RUTGERS
THE STATE UNIVERSITY OF NEW JERSEY

Rutgers Discovery Informatics Institute (RDI²)
New Jersey's Center for Advanced Computation

Staying Green at the Extreme : Exploring Energy Challenges and Tradeoffs for Science Workflows at Extreme Scales

Ivan Rodero, Marc Gamell, **Manish Parashar**
Director, Rutgers Discovery Informatics Institute (RDI²)
Professor, Department of Electrical & Computer Engineering

Rutgers Discovery Informatics Institute

The Applied Software Systems Laboratory

CAC

RUTGERS

RDI² Rutgers Discovery Informatics Institute

Abstract

- As scientific applications target extreme scales, energy-related challenges are becoming dominating concerns. As a result, it is critical to explore emerging architectures (e.g., with multiple cores and deep memory hierarchies) and applications (e.g., coupled simulation workflows) from an energy perspective and investigate associated overheads and tradeoffs. For example, energy/power-efficiency have to be addressed in combination with quality of solution, performance and reliability, and other objectives, and achieving the desired levels of reduction in power consumptions requires a comprehensive cross-layer and application-aware strategy. In this talk I will explore these issues and will describe recent related research efforts at the Rutgers Discovery Informatics Institute (RDI2).

RUTGERS RDI² Rutgers Discovery Informatics Institute

Outline

- Power/Energy Challenges at Extreme Scale
  - Research landscape
- The GreenHPC Project @ Rutgers
  - Application-aware power management
  - Programming support for GreenHPC
  - Power/Energy-aware data management
- Concluding Remarks and Open Challenges

RUTGERS RDI² Rutgers Discovery Informatics Institute

Rutgers Discovery Informatics Institute (RDI²)

Driving Innovation through Advanced Computing

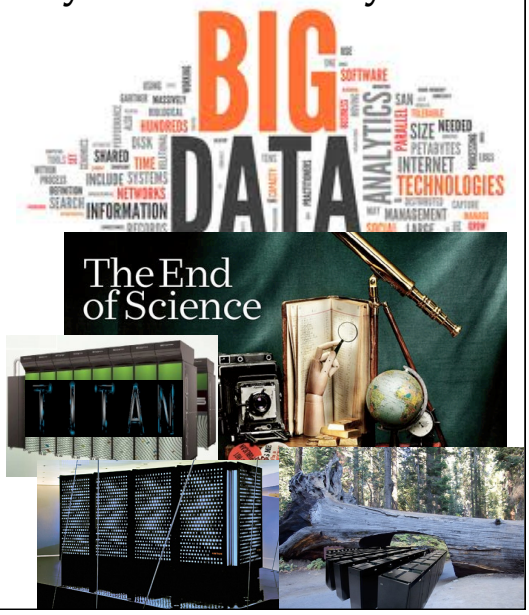
- Established in March 2012 as New Jersey's Center for Advanced Computation with an overarching goal to create a world-class institute focused on computational and data sciences
- Fundamentally integrate research, education, ACI and industry partnerships to address core CDS&E / BigData challenges
- Broaden industry and academic access to state-of-the-art computing technology and expertise
  - NSF Cloud and Autonomic Computing IUCRC
- Integrate multidisciplinary research with ACI and industry partnerships

http://rdi2.rutgers.edu

RUTGERS RDI² Rutgers Discovery Informatics Institute

Modern Science & Society Transformed by Compute & Data

- New Paradigms & Practices
  - End-to-end: Seamless access, aggregation, interactions
  - Data-driven, Data/Compute-intensive; Age of Digital Observation
  - Integrative, multi-scale, online
- Multi-disciplinary collaborations
  - Individuals, groups, teams, communities, networks
  - New global science culture
- Unprecedented opportunities, challenges



RUTGERS RDI² Rutgers Discovery Informatics Institute

Advanced Computing Infrastructure

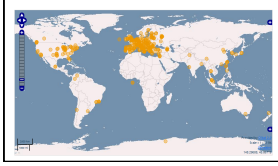
- Large scale, distributed, heterogeneous, multicore/manycore, accelerators, deep storage hierarchies, experimental systems

Titan - Cray XK7
- 20 PF / 300 K cores
- 16-core CPU + GPU
- Gemini 3D torus
- 600 TB memory

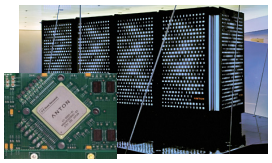
XSEDE
- Worlds Largest Grid
- 11 Resource Providers

Sequoia – IBM BG/Q
- 20 PF / 1.5 M cores
- 18-core processor
- 5D torus
- 1.5PB memory

Modern Datacenters
- 1M servers
- 50-100 MW



Worldwide LHC Computing Grid
- >140 sites;
- ~250k cores;
- ~100 PB disk



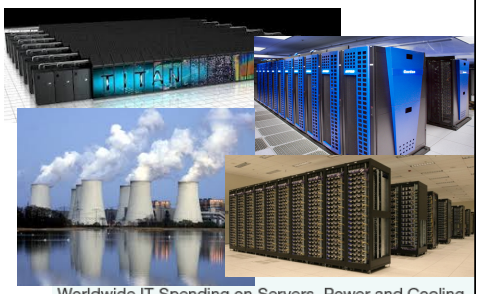
Special Purpose HW (Anton)
- > 100 time acceleration of MD simulations

RUTGERS RDI² Rutgers Discovery Informatics Institute

Energy Efficiency Critical at Extreme Scales

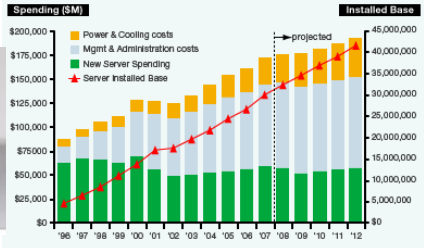
- Power has become a critical concern for HPC/supercomputing
  - Impacts operational costs, reliability, correctness
  - End-to-end integrated power/energy management essential
- Increasing scale towards **exascale**
  - Using existing technology would require gigawatts
    - Multiple nuclear reactors
    - > $2.5B annual power cost

Target < 20MW !!



- Power/energy **not only** from computation
  - Data transfer
    - Memory, I/O, HPC network (e.g., 3D torus)
  - Data storage
    - Disk, NVRAM, etc.

