Japan's post K Computer

Yutaka Ishikawa
Project Leader
RIKEN AICS

HPC User Forum, 7th September, 2016

Outline of Talk

- Introduction of FLAGSHIP2020 project
- An Overview of post K system
- Concluding Remarks

An Overview of Flagship 2020 project

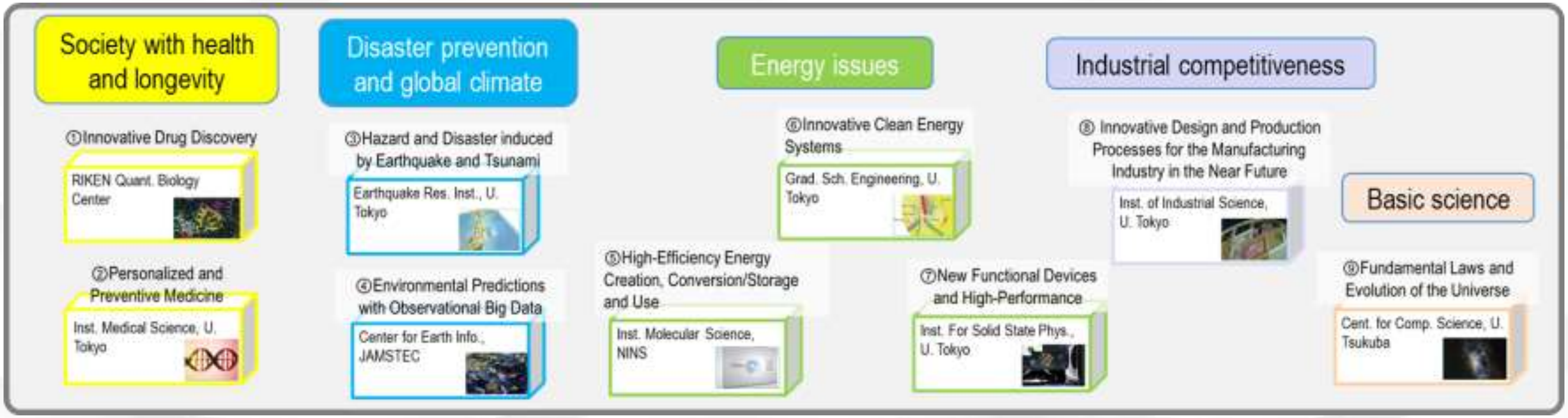
- **Developing the next Japanese flagship computer, so-called "post K"**

Vendor partner

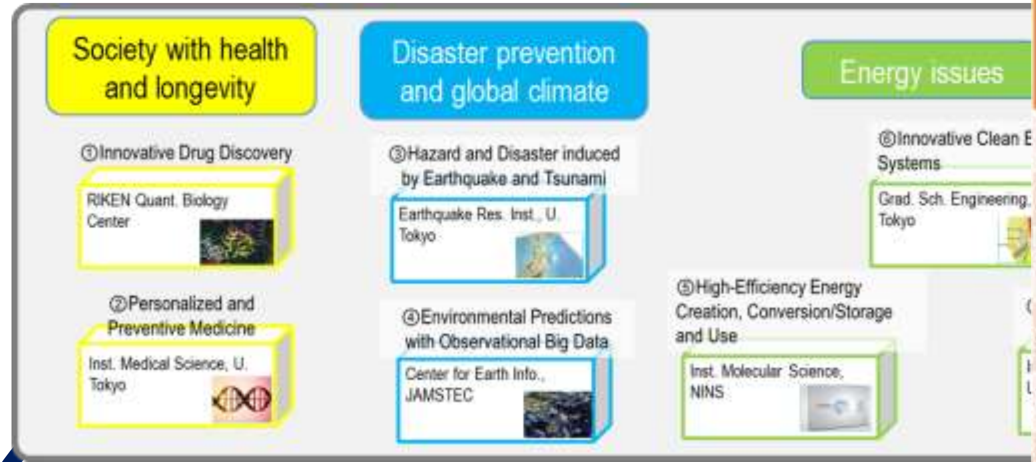
FUJITSU

- **Developing a wide range of application codes, to run on the "post K", to solve major social and science issues**

The Japanese government selected 9 social & scientific priority issues and their R&D organizations.



