Report of the Science of Power Management Workshop
April 9-10, 2009
Arlington, VA

Kirk W. Cameron, Virginia Tech

Kirk Pruhs, University of Pittsburgh

Sandy Irani, University of California, Irvine

Partha Ranganathan, Hewlett-Packard

David Brooks, Harvard University

This workshop was sponsored by the National Science Foundation (www.nsf.gov). The views expressed in this report are those of the individual participants and are not necessarily those of their respective organizations or the workshop sponsor.

Version 1.0 Technical Report No. VT/CS-09-19 August 31, 2009

Table of Contents

> **"**The energy used by the nation's servers and data centers is significant...more than the electricity consumed by the nation's color televisions and similar to the amount of electricity consumed by approximately 5.8 million average U.S. households.**"**
>
> EPA Report to Congress on Server and Data Center Energy Efficiency.
> In response to Public Law 109-431, August 2, 2007.

Preface

A number of reports in the past several years have questioned the sustainability of the computing infrastructure of the United States. Reports by the U.S. EPA and others have concluded that in order for the U.S. to maintain its competitiveness the power and energy consumption challenges facing our IT infrastructure must be addressed.

Power consumption of IT equipment begins with the design of a microchip and continues across the traditional technological boundaries of system design integration and design of the facilities that house them. Since most techniques have been developed in isolation, there are serious gaps in our understanding of the "science" behind these complex systems across and within these boundaries.

In recognition of recent developments, NSF sponsored a Workshop on the Science of Power Management on April 9-10 in Arlington, Virginia. The intent of the workshop was to bring together leading thinkers in the area of power and thermal management from chips to systems to facilities and integrate them with algorithm and theory experts to identify, prioritize and recommend promising research directions in the hope of incubating development of a science of power management.

The format of the workshop was a series of keynote talks from industrial experts and academic leaders followed by breakout sessions focusing on software, hardware, networks, storage, and physicals. Break out groups met twice and group leaders presented their findings to the committee and attendees to close the workshop. The steering committee was tasked with authoring this report and releasing it to the public upon delivery to NSF.

This document contains an executive summary of the key findings of the workshop and the key recommendations for future research to support the development of a science of power management. This workshop would not have been possible without the hard work and diligence of the breakout group leaders and the workshop attendees. We would also like to thank the steering committee members for the additional time and effort they volunteered despite their intense schedules. And finally, a word of thanks to the National Science Foundation for sponsoring this workshop without which this report would have been impossible.

Kirk W. Cameron, Virginia Tech
Kirk Pruhs, University of Pittsburgh
Krishna Kant, NSF

SciPM Workshop Co-chairs

Executive Summary

We believe that there is a need for a consolidated effort to establish a Science of Power Management, or comprehensive set of principles and techniques that provide practical solutions to the power issues facing the information technology community.

Background

There is clear consensus that one of the most important grand challenges facing humanity in the next century is to develop technologies that will allow us to continue advancement in a sustainable manner. There is increased scrutiny on the national and international stage for the United States to curb its energy use and thus carbon emissions. Towards this end, recent studies by the US EPA and Department of Energy have concluded that in particular more effort is needed to curb the power consumption of data centers. With the recent election of Barack Obama, the US is more likely to sign a version of the Kyoto treaty that commits the US to reduce emissions further. Currently IT devices consume about as much energy and produce about as much carbon dioxide as the airline industry. However, because use of IT technology is still growing exponentially (centralized deployments of enterprise volume servers in data centers are growing 12% annually), and because energy and power have not traditionally been first order design constraints for IT technology, improvements in the energy efficiency of IT devices will be much more dramatic, and eventually have much greater impact than in other areas of technology, such as aircraft technology. Some progress is already being made towards these goals. For example, the IT industry has formed groups such as the Green Grid and SPECpower aimed at self-regulation through establishment of best practices for energy efficient data centers.

While it is important to address power management issues in every aspect of IT use, improving the energy efficiency of large data centers is a particularly critical need. If the power consumption of data centers goes unchecked, the sustainability of our national computing infrastructure is in question. These servers support the electronic infrastructure critical to enterprise use for businesses, e-commerce and the Internet. Power consumption and heat production lead to increased cost and reduced reliability in current data centers which in turn amplifies the need to build more.

There has then been a dramatic growth recently in the scope and diversity of research addressing power management from chip and data center design to facility management. Nearly every major conference across the discipline of computer science includes sessions related to power and thermal management from computational theory to compilers and systems to software engineering. New workshops, conferences, and special issue journals are emerging that solicit papers on power and thermal management. Professional magazines such as IEEE Computer have created ongoing columns related to greener computing including power and thermal management. Architects investigate energy-efficient microarchitectures, system researchers investigate operating system power scheduling policies, and thermal engineers investigate cooling systems to address server density issues. However, experts in each of these areas are often isolated from each other which make collaboration difficult. Currently power and thermal management techniques are generally designed in isolation for each type of device, such as clock gating on chips, power state management of a laptop, and load management across servers. The

lack of coordination among techniques can cause missed power management opportunities as well as power management policies that conflict with one another.

The time is appropriate to consider a "science" of power management. Quoting Webster's dictionary, "science is knowledge or a system of knowledge covering general truths". For example, the main purpose of the science of computers (i.e. computer science) is to establish a framework to reason about computation, and to develop a collection of techniques that are applicable in solving a broad range of computational problems. Some examples of techniques developed by computer scientists that have found uses in a variety of settings include hashing, public-key cryptography, and latency hiding with predictive prefetching. Analogously, a science of power management would establish a framework to reason about power/energy/temperature, and to develop a collection of widely applicable power management techniques. It should be emphasized that we can not reason about energy usage in isolation in the same way that we can formally reason about time in isolation as a computational resource. One interpretation of the Church-Turing thesis is that physical laws impose lower bounds on the time required to solve certain problems; as all computation can be made reversible, it seems that physical laws do not impose any inherent lower bound on the energy required to solve any problem. Therefore, the energy characteristics of abstract models will have to be based on characteristics of current and conceivable technologies, not on physical laws. Of course, we also ask that new models be robust to changes in technology so that principles and techniques will still be applicable even as the specific parameters of computer systems evolve over time. Furthermore metrics used to evaluate algorithms will necessarily incorporate trade-offs between energy, time, communication, etc. A formal framework would ideally enable algorithm developers to design algorithms that balance the use of CPU cycles, communication and memory to optimize energy usage and performance in completing a specific task. For example, under a particular model one might hope to identify what level of compression optimally trades off the energy savings of communication with the additional energy costs for compression and decompression at the end points.

While algorithms designers are concerned with solving specific problems in a way that minimizes the use of limited resources, systems designers must devise policies that take a set of tasks whose resource needs are (at least partially) determined and decide how to allocate an ensemble of resources among these tasks. Energy-aware policies will make use of flexibility in load balancing, processor speed, sleep states and other tunable parameters to allocate different resources within a system. In order to formally reason about these tradeoffs, theoretical models will need to balance the requirements of different tasks as well as the availability of resources just as, for example, abstract models for parallel computation incorporate resources such as CPU cycles, communication and storage. A formal framework of this kind would also be useful in designing the optimal distribution of components at design time given some knowledge of the expected workload of the system, and some specified balance between performance and energy conservation. These problems will be especially challenging because systems designed to conserve energy will likely be composed of resources with heterogeneous characteristics. For example, systems may consist of high power and high performance resources for critical tasks and lower power and performance resources for noncritical tasks. Past theoretical research on resource management in heterogeneous systems has not considered the energy characteristics of

resources. Given that energy usage has significantly different mathematical characteristics than time and space, a new theory of energy-heterogeneous systems is needed.

Key Findings & Recommendations to NSF: Vision for a Science of Power Management

Finding #1: The need for further scientific observation.

