(Unorthodox?) Use of Non-Volatile Memory Technology for Green Computing

April 20, 2010

Sam H. Noh

NECSST Lab.
Hongik University

(Unorthodox?) Use of Non-Volatile Memory Technology for Green Computing

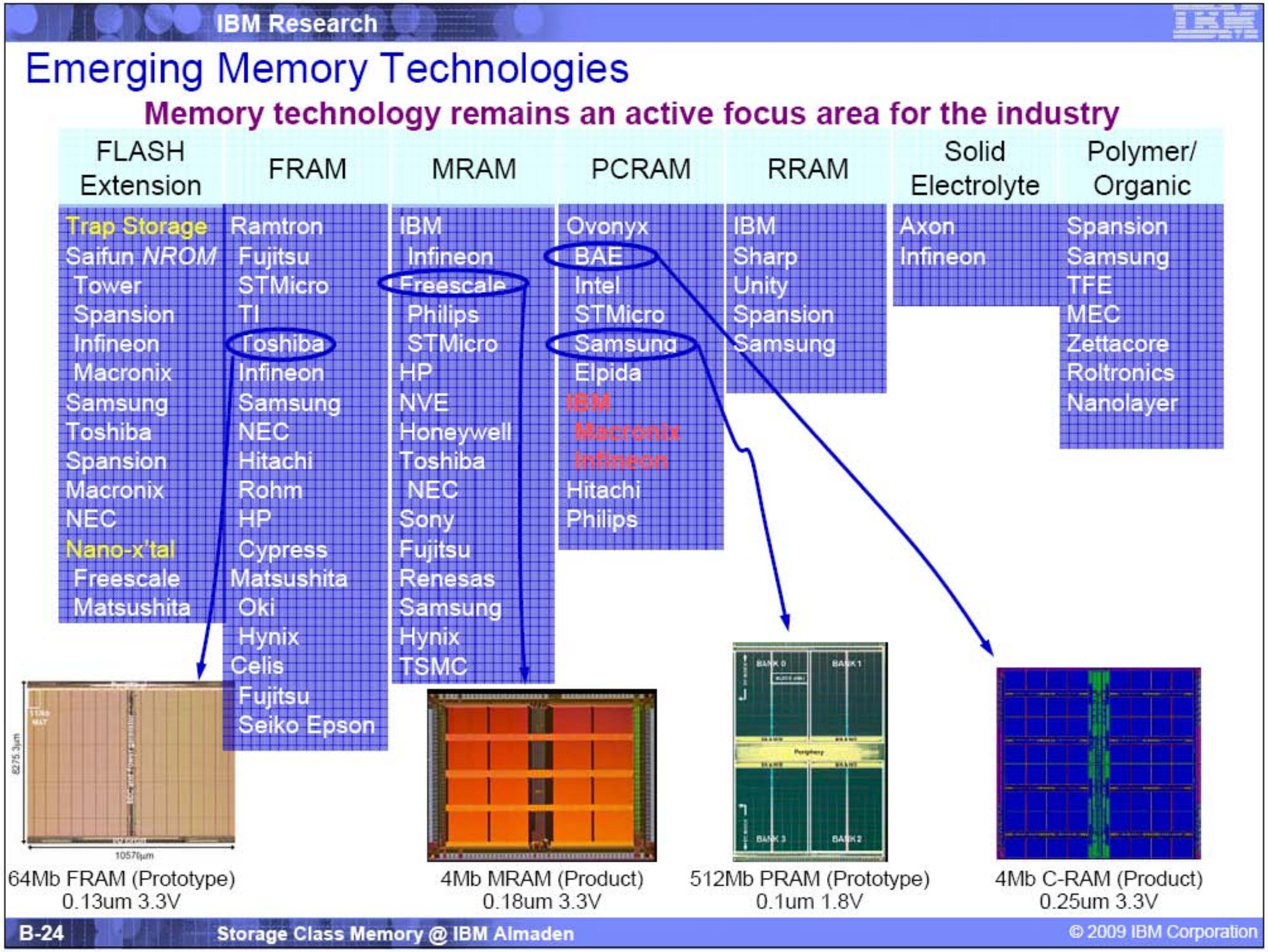
Part I: Non-Volatile Memory Technology

Part II: Green Computing

Storage Class Memory (SCM)

- **Definition**
  - A class of data storage/memory device
    - FAST’09 tutorial
      - Freitas, Wilcke, Kurdi, and Burr
- **Characteristics**
  - Nonvolatile
    - Storage property such as HDD and Flash
  - Random byte addressable
    - RAM property such as DRAM and SRAM

SCM development trend (FAST'09 Tutorial)



IBM Research

Emerging Memory Technologies

Memory technology remains an active focus area for the industry

FLASH Extension	FRAM	MRAM	PCRAM	RRAM	Solid Electrolyte	Polymer/ Organic
Trap Storage	Ramtron	IBM	Ovonyx	IBM	Axon	Spansion
Saifun *NROM*	Fujitsu	Infineon	BAE	Sharp	Infineon	Samsung
Tower	STMicro	Freescale	Intel	Unity		TFE
Spansion	TI	Philips	STMicro	Spansion		MEC
Infineon	Toshiba	STMicro	Samsung	Samsung		Zettacore
Macronix	Infineon	HP	Elpida			Roltronics
Samsung	Samsung	NVE	IBM			Nanolayer
Toshiba	NEC	Honeywell	Macronix			
Spansion	Hitachi	Toshiba	Infineon			
Macronix	Rohm	NEC	Hitachi			
NEC	HP	Sony	Philips			
Nano-x'tal	Cypress	Fujitsu				
Freescale	Matsushita	Renesas				
Matsushita	Oki	Samsung				
	Hynix	Hynix				
	Celis	TSMC				
	Fujitsu					
	Seiko Epson					

64Mb FRAM (Prototype)
0.13um 3.3V

4Mb MRAM (Product)
0.18um 3.3V

512Mb PRAM (Prototype)
0.1um 1.8V

4Mb C-RAM (Product)
0.25um 3.3V

B-24 Storage Class Memory @ IBM Almaden

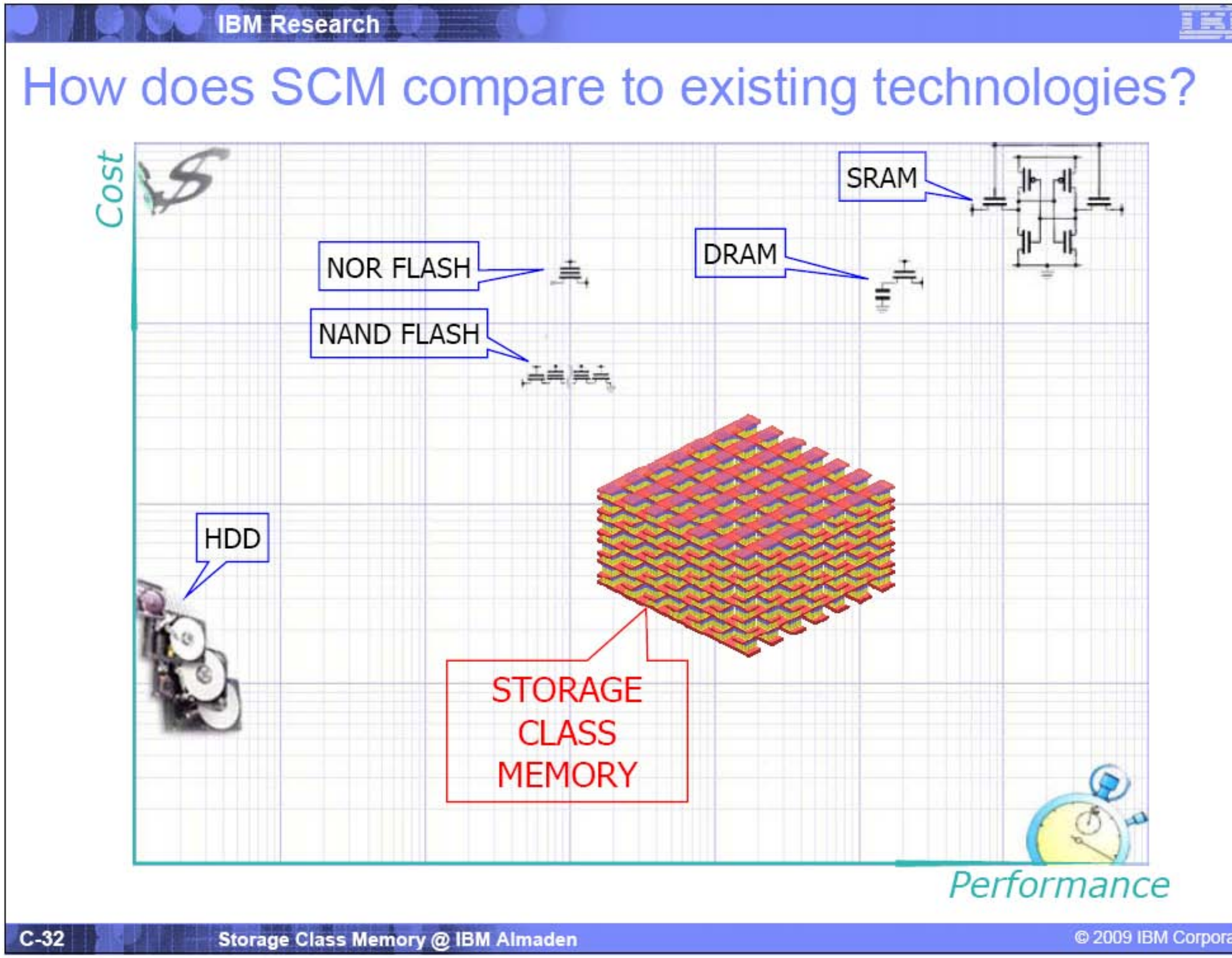
SCM perspectives

- **Current technology in the market**
  - Numonyx: 128Mb PCM
  - Ramtron: 4Mb FeRAM
  - Freescale: 4Mb MRAM