Co-design



	Target Application	
	Program	**Brief description**
①	GENESIS	MD for proteins
②	Genomon	Genome processing (Genome alignment)
③	GAMERA	Earthquake simulator (FEM in unstructured & structured grid)
④	NICAM+LETK	Weather prediction system using Big data (structured grid stencil & ensemble Kalman filter)
⑤	NTChem	molecular electronic (structure calculation)
⑥	FFB	Large Eddy Simulation (unstructured grid)
⑦	RSDFT	an ab-initio program (density functional theory)
⑧	Adventure	Computational Mechanics System for Large Scale Analysis and Design (unstructured grid)
⑨	CCS-QCD	Lattice QCD simulation (structured grid Monte Carlo)

Architectural Parameters

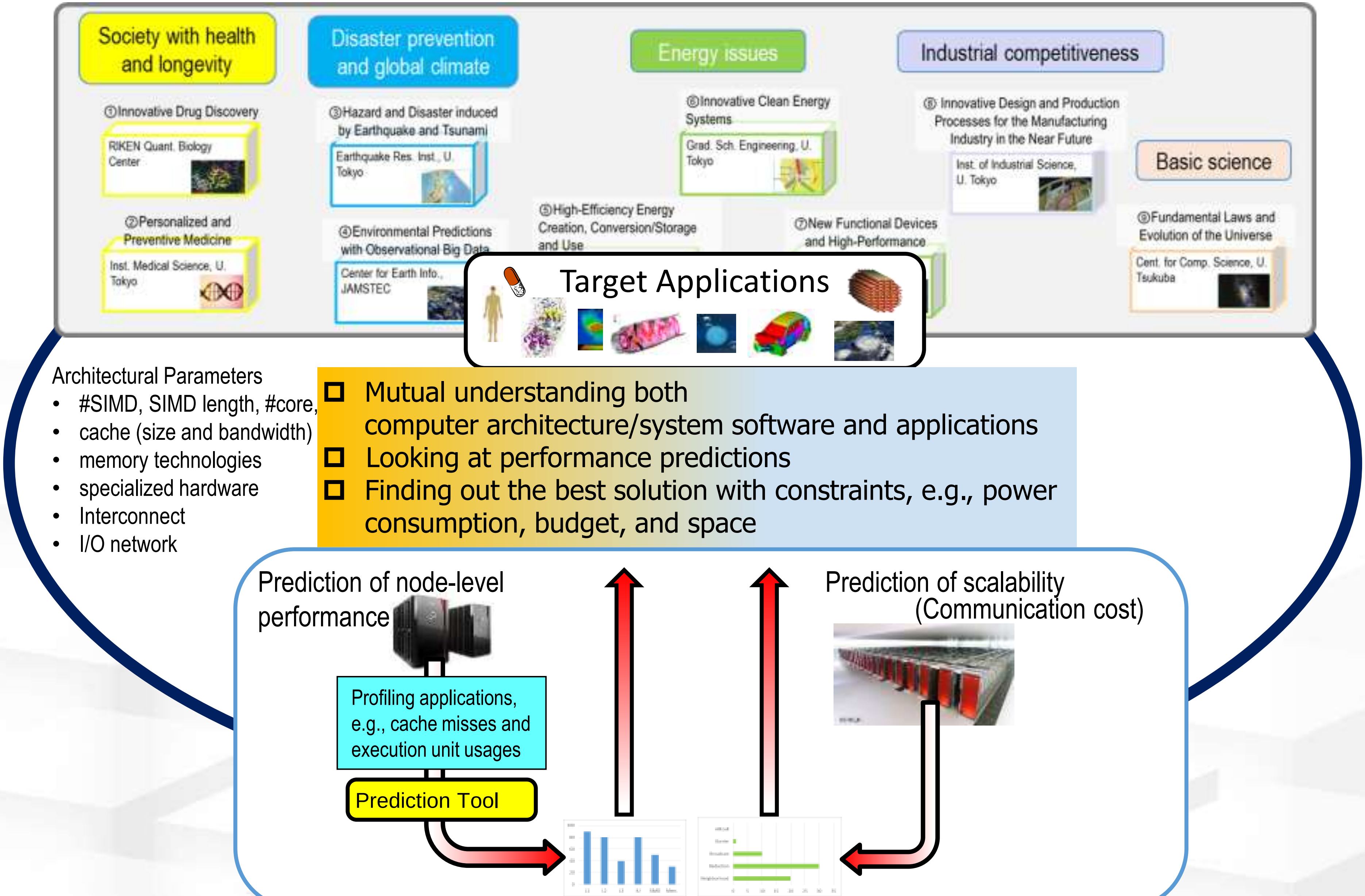
- #SIMD, SIMD length, #core, #NUMA node
- cache (size and bandwidth)
- memory technologies
- specialized hardware
- Interconnect
- I/O network