Empirical observations are the cornerstone of the scientific method and the beginnings of the sciences we utilize today. We need to observe systems to see how they perform in various situations. This includes not only current commercially available systems, but systems purposely constructed as prototypes of new technologies. We need to observe phenomena like how subthreshold leakage is affected by temperature, how one core's temperature affects neighboring cores temperatures, and how servers cool in various situations and with various cooling technologies. It is important that these measurements be widely disseminated. This information will be needed to construct appropriate models. For example, do cores, chips, and servers cool in fundamentally the same way, and thus might reasonably be modeled by a single model, or are they fundamentally different requiring different models?

Recommendation to NSF: Research should be encouraged that includes a strong prototyping component. Prototyping at the chip, board, rack, and datacenter levels can provide a solid grounding for modeling and power management techniques that span the fields engaged in these problems (e.g. architecture, system software, networking, algorithms).

Finding #2: Identify the problems.

There are numerous challenges in identifying metrics that are of greatest importance to power, energy, and thermal optimizations. Identifying metrics that are applicable across the boundaries of chips, systems, and facilities is especially challenging. Since scientific approaches often require repeatability, another important component is to ensure the metrics themselves can be measured and the experiments can be recreated.

Metrics are critical to establishing goals and evaluating outcomes. They serve as a means of systematically evaluating new techniques. Perhaps even more importantly, observations and measurements are the first step to analytical analyses than can lead to theoretical concepts such as optimization boundaries.

Recommendation to NSF: Research should be encouraged that defines appropriate standard metrics. These metrics can provide a clear target for success and enable comparisons across fields.

Finding #3: The development of good models.

Models are typically developed to explain phenomenon experienced through observations. Participants of the workshop agreed that once consensus metrics for observations were established, the next big challenge was to develop models that explain observations across the complex systems being observed. Examples include interactions at the boundaries of hardware and software and as well the boundaries between IT equipment and facilities. There are currently few, if any, models that capture these interactions and the need is critical if we are

ultimately to optimize these complex systems. Different types of models will be needed, from abstract simple models used for high-level heuristic reasoning during the design of technologies, to detailed low-level models for accurate predictions of the performance of particular technologies.

A good abstract model is one that is complex enough to capture the most important factors in the application, while being simple enough to allow abstract reasoning. Thus the purpose of a good abstract model is not to model exactly all aspects of reality, but to be used as a heuristic tool by an engineer when searching for engineering solutions. The establishment of a good model can be the difference between success and failure of a science as a foundation of an engineering discipline. Historically the success of algorithmics as a science underlying software engineering relied on the utility of the Random Access Model (RAM) as a model for computation. While the RAM model ignores issues such as memory hierarchies, it captures enough reality to be an extremely useful model for software engineers. In contrast, it has proven to be more difficult to find an equally useful model for parallel computation. Many breakout groups mentioned the need to develop a general model of computation, akin to the RAM, that accounts for energy and power.

There is a need for models to be developed at all abstraction levels and granularities. Taking the microprocessor as an example, there is a need for accurate projections of device scaling characteristics, basic memory and logic components, core-level models, core-to-core interconnect, and off-chip interconnect. These models must be developed for different levels of fidelity and speed, which will allow a principled evaluation of the appropriate model for individual explorations. These models will provide a solid baseline for higher level models (along the lines of the RAM model discussed above), and many intermediate levels that can be of use to various layers in the abstraction hierarchy.

Recommendation to NSF: Research should be encouraged to explore new models that cross discipline boundaries to understand the fundamental limitations and properties of power, energy, and thermal management.

Finding #4: The need for scientific optimization.

There was consensus among workshop participants that the development of short-term piecemeal solutions to particular engineering problems is not a long term solution. Many of these local optimizations at different levels actually interfere with each other. There needs to be a global framework so that the optimizations at the various levels work in concert. At this point, it is not at all clear what such a framework will look like. Evidence to date suggests that the mathematical optimization problems that will arise from such a framework, and from the models developed, will require novel optimization techniques. For example, the optimization problems that arise from speed scaling with energy and/or temperature objectives are commonly nonlinear convex optimization problems. The efficient solution of such problems will likely require novel algorithmic techniques. Ideally these techniques will be widely applicable at various levels of the systems hierarchy.

Recommendation to NSF: Research should be encouraged to explore new optimization techniques based in scientific approaches to exploit the analytical properties of power, energy, and thermal models.

Finding #5: The need for integration of power, energy and thermal management in curricula.

Most students are not introduced to the limitations of power, energy, and thermals until graduate school. There was consensus among workshop participants that practical and theoretical concepts surrounding power, energy, and thermal management be introduced early and often in computer science and electrical engineering curricula. One long-term goal is to teach students to reason about power, energy and temperature as naturally as current software engineers reason about the time and space used by computation. This will lead to the production of engineers that are better able to solve power related problems. Since the leaders in the field themselves are still developing the scientific framework underlying power management, these goals are far off. However, the development of an upper division undergraduate class that cuts across traditional boundaries (theory, systems, software, etc.) would be useful at this stage. Eventually, as the science matures, this material will filter down to core introductory courses.

Recommendation to NSF: The integration of power, energy, and thermal techniques and the associated formalisms into educational curricula should be encouraged. Research to develop new curricula focusing on metric development, model development, formal optimization techniques, and cross-discipline integration should be supported.

Finding #6: The development of a scientific community of power management

Power related issues are being investigated by almost all subdisciplines of computer science. But experts in these different subdisciplines are separated by cultural and language barriers that make communication and collaboration difficult. Encouragingly, the participants at the workshop showed a genuine interest in communicating with experts in the other areas. Further collaboration and communication would surely enhance the developments outlined above.

Recommendation to the NSF: Support further cross disciplinary workshops and collaborations. Provide specific encouragement for cross-disciplinary funding opportunities that researchers from multiple domains can target.

Finding #7: IT power management should look outwards as well as inward

As discussed in this document, there are many opportunities for the computing industry to reduce power consumption of IT components and to provide better compute performance-per-watt in the coming decade. However, energy has become a global issue and IT should have a large role to play looking outward – specifically, in reducing the energy footprint of many other industries. A few examples include smart building energy management, climate and weather modeling, smart power grids that incorporate alternative energy sources, and remote telepresence. IT has transformed many aspects of society over the last 20 years, and IT has the potential to provide additional transformative effects through smart applications of this technology.

Recommendation to the NSF: Encourage CISE researchers to actively seek opportunities for their research to have broad impact on energy issues across society.

Conclusions

A science of power management should unify power/thermal management techniques across a broad range of system components, physical and logical hierarchies, and control domains, and develop theoretical models of power-performance tradeoffs at multiple levels of granularity. The expected benefits of a science of power management are many-fold. Consolidating basic models for temperature, energy and power will give algorithm developers a common platform to work with. The development of a set of canonical algorithmic techniques will make it easier for new methods and protocols to filter into active use. All of these efforts will increase awareness within the scientific community that power is now a matter of central importance to both the developers and users of information technology which will in turn feed more progress.

