Report on Exascale Architecture Roadmap in Japan

Masaaki Kondo (UEC-Tokyo)

(presented on behalf of SDHPC architecture WG)

Our Mission

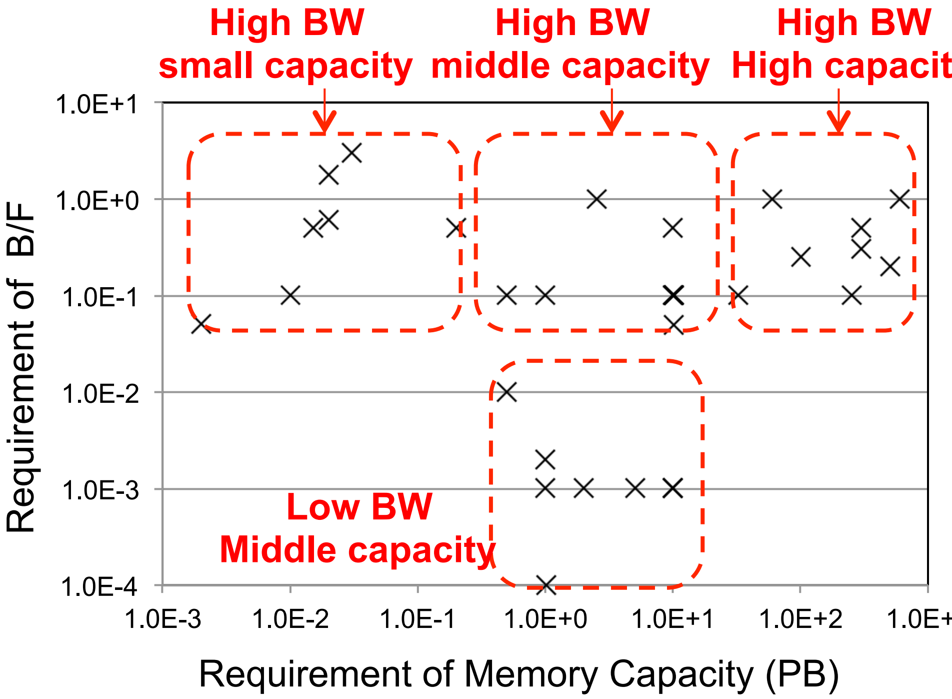
- Studying key technologies in achieving Exascale systems available in 2018-2020
- Investigating effective Exascale architectures for target sciences in collaboration with application WG
- Developing roadmap towards Exascale systems
  - Performance prediction based on technological trends
  - Listing technological challenges to Exascale systems
  - Breaking down R&D issues
    - Processor architecture
    - Memory subsystem
    - Managing huge-scale parallelism, Interconnection network
    - Power efficiency
    - Dependability
- Presenting an image of Exascale systems

Strategic Development of Exascale Systems

- Exascale systems
  - Cannot be built upon traditional technological advances.
  - Needs special efforts in architecture / system software for developing effective (useful) Exascale systems
- Strategy
  - HW/SW/Application co-design
  - Close cooperation with the application WG
  - Architecture design suited for target application requirements
  - Exploring best-matching between available technologies and application requirements

System Requirement for Target Sciences

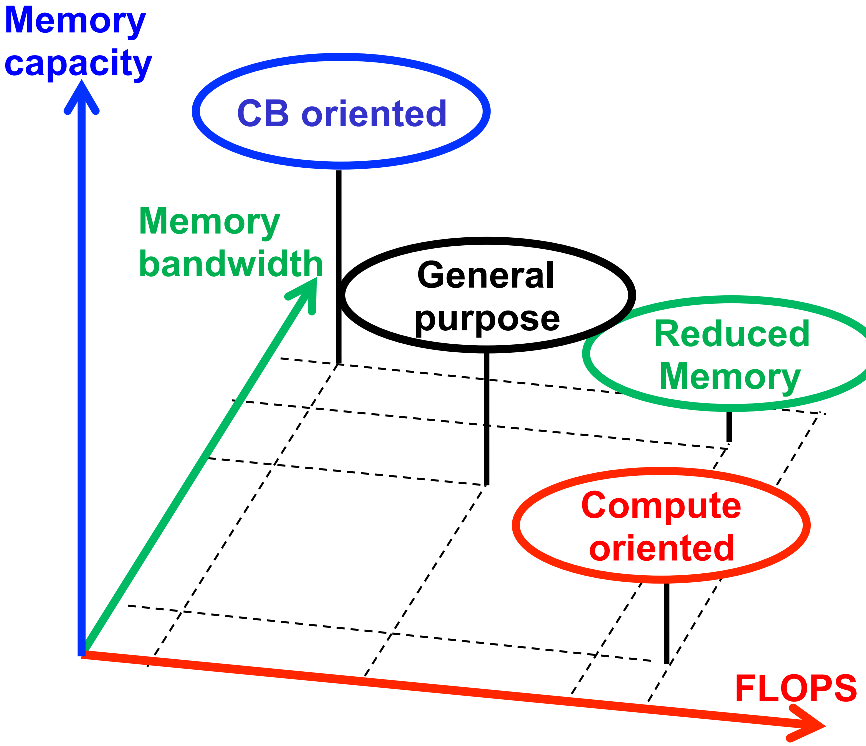
- System performance
  - FLOPS: 800 – 2500PFLOPS
  - Memory capacity: 10TB – 500PB
  - Memory bandwidth: 0.001 – 1.0 B/F
  - Example applications
    - Small capacity requirement
      - □ MD, Climate, Space physics, …
    - Small BW requirement
      - □ Quantum chemistry, …
    - High capacity/BW requirement
      - □ Incompressibility fluid dynamics, …