- **Development trend**
  - More than 30 major IT companies such as Intel, Samsung, Toshiba, etc.
  - Samsung : start production of 512Mb PCM
    - Press Release, Sept. 22, 2009
  - Toshiba: 128Mb FeRAM prototype
    - Press Release, Feb. 8, 2009
- **Perspectives (IBM Almaden Research Center)**
  - Widely deployed in data center systems by 2012
  - Fully replace HDD by 2020

SCM performance and cost

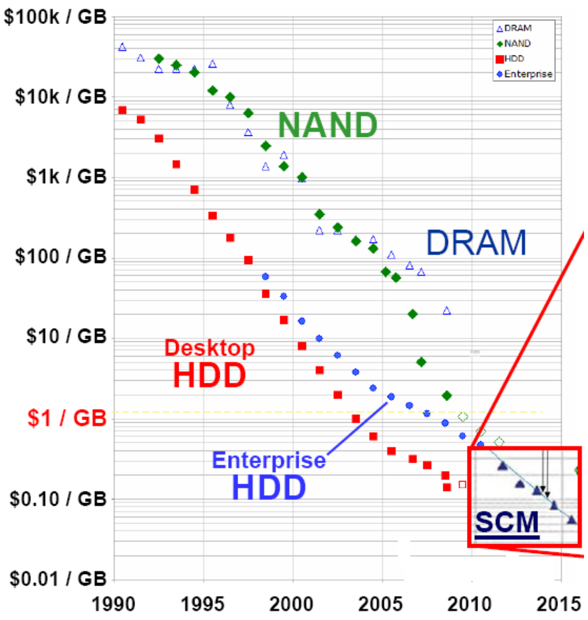
(FAST'09 Tutorial)



IBM Research

How does SCM compare to existing technologies?

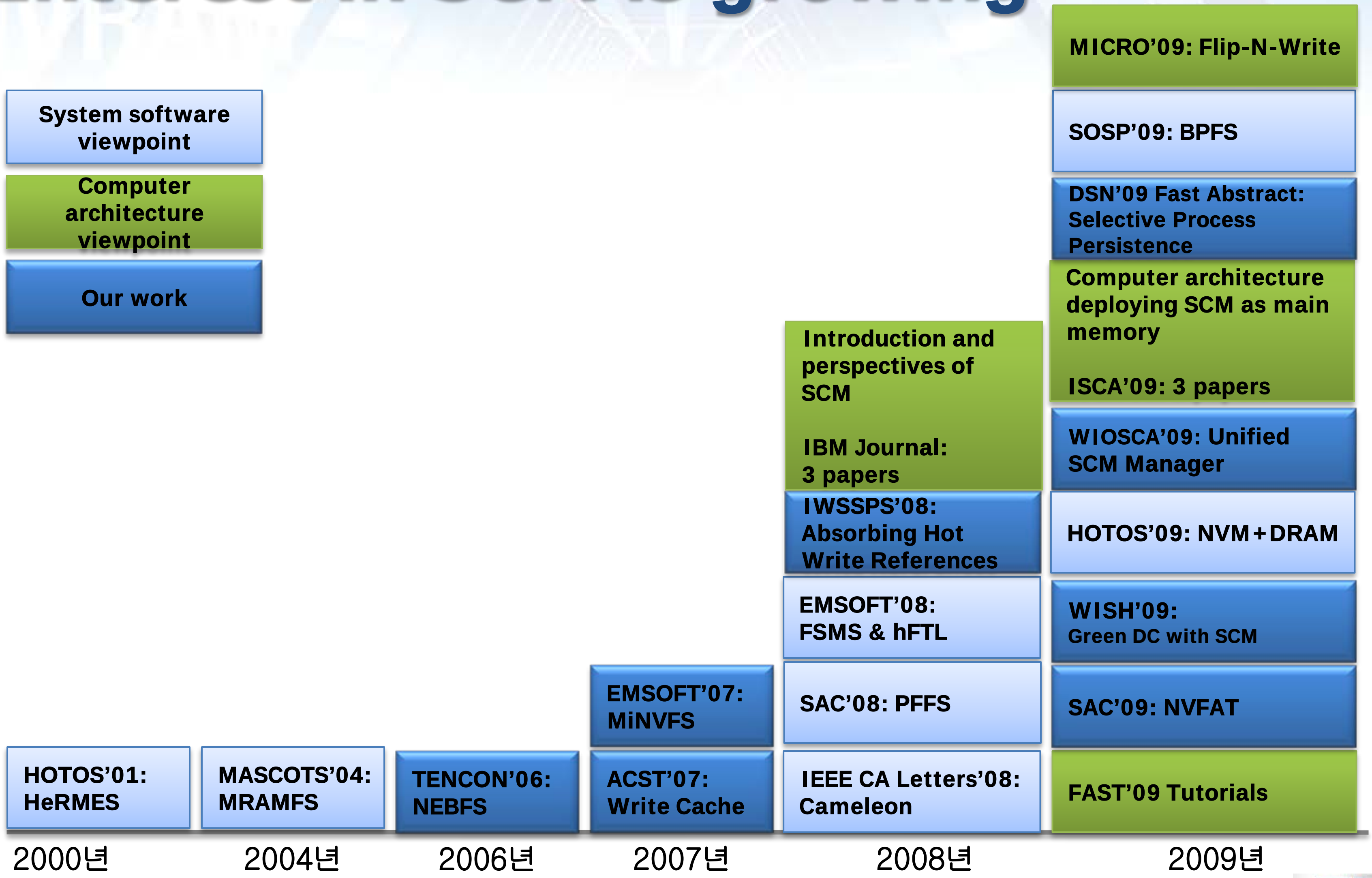
C-32 Storage Class Memory @ IBM Almaden © 2009 IBM Corporation



SCM in the market

		Flash memory		RAM	SCM		
		NAND	NOR	SDRAM	PCM	FeRAM	MRAM
Non-volatility		Yes	Yes	No	Yes	Yes	Yes
Access units	**Read / Write**	Page	Byte	Byte	Byte	Byte	Byte
	Erase	Block	Block	-	-	-	-
Access speed	**Read**	12 us	110 ns	~20-50 ns	>	55 ns	35 ns
	Write	200 us	80 us	~20-50 ns	>>	55 ns	35 ns
	Erase	2 ms	0.6 s	-	-	-	-
Endurance (# of writes)		10^5	10^5	10^{15}	10^8	10^{12}	10^{15}
Active Current		10~30 mA	14~25 mA	70~110 mA (+ Refresh: 160mA)	<	18 mA	55~155 mA
Standby Current		10~50 uA	5 uA	2~28 mA (+ Self-refresh: 1.5~3mA)	<<	5 uA	9~28 mA

Interest in SCM is growing



System software viewpoint

Computer architecture viewpoint

Our work

HOTOS'01: HeRMES

MASCOTS'04: MRAMFS

TENCON'06: NEBFS

EMSOFT'07: MiNVFS

ACST'07: Write Cache

Introduction and perspectives of SCM

IBM Journal: 3 papers

IWSSPS'08: Absorbing Hot Write References

EMSOFT'08: FSMS & hFTL

SAC'08: PFFS

IEEE CA Letters'08: Cameleon

MICRO'09: Flip-N-Write

SOSP'09: BPFS

DSN'09 Fast Abstract: Selective Process Persistence

Computer architecture deploying SCM as main memory

ISCA'09: 3 papers

WIOSCA'09: Unified SCM Manager

HOTOS'09: NVM+DRAM

WISH'09: Green DC with SCM

SAC'09: NVFAT

FAST'09 Tutorials

2000년

2004년

2006년

2007년

2008년

2009년

SCM Viewpoints

- **Storage**
  - Non-volatile characteristic
    - Storage Class Memory
  - Conventional storage devices are block devices
    - But, SCM is byte addressable
- **Memory**
  - Byte addressable characteristic
  - Conventional memory is volatile
    - But, SCM is non-volatile ← our emphasis

(Unorthodox?) Use of Non-Volatile Memory Technology for Green Computing

Part I: Non-Volatile Memory Technology

Part II: Green Computing

Green…

- Technological buzz word
- Socially correct
- Politically…
- Various approaches

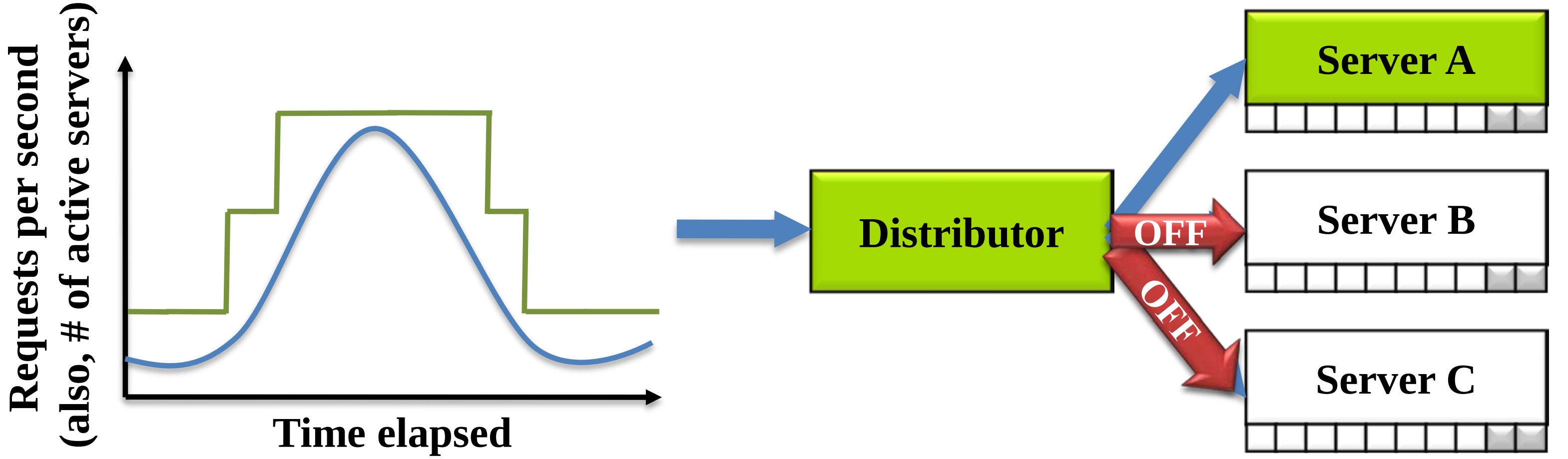
Not just lights…idle…shut it down

...even data centers

- **Tremendous energy consumed by data centers**
  - Billions of kWh consumed by U.S. data centers
  - Billions of dollars per year for maintaining data centers
- **Focus on idleness of systems**
  - Average 20~30% system utilization in data centers
  - Idle servers consume 60% of peak power