Worldwide IT Spending on Servers, Power and Cooling, and Management/Administration

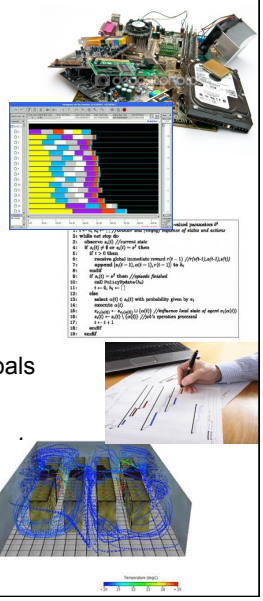


Source: IDC, Transforming the Datacenter: Consolidation, Pervasive Virtualization and Energy Optimization, Doc #DIR2009_T4_MB, Mar 2009

RUTGERS RDI² Rutgers Discovery Informatics Institute

Managing Energy/Power – A Wide Range of Techniques

- Aggressive component-level power management
  - CPU (e.g., DVFS), memory, disk, NIC, etc.
- Run-time power management
  - System level, exploiting slack in MPI programs, etc.
- Application/workload power management
  - Workload profiling/characterization, consolidation, etc.
  - Application, algorithm adaptations
- Autonomic policy adaptation
  - Characterize operational state and adjust management goals
- Cooling and thermal management
  - React to thermal hotspots, proactive workload placement
- …., etc.



Can we use these together in a cross-layer, coordinated, and consistent manner?

RUTGERS RDI² Rutgers Discovery Informatics Institute

Green HPC: Landscape (I /III)

- Dynamic use of low power modes (sleep, suspend, hibernate)
  - Advanced Configuration and Power Interface (ACPI)
    - Pallipadi et al. [OLS'07]
  - Switching on/off servers for energy conservation
    - Heath el al. [PPoPP'05]
- Dynamic scaling (usually impacts latency)
  - Processor Dynamic Voltage and Frequency Scaling (DVFS)
    - During communication (MPI slack)
      - Kappiah et al [SC'05], Lowenthal et al. [SC'06, SC'07], Freeh et al. [IPDPS'05, PPOPP'06]
    - Using application profiles
      - Cameron et al. [SC'05, Computer'05]
    - Using counters
      - Hsu et al. [SC'05], Rountree [ICS'09]

RUTGERS RDI² Rutgers Discovery Informatics Institute

Green HPC: Landscape (II/III)

- Dynamic scaling (contd.)
  - Scaling other components/subsystems
    - Dynamic memory frequency
      - Delaluz et al. [IEEE TC'01, DAC'02], Fradj et al. [DSD'06], Bianchini et al. [ASPLOS'11]
    - Storage subsystem -- Multi-speed disks and RAIDS
      - Rotem et al. [IPDPS'09], Pinheiro et al. [ICS'04]
    - NVRAM – PCM/STTM, memristors, flash-based
      - Caulfield et al. [ASPLOS'09, MICRO'10], Bianchini et al. [SC'11]
    - Flash-based SSD
      - Urgaonkar et al. [OSDI'08]

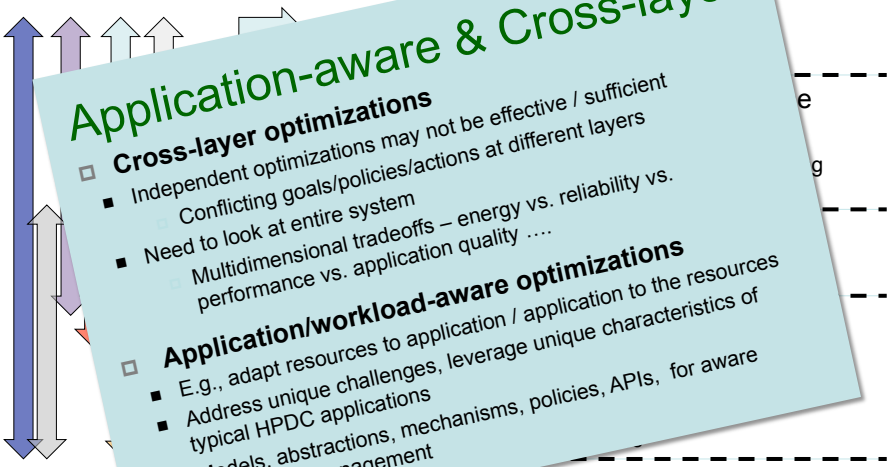
RUTGERS RDI²

Green HPC: Landscape (III/III)

- Other approaches
  - Architecture/Instruction-set based
    - Martonosi et al. [ISCA'00, ISCA'01,HPCA'12]
  - Cluster-based power management controller
    - Ranganathan et al. [SIRARCH'06]
  - Job allocation and scheduling for power conservation
    - Chase et al. [SOSP'01], Pinheiro et al. [DCS-TR-440], Zong et al. [ICPP'07]
  - Network power management
    - Lefevre et al. [IPDPS'09, ISPA'11]
  - Thermal-aware
    - Skadron et al. [TACO'04]
  - Virtualized environments, programming languages, etc.
  - …

RUTGERS RDI²

Energy/Power Management at M…ers



Application-aware & Cross-layer

- **Cross-layer optimizations**
  - Independent optimizations may not be effective / sufficient
    - Conflicting goals/policies/actions at different layers
  - Need to look at entire system
    - Multidimensional tradeoffs – energy vs. reliability vs. performance vs. application quality ….
- **Application/workload-aware optimizations**
  - E.g., adapt resources to application / application to the resources
  - Address unique challenges, leverage unique characteristics of typical HPDC applications
  - Models, abstractions, mechanisms, policies, APIs, for aware cross-layer management

…ucture – sensing environment

- Infrastructure: thermal sensors, instrumentation, monitoring, cooling

RUTGERS RDI² Rutgers Discovery Informatics Institute

GreenHPC – Overarching Research Goals

- Multiple orders of magnitude power/energy reduction are required – a cross-layer and application-aware strategy is necessary
  - Innovations necessary at multiple levels (algorithms, programming, runtime, OS, …)
- Energy/power-efficiency in combination with other objectives – Tradeoffs are essential
  - Quality of solution, Performance, Reliability, etc.
- Transporting, processing and analyzing increasing data volumes and rates is becoming the dominant challenge

RUTGERS RDI² Rutgers Discovery Informatics Institute

GreenHPC @ Rutgers – Selected Research Efforts

1. **Application-aware** power management
   - Using application behavior for aggressive proactive management
2. **Programming Support** for GreenHPC
   - Application-driven management using *hints*
3. Power-aware **data management**
   - Managing power for end-to-end application workflows