Target Application

Target Applications' Characteristics

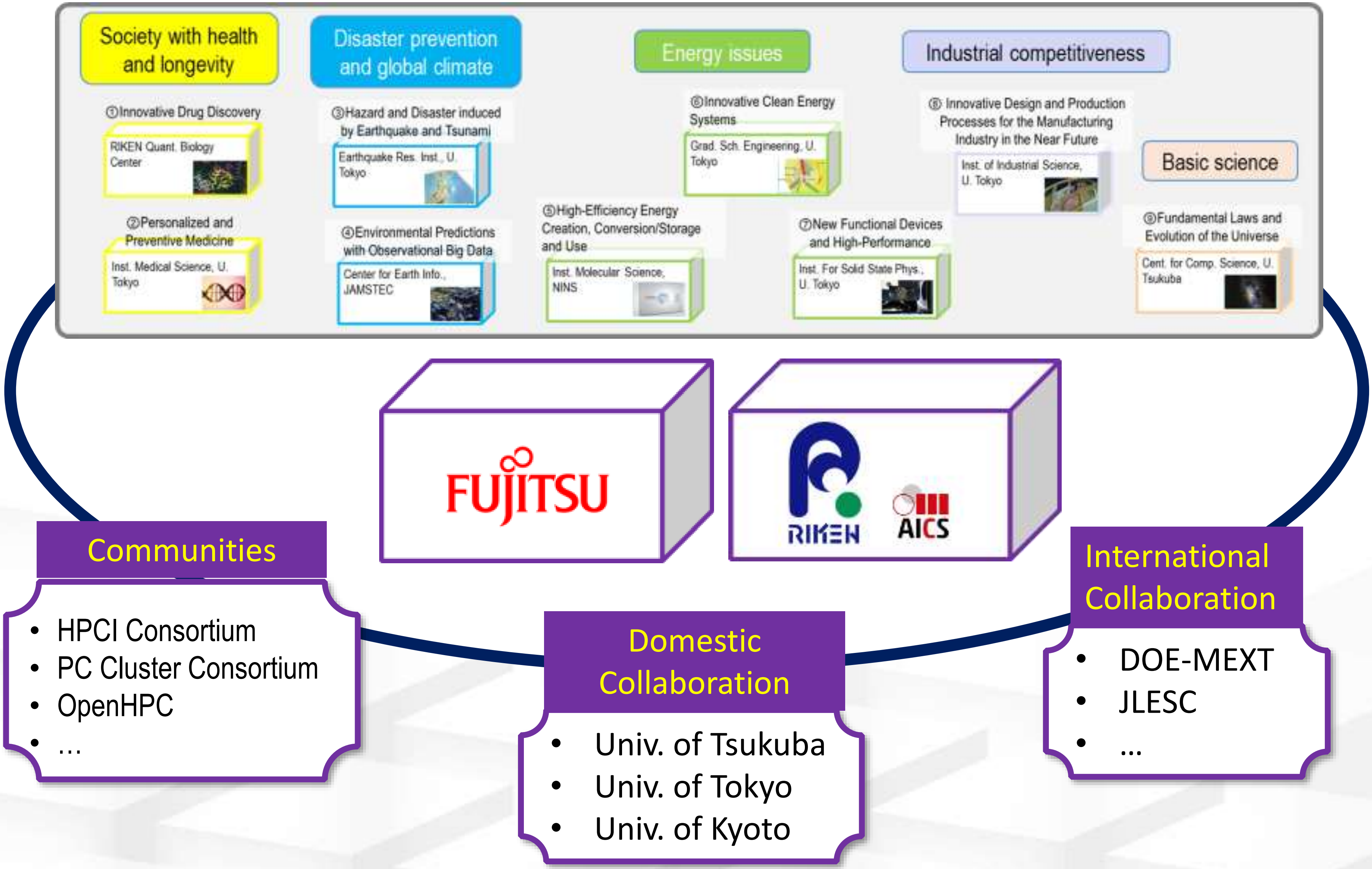
	Target Application		
	Program	Brief description	Co-design
①	GENESIS	MD for proteins	Collective comm. (all-to-all), Floating point perf (FPP)
②	Genomon	Genome processing (Genome alignment)	File I/O, Integer Perf.
③	GAMERA	Earthquake simulator (FEM in unstructured & structured grid)	Comm., Memory bandwidth
④	NICAM+LETK	Weather prediction system using Big data (structured grid stencil & ensemble Kalman filter)	Comm., Memory bandwidth, File I/O, SIMD
⑤	NTChem	molecular electronic (structure calculation)	Collective comm. (all-to-all, allreduce), FPP, SIMD,
⑥	FFB	Large Eddy Simulation (unstructured grid)	Comm., Memory bandwidth,
⑦	RSDFT	an ab-initio program (density functional theory)	Collective comm. (bcast), FFP
⑧	Adventure	Computational Mechanics System for Large Scale Analysis and Design (unstructured grid)	Comm., Memory bandwidth, SIMD
⑨	CCS-QCD	Lattice QCD simulation (structured grid Monte Carlo)	Comm., Memory bandwidth, Collective comm. (allreduce)