Ideal data center

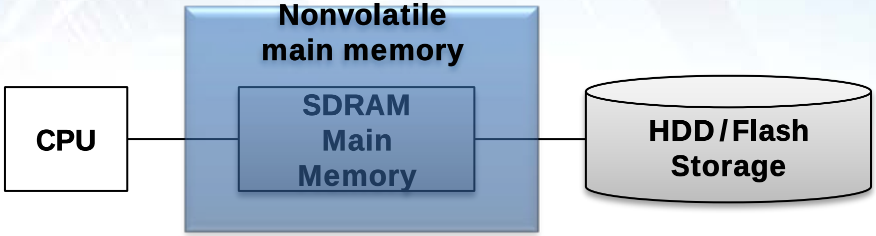
- **Turn off idle servers**
  - turn on only to satisfies QoS



Challenges in realizing the ideal DC

- <u>Systems not responsive enough for practical use</u>
- Compromises encapsulation
- Very short bursts of idleness
- Need for replication
- Inactive-to-active energy penalty

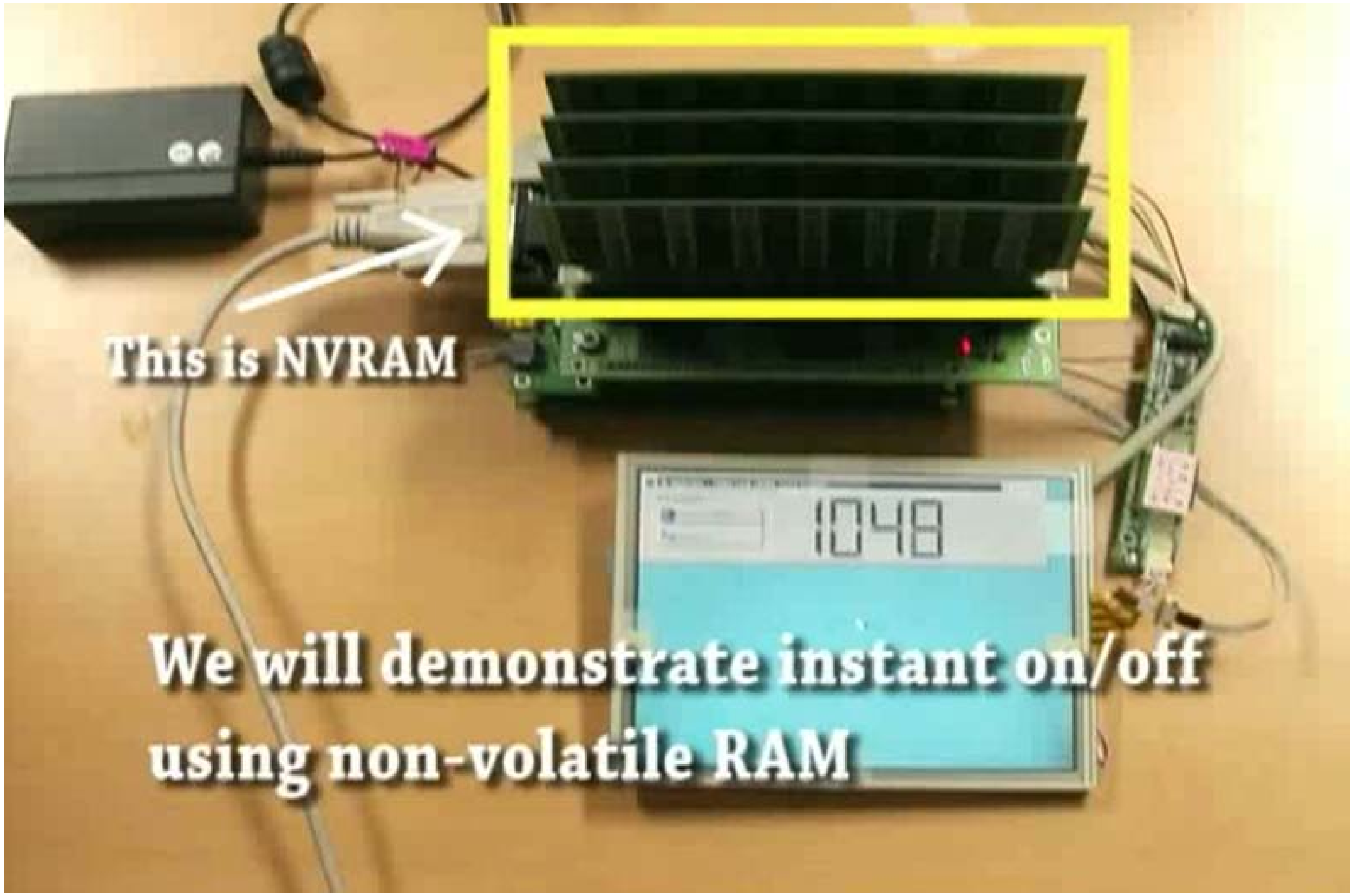
Use SCM to realize ideal data center



- **SCM as main memory**
  - plus, non-volatile
    - Unorthodox?
    - Process state is preserved with power off
- **Notion of Persistent Computing**
  - "as if the system had always been on"
    - Despite power off period in between
- **Turn off when idle; turn back on when needed**
  - must be fast…but, how fast is fast enough?

SOONN (System On/Off iNstaNtly)

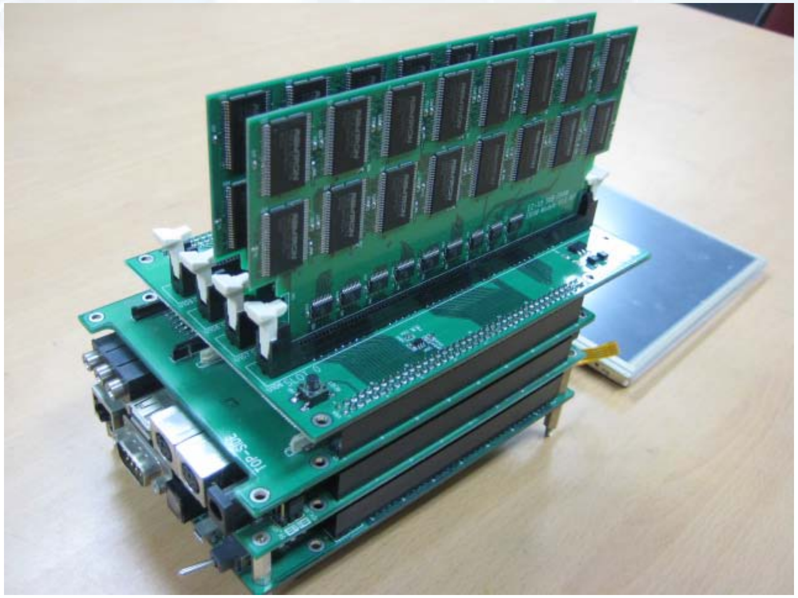
瞬 : Blink of an eye



We will demonstrate instant on/off using non-volatile RAM

Experimental platform

Component	Specification
CPU	PXA255 400MHz
SDRAM	32MB X 2ea
NAND Flash	32MB X 2ea
Boot Flash	512KB
SCM(FeRAM)	0.5MB X (up to) 128ea
Peripherals	UART, LCD, JTAG, …