RUTGERS RDI² Rutgers Discovery Informatics Institute

1. Application-aware power management

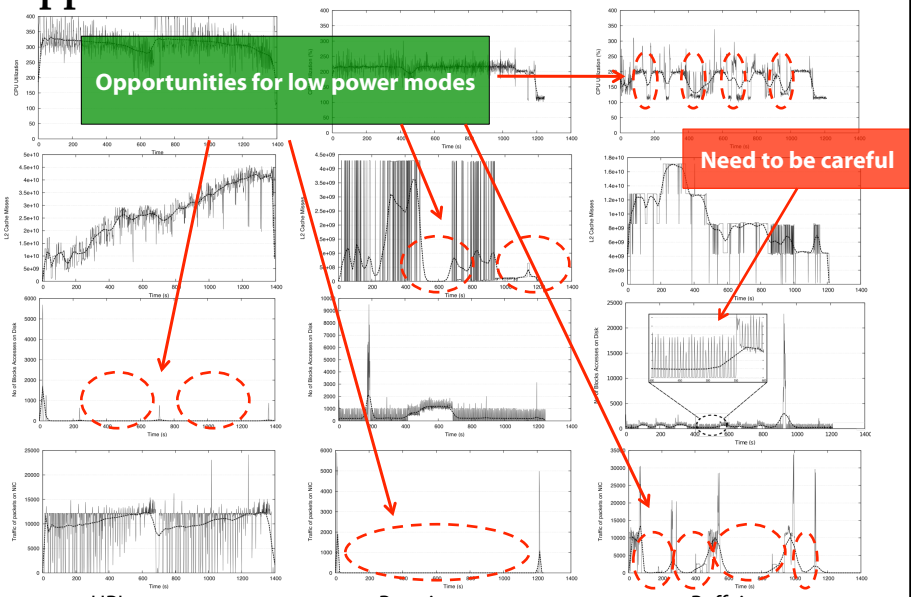
RUTGERS RDI² Rutgers Discovery Informatics Institute

Application-aware Power Management

- Objective: Use application behaviors to enable aggressive and proactive power management
  - Reactive power management is not always optimal and can result in large overheads
  - Proactive adaptation of resources based on application behavior and requirements (e.g., subsystems demand over time)
  - Maintain performance to the extent possible
- Overall approach [HiPC10, HiPC11]
  - Application characterization (subsystem demand)
  - Empirical quantification of possible power savings (upper bound)
  - Proactive subsystem power management at system level

RUTGERS RDI² Rutgers Discovery Informatics Institute

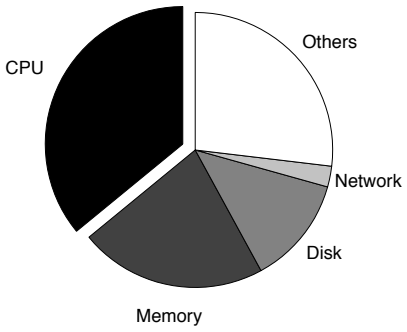
Application Characterization



RUTGERS RDI² Rutgers Discovery Informatics Institute

Component-based Power Management

- Application profiling
  - CPU utilization
  - Memory utilization (L2 cache misses is an indicator)
  - Disk utilization
  - Network traffic
- Proactive power management
  - Based on the application profile
  - Using component/subsystems low power modes
    - CPU DVFS
    - Memory (e.g., switching off banks)
    - Disk (spin down/up)
    - NIC (sleep modes)



Typical system power breakdown

RUTGERS RDI² Rutgers Discovery Informatics Institute

Application-Aware Proactive Aggressive Power Management (PAPM)

Energy consumed in an interval of time t_i for a subsystem *sys:*

$$E^{sys}_{t_i} = P^{sys}_{active} \cdot t_{i,active} + P^{sys}_{idle} \cdot t_{i,idle} + E_{t_{on/off}}$$

- $E_{t_{on/off}}$ → Energy cost associate to a transition
- $t_{i,idle}$ → Time duration in idle mode
- P^{sys}_{idle} → Power if the subsystem sys is idle (low power mode)
- $t_{i,active}$ → Time duration in active mode
- P^{sys}_{active} → Power if the subsystem sys is active

where *sys={mem, disk, nic}* and $P = P_{static} + P_{dynamic}$

$$P_{dynamic} = P_{cpu} + P_{mem} + P_{disk} + P_{nic}$$

RUTGERS RDI² Rutgers Discovery Informatics Institute

PAPM Algorithm

Algorithm
Input: Physical memory, disk, NIC
Output: Power state for a given subsystem
for *time = 0 to i* **do**
  for *sys = disk, memory, NIC* **do**
    if *Idle-Condition* **then**
      if *Time-Condition* **then**
        if *Energy-Condition* **then**
          $P^{sys}_{PwrSt} = P^{sys}_{idle}$;
        end
      end
      else
        $P^{sys}_{PwrSt} = P^{sys}_{active}$;
      end
    end
  end
end

- P^{sys}_{idle} → Power if the subsystem sys is idle (low power mode)
- P^{sys}_{PwrSt} → Current power for subsystem sys
- P^{sys}_{active} → Power if the subsystem sys is active

- **Idle-Condition** checks if the subsystem is going to be idle in the next time interval
- **Time-Condition** checks if it is feasible to transition to any of the available lower power states based on the latency of the subsystems and the idle time between the two active periods
- **Energy-Condition** checks if it is worthwhile to transition the system to a low power state and if any energy savings will be achieved

RUTGERS RDI² Rutgers Discovery Informatics Institute

Subsystems Energy Contribution – Simulation

Benchmark	DVFS	Run Time (s)	Energy (J)	Energy Savings CPU	Memory	Disk	NIC	Total (J)	%
HPL	×	1,382 s	298,546 J	-	1,380 J	5,338 J	240 J	6,958 J	**2.33%**
	√	1,383 s	292,824 J	5,722 J				12,680 J	4.33%
b_eff_io	×	1,206 s	164,224 J	-	5,297 J	-	1,124 J	6,421 J	**3.90%**
	√	1,212 s	161,460 J	2764 J				9,185 J	5.69%
bonnie++	×	1,247 s	190,613 J	-	3,263 J	574 J	3,841 J	7,678 J	**4.03%**
	√	1,248 s	187,533 J	3,080 J				10,758 J	5.73%
TauBench	×	1,134 s	251,904 J	-	3,377 J	7,297 J	1,979 J	12,653 J	**5.02%**
	√	1,136 s	244,473 J	7,431 J				20,084 J	8.21%
FFTW	×	1,052 s	198,621 J	-	2,112 J	6,927 J	297 J	9,336 J	**4.70%**
	√	1,055 s	193,146 J	5,475 J				14,811 J	7.67%

