by
 - HPC-ACE2 instructions
 - 256-bit wide SIMD
 - Rounding / Bit manipulation / Exponential auxiliary instructions
 - Microarchitectural enhancements

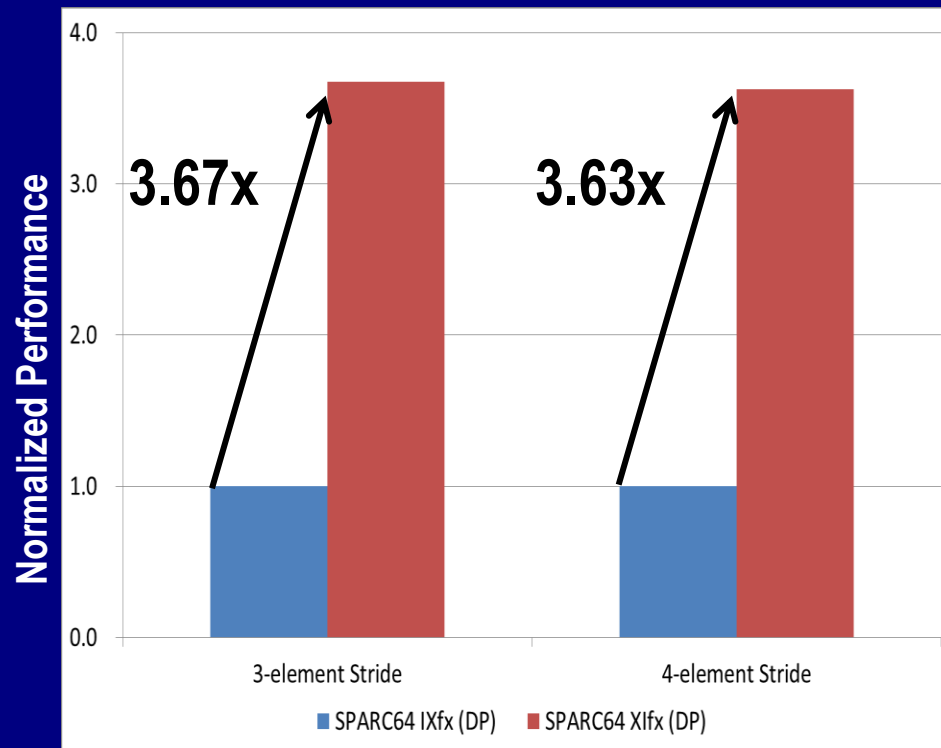
Built-in Functions Performance per Core



Stride Load/Store Instructions

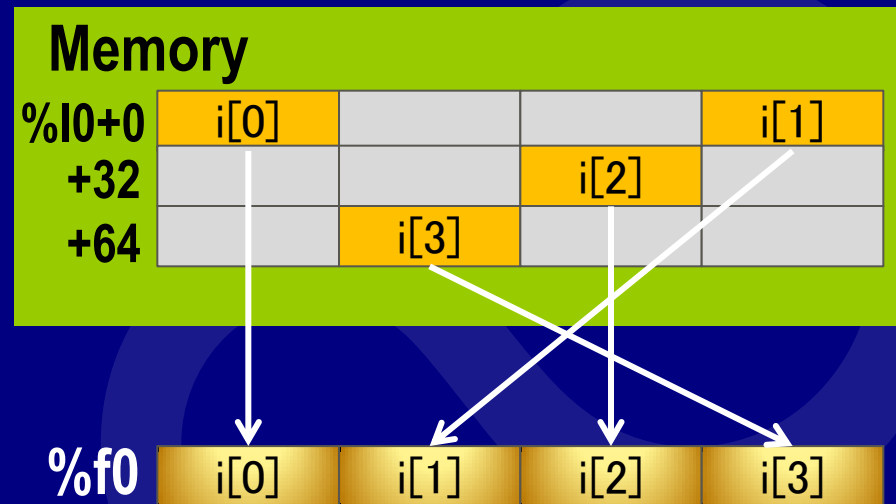
- Stride access is frequently used in various HPC apps.
 - Support from 2 to 7-element stride width
- 3.6x faster than SPARC64™ IXfx