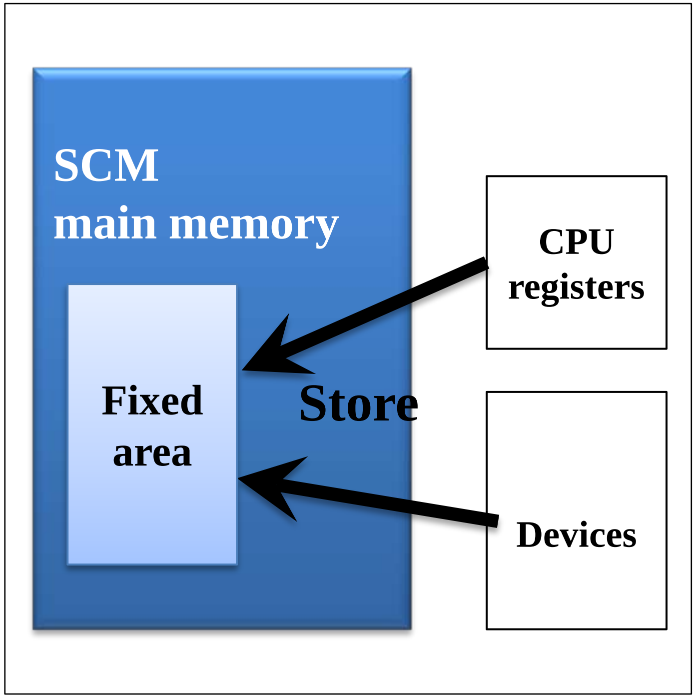


- Linux 2.6.21

SOONN (System On/Off iNstaNtly)
瞬 : Blink of an eye

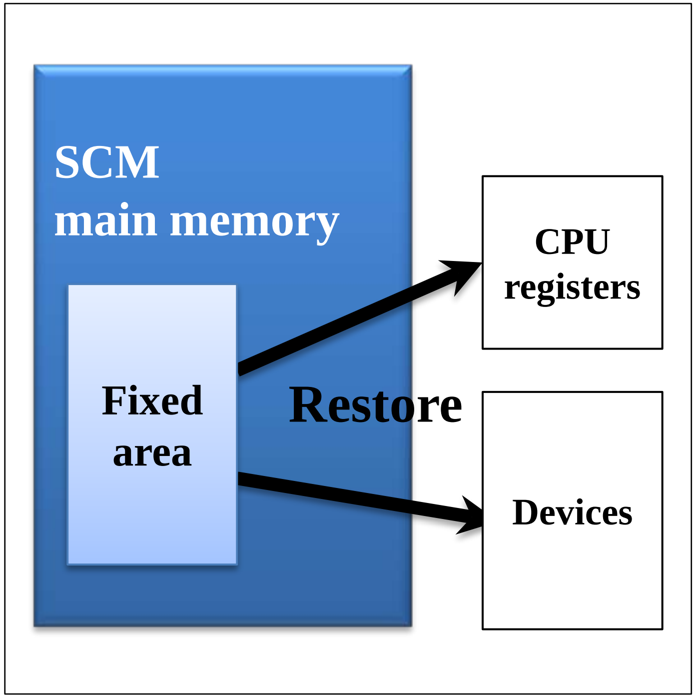
- **Persistent computing**

When turned off



SCM
main memory
Fixed
area
Store
CPU
registers
Devices

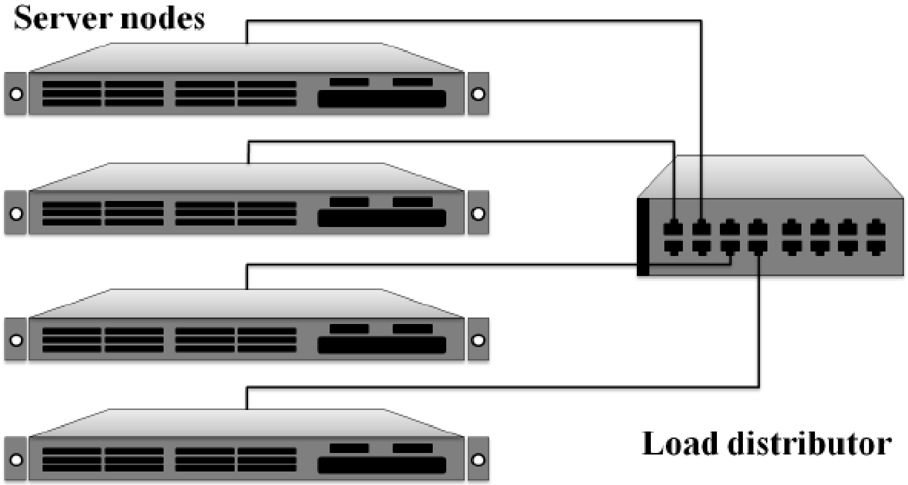
When turned on



SCM
main memory
Fixed
area
Restore
CPU
registers
Devices

Implementation of ZEUS

- **ZEUS: Zero Energy for Unused Servers**
  - Server: SOONN system
  - Distributor: Linux Virtual Server project based IPVS software



Experimental environments

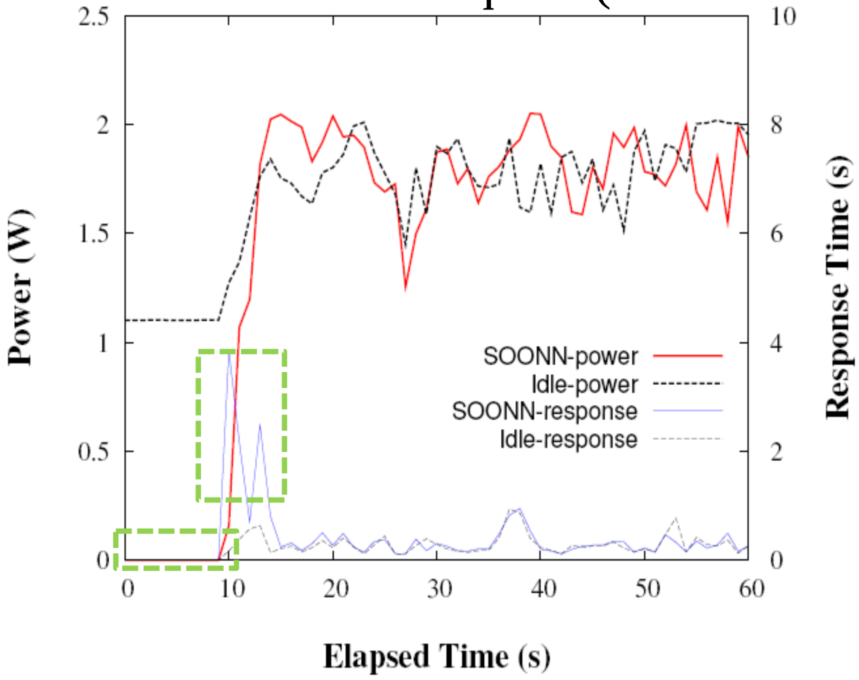
- **Hardware setup**

	CPU	Main memory	NIC	Notes
4 Servers	ARM PXA255 XScale 400MHz	32MB FeRAM	10Mbps	Apache Web server
1 Distributor	x86 Intel 2GHz	2GB DDR3 SDRAM	100Mbps	
1 Client	x86 Intel 2.4GHz	2GB DDR3 SDRAM	100Mbps	Proxycizer trace replayer

- **Server configuration**
  - Servers only execute a CPU-bound CGI document that consumes 0.05 seconds

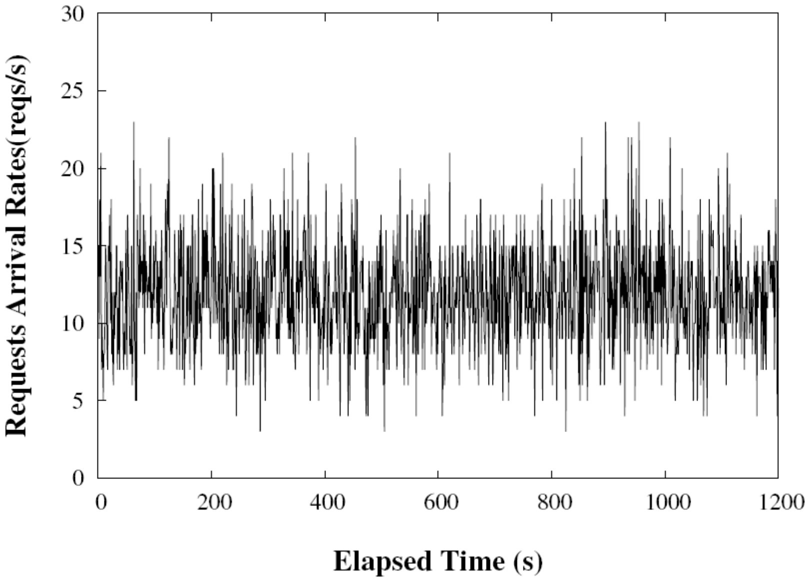
Single SOONN server

- **Power and response time**
  - Zero power for SOONN server at idle
  - Send requests at maximum rate from time 10
    - Roughly 4 seconds for the first request (1.17 seconds for “ready to respond”)



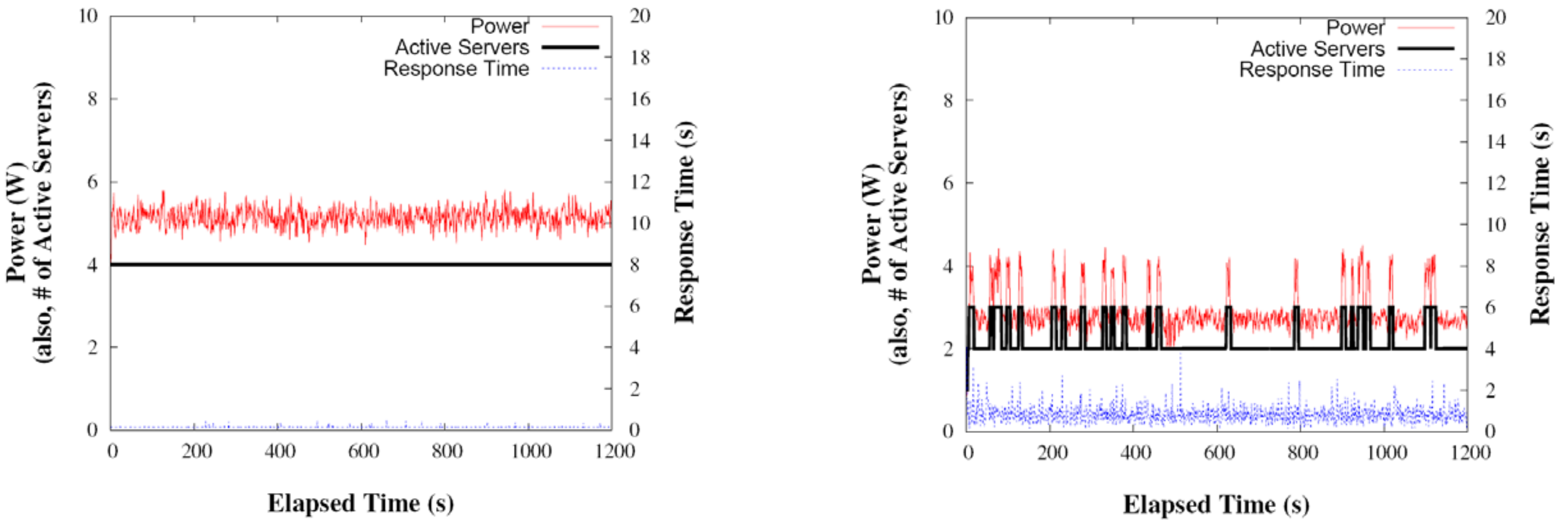
Synthetic workload

- **Requests sent with Poisson distribution**
  - 14,400 requests during a 20 minute long experiment
    - Average of 12 requests per second