- Minimal performance penalty – 0.22% on an average
- Can improve the energy efficiency by up to ~8%, depending on the workload profile for these benchmarks (potentially more for other workloads)

RUTGERS RDI² Rutgers Discovery Informatics Institute

Energy Savings – Experimental Validation

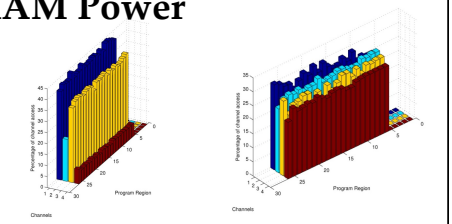
Benchmark	Configuration	Run Time (s)	%	Energy (J)	%	EDP	%
HPL	*Reference*	1,383 s	-	292,824 J	-	404,975,592	-
	PAPM	1,385 s	+0.14%	287,906 J	-1.67%	398,749,810	-1.53%
	PAPM+SSD	1,385 s	+0.14%	281,559 J	-3.84%	389,959,215	-3.70%
b_eff_io	*Reference*	1,212 s	-	161,460 J	-	195,689,520	-
	PAPM	1,217 s	+0.41%	157,335 J	-2.55%	191,476,695	-2.15%
	PAPM+SSD	1,134 s	-6.43%	143,768 J	-10.95%	163,032,912	-16.68%
bonnie++	*Reference*	1,248 s	-	187,533 J	-	234,041,184	-
	PAPM	1,249 s	+0.08%	182,904 J	-2.47%	228,447,096	-2.39%
	PAPM+SSD	1,169 s	-6.33%	168,606 J	-10.09%	197,100,414	-15.78%
TauBench	*Reference*	1,136 s	-	244,473 J	-	277,721,328	-
	PAPM	1,139 s	+0.26%	236,496 J	-3.26%	269,368,944	-3.00%
	PAPM+SSD	1,137 s	+0.08%	229,446 J	-6.14%	260,880,102	-6.06%
FFTW	*Reference*	1,055 s	-	193,146 J	-	203,769,030	-
	PAPM	1,057 s	+0.19%	187,677 J	-2.83%	198,374,589	-2.64%
	PAPM+SSD	1,051 s	-0.38%	177,071 J	-8.32%	186,101,621	-8.67%

- Reference: State-of-the-art DVFS
- PAPM: Proactive Aggressive Power Management
- PAPM+SSD: PAPM with the use of SSD rather than disk

RUTGERS RDI² Rutgers Discovery Informatics Institute

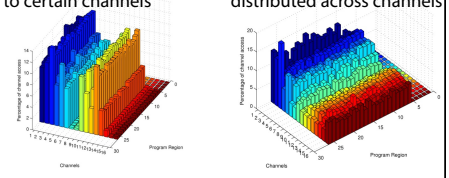
Use Case: Multi-channel DRAM Power Management [HiPC'11]

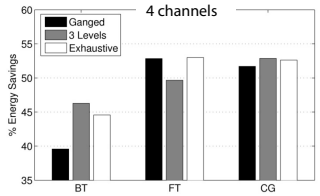
- Applications exhibit heterogeneous BW requirements over time
- Approach:
  - Apply frequency and voltage scaling per channel
  - Affinity to channels: more channels – more opportunities
  - Controlling channels independently – more energy savings
- Application-awareness provides higher energy conservation with low penalty in performance



Default: accesses clustered to certain channels

Interleaved: accesses distributed across channels





RUTGERS RDI² Rutgers Discovery Informatics Institute

Application-aware Power Management: Lessons Learned

- Application-aware power control can lead to improved energy efficiency without significant performance penalty
- Power management at the subsystem level can not be neglected
- Low power devices and upcoming technologies (e.g., SSD, storage-level memory) can significantly increase energy efficiency

- However, power management at a single level is not enough
- A *cross-layer* approach is required!

RUTGERS RDI² Rutgers Discovery Informatics Institute

2. Programming Support for GreenHPC

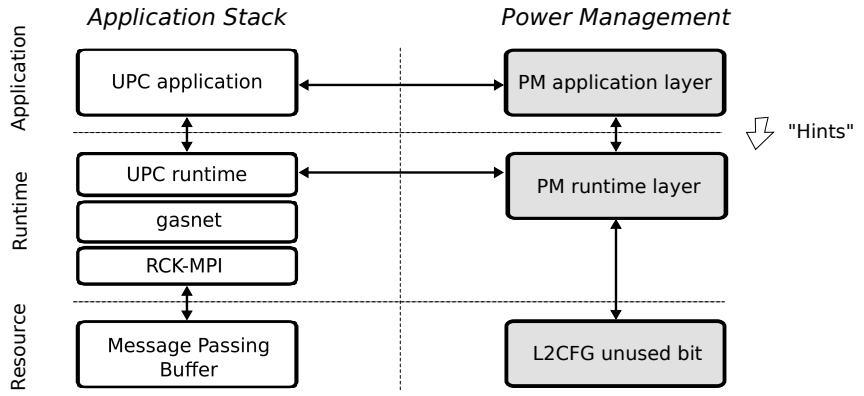
RUTGERS RDI² Rutgers Discovery Informatics Institute

Programming Support for GreenHPC

- Objective: Develop programming support to enable applications to provide hints to drive runtime power management
- Overall approach [HPDC12]
  - Language extensions ("hints") to drive/tune runtime power management
  - Runtime middleware for application-aware cross-layer power management
  - Implementation: PGAS applications on many-core platform (i.e., Intel SCC)

RUTGERS RDI² Rutgers Discovery Informatics Institute

Architecture for PGAS Programming Support

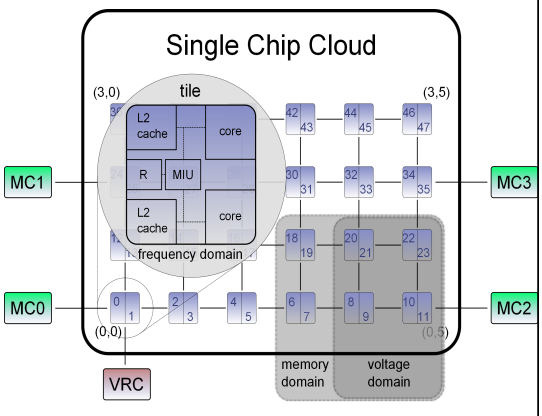


- Implementation using:
  - Berkeley UPC runtime
  - Intel SCC platform

RUTGERS RDI² Rutgers Discovery Informatics Institute

Experimental Platform

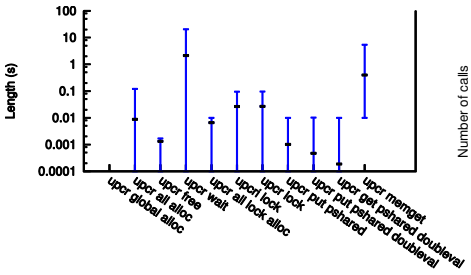
- Intel SCC platform
  - Many-core prototype by Intel Labs Tera-scale Program
  - 48 x86 P54C Pentium cores
  - 24-router on-die mesh network, hardware message-passing
  - Per-core 16 KB L1 cache, 256 KB L2 cache
  - Metering infrastructure
  - Frequency (per 2 core groups) and voltage (per 8 core groups) scaling
  - Power ranges from 25W to 125W, i.e., 0.5W-2.6W per core