Co-design



Society with health and longevity
Disaster prevention and global climate
Energy issues
Industrial competitiveness
Basic science
①Innovative Drug Discovery
RIKEN Quant. Biology Center
②Personalized and Preventive Medicine
Inst. Medical Science, U. Tokyo
③Hazard and Disaster induced by Earthquake and Tsunami
Earthquake Res. Inst., U. Tokyo
④Environmental Predictions with Observational Big Data
Center for Earth Info., JAMSTEC
⑤High-Efficiency Energy Creation, Conversion/Storage and Use
⑥Innovative Clean Energy Systems
Grad. Sch. Engineering, U. Tokyo
⑦New Functional Devices and High-Performance
⑧ Innovative Design and Production Processes for the Manufacturing Industry in the Near Future
Inst. of Industrial Science, U. Tokyo
⑨Fundamental Laws and Evolution of the Universe
Cent. for Comp. Science, U. Tsukuba
Target Applications
Architectural Parameters
• #SIMD, SIMD length, #core,
• cache (size and bandwidth)
• memory technologies
• specialized hardware
• Interconnect
• I/O network
Mutual understanding both computer architecture/system software and applications
Looking at performance predictions
Finding out the best solution with constraints, e.g., power consumption, budget, and space
Prediction of node-level performance
Profiling applications, e.g., cache misses and execution unit usages
Prediction Tool
Prediction of scalability (Communication cost)