BUT: Biasing with Upper Threshold

- **Significant energy savings**
  - ZEUS is responsive
  - Performance degradation from user response time viewpoint

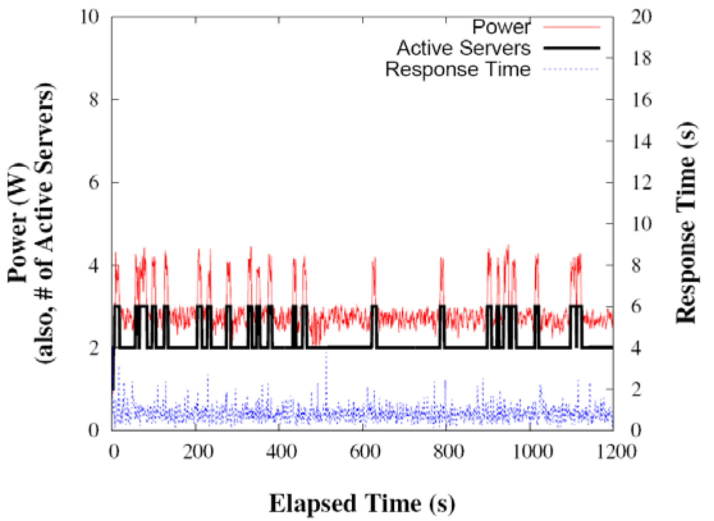


(a) BASE: Conventional data center

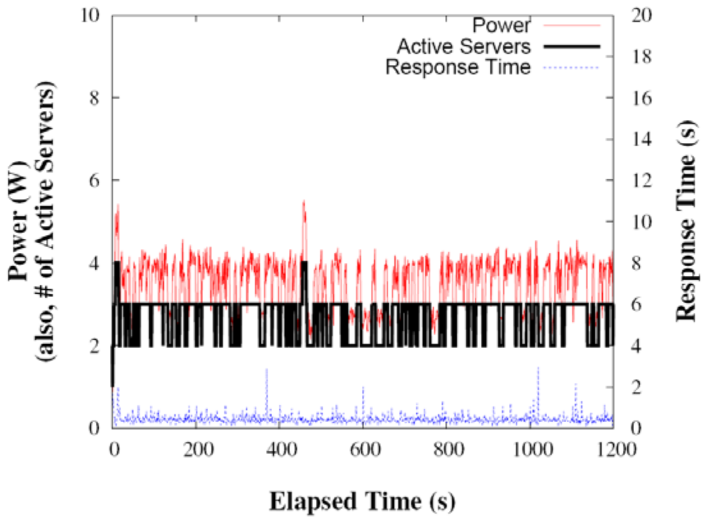
(b) ZEUS: BUT Threshold 10 (BUT10)

Effect of BUT threshold

- **Lowering upper threshold turns systems on earlier**
  - Increase power usage, but improve response time



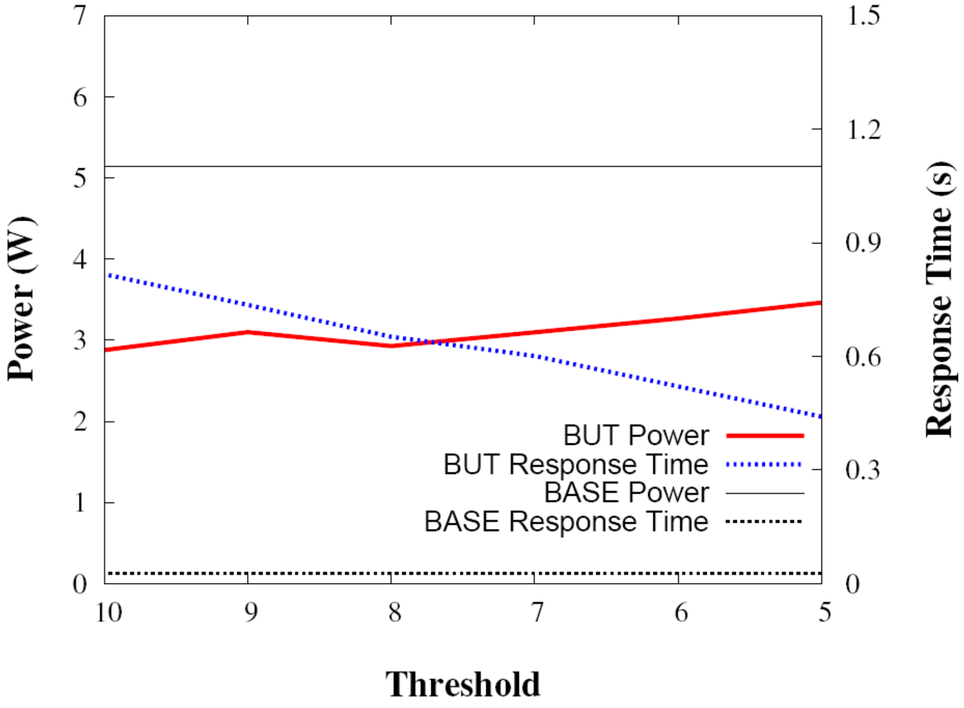
(b) ZEUS: BUT Threshold 10 (BUT10)



(c) ZEUS: BUT Threshold 5 (BUT5)

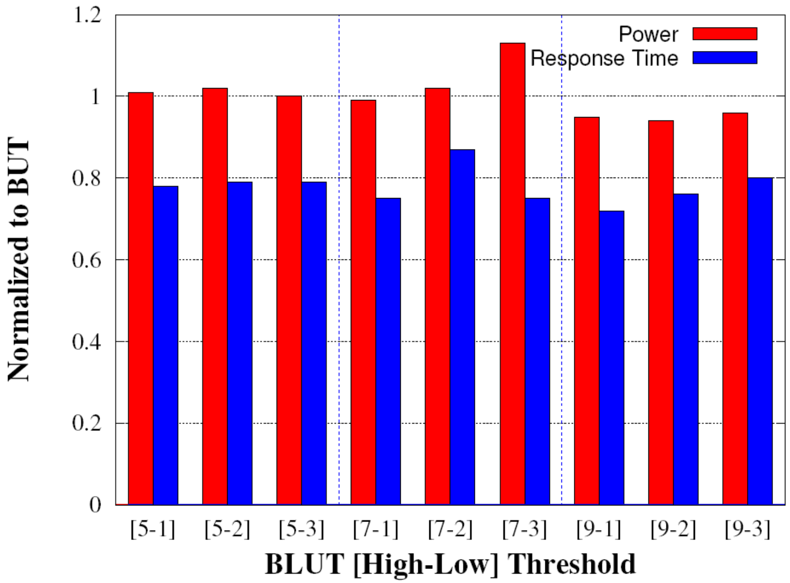
Trade-off: power vs. response time

- **Power savings with configurable service of quality**



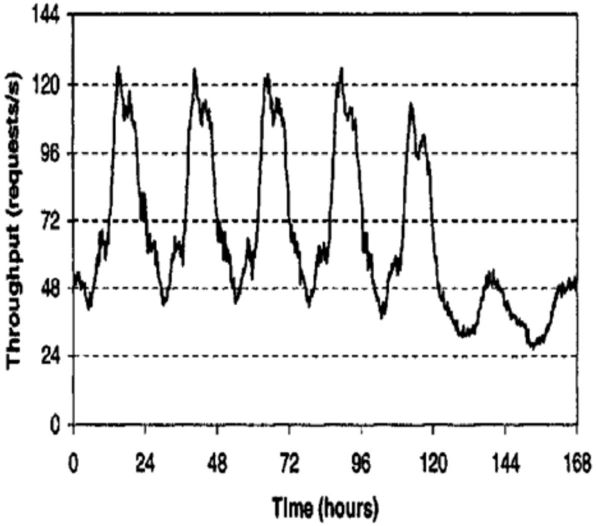
BLUT: BUT with Low Threshold

- **BUT extension to adopt load balancing feature**
  - Decide the number of servers currently needed
  - Balance requests among turned on servers