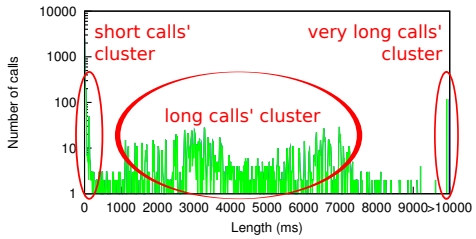


RUTGERS RDI² Rutgers Discovery Informatics Institute

Application in the Loop

- Application-awareness can improve runtime power management (in combination with system level power controls)
- Example: NAS FT profiling
  - Only certain operations are candidate for power management
    - Remote memory access (memget) and barrier (wait) operations
  - Both operations have lots of very-short calls and several med-long operations (short calls result in large overheads)





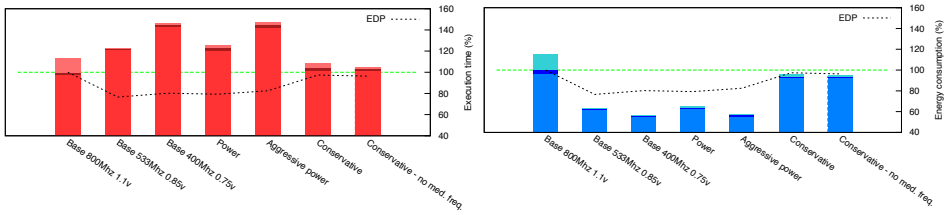
RUTGERS RDI² Rutgers Discovery Informatics Institute

Language Extensions Provide User "hints"

- Analogous to processor DVFS governors at the OS level
  - PM PERFORMANCE, maximum performance
    - full power
  - PM CONSERVATIVE, balance power/performance
    - low power modes during slack
  - PM POWER, minimum power, limited delay penalty
    - low voltage level, regardless application's slack periods
  - PM AGGRESSIVE POWER, maximum energy save
    - lowest voltage and frequency levels

RUTGERS RDI² Rutgers Discovery Informatics Institute

Use Case 1: NAS FT – UPC implementation



- Runtime power management policy selects appropriate tradeoff between energy and delay
  - Larger energy savings -> higher time penalties
- Energy reduction with little time penalty is possible
  - PM CONSERVATIVE 7% results in energy savings with a 0.4% time penalty
- Language extensions provide hints to guide runtime policy selection
  - Other hints can obtain higher energy savings (45%) with a higher time penalty cost

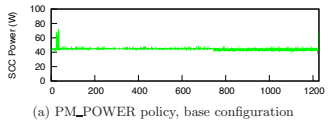
RUTGERS RDI² Rutgers Discovery Informatics Institute

Use Case 2: Sobel Filter Application

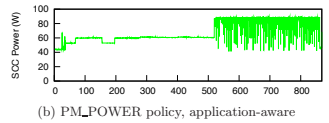
- Sobel filter has two different phases
- Two studied strategies
  - Using the same policy during the whole application execution
  - Using hints during the initial phase (e.g., PM_POWER)
- Hints avoid time delay and maintains energy savings
  - The right power mode cannot be identified at runtime or system level

Language extension (user hints)	Sobel - fix policy		Sobel – dynamic policy	
	Time (%)	Energy (%)	Time (%)	Energy (%)
PM_PERFORMANCE	100.0	100.0		
PM_POWER	147.6	75.0	101.8	79.4
PM_AGGRESSIVE_POWER	193.8	72.8	101.9	73.7
PM_CONSERVATIVE	**118.0**	**75.7**	**101.5**	**73.7**

Power over time (Sobel)



(a) PM_POWER policy, base configuration



(b) PM_POWER policy, application-aware

RUTGERS RDI² Rutgers Discovery Informatics Institute

Programming Support for GreenHPC: Lessons Learned

- Cross-layer power management enables a wider range of energy and performance behaviors
- Application-awareness improves runtime power management
- Language extensions (via user-level hints) enable a better selection of the appropriate energy/performance tradeoff
- Cross-layer decisions can help address hardware power management limitations
  - E.g., Intel SCC provides per-tile frequency scaling and per-voltage domain voltage scaling

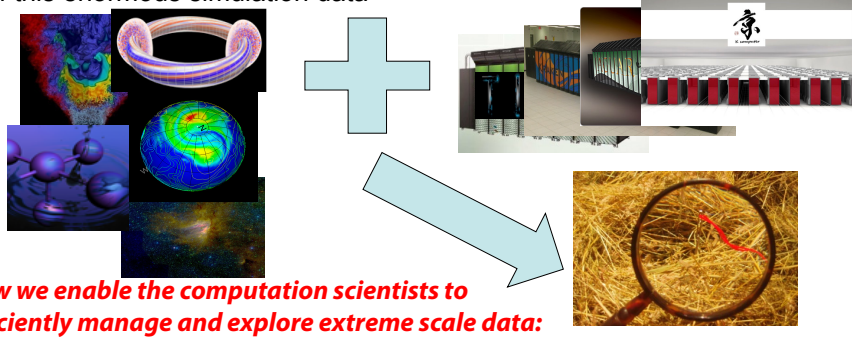
RUTGERS RDI² Rutgers Discovery Informatics Institute

3. Power-aware data management

RUTGERS RDI² Rutgers Discovery Informatics Institute

Scientific Discovery through Simulations - III

- Scientific simulations running on high-end computing systems generate huge amounts of data!
  - If a single core produces 2MB/minute on average, one of these machines could generate simulation data between ~**170TB** per hour -> ~**700PB** per day -> ~**1.4EB** per year
- Successful scientific discovery depends on a comprehensive understanding of this enormous simulation data



How we enable the computation scientists to efficiently manage and explore extreme scale data: "find the needles in haystack" ??