R&D Organization

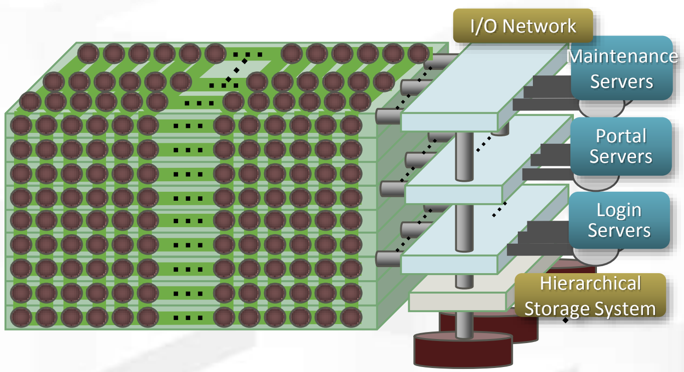


Outline of Talk

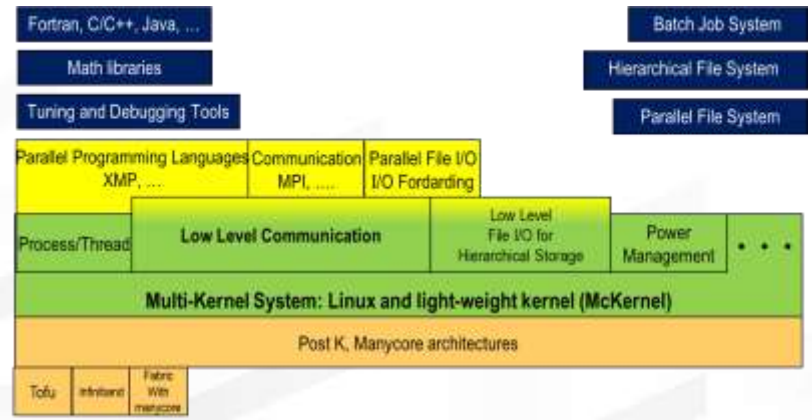
- Introduction of FLAGSHIP2020 project
- **An Overview of post K system**
- Concluding Remarks

An Overview of post K

- **Hardware**
  - Manycore architecture
  - 6D mesh/torus Interconnect
  - 3-level hierarchical storage system
    - Silicon Disk
    - Magnetic Disk
    - Storage for archive

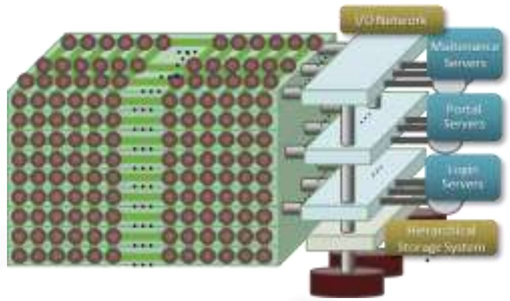


- **System Software**
  - Multi-Kernel: Linux with Light-weight Kernel
  - File I/O middleware for 3-level hierarchical storage system and application
  - Application-oriented file I/O middleware
  - MPI+OpenMP programming environment
  - Highly productive programing language and libraries



What we have done

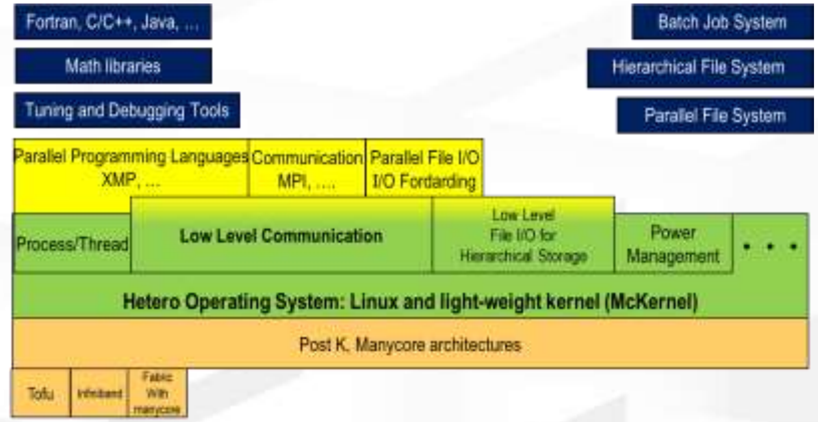
- **Hardware**
  - Instruction set architecture



Continue to design

- Node architecture
- System configuration
- Storage system

- **Software**
  - OS functional design
  - Communication functional design
  - File I/O functional design
  - Programming languages
  - Mathematical libraries



Instruction Set Architecture

- **ARM V8 HPC Extension**
  - Fujitsu is a lead partner of ARM HPC extension development
  - Detailed features were announced at Hot Chips 28 - 2016

    http://www.hotchips.org/program/

    Mon 8/22 Day1 9:45AM GPUs & HPCs

    ARMv8-A Next Generation Vector Architecture for HPC

    SVE (Scalable Vector Extension)

- **Fujitsu's inheritances**
  - FMA
  - Math acceleration primitives
  - Inter core barrier
  - Sector cache
  - Hardware prefetch assist