Appendix A: Organizing and Steering Committee

Kirk Pruhs, U/Pitt
Kirk Cameron, Virginia Tech
Krishna Kant, NSF
Sandy Irani, UC/Irvine
Partha Ranganathan, HP
David Brooks, Harvard

Appendix B: List of Attendees

First Name	Last Name	University/Company	Category
Rajesh	Gupta	University of California, San Diego	Invited speaker
Daniel	Reed	Microsoft	Invited speaker
Larry	Smarr	University of California, San Diego	Invited speaker
Jeanette	Wing	Carnegie Mellon University	Invited speaker
Taieb	Znati	University of Pittsburgh	Invited speaker
Tarek	Abdelzaher	Univ. of Illinois-Urbana-Champaign	Participant
Divy	Agrawal	Univ. of California at Santa Barbara	Participant
Dave	Albonesi	Cornell University	Participant
Nikhil	Bansal	Walton Research Center	Participant
Michael	Bender	Stony Brook University	Participant
Ricardo	Bianchini	Rutgers University	Participant
Maciej	Brodowicz	Louisiana State University	Participant
David	Bunde	Knox College	Participant
Ali	Butt	Virginia Tech	Participant
John	Carter	IBM Austin	Participant
Chandra	Chekuri	Univ. of Illinois-Urbana-Champaign	Participant
Sangyeun	Cho	Univ. of Pittsburgh	Participant
Ken	Christensen	Univ. of S. Florida	Participant
Panos	Chrysanthis	University of Pittsburgh	Participant
Almadena	Chtchelkanova	NSF CISE CCF	Participant
Song	Ci	Univ. of Nebraska-Lincoln	Participant
Chris	Dafis	NAVSEA Warfare Center	Participant
Rajarshi	Das	IBM	Participant
David	Du	Univ. of Minnesota	Participant
Rajiv	Gandhi	Rutgers University	Participant
Rong	Ge	Marquette University	Participant
Chris	Gniady	Univ. of Arizona	Participant
Sudipto	Guha	Univ. of Pennsylvania	Participant
Sandeep	Gupta	Arizona State Univ.	Participant
Anupam	Gupta	CMU	Participant
Mor	Harchol-Balter	CMU	Participant
Howie	Huang	The George Washington Univ.	Participant
Wei	Huang	University of Virginia	Participant

Bala	Kalyanasundaram	Georgetown University	Participant
Jeff	Kephart	IBM Austin	Participant
Samir	Khuller	University of Maryland	Participant
Tracy	Kimbrel	IBM	Participant
Christos	Kozyrakis	Stanford University	Participant
Rakesh	Kumar	Univ. of Illinois-Urbana-Champaign	Participant
Jim	Kurose	Univ. of Massachusetts, Amherst	Participant
Hsien-Hsin (Sean)	Lee	Georgia Inst. of Technology	Participant
Vitus	Leung	Sandia Labs	Participant
David	Lowenthal	The University of Arizona	Participant
Jose	Martinez	Cornell University	Participant
Trevor	Mudge	University of Michigan	Participant
Kamesh	Munagala	Duke University	Participant
Jose	Munoz	NSF-OCI	Participant
Gopal	Pandurangan	Purdue University	Participant
Massoud	Pedram	USC	Participant
Sriram	Pemmaraju	Univ. of Iowa	Participant
Robert	Pennington	UIUC	Participant
Steven	Phillips	AT&T Labs	Participant
Qinru	Qiu	Binghamton University	Participant
Jose	Renau	University of Santa Cruz	Participant
Tajana	Rosing	UCSD	Participant
P.	Sadayappan	The Ohio State University	Participant
Prashant	Shenoy	Univ. of Massachusetts, Amherst	Participant
David	Shmoys	Cornell University	Participant
Mark	Smith	Univ. of Illinois-Urbana-Champaign	Participant
Ankur	Srivastava	Univ. of Maryland	Participant
Mircea	Stan	University of Virginia	Participant
Thomas	Sterling	Louisiana State University	Participant
Dimitri	Stiliadis	Bell Labs	Participant
Xian-He	Sun	Illinois Inst of Technology	Participant
Eric	Torng	Michigan State University	Participant
Josep	Torrellas	Univ. of Illinois-Urbana-Champaign	Participant
Eli	Upfal	Brown University	Participant
Bhuvan	Urgaonkar	Penn State Univ.	Participant
Peter	Varman	Rice University	Participant
Vijay	Vazirani	Georgia Institute of Technology	Participant
Sarma	Vrudhula	Arizona State University	Participant
Lisa	Zhang	Bell Labs	Participant
Feng	Zhao	Microsoft	Participant
David	Brooks	Harvard	Committee
Kirk	Cameron	Virginia Tech	Committee
Sandy	Irani	University of California, Irvine	Committee
Krishna	Kant	Intel	Committee
Kirk	Pruhs	University of Pittsburgh	Committee
Partha	Ranganathan	HP Labs	Committee

Christine	Chung	University of Pittsburgh	Student volunteer
Daniel	Cole	University of Pittsburgh	Student volunteer
Michael	Dinitz	CMU	Student volunteer
Roxana	Gheorghiu	University of Pittsburgh	Student volunteer
Sriram	Govindan	Penn State Univ.	Student volunteer
Varun	Gupta	CMU	Student volunteer
Sungjin	Im	UIUC	Student volunteer
Matthew	Johnson	City Univ. of New York	Student volunteer
Jian	Li	University of Maryland	Student volunteer
Benjamin	Moseley	Univ. of Illinois-Urbana-Champaign	Student volunteer
Panickos	Neophytou	University of Pittsburgh	Student volunteer
Barna	Saha	Univ. of Maryland, College Park	Student volunteer
Brian	Wongchaowart	University of Pittsburgh	Student volunteer

Appendix C: Detailed Reports by Break-Out Groups

Report from Group on Software

David K. Lowenthal
Department of Computer Science
The University of Arizona
dkl@cs.arizona.edu

Abstract
The SciPM workshop was held NSF April 9-10, 2009. The goal of the workshop was to identify issues in power management that could benefit from formal treatment. This report summarizes the ideas discussed by the Software breakout group along with our recommendations to NSF.

Introduction
Throughout the history of computing, software has always had formal underpinnings. For example, software systems contain algorithms whose characteristics are known to be good as the problem size scales, based on its order of complexity. All key areas of experimental systems have a theoretical backing. Operating systems has queuing theory and scheduling theory; compilers has automata and formal languages; networks has statistical distributions, and programming languages has axiomatic and denotational semantics.

In the last decade, power management has become an extremely important topic. The reasons for this include saving money, saving the environment, extending battery life, increasing reliability, increasing density, and staying within a fixed power budget. However, as the power management arena is fairly new, there is not a formal treatment of the area akin to the treatment of non-power aware computing (which we will denote traditional computing in this document). From the point of view of power-aware software systems, a formal treatment would help greatly in designing and implementing them effectively.

As an example, most power management schemes trade performance of one or more system components for reduced power. However, when performance decreases, the program takes longer. This could in turn cause the program to consume more energy than if no power management scheme was used. A theoretical framework for power management could conceivably expose the tradeoffs to software to avoid such situations.

Ideas Towards a Science of Power Management

The following items briefly describe ideas from the Software breakout group towards the goal of a science of power management.

- Creation of a power-aware abstract machine. Our first recommendation is to create an abstract machine with which to reason about power management. Such machines, of course, exist for traditional computing: the RAM model is an example. The idea is that a framework is needed to reason about power management schemes. Perhaps a power-aware abstract machine would contain everything a RAM has, plus some notion of power-scalable components.

  We note that parallel computing has been searching for the "right" abstract machine for some time. It involves a tradeoff between simplicity of algorithm design and fidelity of the result. The PRAM model is often too unrealistic to be useful (e.g., unit-time access to any memory location) and can result in poorly-performing parallel programs. A power-aware abstract machine needs to be simple to use, but cannot fall into the same trap as PRAM; perhaps the LogP family of models for parallel computing is a good analogy.

  Additionally, models for newer, heterogeneous machines are needed (e.g., Cell, GPGPUs, etc). These machines promise low power/energy consumption, but are notoriously hard to program.

- Study of power-aware algorithm analysis. Similar to abstract machines, algorithm analysis of traditional computing has a rich and extremely useful history with O(:::) notation, which tells us about asymptotic behavior of a given algorithm. There is a likelihood that certain algorithms lend themselves to lower power usage than others. Perhaps it is time for a new type of algorithm analysis, P (:::), which would encapsulate algorithm behavior from a performance and power standpoint.

  In addition, perhaps such analysis could assist in determining a crucial question when saving power—when performance is lowered to save power, how much is it lowered? While this question seems simple, it has not had a satisfactory answer to date.