RUTGERS RDI² Rutgers Discovery Informatics Institute

Traditional *Simulation -> Insight* Pipelines Break Down

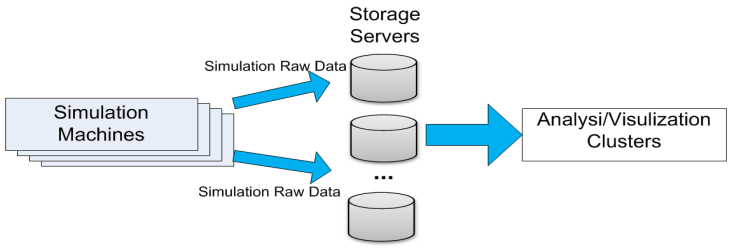


Figure. Traditional data analysis pipeline

- Traditional ***simulation -> insight*** pipeline:
  - Run large-scale simulations on large supercomputers
  - Dump data on parallel disk systems
  - Export data to archives
  - Move data to users' sites – usually selected subsets
  - Perform data manipulations and analysis on mid-size clusters
  - Collect experimental / observational data
  - Move to analysis sites
  - Perform comparison of experimental/ observational to validate simulation data

RUTGERS RDI² Rutgers Discovery Informatics Institute

Challenges Faced by Traditional HPC Data Pipelines

- **Data analysis challenge**
  - Can current data mining, manipulation and visualization algorithms still work effectively on extreme scale machine?
- **I/O challenge**
  - Increasing performance gap: disks are outpaced by computing speed

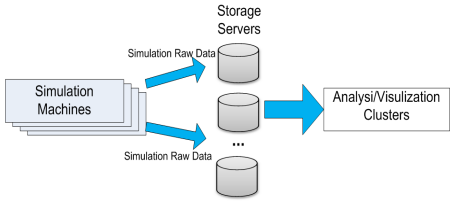


Figure. Traditional data analysis pipeline

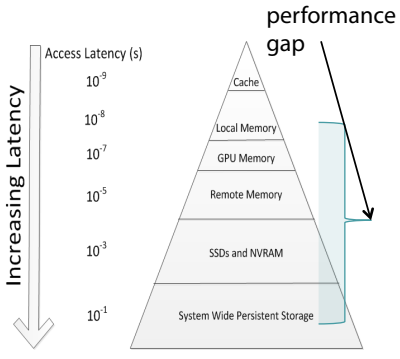
- **Data movement challenge**
  - Lots of data movement between simulation and analysis machines, between coupled mutli-physics simulation components -> longer latencies
  - Improving data locality is critical: do work where the data resides!
- **Energy challenge**
  - Future extreme systems are designed to have low-power chips – however, much greater power consumption will be due to memory and data movement!

*The **costs of data movement** are increasing and dominating!*

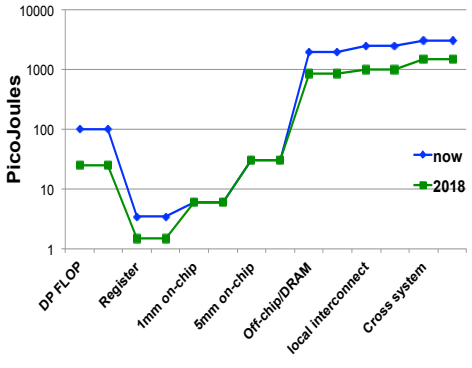
RUTGERS RDI² Rutgers Discovery Informatics Institute

The Cost of Data Movement

- Moving data between node memory and persistent storage is slow!



- The energy cost of moving data is a significant concern

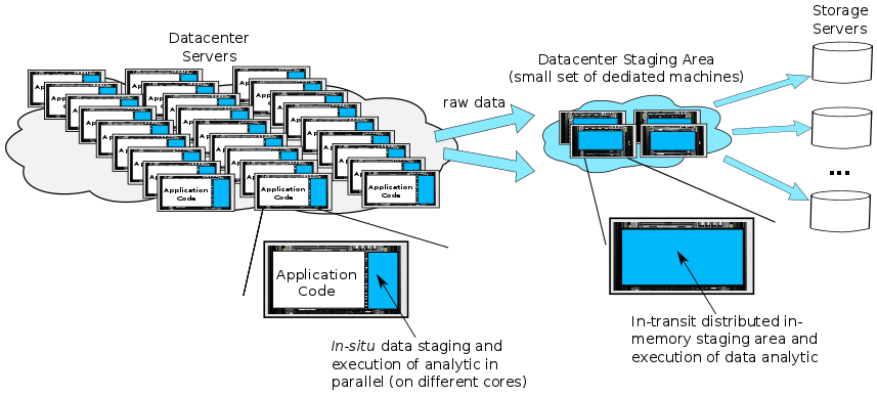


$$Energy\_move\_data = \frac{bitrate * length^2}{cross\_section\_area\_of\_wire}$$

From K. Yelick, "Software and Algorithms for Exascale: Ten Ways to Waste an Exascale Computer"

RUTGERS RDI² Rutgers Discovery Informatics Institute

Rethinking the Data Management Pipeline – Hybrid Staging + In-Situ & In-Transit Execution



- Exploit multi-levels in-memory **Hybrid Data Staging** to:
  - Decrease the gap between CPU and IO speeds
  - Dynamically deploy and execute data analytical or pre-processing operations either in-situ or in-transit
  - Improve IO write performance

RUTGERS RDI² Rutgers Discovery Informatics Institute

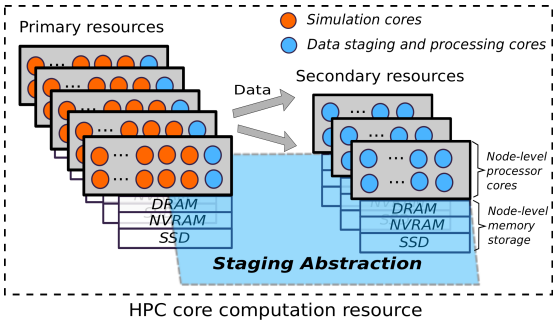
Data Staging over Deep Memory Hierarchy

Motivation

- Small DRAM capacity per core – even aggregated memory on dedicated nodes can hardly keep all coupled data (given the ratio of resource allocations for compute nodes and dedicated nodes)