Stride load Performance



E.g. Stride load @ stride width = 3

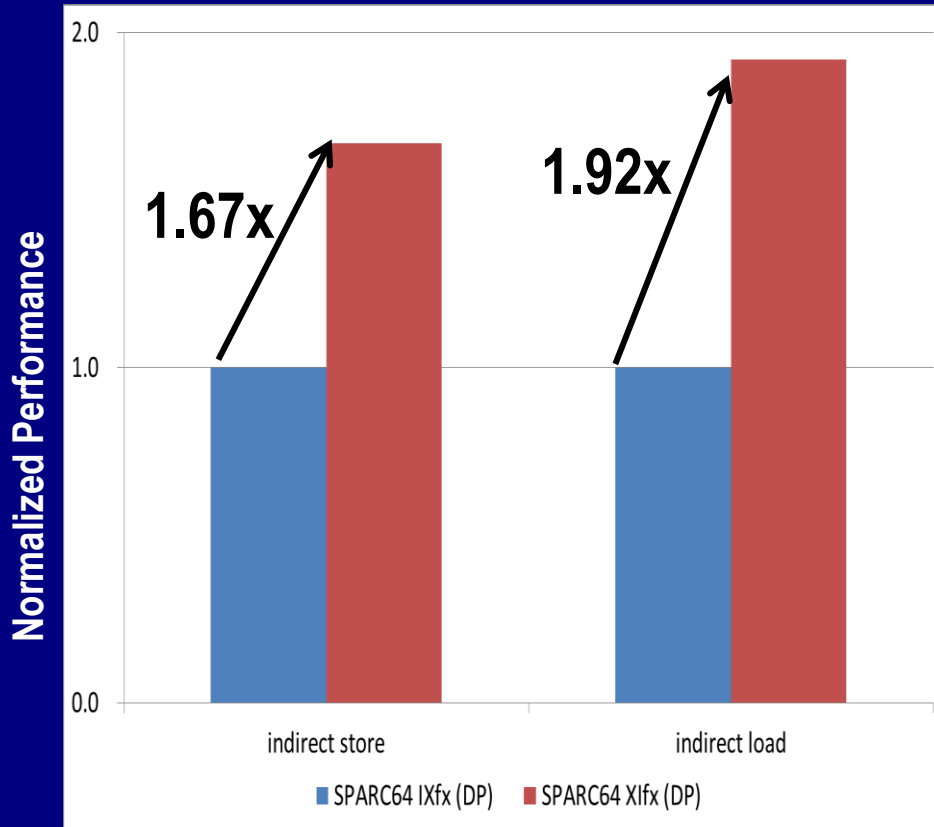
`lddst,s [%l0]@stride 3, %f0`



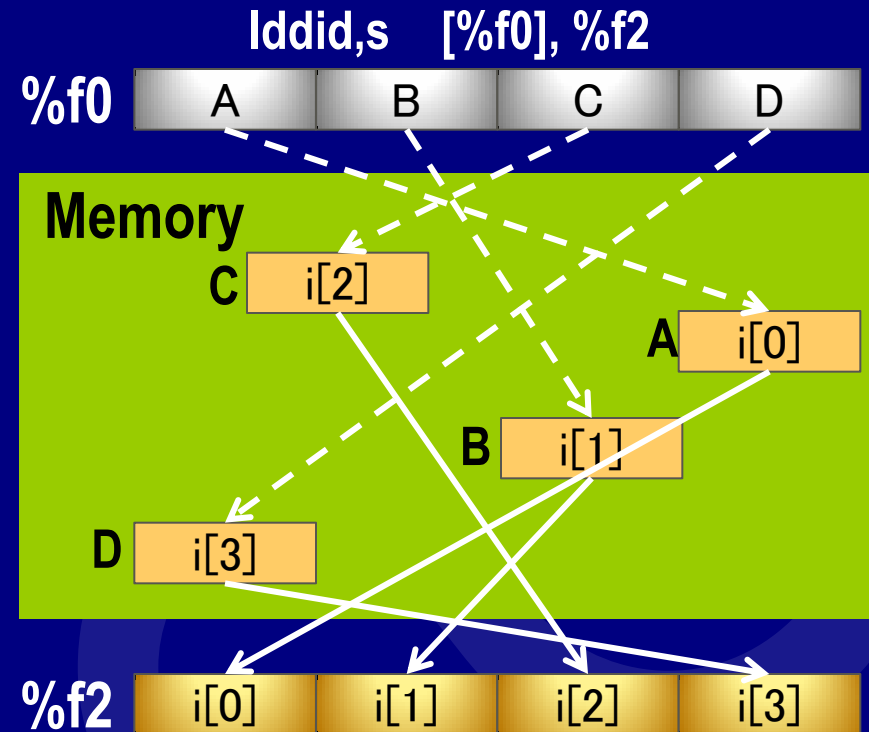
Indirect Load/Store Instructions

- Indirect load and store instructions for list accesses
 - List accesses appear in wide ranges of HPC apps.
- More than 1.6x faster than SPARC64™ IXfx