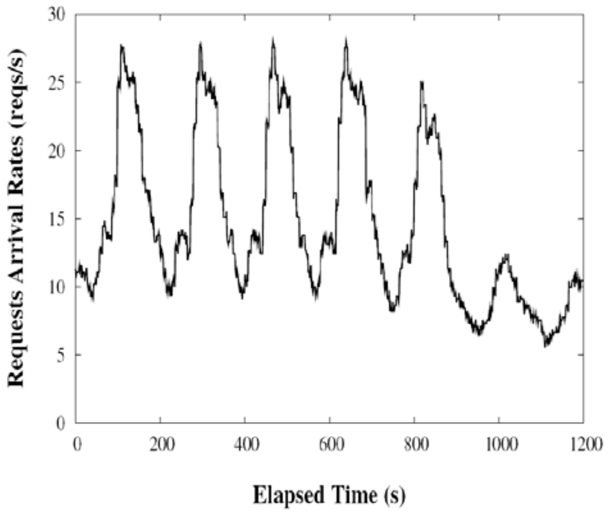


Realistic workload #1

- **“Real IBM trace”-like**
  - Maximum 28 requests/sec (scale down of 40)
  - 20 minutes long (scale down of 500)



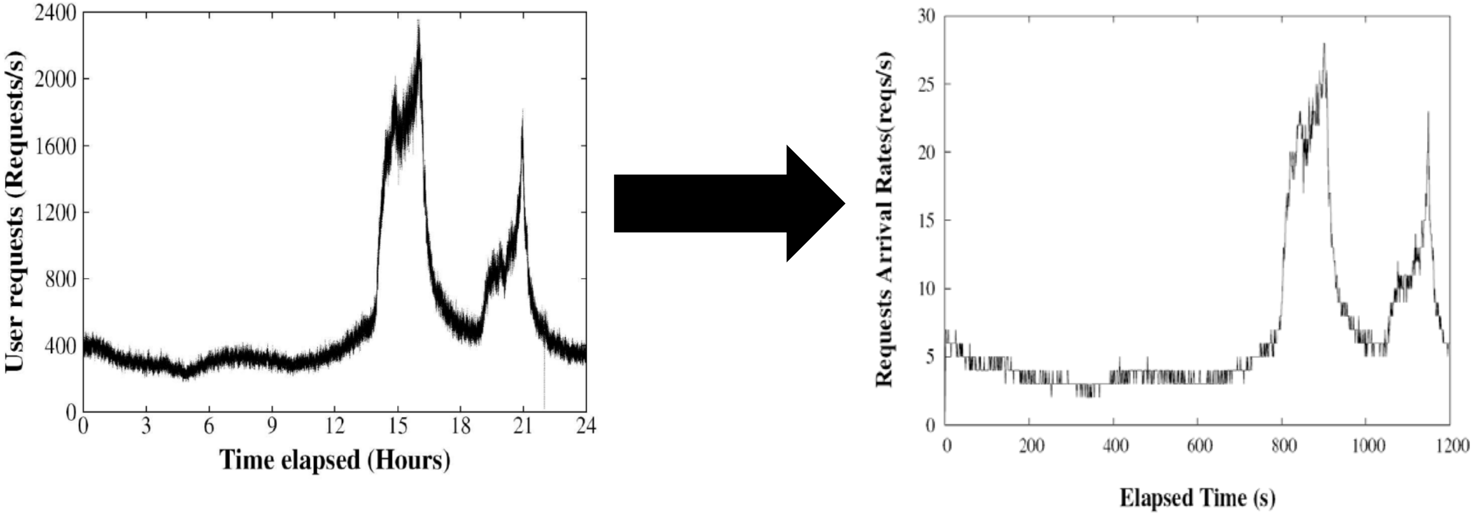
IBM Trace



IBM-like Trace

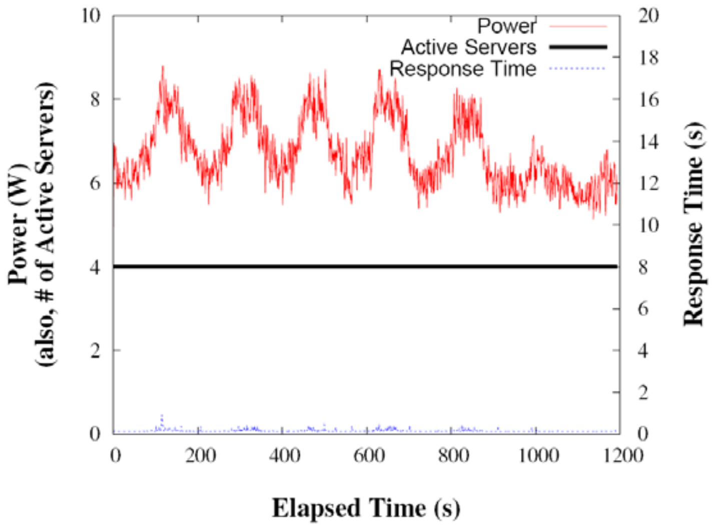
Realistic workload #2

- **“WorldCup98 trace”-like**
  - A day long trace in the 1998 World Cup
    - June 24, 1998

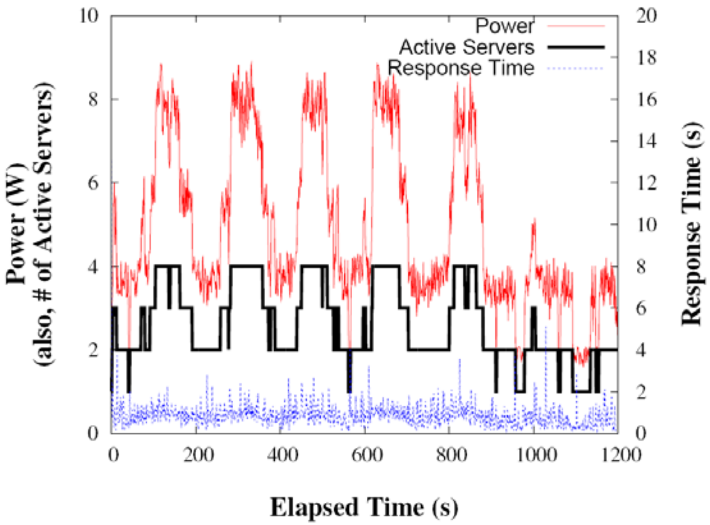


Adaptiveness of ZEUS (1/2)

- **“Real IBM trace”-like**
  - Provide service with adaptive # of active servers
    - Highly fluctuating user request patterns



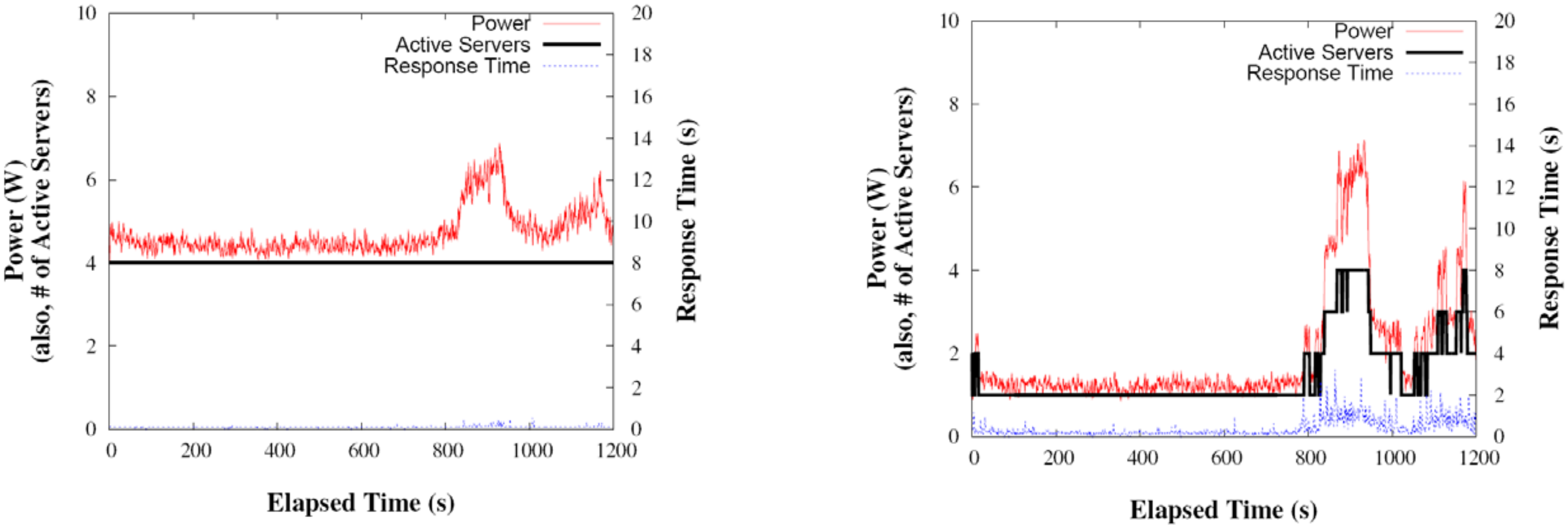
(a) BASE: Conventional data center



(b) ZEUS: BUT Threshold 10 (BUT10)

Adaptiveness of ZEUS (2/2)

- **“WorldCup98 trace”-like**
  - Energy consumed proportional to system utilization



(a) BASE: Conventional data center

(b) ZEUS: BUT Threshold 10 (BUT10)

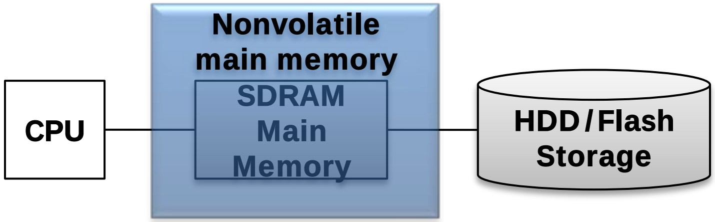
Green data center summary

- **First empirical proof-of-concept study using SCM**
- **SOONN**
  - Swift on/off system
- **ZEUS**
  - Data center that reacts nimbly to workload dynamics

- **However,…**

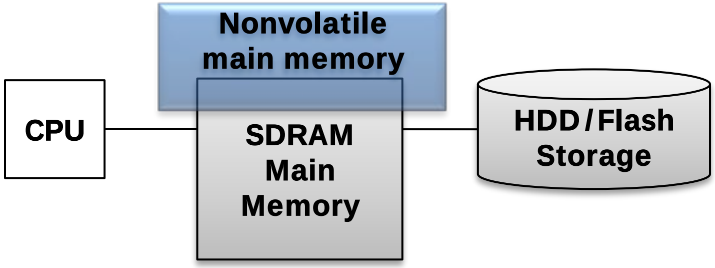
Is this realistic?