Hybrid Staging

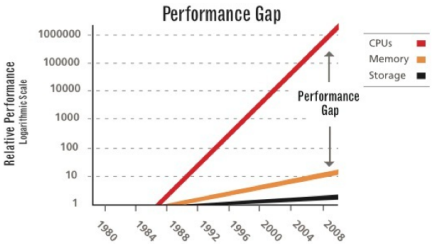
- Spans horizontally across compute nodes of both primary and secondary resources
- Spans vertically across the multi-level memory hierarchy, e.g. DRAM/NVRAM/SSD, to extend the capacity of in-memory data staging

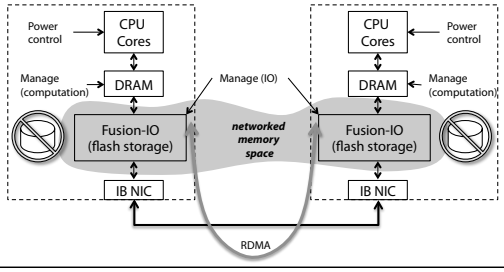


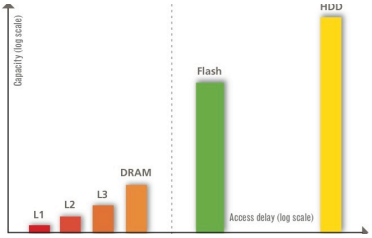
RUTGERS RDI² Rutgers Discovery Informatics Institute

Deep Memory Hierarchy

- NVRAM bridges latency/performance gap between DRAM & disk
- NVRAM usage modes
  - I/O staging and data "caching"
  - Out-of-core computations (virtual memory)
  - "In-the-space" data manipulation and/or reduction
- Fusion-io technology (ioDrive)





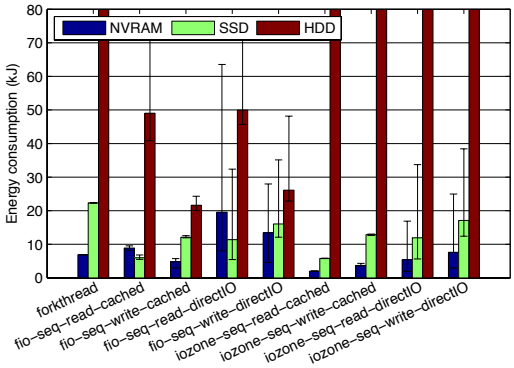


RUTGERS RDI² Rutgers Discovery Informatics Institute

Energy-Performance Tradeoffs in the Data Analytics Pipeline

Characterize power/performance behaviors of deep memory hierarchy using benchmarks

Lessons learned: Storage-class NVRAM memory is a good candidate as intermediate storage



RUTGERS RDI² Rutgers Discovery Informatics Institute

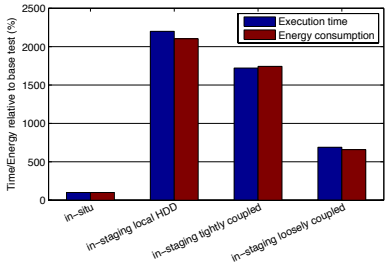
Energy-Performance Tradeoffs in the Analytics Pipeline

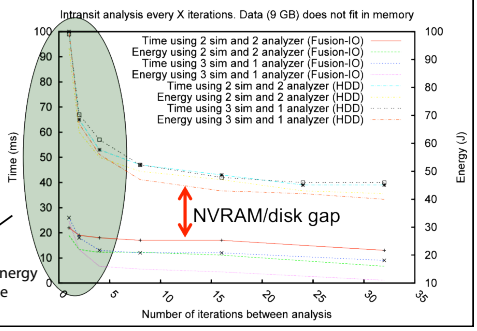
Impact of data placement and movement

- Canonical system architecture composed of DRAM, NVRAM, SSD and disk storage levels and networked staging area
- Canonical workflow architecture that can support in-situ and in-transit analytics
- **Lessons learned:** Data placement/ movement patterns significantly impact performance and energy

Role of the quality of solution

- **Lessons learned:** Frequency of analytics is driven by the dynamics of features of interest, but is limited by node architecture.





Large impact on energy and execution time

Power Behavior of In-situ Analytics Pipeline

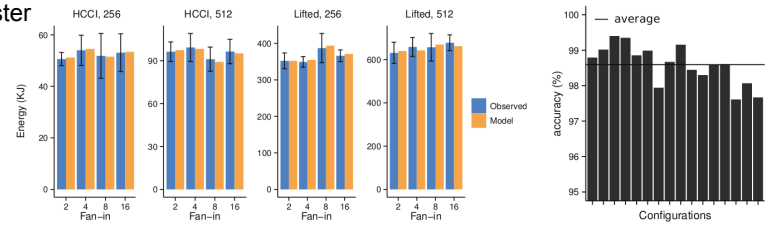
(with J. C. Bennett, H. Kolla, J.Chen, T. Bremer, A. G. Landge, A. Gyulassy, P. McCormick, S. Pakin, V. Pascucci)

- **Motivation.** Coupled simulation workflows use **online data processing to reduce data movement**. Need to explore **energy/performance** tradeoffs.
- **Use case.** Combustion simulation workflow with an **in-situ data analytics** pipeline.
- **Energy model** based on machine-independent application profile

$$E = E_{system} + E_{comm}.$$
$$E_{system} = T \cdot (P^{static}_{system} + P^{dynamic}_{system}).$$
$$P^{static}_{system} = (P^{idle}_{cpu} + P^{idle}_{mem} + P^{idle}_{nic} + P^{idle}_{misc}).$$
$$P^{dynamic}_{system} = \alpha_{cpu} \cdot P^{active}_{cpu} + \alpha_{mem} \cdot P^{active}_{mem}.$$

$$E_{comm} = \begin{cases} \sum_{i=1}^{M} \frac{data_i}{BW_{net}} \cdot P_{transfer} & \text{if } smp(src_i) \neq smp(dest_i) \\ \sum_{i=1}^{M} \frac{data_i}{BW_{mem}} \cdot (P^{active}_{cpu} + P^{active}_{mem}) & \text{if } smp(src_i) = smp(dest_i) \end{cases}$$

- Model **validation** on small cluster

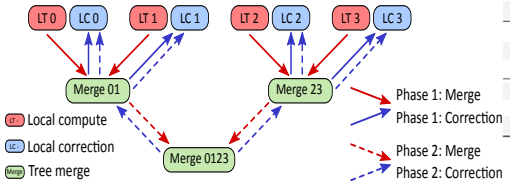


RUTGERS RDI² Rutgers Discovery Informatics Institute

Power behavior of the in-situ analytics pipeline

- Study various **co-design trade-offs**
  - Algorithm design choices
  - Runtime application deployment
  - System architecture
- Algorithm parameter impact is quantified