Post-K: Fujitsu HPC CPU to Support ARM v8 ARM FUJITSU

Post-K fully utilizes Fujitsu's proven supercomputer microarchitecture

Fujitsu, as a "lead partner" of ARM HPC extension development, is working to realize an 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	✔Enhanced	✔	✔
Inter core barrier	✔	✔	✔	✔
Sector cache	✔Enhanced	✔Enhanced	✔	✔
Hardware prefetch assist	✔Enhanced	✔Enhanced	✔	✔
Tofu interconnect	✔Integrated	✔Integrated	✔	✔

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

2

Copyright 2016 FUJITSU LIMITED

OS Kernel

- **Requirements of OS Kernel targeting high-end HPC**
  - Noiseless execution environment for bulk-synchronous applications
  - Ability to easily adapt to new/future system architectures
    - E.g.: manycore CPUs, heterogenous core architectures, deep memory hierarchy, etc.
      - New process/thread management, memory management, …
  - Ability to adapt t
    - Big-Data, in-s
      - Support data
      - Optimize data movement

Our Approach:
Linux with Light-Weight Kernel

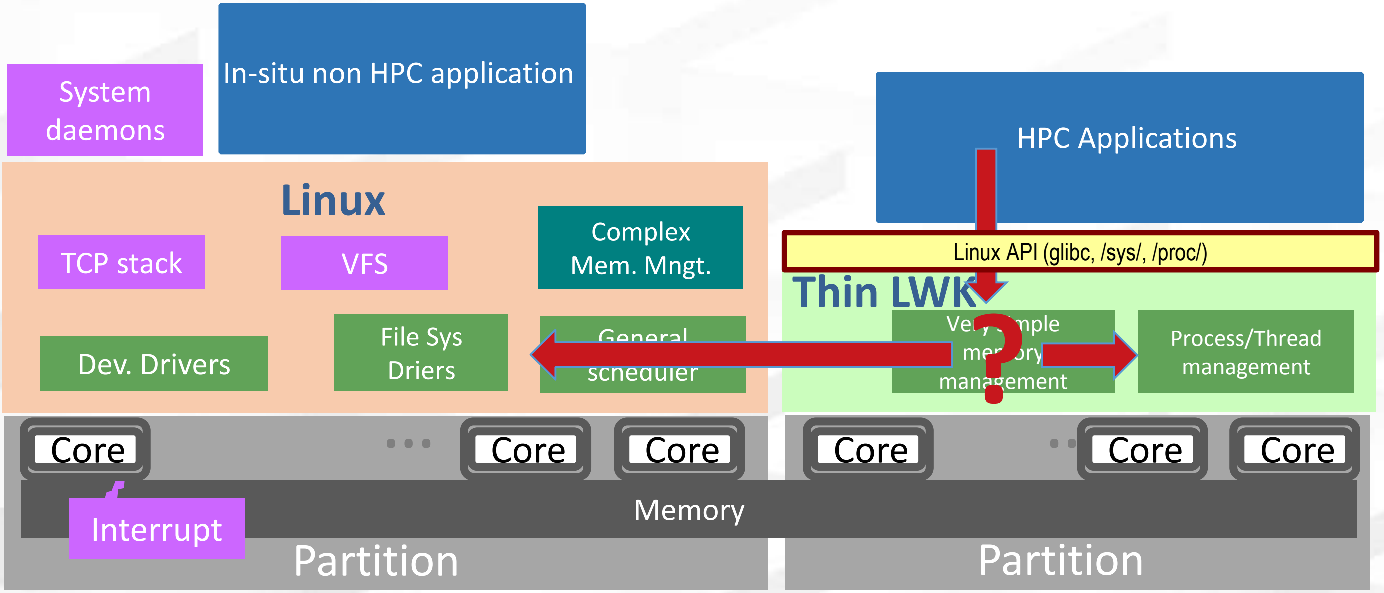
	Approach	Pros.	Cons.
Full-Weight Kernel (FWK) e.g. Linux	Disabling, removing, tuning, reimplementation, and adding new features	Large community support results in rapid new hardware adaptation	• Hard to implement a new feature if the original mechanism is conflicted with the new feature • Hard to follow the latest kernel distribution due to local large modifications
Light-Weight Kernel (LWK)	Implementation from scratch and adding new features	Easy to extend it because of small in terms of logic and code size	• Applications, running on FWK, cannot run always in LWK • Small community maintenance limits rapid growth • Lack of device drivers