- **SCM only main memory**
  - Instant on/off done really fast
  - Persistent computing
  - But, for the immediate future
    - a hefty price tag
    - unfriendly performance numbers



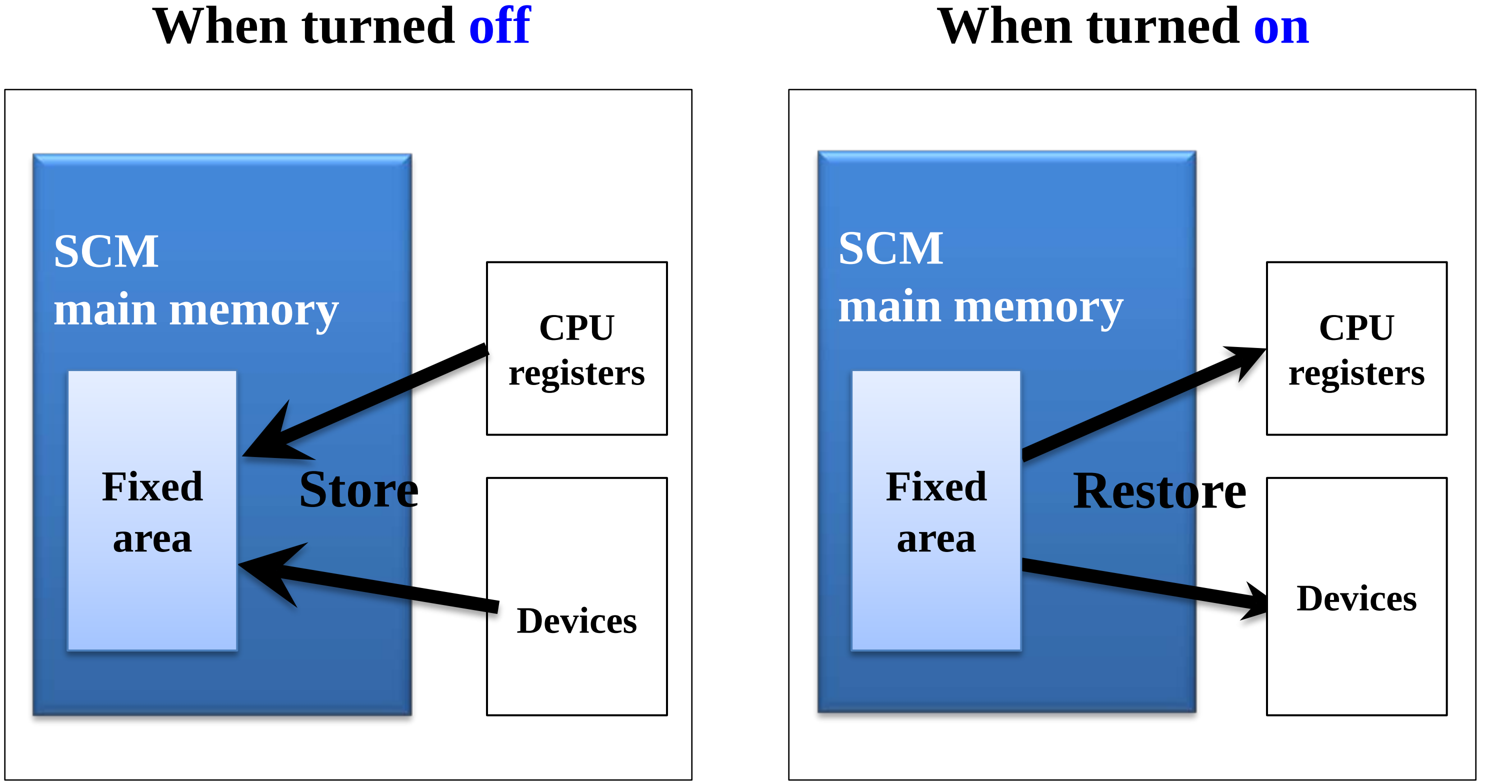
A more feasible solution?

- **SCM + DRAM main memory**
  - Still, provide instant on/off
  - Persistent computing, but only partially
    - lose some processes, but by choice

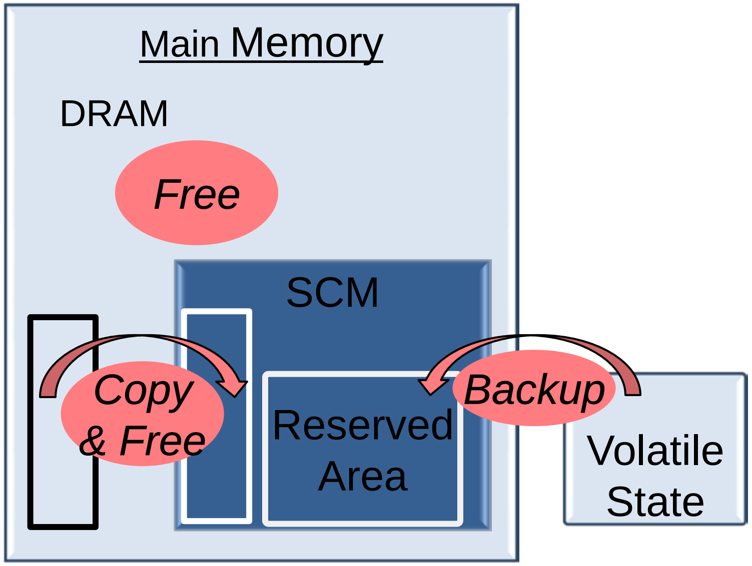


Recall SOONN (System On/Off iNstaNtly)

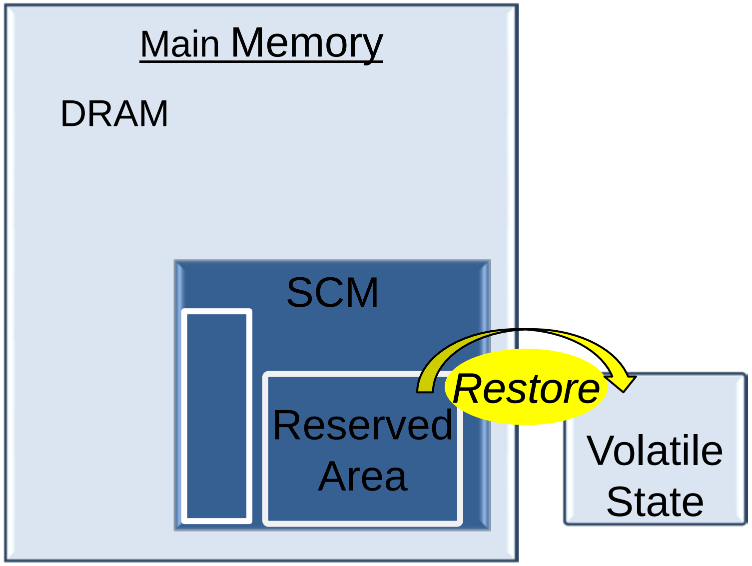
- **Persistent computing**



Instant off/on with SCM+DRAM



Instant Off



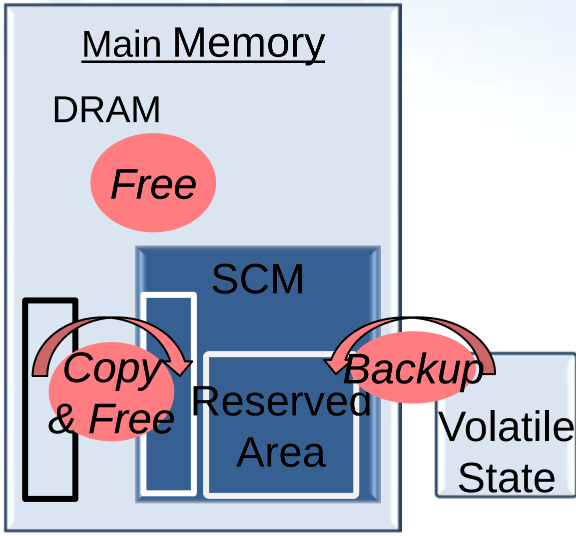
Instant On

What tasks are kept in SCM?