- Education. We believe that computer science curricula, from the ground up, should reflect the new reality that power management is critical. Recently with the multicore revolution, several in the community have suggested integrating parallelism into the curriculum from introductory computer science courses all the way through

graduation. A similar approach may be appropriate for power management. Some questions that could be addressed are: what data structures and algorithms lead to better power characteristics, and what can compilers, operating systems, distributed systems, networks do to lower power?

- Conserving power at data centers and high-performance computing installations. Both types of installations consume significant power. For data centers, queuing theory has been applied to understand best how to limit power (e.g., by shutting machines off when job arrival patterns allow it). Control theory is likely to also be useful. In addition, accurate models of I/O can help utilize storage devices in a power-efficient manner. Generally speaking, optimizing data centers is nontrivial because of conflicting concerns between customers, operators, and government.

  High-performance computing can be viewed as a subset of data centers, but a subset that has additional performance-based constraints and, despite the relatively small user base, accounts for a disproportionally large power/energy cost. We also note that there is a history of innovations in HPC making it into the mainstream (e.g., parallelism). Here, whole-system methods are needed to lower power usage, from the processor all the way down to the power supply. Statistical methods based on large amounts of component sensor data may assist in this regard. Also, currently scalability is measured strictly as a performance-based metric; perhaps it is time for programs to be evaluated based on "energy scalability", based on some metric that incorporates power and performance. We note that architectures are already being evaluated in this manner; there is a "Green Top 500" in addition to the traditional "Top 500" supercomputers.

- Making users power efficient. It is important that users participate in power efficiency; however, users often want performance and do not care about power. Here, we can draw on several scientific disciplines; first, we can use statistical characterization to determine user behavior. This can lead to lowering component power; for example, dimming the screen if, probabilistically, it will not affect the user. It may also lead to smart defaults (e.g., should the wireless be on or off by default?). In addition, economic theory may be applicable to provide an incentive-based scheme to make users power efficient.

- Handling middleware interfaces. Middleware layers should be explicitly designed to incorporate power management decisions. These layers operate on possibly different time scales, so specific information needs to be passed up and down these layers. However, what information that should be is an open question. Here, middleware can potentially draw from the field of combinatorial optimization.

  Modeling may also assist with middleware in that there are many data dimensions (e.g., performance, power) and the data is needed in a short timeframe.

- Measuring effectively. Generally, measurement is needed so that the current state of the system can be determined. Challenges here include how to capture and process large streams of data with the appropriate frequency. More challenging is how to capture resource consumption in virtualized environments, which are the norm in data centers.

Recommendations to NSF

The software group makes the following recommendations to NSF.

1. There should be a new crosscutting NSF program to support progress towards a science of power management.

2. If there cannot be a new crosscutting program, there should be a focus area within the CNS/CCF/IIS core for a science of power management.

3. NSF should encourage the broader impact section of standard NSF proposals to, if appropriate, mention how the proposed work impacts power/energy.

4. NSF should consider constructing a hardware/software test bed for researchers interested in scientifically-driven power management research. Currently, researchers have to build their own power-aware infrastructure, which may drive away more theoretically-based researchers. The national labs are building a cluster for high-performance computing that will allow researchers to test new hardware and software technologies. It would be quite useful to have an analogous infrastructure for power research.

Report from Group on Software Science Challenges in Data Center Energy Management

Jeffrey O. Kephart (IBM)

Introduction

In this report, we outline some of the significant science challenges identified by one of the SciPM2009 breakout teams that discussed the area of software and middleware. Broadly speaking, there are two main categories of interest:

- Software and middleware have an important role to play in managing computing systems, of which data centers are one very interesting and important large-scale example, to specified goals, tradeoffs and constraints pertaining to performance, availability, energy, and other management concerns. This was the main focus of the breakout discussions.

- Software and middleware could be made more efficient and parsimonious in their use of computing resources, resulting in improvements in both performance and in power consumption. This avenue was not explored in the breakout sessions. However, during the Q&A following the software breakout session report, NSF Division Director Taieb Znati pointed out that power-aware compilers and tools that try to address software bloat are an important realm that warrants new exploratory science.

In the remainder of this report, we explore these two categories, identifying science challenges for each.

We interpret the term "science challenges" broadly, taking it to mean significant problems, solutions to which would likely yield a set of highly-cited publications in top-quality computer science (or other science) conferences, and would be likely to influence or be adopted into technologies that would be deployed widely in data centers and other computing systems, and in industry products.

Managing to specified goals, tradeoffs and constraints

It is natural to conceptualize the task of data center energy management as an optimization problem. Expressed this way, there are two major questions:

- What is the objective function?
- How do we optimize the objective function?

Thus there are two major areas of investigation, each of which has several associated science challenges. First, we must develop means for establishing what are the goals,

tradeoffs and constraints to which the data center[1] must be managed; and second, we must develop architectures and algorithms for managing to those objectives. We explore each of these in turn.

Establishing objectives

While there is some merit in trying to define standard metrics that go beyond MIPs/watt, and objectives that go beyond "maximize MIPs/watt", there is no one universally applicable metric or objective, and we must recognize and cope with that fact by developing means for establishing and/or eliciting goals, tradeoffs, and constraints.

It is worth noting that middleware is ideally positioned to be aware of many of the goals, tradeoffs and constraints, particularly at the application and service level, so many of the innovations in establishing objectives that we seek would find a natural home in middleware.

One challenging aspect of establishing objectives is that there are multiple sources of them, and many conflicting concerns that need to be resolved.

- **Customers** want their applications to run well. But what does that mean? What constitutes "running well" can be complex, and very dependent on the application and the customer. The broad categories of service attributes that are of greatest interest include performance, availability, reliability, and security. Within each of these categories, there may be myriad details. For example, performance may be based on application response time, or throughput, and the specification may include expected values or statistical distributions with accompanying time windows (e.g. within a 5-minute window at least 95% of all transactions should take less than 2 seconds, and the throughput must be sustained at a level of at least 100 transactions per second).

- **Data center operators** want to maximize net profit by satisfying Service Level Agreements while minimizing operating expenses that include energy costs and capital expenses that include the cost of new data center construction, which is influenced directly by power consumption considerations.

- **Government** may wish to reduce emissions, or ensure safe operating conditions for businesses and humans. They may try to enforce objectives (by introducing tax incentives, or defining standards or ratings akin to "Energy Star"). They may try to enforce constraints by enforcing regulations, or introducing markets for carbon emissions trading.

- **Device designers** may build into physical devices or firmware certain physical constraints and objectives that must be respected by the software stack.

[1] We shall use the term "data center" for convenience in this report, but please understand that the discussion may apply to other computing systems as well.

Some key science challenges that arise in this context include:

- **What combination of preference elicitation algorithms and interfaces based on human factors considerations and studies are best able to elicit individual objectives, tradeoffs and constraints from humans in each of these roles?** Humans often have great difficulty articulating their desires up front, but are better at reacting to observed behavior, which will likely necessitate innovative work on iterative preference elicitation. Such work could be informed by some studies in the literature of economics, decision theory, and human-computer interaction, and will likely entail cross-disciplinary work among these fields.

- **How and where are these objectives, tradeoffs and constraints to be represented?** There must be some translation from the way humans understand objectives into a form that is suitable for machine manipulation and calculation. Alternatives to be explored include a) combining the objectives into a single large .xml file and propagating them in various translated form through the system; and b) distributing pieces of the objective function throughout the system, and using agent-based or other forms of multi-lateral or mediated negotiation to resolve conflicts. This of course depends very much on the architecture and algorithms used to perform the optimization – the subject of the next sub-section.