Indirect Load/Store Performance



E.g. Indirect load

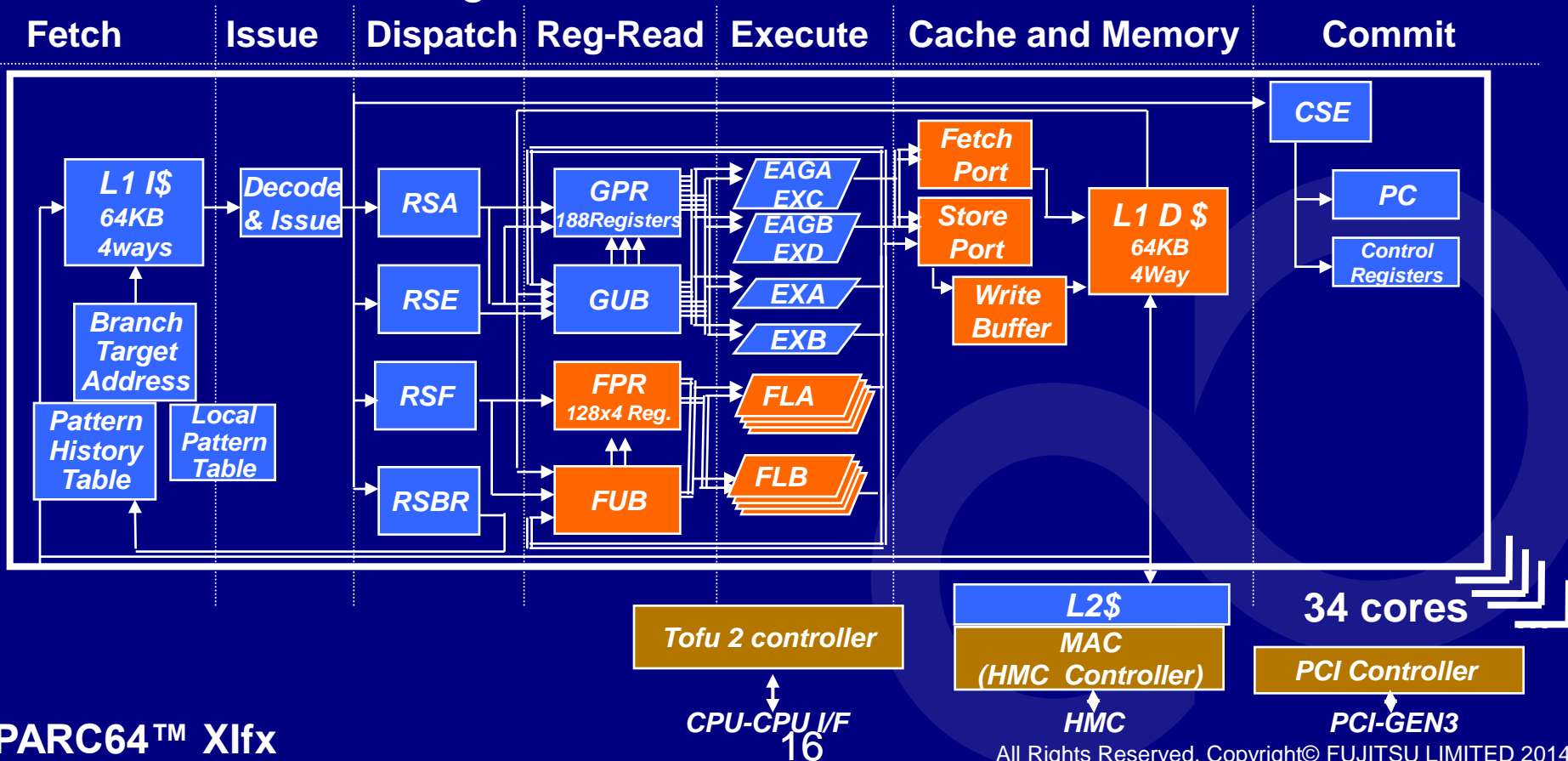


Agenda

- ◆ Fujitsu Processor Development
- ◆ SPARC64™ Xlfx
 - Design Concept and Processor Overview
 - Node Architecture
 - HPC-ACE2: ISA enhancements
 - Microarchitecture
 - Enhanced VISIMPACT and Sector Cache
 - Assistant Core
 - Performance
 - RAS
- ◆ Summary