Task Class			Necessity	Choice
Kernel Thread			Mandatory	O
User Process	Spawned During Boot	System Request	Mandatory	O
		User Request	Optional	O
	Spawned After Boot		Optional	X

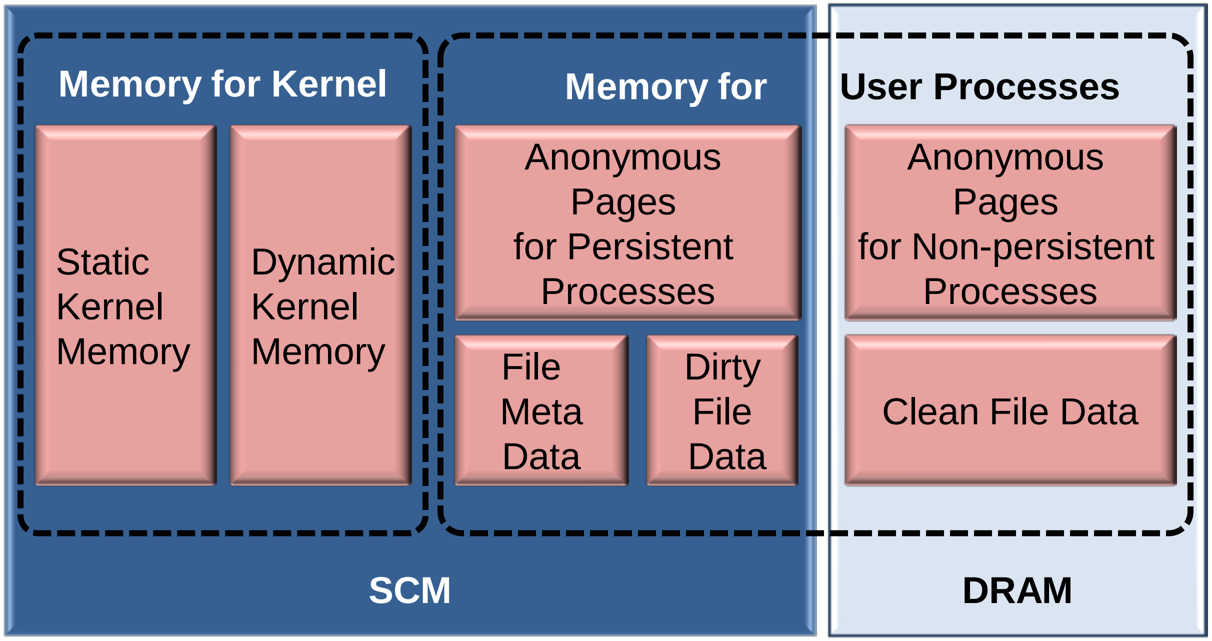
- **Persistent process**
  - processes persistent after instant off/on sequence
- **Non-persistent process**
  - processes unavailable after instant off/on sequence

Effect of losing non-persistent processes



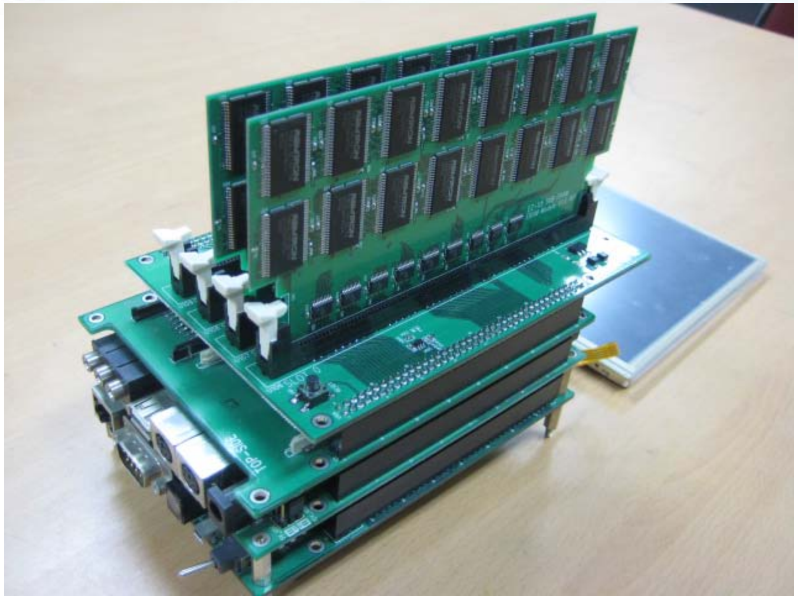
- **Negative**
  - Lose part of previous existing state
- **Positive**
  - Reduced instant on/off time
    - Decrease housekeeping overhead to turn off
  - Reduced SCM capacity
    - Resulting in lowered cost
  - Quick reclamation of free DRAM pages
    - Pseudo-rebooting
    - Start again from clean slate

Memory Objects in SCM vs DRAM



Implementation Platform

Component	Specification
CPU	PXA255 400MHz
SDRAM	32MB X 2ea
NAND Flash	32MB X 2ea
Boot Flash	512KB
SCM(FeRAM)	0.5MB X (up to) 128ea
Peripherals	UART, LCD, JTAG, …



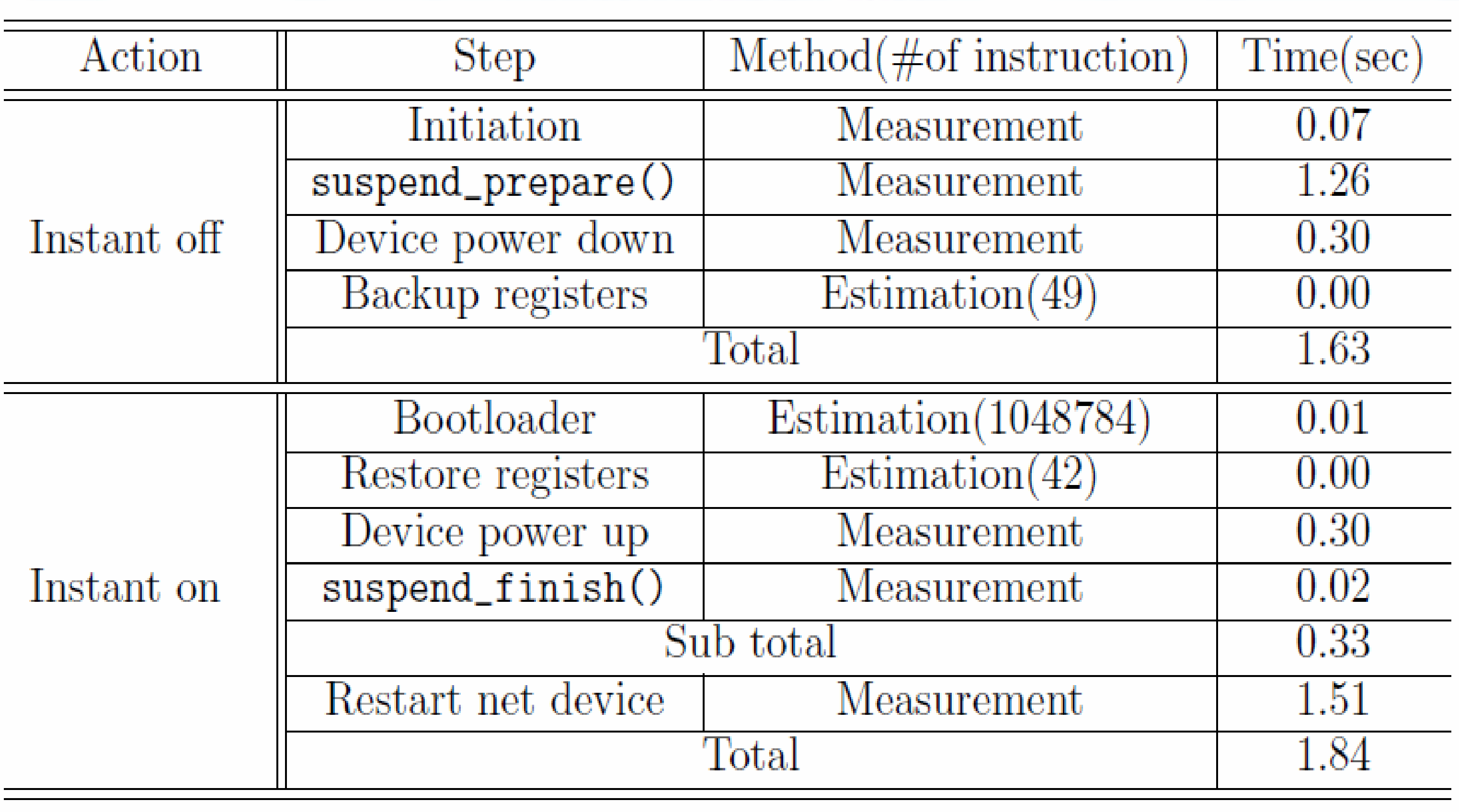
- **Linux 2.6.21**

Experimental Measurements

- **Instant off/on sequence**
  - Measurable steps
    - user space: **`gettimeogday()`**
    - kernel space: **`sys_gettimeofday()`**
  - Use estimated value when steps cannot be measured

$$\boxed{\frac{1}{400MHz} \times \text{Number of Instructions} \times \text{Cycles per Instruction}}$$

Experimental Measurements



Action	Step	Method(#of instruction)	Time(sec)
Instant off	Initiation	Measurement	0.07
	`suspend_prepare()`	Measurement	1.26
	Device power down	Measurement	0.30
	Backup registers	Estimation(49)	0.00
	Total		1.63
Instant on	Bootloader	Estimation(1048784)	0.01
	Restore registers	Estimation(42)	0.00
	Device power up	Measurement	0.30
	`suspend_finish()`	Measurement	0.02
	Sub total		0.33
	Restart net device	Measurement	1.51
	Total		1.84

Summary

- **SCM overview**
- **Green computing using SCM**
  - Novel notion of persistent computing
  - SOONN and ZEUS as persistent computing application
    - Swift on/off system
    - Data center reacts nimbly to workload
  - First empirical proof-of-concept study using SCM
  - Hybrid SCM + DRAM configuration
- **Future work**

Future work

- **Short/mid term**
  - Dependability
  - Transactions
  - Apply SOONN to realistic applications
  - Develop real ZEUS

- **Long term**
  - All other fields of Computer Science?

Conclusion

- **SCM: Interesting?**
- **SCM: Simply new storage?**
- **SCM: Simply a replacement of RAM?**
- **SCM: Revolutionary technology?**
  - Implication on various computing fields?
  - Architecture?
  - Operating systems?
  - Database?
  - A whole new field of applications?

Q/A

Thank you!!

http://next.hongik.ac.kr