- Interconnection Network
  - Not enough analysis has been carried out
  - Some applications need >1us latency and large bisection BW
- Storage
  - There is not so big demand

Candidate of ExaScale Architecture

- Four types of architectures are considered
  - General Purpose (GP)
    - Ordinary CPU-based MPPs
    - e.g.) K-Computer, GPU, Blue Gene, x86-based PC-clusters
  - Capacity-Bandwidth oriented (CB)
    - With expensive memory-I/F rather than computing capability
    - e.g.) Vector machines
  - Reduced Memory (RM)
    - With embedded (main) memory
    - e.g.) SoC, MD-GRAPE4, Anton
  - Compute Oriented (CO)
    - Many processing units
    - e.g.) ClearSpeed, GRAPE-DR



Performance Projection

- Performance projection for an HPC system in 2018
  - Achieved through continuous technology development
  - Constraints: 20 – 30MW electricity & 2000sqm space

Node Performance

	Total CPU Performance (PetaFLOPS)	Total Memory Bandwidth (PetaByte/s)	Total Memory Capacity (PetaByte)	Byte / Flop
General Purpose	200~400	20~40	20~40	0.1
Capacity-BW Oriented	50~100	50~100	50~100	1.0
Reduced Memory	500~1000	250~500	0.1~0.2	0.5
Compute Oriented	1000~2000	5~10	5~10	0.005

Network

	Injection	P-to-P	Bisection	Min Latency	Max Latency
High-radix (Dragonfly)	**32 GB/s**	**32 GB/s**	**2.0 PB/s**	**200 ns**	**1000 ns**
Low-radix (4D Torus)	**128 GB/s**	**16 GB/s**	**0.13 PB/s**	**100 ns**	**5000 ns**

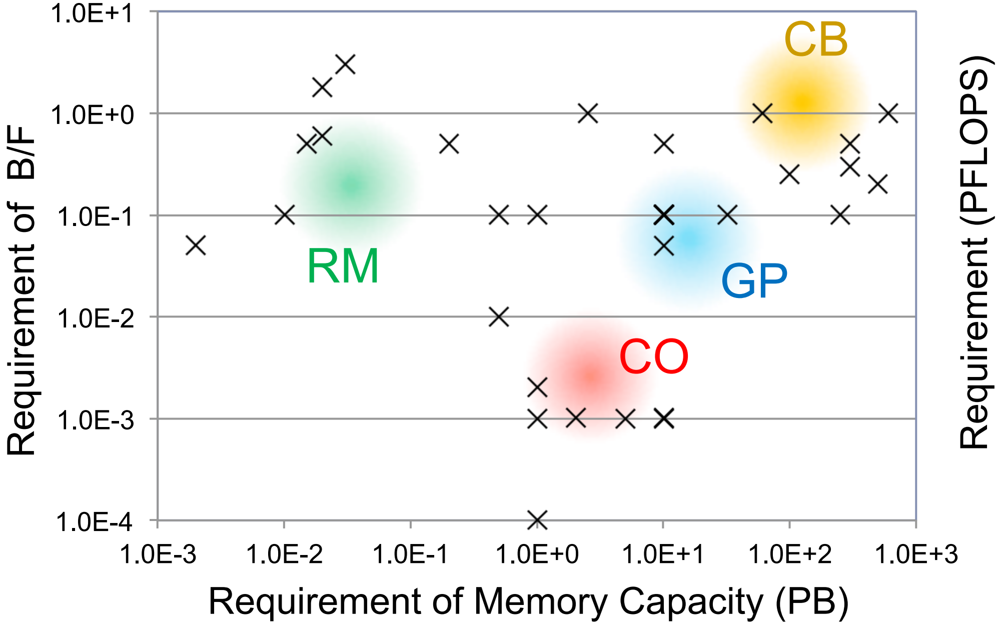
Storage

Total Capacity	**Total Bandwidth**
1 EB	10TB/s
100 times larger than main memory	For saving all data in memory to disks within 1000-sec.