McKernel developed at RIKEN

- **Partition resources (CPU cores, memory)**
- **Full Linux kernel on some cores**
  - System daemons and in-situ non HPC applications
  - Device drivers
- **Light-weight kernel(LWK), McKernel on other cores**
  - HPC applications

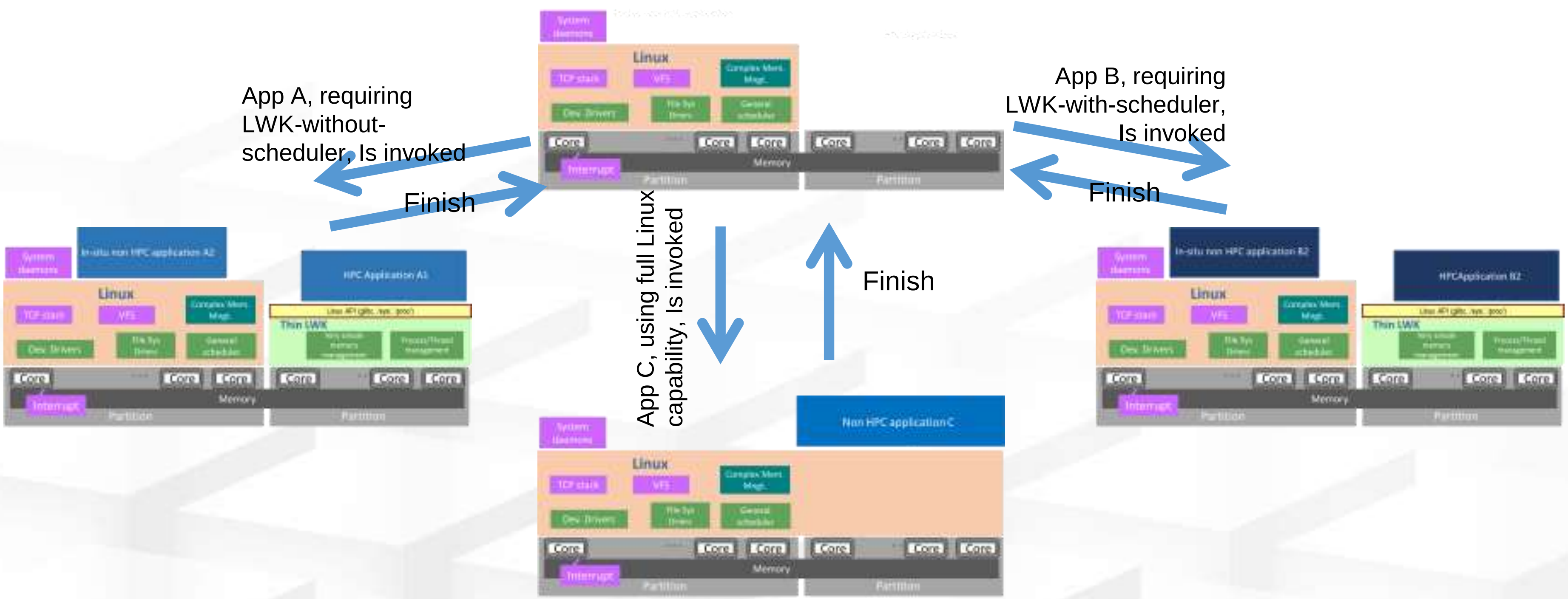
- **McKernel is loadable module of Linux**
- **McKernel supports Linux API**
- **McKernel runs on**
  - Intel Xeon and Xeon phi
  - Fujitsu FX10

McKernel is deployed to the Oakforest-PACS supercomputer, 25 PF in peak, at JCAHPC organized by U. of Tsukuba and U. of Tokyo