- **How do we combine the objectives, tradeoffs and constraints from people in each of the roles identified above?** Is the collective welfare best satisfied by creating a single complex objective function formed by weighting each goal, or by prioritizing the goals, or through some other nonlinear combination of individual goals – and what tools are needed to establish a global objective function from the individual ones in this case? Or is it better to let the individual objectives be represented by software agents, and use game-theoretic or market-based techniques to determine the system behavior? In the latter case, new science is needed to explore whether existing market and/or multi-agent approaches can be extended, or whether fundamentally new approaches are needed for this new data-center context. Related to this, we anticipate new challenges in mechanism design.

Managing to objectives

While middleware is ideally positioned to be aware of (and elicit) application-and service-level objectives, it cannot control everything directly. How then can the objectives be conveyed to the rest of the software stack, to the hardware, and to the physical infrastructure? There are several key challenges in the realms of monitoring, modeling, architecture, and algorithms.

Monitoring serves several purposes. Control algorithms need to assess the current state of the system, and learning algorithms need to correlate past actions and environmental conditions with observed behavior to help develop models of system behavior. Monitoring is needed to detect when things are going awry. Furthermore, monitoring is needed to measure how much compute and energy resource has been consumed by various

applications, individuals, departments, or companies, to support providing the right incentives. Some key monitoring challenges include:

- What are the most effective ways to capture and process large streams of data with frequencies that are appropriate for the purposes to which they are being put?
- Capturing resource consumption data in virtualized environments.

Some key **modeling** challenges include:

- How can models be used to help transform information pertaining to objectives as it propagates up and down the stack?
- How can we learn good models of performance, power consumption, etc., given the complexity of the environment and the resulting high dimensionality of the data (which causes even a flood of data to become sparse), and also the rapid timeframe in which learning may need to take place? (This could also be regarded as an algorithmic problem.)

Some key **architectural** challenges include:

- For what control knobs should each level of the software stack (including middleware) be responsible for manipulating directly? We believe the answer has much to do with the inherent time scales on which each level is able to act: microseconds for firmware, milliseconds for the operating system, and seconds to minutes for various middleware depending on function. At the level of the physical infrastructure, the appropriate time scales may range from a few minutes to several hours.
- For each layer of the software stack, should it interact solely with its immediate neighbor "above" and "below" it, or is there merit in having a more tangled pattern of interconnectivity among layers?
- Can a multi-agent architecture in which self-motivated, semi-autonomous agents interact be effective for managing joint performance, power and other objectives? In such a case, what are the boundaries of the agents? Is it better for them to represent different machine groups, or different management disciplines (e.g. performance, availability, power), or different levels of the software stack?
- What essential information needs to be exchanged among layers of the stack? We believe one should strive to minimize the need for one layer to understand the inner workings of the others, so the terms shared in common by two layers should be chosen judiciously. Can the terms be derived from machine learning? How can information pertaining to objectives and constraints be conveyed to layers of the stack that do not understand the terms in which the objectives and constraints were expressed originally? (For example, response time and throughput are not meaningful to the OS.) A strict command hierarchy will surely not work, as "lower" layers may have inviolable constraints (e.g. on operating temperature) that must be respected despite demands made by "higher" layers.

Finally, several key **algorithmic** challenges that arise in the context of managing to objectives are expected to require advances in the fields of:

- Economics, mechanism design and game theory

- Decision theory
- Negotiation
- Machine learning (individual agents) in complex, high-dimensional environments
  - Learning models
  - Learning decisions
- Multi-agent learning
- Feedback control theory in systems with multiple interacting feedback loops
- Several additional areas identified by the breakout team led by David Lowenthal

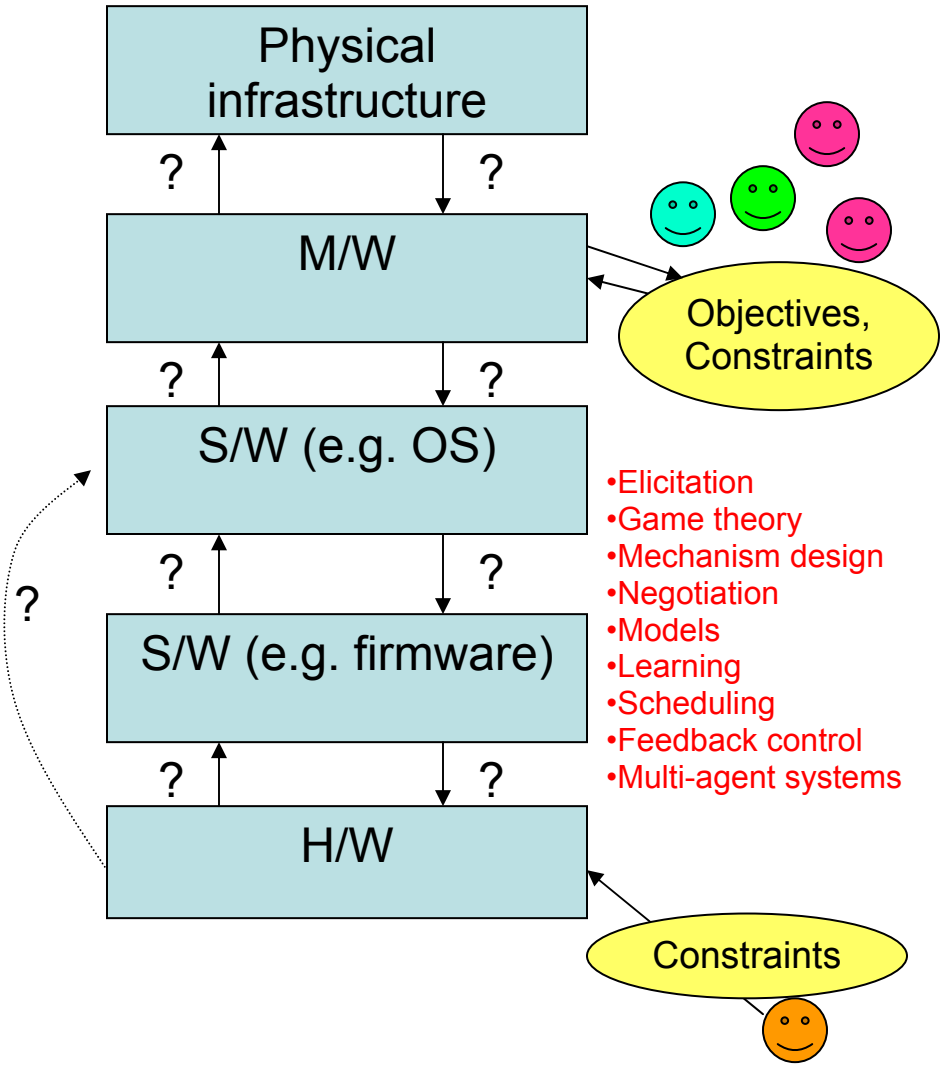

Figure 1. Role of software and middleware in the larger data center context. Some of the fields in which science advances are expected to be necessary are listed in red on the right side.

Tools for improving software efficiency

Tools for improving software efficiency are an avenue that we did not explore in our brainstorming sessions. In contrast to the questions we considered in our breakout session,

which centered on tradeoffs among performance, energy consumption, availability, etc., this is an interesting case where performance and energy considerations are aligned, rather than in conflict. In other words, any gains in software runtime efficiency that result in fewer instructions being executed for a given user experience result in faster performance *and* lower energy consumption.

One observation is that any techniques that can reduce the number of instructions required to perform a given function, and thereby lead to better performance, would already seem to have had a strong rationale for serious attention by the computer science community. Energy efficiency could be regarded as a side effect of achieving greater performance. If the prospect of enormous gains in computing performance has not motivated a lot of research in this area, might the prospect of greater energy efficiency provide inspiration where the prospect of better performance did not?