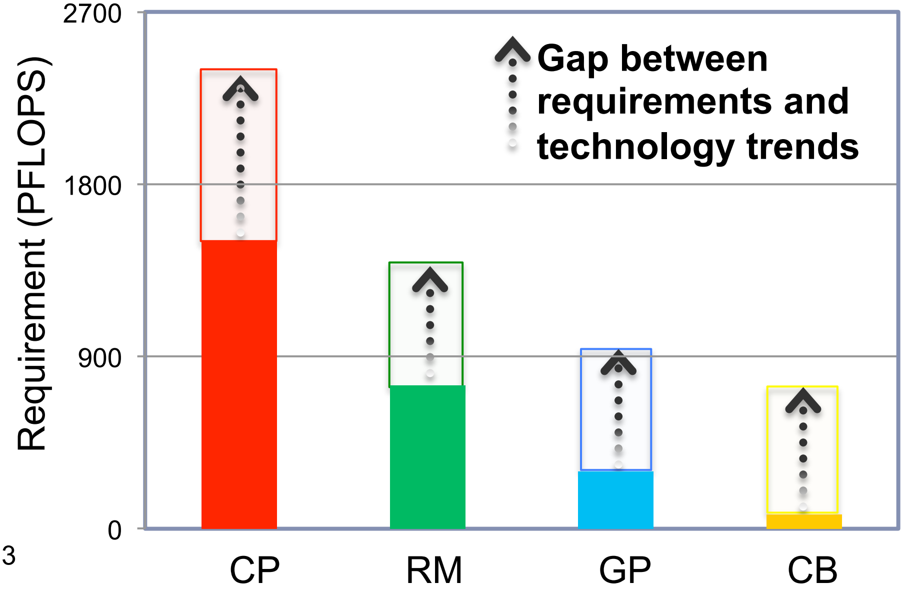
Gap Between Requirement and Technology Trends

- Mapping four architectures onto science requirement
- Projected performance vs. science requirement
  - Big gap between projected and required performance

Mapping of Architectures



Projected vs. Required Perf.



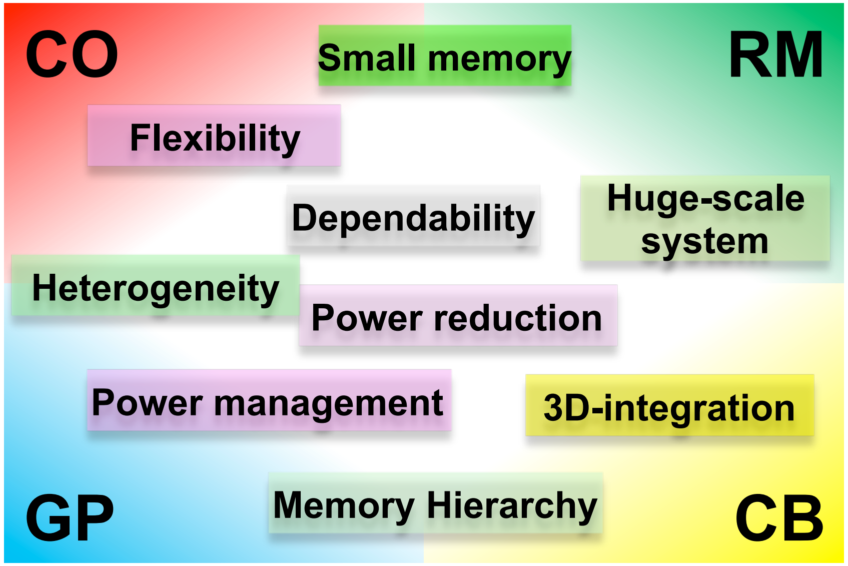
Needs national research project for science-driven HPC systems