OS: McKernel

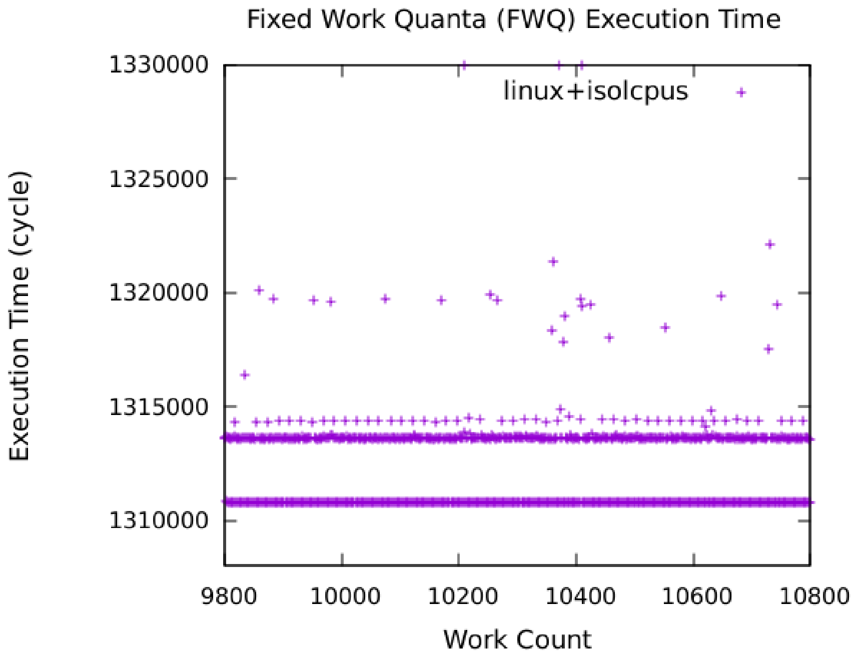
- Linux Kernel＋Loadable LWK, McKernel
  - Linux Kernel is resident, and daemons for job scheduler and etc. run on Linux
  - McKernel is dynamically reloaded (rebooted) for each application



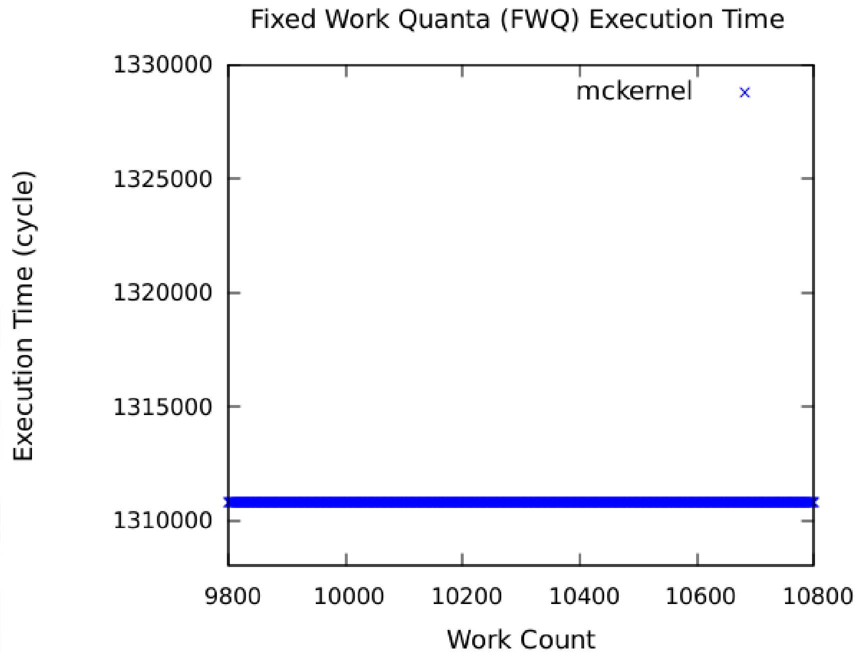
OS: McKernel

Results of FWQ (Fixed Work Quanta)

Linux with isolcpus



McKernel



https://asc.llnl.gov/sequoia/benchmarks

Concluding Remarks

- **Post K's CPU is based on ARM V8 with HPC extension**
- **The usability will be improved than the K computer by changing architecture**
  - More wide-range community support

- **The system software stack for Post K is being designed and implemented with the leverage of international collaborations**
  - The software stack developed at RIKEN is Open source
  - It also runs on Intel Xeon and Xeon phi
  - RIKEN would like to contribute to OpenHPC