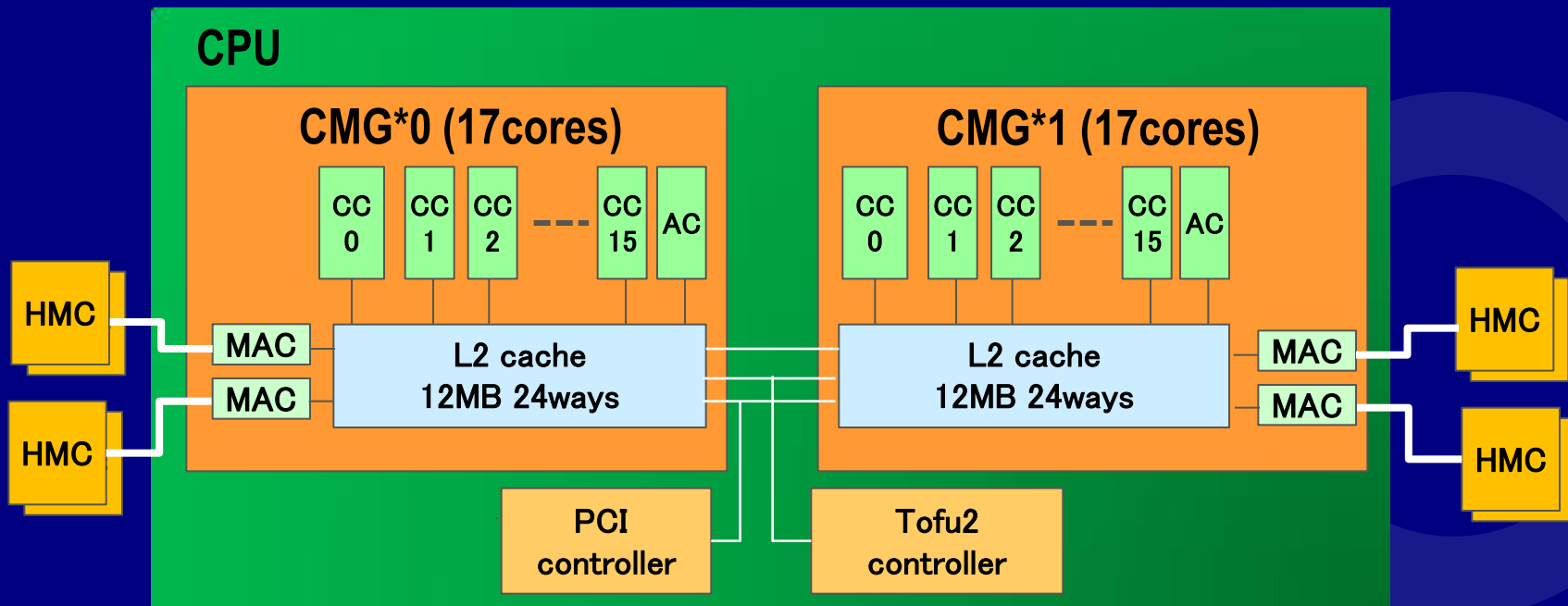
SPARC64™ Xifx Core Pipeline

- 2x 256-bit SIMD FMAs + 4x ALUs (shared with 2 AGENs)
- 2x 256-bit SIMD LOADs or 1x 256-bit SIMD STORE
- Fundamental pipelines are based on SPARC64™ X+
 - Superscalar, Out-of-Order, branch prediction, etc.
- No multithreading



Many-Core Architecture

- SPARC64™ Xlfx has 2 CMGs (Core Memory Group)
 - CMG consists of 17 cores, L2 cache and 2 memory controllers (MAC)
 - Two CMGs keep cache coherency by ccNUMA with on-chip directory
 - 32GB memory capacity
 - To bind a process in a CMG is recommended