Issues Towards Exascale Systems

- There are several issues for developing science-driven Exascale Systems
- Common issues
  - Limitation of power consumption, system footprint, cost
- General Purpose (GP)
  - Needs to augment advantages compared to commodity machines
- Capacity-Bandwidth oriented (CB)
  - Currently, no clear benefit compared to GP in terms of power & cost
  - Needs to improve power-performance efficiency
- Reduced memory (RM) & Compute oriented (CO)
  - Application range is limited due to memory constraints
  - Co-design with application people is important

Challenges Toward Exascale System Development

- Challenges in all architectures
  - Power efficiency, Power management, Dependability
- Challenges in each architecture
  - General Purpose (GP)
    - Multi-level memory hierarchy
    - Management of heterogeneity
  - Capacity-Bandwidth oriented (CB)
    - Memory system power reduction (3D-ICs, smart memory)
  - Reduced Memory (RM)
    - On-chip network
    - Small memory algorithm
    - Huge-scale system management
  - Compute Oriented (CO)
    - Flexibility to wide variety of sciences

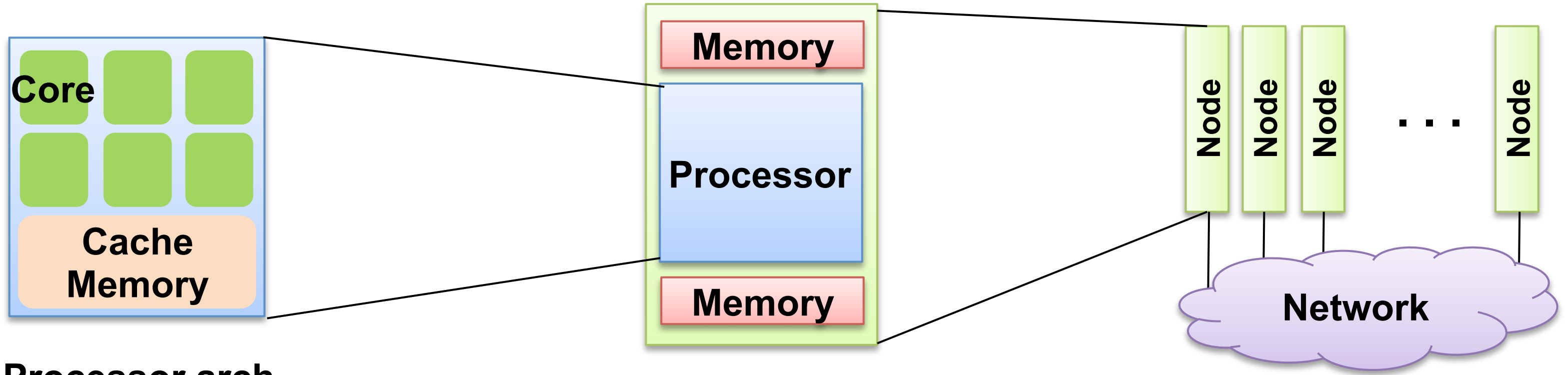


Research Directions (in part)

- Power reduction
  - About 60x performance-power improvement is required beyond traditional CMOS scaling
  - Possible technology candidates
    - New devices: SOTB, 3D-IC, Near threshold Vdd
    - Low-power memory: NVRAM, Wide-I/O, Hybrid memory cube
    - Low-power Interconnect: power-efficient topology & switches
    - System-level power management: power-capping, power monitoring
- Heterogeneous architecture
  - Providing flexibility and high effective performance is important
  - Data-sharing between latency and throughput cores or among throughput cores
    - Implicit data transfer or explicit sharing, cache coherence, etc.
  - Communication network between latency and throughput cores

Overview of an Exascale System

- An example system image of GP architecture
  - GP is a basis of all types of architectures
- Explored each of the following system layers
  - Processor arch. (core and cache configuration)
    - Latency / throughput core, on-chip main memory
  - Node arch. (connection between processor and memory)
    - CPU-memory 3D integration, #CPUs per node
  - System arch. (interconnection network)
    - High-radix / Low-radix network



Processor arch.
(core, cache, …)

Node arch.
(Processor, Memory)

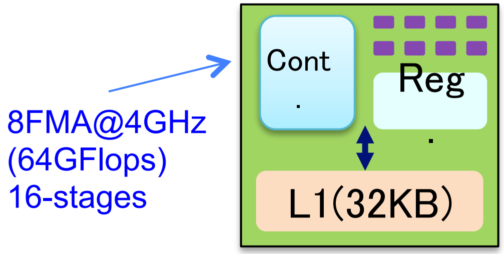
System arch.
(interconnection NW)

Processor Architecture

Latency Core (LC)