Report from Group on Hardware

John Carter (IBM)
Massoud Pedram (USC)

Vision

Develop (1) technologies that create energy-efficient devices, logic cells and memory, integrated circuits, architectures, and systems; (2) circuit solutions that enable efficient on-chip power distribution and conversion, energy storage and recovery, and energy-efficient clocking and signaling; (3) control methodologies that facilitate adaptive power/thermal management and speed scaling so as to minimize the energy consumption per computational work; (4) address the fundamental tradeoffs between energy, performance, and reliability.

Challenges

1. Investigate the fundamental tradeoffs between energy efficiency, performance and reliability for different types of information processing circuits and systems.

2. Investigate the role and value of heterogeneity in this 3-D tradeoff space, addressing issues such as the degree, type, and granularity of heterogeneity in designs.

3. Develop high-level, abstract models of energy efficiency (similar to big O notation or based on information theoretic concepts) that can be used by system and application software developers. Address model composition and information flow across different levels of design hierarchy and different domains.

4. Formalize the specification, translation and propagation of various top-level specifications and requirements to lower level design steps including RTL synthesis and physical design.

5. Investigate how to reduce power consumption of both end-user systems such as laptops and PDAs as well high end computing clusters such as data and switching centers subject to performance and reliability constraints. Do this while paying attention to issues related to validating and verifying designs, scalability of proposed solutions, and hardware cost.

6. Provide support and infrastructure for building prototype chips/systems to study low power design issues.

Advancements Needed:

1. What are the abstracts measures and models, as well as utility functions and metrics that are self-consistent, efficiently computable/executable, and which can be used for reasoning about fundamental tradeoffs between energy, performance, and reliability subject to various sources of noise/variability in today's designs? These models and metrics should be equally applicable to different levels of the design hierarchy (system, architecture, circuit, and layout), to different domains (hardware, hypervisor, operating system software, middleware), to different tiers (gate, module, chip, board, rack, and system) and to different systems (e.g., processing, memory, storage, networking, embedded designs).

2. What are the optimization methods and algorithms that radically expand the dynamic range of feasible/operable solutions in the aforesaid 3-D tradeoff space? What are the fundamental obstacles, if any, to achieving a nearly continuous solution space covering a wide dynamic range? How to analytically model and efficiently search for good alternatives (deep local minima) in this multi-dimensional space? How to design new computing, networking, and storage solutions that exhibit energy-proportionality?

3. What is the minimal set of basic hardware acceleration capabilities that can significantly benefit a large class of applications and how these will interface with the modern compute servers, network and storage? What role does heterogeneity play in achieving energy efficiency?

4. Can we achieve near lossless power conversion, delivery, and distribution in electronic circuits and system from the point of generation to the point of consumption? What factors (physical laws, parasitic-related, or cost-driven) determine or contribute to power losses during distribution and conversion? What limits are imposed on the performance and reliability of electronic circuits and systems if the available power is constrained in terms of its peak value or rate of delivery? What is the minimum set of demand shaping methods that can avoid service disruption or catastrophic failure when the supply becomes constrained or stressed?

5. How to continue functional scaling without physical scaling? What technologies (and possibly models of computation) would allow us to increase our information processing capacity in spite of the Moore's law coming to an end? How much can Shannon's law help by giving us higher bandwidth to utilize "distant" processing and "over-the-net" memory? How do we cope with quantum devices and stochastic models of computation? Which beyond-CMOS devices exhibit the desirable features of CMOS switches that made CMOS such a big success?

6. What are the key advances needed to enable full virtualization of hardware resources (organized into standard "compute" and "storage" servers, or disaggregated and arranged as pools of compute, memory, network bandwidth and storage resources) within a large enterprise computing environment? How many computing cores, how much memory and disk space, and how much network bandwidth are needed in order to energy-efficiently execute some application in the aforesaid 3-D tradeoff space under partial or full virtualization strategies?

7. What reduced subset of a system's functionality is essential to provide minimum acceptable level of service or to wake up a partially sleep system in order to provide full functionality on demand? What functions are amenable to successful reductions or wakeup-on-demand capability? What reduction functions are most suitable for which class of applications/services? How do we translate designer's intent into a set of essential (always awake or watchful of wakeup requests) and enhanced (potentially) functionality so as to achieve energy proportionality?

8. By how much and in what ways do the application type (e.g., scientific computation, transaction processing, database search), resource demands (memory vs. I/O, FP vs. Integer operations) and/or runtime characteristics (e.g., cpu bound vs. memory bound vs. I/O bound,) affect the energy-performance tradeoff of the corresponding code being executed on a given (statically provisioned) computer system? Knowing that the resource pressures and runtime characteristics of the application change as a function of the program phase, how should one go about modeling and dealing with these variability effects? What is the minimum set of dynamic provisioning capabilities and/or dynamically-provisioned additional resources that help greatly improve the energy-performance tradeoff solution?

Recommendations to NSF

Support development of abstract models, real or synthetic workloads, standardized test benches, and hardware test beds that can be used by researchers who are interested in energy-efficient and reliable system design and to encourage integrative efforts that produce a power/thermal-aware design flow and a design methodology that considers energy-performance-reliability tradeoffs early in the design process rather than merely point tools and techniques.

Help unify the diverse set of point tools and solutions for energy efficient and energy proportional designs into a unified framework and tool flow. Create a multi-university, collaborative research project addressing all aspects of the Science of Energy Governance from circuits and micro-architectures to embedded systems and data center design and operation. NSF may consider putting together an at-scale virtualized hardware test bed for

the architecture and computing communities. This test bed, for example, may comprise of a number of internet-connected enterprise computing facilities (**federated data centers**) placed and operated at a few select universities and made open to and remotely accessible by the wider research community for experimentation, data collection and analysis, and proof-of-concept demonstrations. The idea is similar to the Global Environment for Network Innovations (GENI) project.

Report from Networking Breakout Groups N1 and N2 Science of Power Management Workshop

April 9 and 10, 2009
Arlington, Virginia

Version 2.0 – April 23, 2009

Author/editor of this report:
Ken Christensen (USF)

Members of the breakout groups:

Group N1:
Ken Christensen
Panos Chrysanthis
Anupam Gupta
Sriram Pemmaraju

Group N2:
Michael Dinitz
Sungjin Im
Ben Moseley
Steven Phillips
Prashant Shenoy
Dimitri Stiliadis
Gopal Pandurangan
Lisa Zhang

Purpose and Scope

This report is intended to be "a concise write-up on the outcomes of the workshop" for the 2009 Science of Power Management workshop. This report reflects the views of the two networking break-out groups at the workshop. Any errors in this report are solely those of the author. The organization of this report is: Vision, Major Challenges, Scientific Advancements Required to Address the Challenges, Specific Recommendations for NSF, and Issues Not Articulated at the Workshop.

Vision

Communication consumes energy. This energy consumption needs to be understood in order to be able to find methods to reduce it. Power management as a science applied to networks must be able to address the full interdependency of energy and performance. **The vision is that networks should be as energy efficient as possible when providing the services that they are intended to provide.** This vision extends from application-specific, battery-constrained wireless sensor networks to the Internet in its entirety. Sensor networks

must be energy efficient to be feasible to deploy. The network equipment and hosts that comprise the Internet must be energy efficient to reduce their operational cost and minimize their CO_2 footprint.

Major Challenges

Central to a science is the ability to make predictions – predictions typically based on mathematical models. Being able to truly measure and predict energy-performance trade-offs is the major challenge to a science of power management for networks. **Developing models is the key to this understanding and to the ability to make predictions.** Models can enable a deep understanding of energy-performance trade-offs and will allow the research community to match theory to reality. Four key questions are:

1. Is modeling energy use an optimization problem, or something else altogether?
2. What types of models are needed to be able to come-up with new energy-efficient protocols?
3. Do we need to model for parallelizability?
4. What are the hidden (or currently unknown) energy-performance costs in real systems?
5.

Models of energy use would need to be created within a larger framework of models to fully understand the interactions of time, space, and energy. Models would need to strike a balance between tractability and applicability to the real work. Within the scope of model building is determining the correct inputs to be used – this itself is a major challenge.