High Bandwidth

- High bandwidth cache, memory and Tofu2

- 2x Cache bandwidth / Core

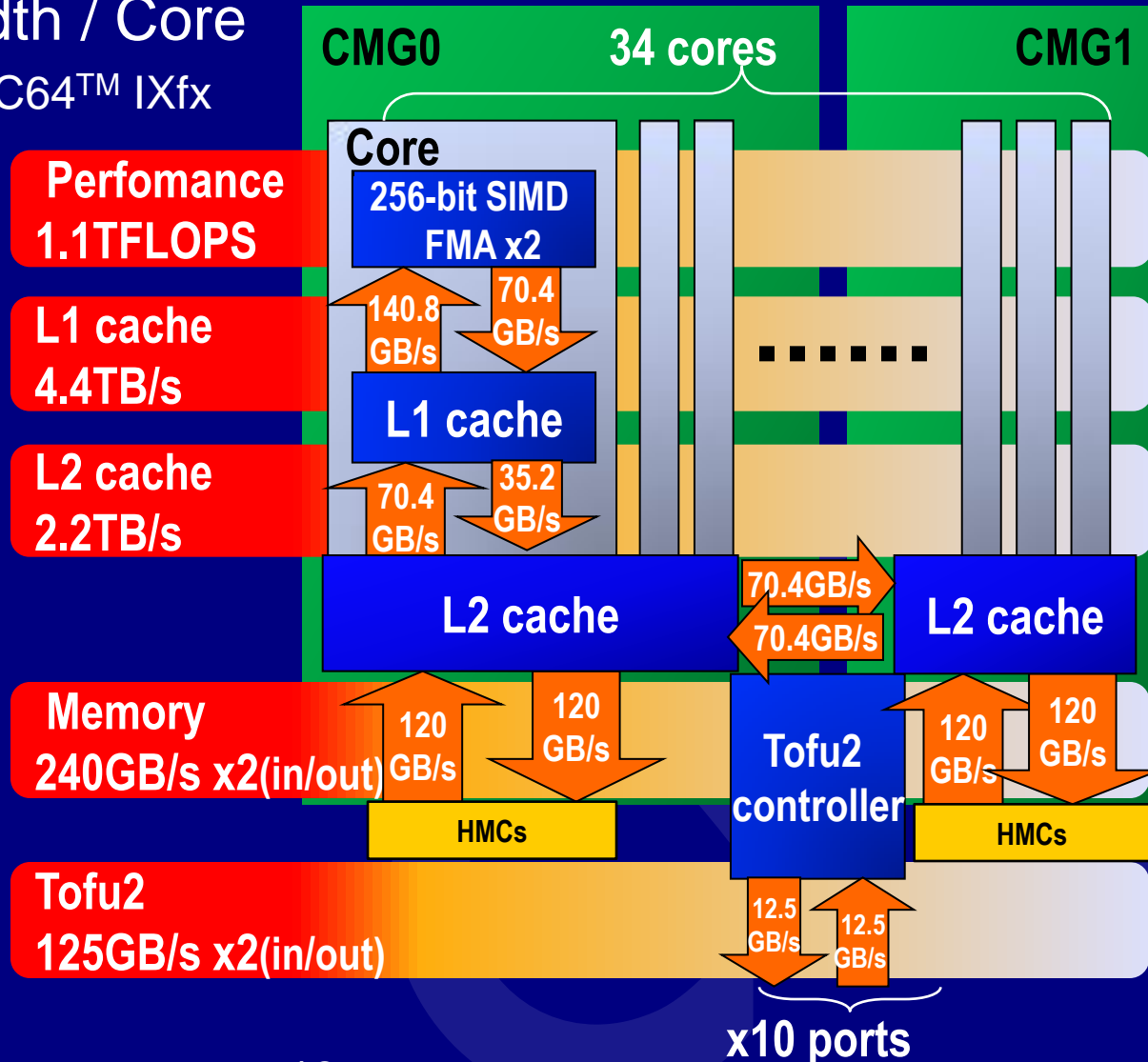
- Compared to SPARC64™ IXfx

- 8x HMC

- 15 Gbps
- 16 lanes
- 8 ports

- Tofu2

- 25 Gbps
- 4 lanes
- 10 ports



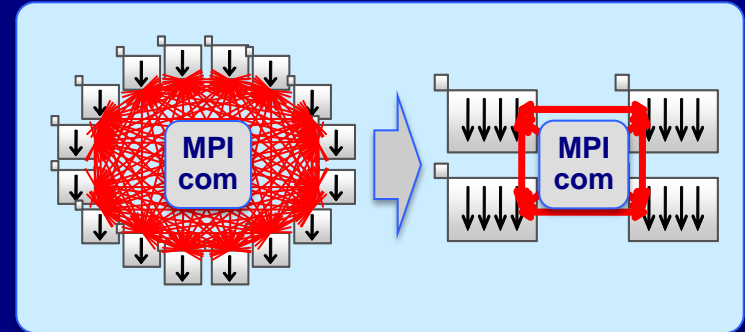
Agenda

- ◆ Fujitsu Processor Development
- ◆ SPARC64™ Xlfx
 - Design Concept and Processor Overview
 - Node Architecture
 - HPC-ACE2: ISA enhancements
 - Microarchitecture
 - Enhanced VISIMPACT and Sector Cache
 - Assistant Core
 - Performance
 - RAS
- ◆ Summary

Enhanced VISIMPACT

- Advantages of Hybrid Parallelization

- To reduce communication cost in highly parallel programs
- To increase user memory space by reducing communication buffer



- VISIMPACT* (introduced in FX1)

- Automatic parallelization technology by Fujitsu's compiler
- Hardware barrier for fast synchronization

- Enabling 8 sets of Hardware barriers between 32 cores

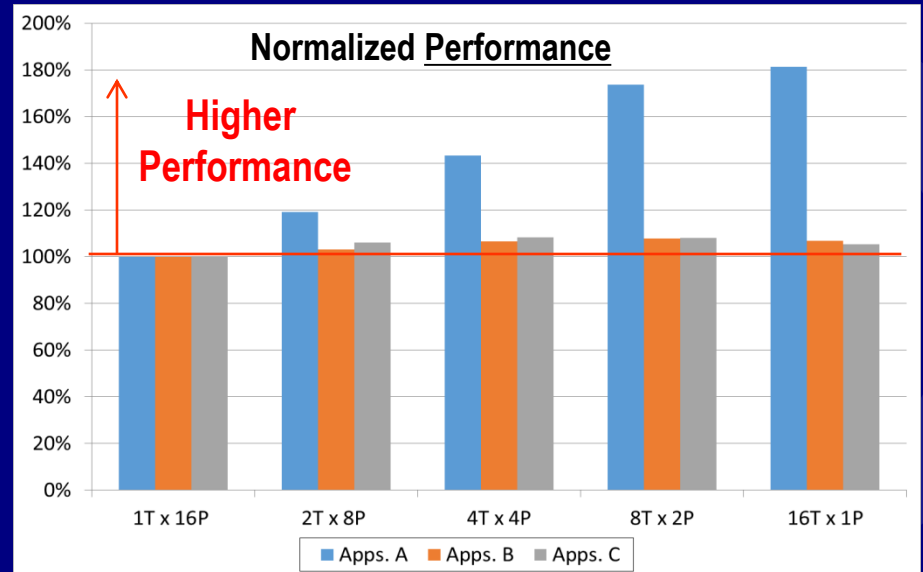
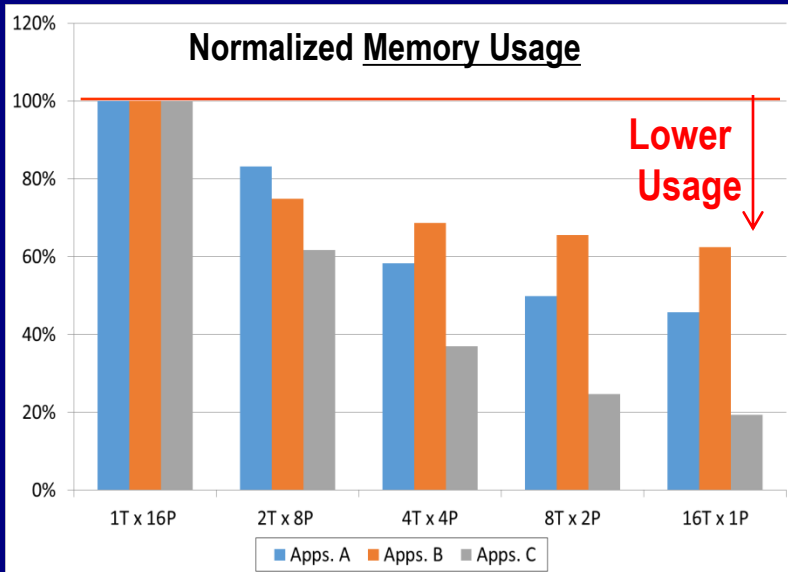
- Optimum combination of # Threads and # Processes depends on apps.
- Any combinations of T(Threads) and P(Processes) are supported
 - 32 T(Thread) x 1 P(Process), 16 T x 2 P, 8 T x 4 P, etc.
- The goal is heterogeneous hybrid parallelization for load imbalance and multi physics