- High clock-speed
- Deep pipeline
- Out-of-order, Branch-prediction
- Cache, Prefeching, …

single-thread performance

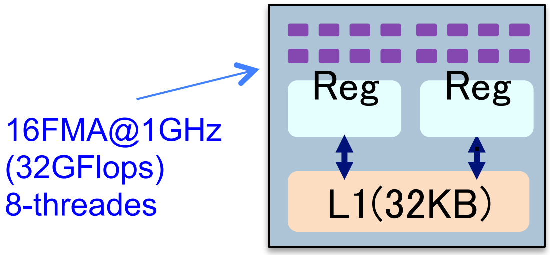


8FMA@4GHz
(64GFlops)
16-stages

Throughput Core (TC)

- Low clock-speed
- Shallow pipeline
- Simple in-order
- Multi-thread support

good power efficiency

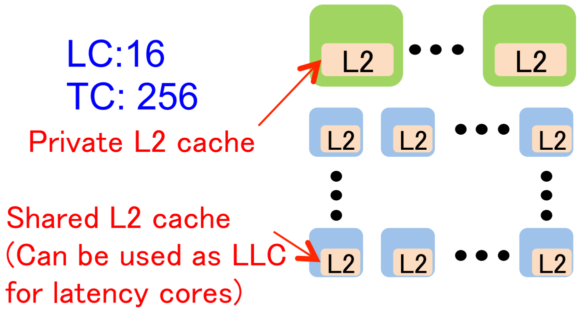


16FMA@1GHz
(32GFlops)
8-threades

Heterogeneous

- Combined LCs and TCs
  (On-chip or Off-chip)
- Complicates programming

both single/multi-thread perf.



LC:16
TC: 256

Private L2 cache

Shared L2 cache
(Can be used as LLC
for latency cores)

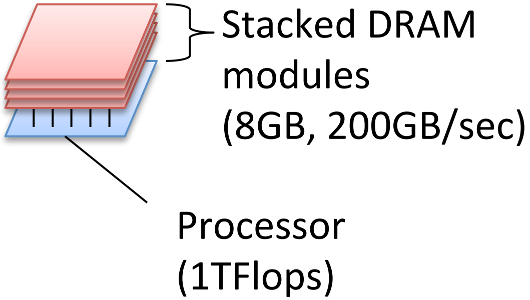
	# cores	FLOPS	Clock speed	LLC
Latency Cores only	32	2TFLOPS	4GHz	128MB
Throughput Cores only	512	16TFLOPS	1GHz	128MB
Heterogeneous (area of LC:TC = 1:1)	16L+256T	9TFLOPS	4GHz/1GHz	128MB
(c.f. K-computer (58W/CPU)	8	128GFLOPS	2GHz	6MB

Assumption: each core consumes 50-200W power

Node Architecture

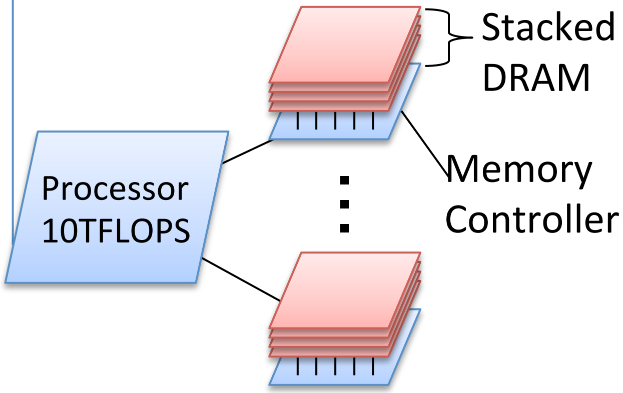
Thin node

- 3D CPU-memory integration with Wide I/O technology
- Power: 2-20W / node
- # of nodes: 1M nodes



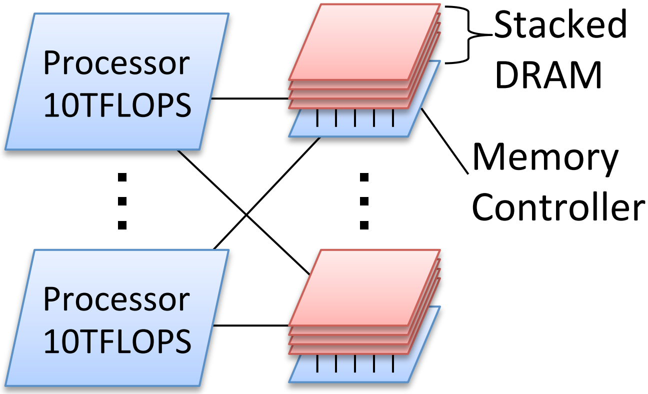
Middle node

- Stacked DRAM with high-speed memory I/O (HMC)
- 1 CPU + Multi memory module
- Power:20-200W / node
- # of nodes: 100K nodes