A major challenge to reducing energy consumption of networks is finding new ways of designing protocols that support and enable energy efficiency. Current protocols are not designed for energy efficiency. One example of this is the reliance of many protocols and network applications on periodic keep-alive messages to maintain soft state. The necessity to generate and/or respond to such periodic keep-alive messages prevents hosts (and links) from going to sleep. Protocols need to be designed from a clean slate to get away from keep-alive messages and device-specific hacks. End-to-end protocols need to be designed to enable and deal with hosts, and/or components within hosts, that go into a sleep mode. A related question is how to provide a service when a lot of nodes or hosts (or subcomponents within a node such as antennas) are asleep? New design methods – such methods closely related to modeling – should support cross-layer optimization to address computation versus communication and latency versus energy trade-offs. This would address a key question on whether it is more efficient to do computation in the cloud (that is, moving bits from my host to the cloud and back) or locally in my host (that is, effectively sending power to me to enable local computation)?

A second major challenge is addressing energy metrics. Energy metrics need to be understood to be able to realize what architecture(s) would optimize energy use. A joule per bit metric may be too simple. Many energy costs are (or maybe) hidden. For example, what levels of user annoyance are acceptable in trade-off to energy use? Knowledge of user annoyance may be hidden to the analyst or modeler in many cases. Uncovering these hidden energy costs is a major part of this challenge. If hidden costs can be discovered and

understood, a cost-benefit analysis can be performed and new approaches to energy efficiency explored. For example, game theoretic approaches for allowing hosts to automatically take action as appropriate (that is, to a given energy-performance trade-off) might be possible. Also possible might be more collaborative approaches to acquiring and storing data – for example, one might choose to avoid one-to-one relationships and effectively "recycle bits" between multiple hosts and users.

A third major challenge is addressing new ways to achieve energy aware network design and traffic engineering. For example, are we better off using 50% of processing (or network) capacity 100% of the time, or 100% of capacity 50% of the time? This question arises in many contexts including utilization of network links, multicore processors, and data center servers. The answer depends on many things including the relationship between capacity and energy consumption. Answering this question could lead to optimal energy-capacity relations and the ability to design network equipment (such as IP routers and Ethernet switches) to be truly energy proportional – that is, no power consumption if no traffic.

Other challenges include:

1. The need to develop techniques for energy-optimizations of disparate network elements that are geographically distributed across multiple sites and built incrementally over time.
2. The need for global addresses to be able to wake-up a host anywhere and at anytime.
3. The notion that we need to move away from a pull model (that is, a client pulls data from a server) to a push model where a client can sleep at all times except when a server has data to send to it. Key to this notion is global addressability, which is (2) above.
4. The need to enhance the sockets interface to provide an interface to new power-related capabilities.
5. The opportunities to explore existing techniques such as tiered/heterogeneous architectures of wireless nodes, caching of data closer to a consumer, distributing applications, using approximate queries, and otherwise exploiting incomplete/partial knowledge of an environment as a means to save energy with some trade-off in performance.

A final major challenge is the broader view of user interfaces and user annoyance with respect to making things work "energy smart". **This challenge encompasses the overall perspective that it is regulatory, economic, and social behavior issues that ultimately drive energy efficiency decisions.** That is, there are obstacles beyond just having the technology to achieving energy savings. How should these kinds of non-technology problems be posed and addressed?

Scientific Advancements Required to Address the Challenges

A key direction – and one needing scientific advancements – is the creation of tools, test beds, and formal methods and models. These advancements are needed to address the challenges described in the previous section of this report. Scientific advancements are needed in areas that include:

- Exploring new formal mathematical modeling methods for analysis.
- Exploring new formal techniques for analysis and synthesis.
- Exploring how existing techniques, such as communicating FSMs, could be utilized for analysis and synthesis.
- Developing new simulation models with the ability to monitor and control (simulated) power use, and have the flexibility to experiment with new energy-efficient protocols.
- Creating new experimental test beds somewhat similar to the existing PlanetLab and SensorLab. These experimental test beds need to have the ability to monitor and control actual power use, and have the flexibility to experiment with new energy-efficient protocols.

Specific Recommendations for NSF

A specific recommendation to NSF is to include an energy-related component in a future CISE program solicitation. Results from NSF funded research in this area would very likely have significant and measurable intellectual and societal impact.

Issues Not Articulated at the Workshop

We can view energy efficiency and networks in two contexts, 1) energy efficiency "of" networks, and/or 2) energy efficiency "by" networks. The latter view might include remote monitoring and control of physical infrastructure (such as smart grids), substitution of virtual travel for actual travel (that is, moving bits instead of atoms), and so on. The workshop focused mostly on energy efficiency "of" ICT and not so much on "by" ICT. If we consider the latter as a significant and relevant opportunity, we may take a different view of – and have different requirements for – energy efficiency and networks.

Report from Group on Storage Systems

Peter Varman (Rice) and David Du (Minnesota)

Vision Statement

The volume of stored data continues to increase at an alarming rate to accommodate new data types (e.g. HD streams), comply with legal requirements to retain data in perpetuity, and meet the demands for high availability and reliability using redundancy. Storage farms continue to become bigger, overwhelming potential gains from server consolidation and consuming a significant fraction of data center power. However, the daily access rate of data increases at a slower pace, especially for persistent data, creating the potential for energy savings in storage systems.

A multi-faceted approach to power and energy conservation in storage systems is described. This involves: (i) leveraging emerging technologies; (ii) developing models, algorithms, and analysis for power management based on a fundamental understanding of I/O applications, QoS requirements, and workload characteristics; and (iii) dealing with and exploiting redundant and geographical dispersion of data.

Challenges and Necessary Scientific Advances

A. Technology Driven Challenges:

Table 1 shows the power consumption characteristics of several storage devices, and indicates the potential of Solid State Disks (SSDs) in reducing storage power. While SSDs can potentially provide performance and power advantages, identifying cost-effective storage architectures to exploit their latent advantages is a significant open problem.

A1: How best to integrate flash-memory based Solid State Drives (SSDs) into the existing storage hierarchy for optimum power and performance?

Since the SSD lies on the storage path between main memory and disk, it may be envisaged either as a front-end for the disk-based storage subsystem, or alternatively as a back-end for main memory. In the first view, the SSD appears as a large disk or buffer cache controlled by OS software; the latter treats the SSD as a large main memory that is addressable by the processor, with main memory DRAM acting as a processor cache for the SSD.

	Approximate Power Consumption	mW/GB

DRAM DIMM Module (1GB)	5W	5000
15K RPM Drive (300GB)	17.2W	57.33
7.2K RPM Drive (750 GB)	12.6W	16.8
High Performance SSD (128 GB)	2W	15.6

Table 1: Power Consumption of Typical Storage Devices
Source: Flash Storage Today, ADAM Leventhal, ACM Queue, 2008

While SSDs are a viable alternative to hard disk drives (HDDs) in certain environments (e.g. laptop computers), it is unlikely that they will totally replace HDDs anytime soon. Power management strategies must continue to address mechanisms to reduce storage power consumptions in HDD-centered storage centers, while remaining cognizant of the increasing role played by Flash SSDs and other non-volatile memory based devices.

A2: Develop Dynamic Power Management schemes that trade off performance, energy and reliability for hybrid (HDD and SSD) data storage centers.

Some of the characteristics of HDDs that make DPM challenging include the following:

- Lack of multiple power states in commodity HDDs restricts DPM strategies
- High ON/OFF switching latency and power draw during the transitioning makes fine-grained power mode switching impractical
- Reliability concerns with frequent power cycling
- Extreme sensitivity of performance and power requirements on workload locality

Scientific Advancements Required

- Modeling of SSD-based storage devices and hybrid storage architectures to assess their power/performance characteristics for different classes of workloads. The problem is compounded by the lack of standardized interfaces and variability among multiple proprietary FTL (Flash Translation Layer) implementations. Progress will benefit from open standard interfaces to permit Operating System intervention and optimization.