# cores	Blocking factor	Fan in	Time (s)	Energy (KJ)	$MIPS$ $(x10^3)$	MEM_{bw} (GB/s)	MPI MSGs
320	(5, 8, 8)	8	55.9	214.4	142	484	3,716
	(5, 8, 8)	16	57.7	221.3	138	473	4,752
	(5, 8, 8)	32	59.3	227.4	139	478	5,946
960	(15, 8, 8)	8	18.8	225.5	423	1,440	10,917
	(15, 8, 8)	16	19.0	228.7	420	1,434	11,116
	(15, 8, 8)	32	25.3	300.6	334	1,151	16,815
1920	(15, 16, 8)	8	11.3	284.5	735	2,517	19,548
	(15, 16, 8)	16	12.4	310.0	700	2,417	24,958
	(15, 16, 8)	32	12.0	301.6	710	2,446	24,644
3840	(15, 16, 16)	8	8.3	447.7	1183	4,135	57,424
	(15, 16, 16)	16	12.7	652.0	864	3,070	73,367
	(15, 16, 16)	32	8.2	440.5	1,172	4,085	50,902
11520	(45, 16, 16)	8	6.9	1,227.1	1,869	6,717	194,043
	(45, 16, 16)	16	6.7	903.2	N/A	N/A	170,912
	(45, 16, 16)	32	9.9	1,668.2	1,526	5,593	194,596
23040	(45, 32, 16)	8	15.7	5,095.2	1,215	4,508	417,360
	(45, 32, 16)	16	7.5	2,861.9	2,457	9,106	360,422
	(45, 32, 16)	32	18.0	6,060.9	1,477	5,641	508,704
30000	(75, 40, 10)	8	8.0	4,104.4	2,777	10,399	600,072
	(75, 40, 10)	16	8.7	4,357.4	2,497	9,360	535,693
	(75, 40, 10)	32	12.3	5,828.7	2,047	7,778	556,098

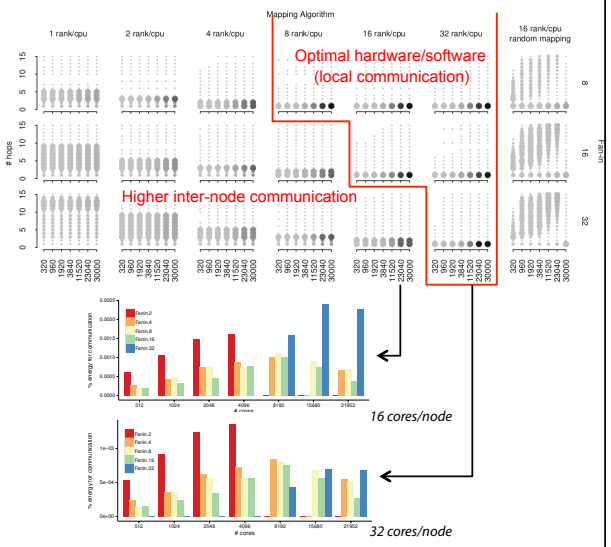


Topology analysis: parallel merge tree algorithm

RUTGERS RDI² Rutgers Discovery Informatics Institute

Power Behavior of In-situ Analytics Pipeline

- Communication patterns depends on:
  - Algorithmic configuration (Fan-in)
  - MPI processes mapping (ranks/cpu)
  - System architecture (e.g., 16 cores/node vs. 32 cores/node)
- Data motion energy depends on transfers



Lessons learned: Energy for data motion is impacted by algorithm and system

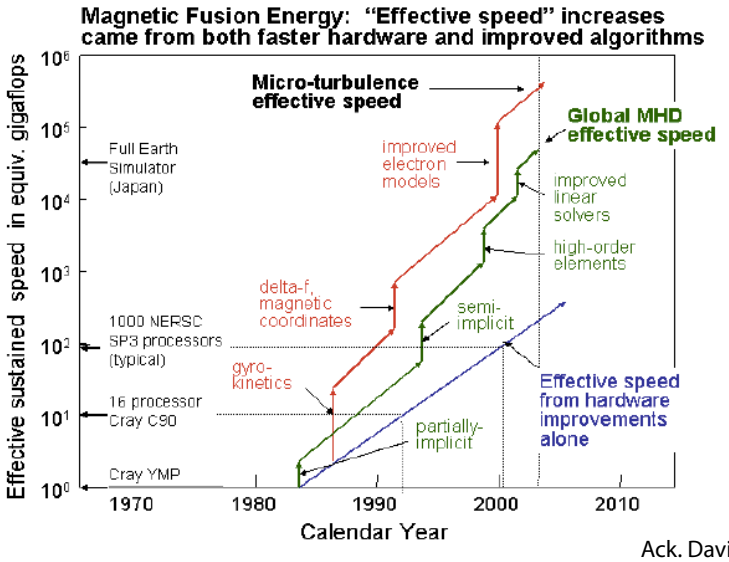
RUTGERS RDI² Rutgers Discovery Informatics Institute

Conclusions

- Energy-related challenges have become dominating concerns for scientific applications at extreme scales
- Costs (energy, latency) related to transporting, processing and analyzing increasing data volumes and rates are limiting the insights from extreme scale applications
- Multiple orders of magnitude power/energy reduction are required – a cross-layer and application-aware strategy is necessary
  - Innovations necessary at multiple levels (algorithms, programming, runtime, OS, etc.)
- Energy/power-efficiency in combination with other objectives – understanding tradeoffs are important
  - Quality of solution, Performance, Reliability, etc.
- Co-design is essential

RUTGERS RDI² Rutgers Discovery Informatics Institute

Not All Improvements Come from HW/System



Ack. David Keyes

RUTGERS

RDI²
Rutgers Discovery Informatics Institute

Thank You!

Green HPC

ADIOS

EPSi
Edge Physics Simulation

SDAV
Scalable Data Management, Analysis, and Visualization

EXACT

SciDAC
Scientific Discovery through Advanced Computing

RDAV

CPES
Center for Plasma Edge Simulation

Manish Parashar, Ph.D.
Prof., Dept. of Electrical & Computer Engr.
Rutgers Discovery Informatics Institute (RDI²)
Cloud & Autonomic Computing Center (CAC)
Rutgers, The State University of New Jersey

Email: parashar@rutgers.edu
WWW: rdi2.rutgers.edu

TASSL The Applied Software Systems Laboratory

CAC Cloud and Autonomic Computing Center

OLCF Oak Ridge Leadership Computing Facility

Oak Ridge National Laboratory