Large Node

- Stacked DRAM with high-speed memory I/O (HMC)
- Multi CPU + Multi memory module
- Power: ~2000W / node
- # of nodes: 10K nodes

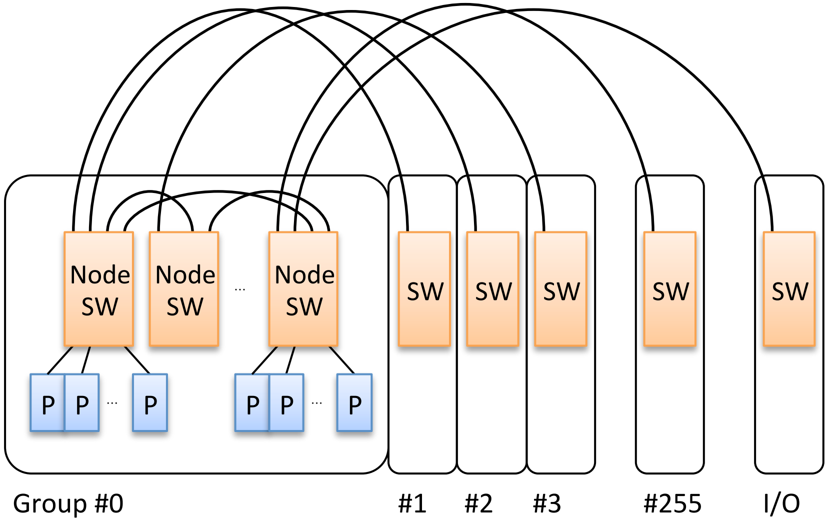


	Performance	Memory Capacity	Memory BW	B/F
Thin Node	1TFLOPS	8GB	200GB/s	0.2
Middle Node	10TFLOPS	128GB	1000GB/s	0.1
Large Node	80TFLOPS	1024GB	8000GB/s	0.1
(c.f.) K-Computer	128GFLOPS	16GB	64GB/s	0.5

(We assume half of the power is consumed by processor)

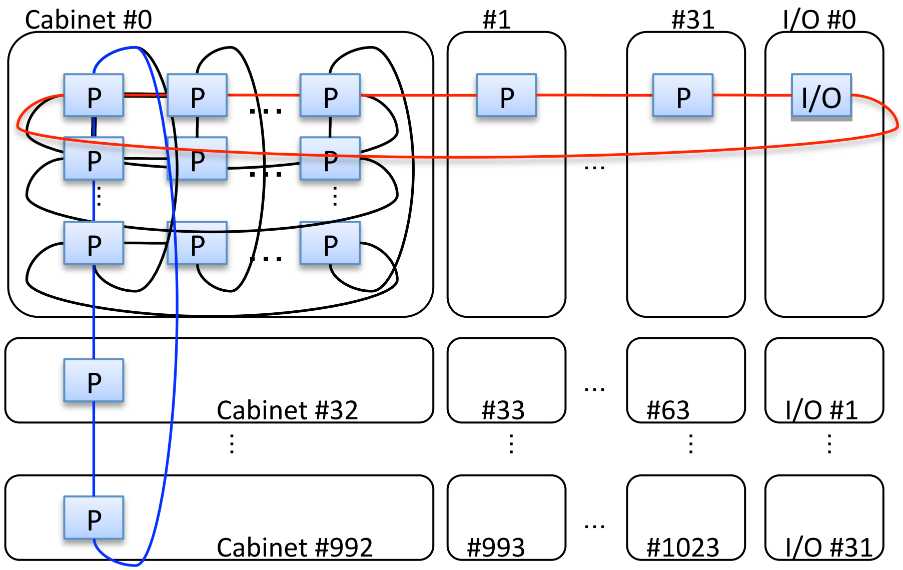
System Architecture

- High-radix NW (e.g. Dragonfly)
  - Latency ☺ latency to farthest node
    ☹ latency to adjacent node
  - Throughput ☺ bisection BW
    ☹ injection BW



(multiple cabinet forms one group)

- Low-radix NW (e.g. 4D-Torus)
  - Latency ☺ latency to adjacent node
    ☹ latency to farthest node
  - Throughput ☺ injection BW
    ☹ bisection BW