*Virtual Single Processor by Integrated Multi-core Parallel Architecture

Effect of VISIMPACT

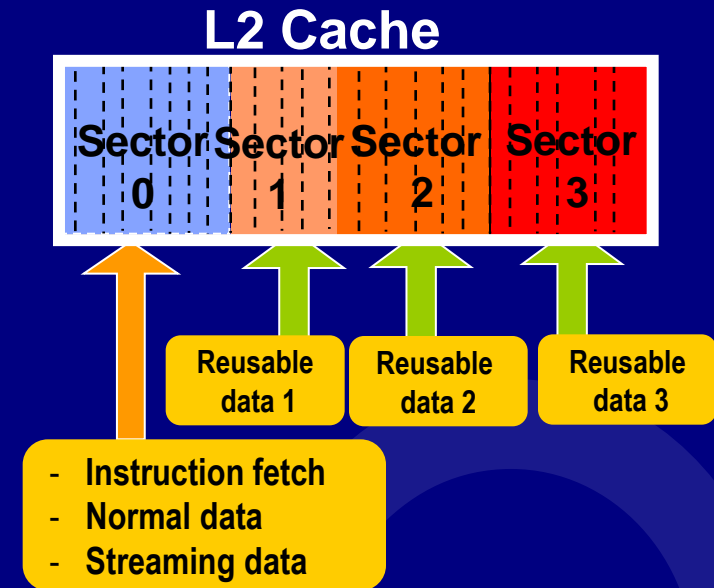
- Lower memory usage
 - By reducing communication buffer for MPI
- Higher performance
 - By reducing MPI communication cost

Memory usage and Performance
of #Threads x #Processes



Enhanced Sector Cache

- Sector Cache (introduced in K computer)
 - Cache line is replaced to keep specified sector size when cache miss occurs
- Like 'Local Memory'
 - Leave the reusable data on cache by dividing cache into segments
- Unlike 'Local Memory'
 - No need for a dedicated address
 - No penalty to save and restore in context switch
- SPARC64™ Xlfx supports 4 sectors in L1 cache (per core) and L2 cache (per CMG) respectively
 - More usable than SPARC64™ IXfx of 2 sectors in L1 and L2 respectively
 - Each sector size can be specified separately



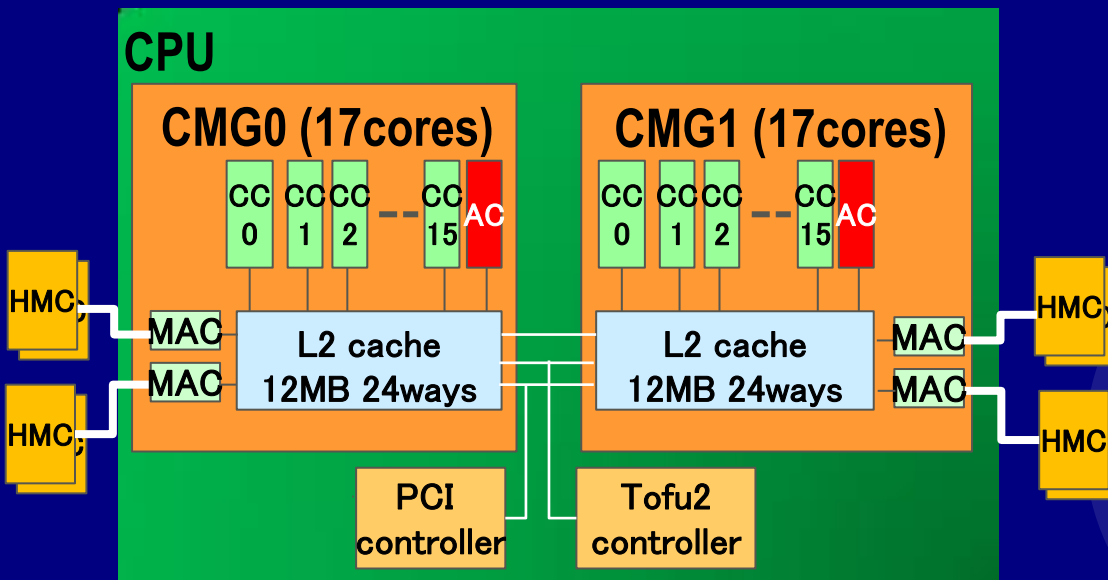
Agenda

- ◆ Fujitsu Processor Development
- ◆ SPARC64™ Xlfx
 - Design Concept and Processor Overview
 - Node Architecture
 - HPC-ACE2: ISA enhancements
 - Microarchitecture
 - Enhanced VISIMPACT and Sector Cache
 - Assistant Core
 - Performance
 - RAS
- ◆ Summary