	P2P	Injection	Bisection	Min-Latency	Max-Latency
High-Radix(Dragonfly)	32GB/s	32GB/s	2.0PB/s	200ns	1000ns
Low-Radix（4D Torus）	16GB/s	128GB/s	0.13PB/s	100ns	5000ns

Research Issues

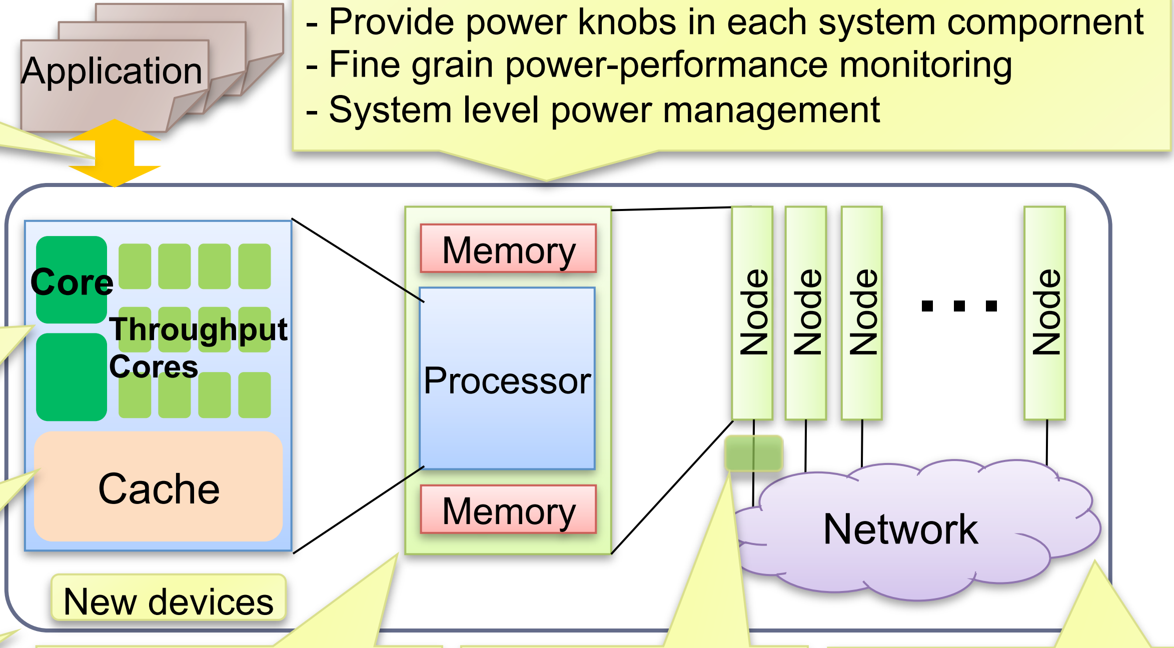
- Key R&D issues in each system component

- Arch. development with co-design
- Dynamic HW-adaptation

- Provide power knobs in each system compornent
- Fine grain power-performance monitoring
- System level power management

-Data-sharing & network between latency&throughput cores
- Data-sharing among throughput cores
- High-perf. &Low-power NoC