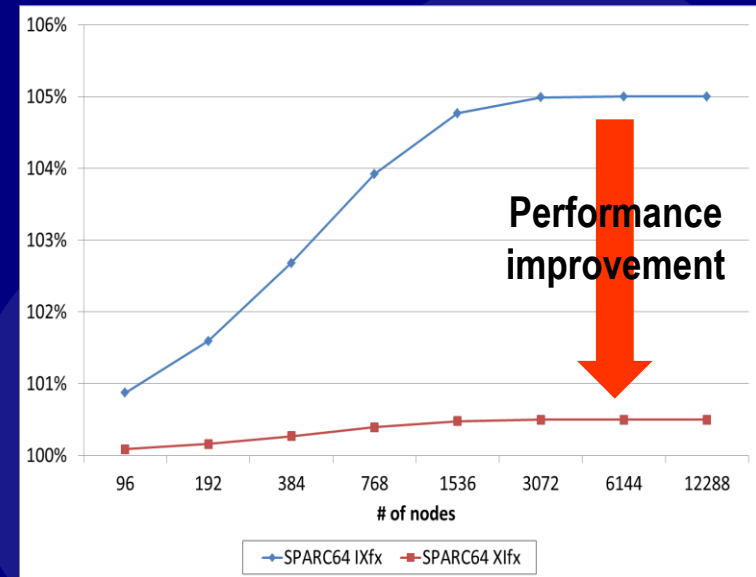
Assistant core

- Assistant core serves Daemon, IO, MPI asynchronous communication instead of computation
 - Each CMG has an assistant core allocated on 17th core
 - Sector cache within L2 cache allocates one sector to assistant core to avoid cache pollution
- Minimize performance degradation in large systems by jitter reduction

CPU block diagram



Perf degradation ratio by jitter (model)



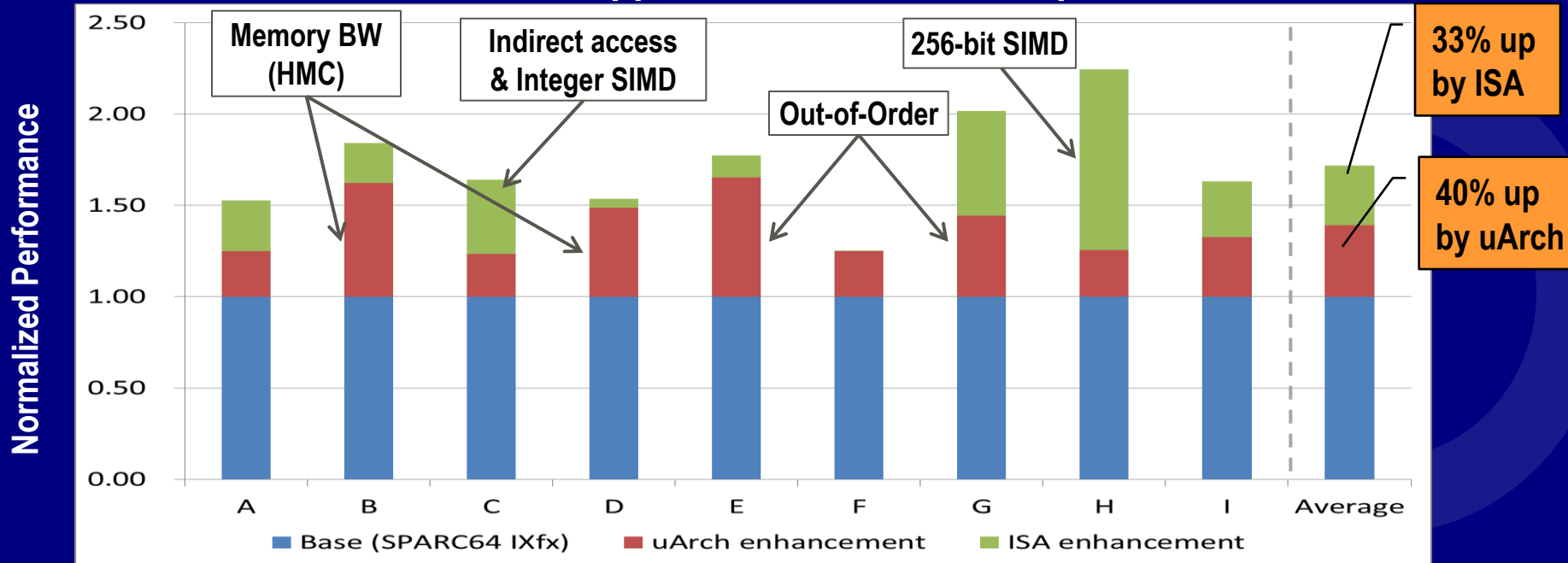
Agenda

- ◆ Fujitsu Processor Development
- ◆ SPARC64™ Xlfx
 - Design Concept and Processor Overview
 - Node Architecture
 - HPC-ACE2: ISA enhancements
 - Microarchitecture
 - Enhanced VISIMPACT and Sector Cache
 - Assistant Core
 - Performance
 - RAS
- ◆ Summary

Performance

- SPARC64™ Xlfx boosts performance up by ISA and microarchitectural enhancements
 - 97% execution efficiency for DGEMM
 - Sector cache realizes the same effect as 2.5x L1 cache size
 - 1.7x faster per core than SPARC64™ IXfx in real HPC applications such as fluid dynamics

Real HPC Applications Performance per Core



Agenda

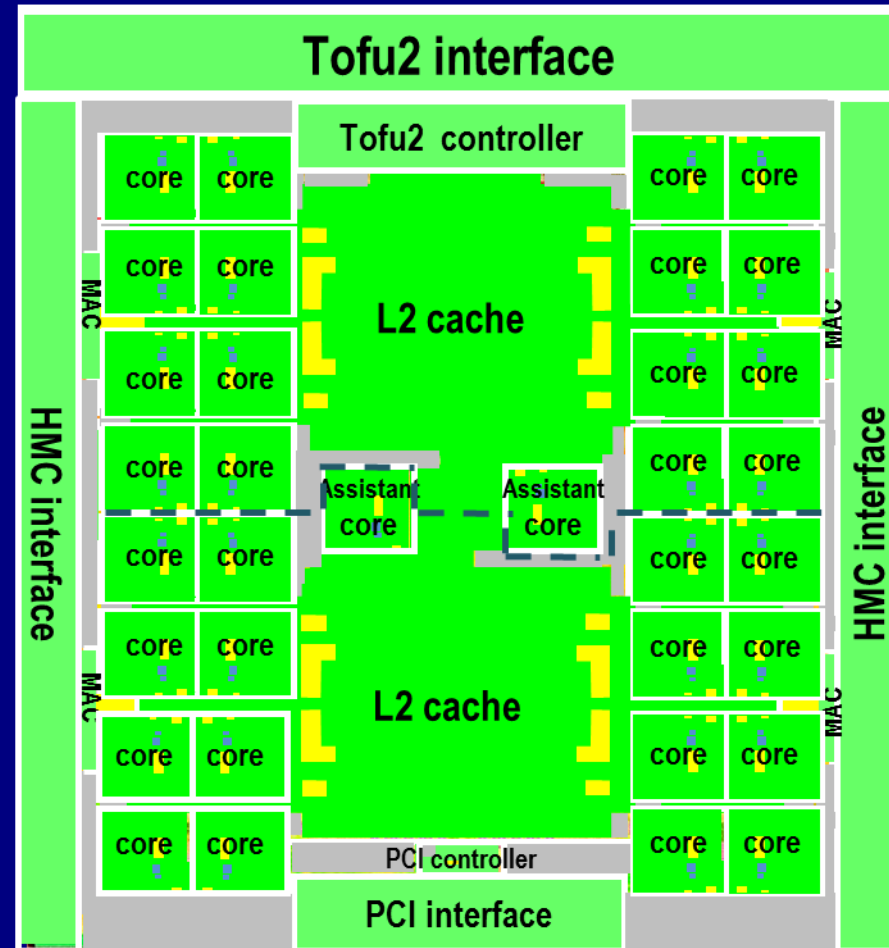
- ◆ Fujitsu Processor Development
- ◆ SPARC64™ Xlfx
 - Design Concept and Processor Overview
 - Node Architecture
 - HPC-ACE2: ISA enhancements
 - Microarchitecture
 - Enhanced VISIMPACT and Sector Cache
 - Assistant Core
 - Performance
 - RAS
- ◆ Summary

Reliability, Availability, Serviceability

- HPC system requires extensive RAS capability of CPU and interconnect
- SPARC64™ Xlfx inherits mainframe-level RAS features
 - # checkers in CPU increased to ~92,900
 - Tofu2 buses support self-recovery and lane dynamic degradation

Units	Error Detection and Correction
Cache (Tags)	ECC, Parity & Duplicate
Cache (Data)	ECC, Parity
Registers	ECC (INT/FP), Parity (Others)
ALUs	Parity, Residue
Other RAS features	
Cache dynamic degradation	
Hardware Instruction Retry	
Lane dynamic degradation for Tofu2	

SPARC64™ Xlfx RAS diagram



Green: 1-bit error Correctable
 Yellow: 1-bit error Detectable
 Gray: 1-bit error Harmless

Agenda

- ◆ Fujitsu Processor Development
- ◆ SPARC64™ Xlfx
 - Design Concept and Processor Overview
 - Node Architecture
 - HPC-ACE2: ISA enhancements
 - Microarchitecture
 - Enhanced VISIMPACT and Sector Cache
 - Assistant Core
 - Performance
 - RAS
- ◆ Summary

Summary

- ◆ SPARC64™ Xlfx is Fujitsu's latest SPARC processor, designed for massively parallel supercomputing systems
- ◆ Enhance and inherit K computer features
 - ◆ Stand alone scalar many-core architecture
 - ◆ VISIMPACT and Sector Cache
 - ◆ On-chip integrated Tofu2
- ◆ Introduce new technologies to EXA scale
 - ◆ HPC-ACE2
 - ◆ HMC
 - ◆ Assistant cores
- ◆ SPARC64™ Xlfx has improved performance of real HPC applications significantly
- ◆ As a next step, Fujitsu goes forward to EXA scale supercomputing

Abbreviations

- SPARC64™ Xlfx
 - RSA: Reservation Station for Address generation
 - RSE: Reservation Station for Execution
 - RSF: Reservation Station for Floating-point
 - RSBR: Reservation Station for Branch
 - GUB: General-purpose Update Buffer
 - FUB: Floating-point Update Buffer
 - GPR: General-Purpose Register
 - FPR: Floating-Point Register
 - CSE: Commit Stack Entry
 - EAG: Effective Address Generator
 - EX : Execution unit (Integer)
 - FL : Floating-point unit
 - HPC-ACE: High Performance Computing-Arithmetic Computational Extensions
 - HMC: Hybrid Memory Cube
 - Tofu: Torus-Fusion