- Memory Hierarchy design
- Refine on-chip memory arch.
- On-chip main memory



- Checkpointing support
- Migration support
- HW monitoring for fault prediction

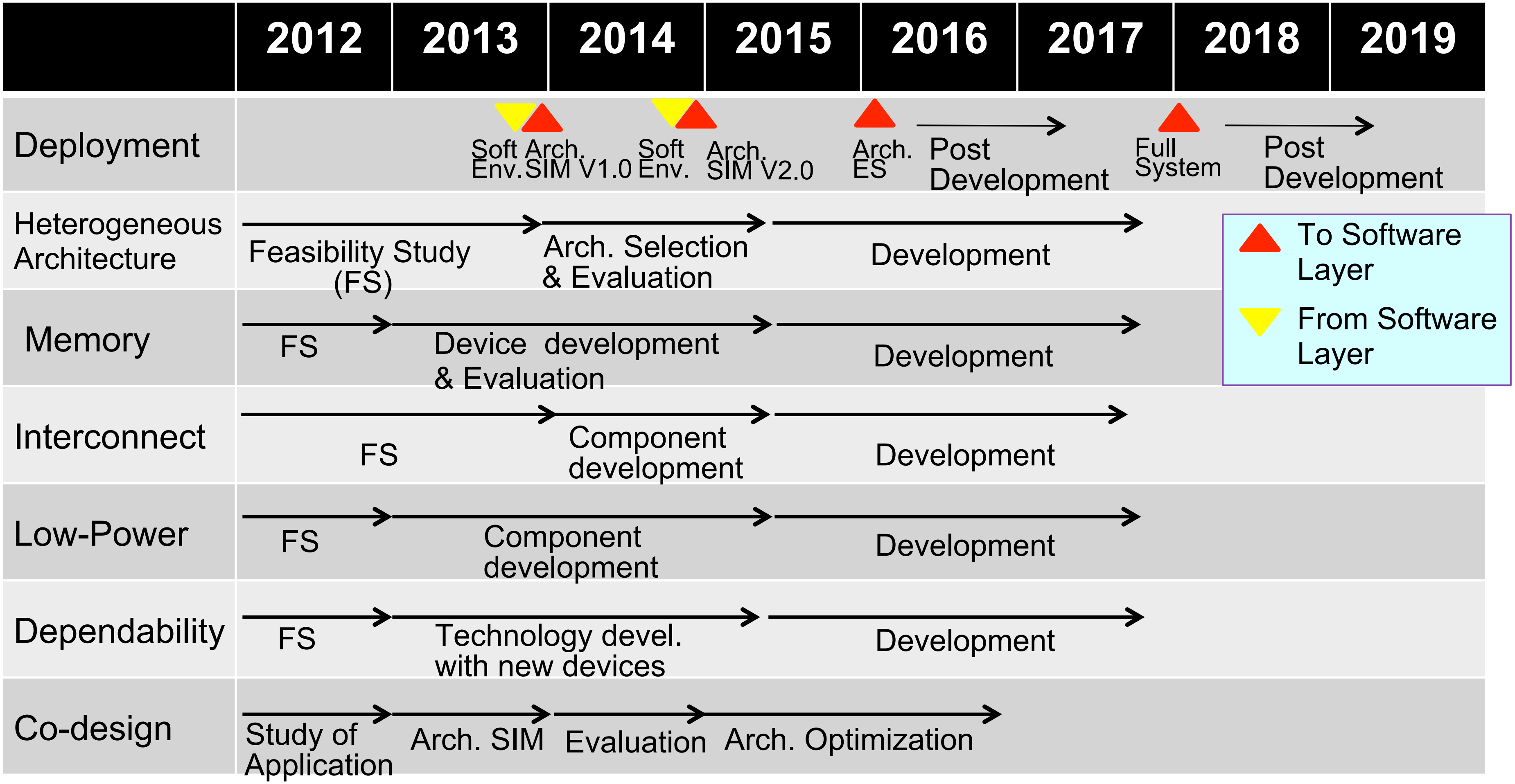
- 3D memory integration
- Wide I/O, HMC
- NVRAM
- Smart Memories

- Intelligent NI
- Collective comm. support
- fine-grain barriers

- Suitability of High/Low-Radix NW
- Optimization for Collective Comm.
- QoS management

Roadmap of Exascale System Development

- Timeline towards deployment of Exascale Systems



2012 2013 2014 2015 2016 2017 2018 2019

Deployment: Soft Env. ▼ Arch. SIM V1.0 ▲ Soft Env. ▼ Arch. SIM V2.0 ▲ Arch. ES ▲ Post Development Full System ▲ Post Development

Heterogeneous Architecture: Feasibility Study (FS) → Arch. Selection & Evaluation → Development

Memory: FS → Device development & Evaluation → Development

Interconnect: FS → Component development → Development

Low-Power: FS → Component development → Development

Dependability: FS → Technology devel. with new devices → Development

Co-design: Study of Application → Arch. SIM → Evaluation → Arch. Optimization

▲ To Software Layer

▼ From Software Layer

Summary

- Exascale architectures required for future sciences
- Roadmap towards Exascale systems
  - Performance projection based on technological trends
  - Technological challenges
  - Breaking down of research issues
- A system image of Exascale systems

- For science-driven Exascale systems, it is necessary to explore system architecture via HW/SW/Application co-design

Acknowledgement

- This material and the document of Exascale architecture roadmap is written in cooperation with the following colleagues
  - Yuichiro Ajima (Fujitsu)
  - Yasuo Ishii (NEC)
  - Koji Inoue (Kyushu Univ.)
  - Toshihiro Hanawa (Univ. of Tsukuba)
  - Michihiro Koibuchi (NII)
  - Yukinori Sato (JAIST)
  - Kentaro Sano (Tohoku Univ.)

  Advisory
  - * Satoshi Matsuoka (Titech)
  - * Hiroshi Nakamura (Univ. Tokyo)
  - * Kei Hiraki (Univ. Tokyo)