- Novel approaches to storage DPM based on coordinated control of ensembles of storage devices.

- Operating System models/mechanisms/data structures/algorithms to optimize power in the presence of device constraints. These include:

    - Energy Aware File Systems: Organization, Meta Data management

- Energy Aware Storage organizations (e.g. Clustering, SSD caching, Prefetching)
- Active Data Placement for energy conservation (for example dynamic reorganization)
- Fundamentally new data structures for energy efficient update and data access
- Workload Intervention and Redirection
- Prediction and pre-fetching data schemes for energy saving

B. Abstract Metrics, Models and Mechanisms

A major issue is to incorporate energy/power as a first-class citizen of QoS specifications in storage applications. There are numerous challenges ranging from identifying the appropriate metrics, modeling, and designing mechanisms to optimize these metrics.

B1: What are the appropriate power and performance metrics in I/O and storage systems?

It is important to develop metrics that allow cross-comparison of solutions in the power-performance design space. Some natural questions in this regard include: Are there analogs to metrics like the energy-delay product that have found use in other domains? How do you compactly model and articulate an energy-friendly workload? Is the number of I/O operations performed a suitable surrogate for the energy consumption? If so, under what circumstances? How does the type of I/O (e.g random versus sequential, read vs. write vs. synch) affect power? Do we have sufficient understanding to distinguish first-order effects from secondary or tertiary effects in terms of workload and device characteristics?

B2: Designing Energy-aware QoS Performance Models

- QoS models that incorporate power consumption in pricing SLAs to encourage clients to provide energy-friendly workloads, and guide workload reshaping for energy conservation at the server

- Energy aware models for data access methods and out-of-core algorithms in backend database systems

B3: Energy-efficient Scheduling Algorithms, Data Access Methods and Data Structures

- In data center environments how do we provision and schedule I/O resources to maximize energy efficiency? How can we explicitly leverage the inherent advantages of statistical multiplexing and apply it to storage workloads sharing data center resources? How do we capture the locality-sensitive nature of storage devices and their power modalities? What are the alternatives?

- Can we design energy-aware and energy-efficient indexing schemes and access structures for supporting data intensive and database applications?

Scientific Advancements Required

- Fundamental understanding and modeling of the relationship and sensitivity between algorithms, workloads and power consumption.

- What is the appropriate abstraction level for power models? Is it sufficient to work with a functional form (e.g. convex function)? Functional relation (e.g. cubic dependence)? Or do we need a detailed operational model?
  - What fundamental algorithm characteristics affect power consumption significantly?
  - What workload characteristics (e.g. burstiness) affect the power performance tradeoff?

- Algorithmic methods to control the power/performance relationship by provisioning, workload shaping, and resource scheduling. Challenges include:

  - Storage workloads tend to be bursty and have stringent response time requirements
  - Peak requirements are many times long-term average rate
  - Servers/disks are not amenable to fine-grained power mode switching to respond to bursts

C. Leveraging Information Redundancy and Geographic Dispersal

Reliability and availability requirements for stored data require the use of redundancy at several levels within and across data centers. Data replication or erasure codes are used to recover from device failures or latent sector errors in the storage array, the data center, and across data centers (for disaster recovery and availability purposes), and provide opportunities for load balancing and optimized access.

C1: Fundamental tradeoffs between power, performance, and reliability for optimizing storage and data access both within and across data centers using redundancy

C2: Models and analysis of energy-aware workload distribution on a global scale

C3: In sensor networks or ubiquitous computing, modeling data storage, access and transmission for energy savings

Scientific Advancements Required

- Decentralized/distributed QoS scheduling algorithms that optimize energy and performance
- Hierarchical redundancy schemes combining replication and erasure coding on a global scale, that permit adaptive reconstruction of the data
- Energy efficient schemes for moving large volume of data from one place to another via Internet (data dedupe)
- Adaptive algorithms (centralized and distributed) that iteratively decide on what parts of the input are realized and what parts of the input are influential in computing the answer (loopback control)
- Energy efficient tradeoffs between storing multiple copies of data, re-computing data and delivering data via networks.

D. Idealized Limit Studies

D1: Developing models for energy cost of data life cycle

D2: Systematic design of codes that simultaneously optimize reliability and power

E. Recommendations for NSF in terms of support of research, infrastructure, education and workforce development.

1. For research support, a new program on power management will be great. If this is not possible, the issue of power management should be clearly included in the solicitation of a number of existing programs in CISE. All three divisions, CCF, CNS and IIS, have relevant programs but currently do not have power management included explicitly in the scope of funding.

2. As for infrastructure, we are lacking a large scale test bed for some of the research issues that have to deal with scalability. The research on individual components can be relatively easy to set up. However data center scale research are not available for systems type of research work. A funding from MRI to set up a shared test bed can be very helpful in this regard.

Report from Group on Physicals

Moderator: Ricardo Bianchini (Rutgers)

Scope

The scope of the discussion in the physicals group involved a very broad range of areas in computer system design and power management (including energy and thermal management): power supplies, materials, packaging, architecture, enclosures, cooling, and even the electricity grid. Much of the discussion centered around data centers, but we tried to keep smaller scale computer systems in mind as well. Throughout the discussion, we focused on the advances and innovations that would be required in these areas to transform power management into a more formal and scientific endeavor.

Vision statement

We believe that power management is currently far from where it needs to be. We envision a future in which (1) we achieve a more formal understanding and representation of power and reliability issues; (2) we produce more accurate static and dynamic predictions of future power (especially thermal) and reliability behavior; (3) we enable the use of formal optimization techniques that can minimize power consumption and/or maximize the benefits of power management without harming reliability; (4) we can more easily integrate new low-power (or at least power-aware) technologies into our prediction and optimization frameworks; and (5) we can more easily repeat power (especially thermal) management experiments within and across research groups.

Major scientific challenges and required advances

The research community must advance in many directions to turn this vision into reality. Specifically, innovations are needed in the following areas at least:

1. Analytical models:

They are needed to better understand and represent power dissipation, thermal behavior within different enclosures (from cellular phone form factors to warehouse-scale data centers), cooling of different types and at different scales, reliability of hardware components, and battery discharge and efficiency. These models should enable us to predict future behaviors system-wide and optimize the power management accordingly. Obviously, all models must account for performance as well.

Existing models fall short in many ways. The power models are typically concerned with a single component (usually the CPU) and embody many simplifications. For example, the effect of I/O, virtualization, and multiprogrammed CMPs on system-wide power consumption has not yet been fully modeled.

The few previously proposed thermal models are also quite restricted. For example, Computational Fluid Dynamics models have been used for data centers. However, those models are static and cannot be used to study dynamic thermal management. New approaches to cooling, such as liquid, spray, and free cooling, will require new thermal models.

To make matters worse, the effect of temperature on reliability is also poorly understood at this point. For example, existing reliability models for hard disks have been called into

question by Google. In addition, the impact of free cooling and high-temperature computing on reliability must be understood before these novel approaches can be widely used.

Finally, new hardware technologies, such as 3D stacking and solid-state storage, will also require new power, thermal, and reliability models. We do not know of any existing efforts in these directions.

2. Cross-area, cross-domain, and cross-tier interactions
The "physicals" aspects of power management span multiple areas (e.g., materials, enclosure design, and cooling), multiple domains (horizontal sections of computer systems), and multiple tiers (vertical sections).

Today's systems suffer dramatically from a poor integration of their power managers. For example, in a server cluster, each CPU, each operating system, and a cluster manager may all independently try to manage power consumption. In fact, it has been observed that such independent managers may drive servers to shut down, as they make conflicting management decisions. Similarly, it is possible that independent managers acting in different horizontal sections of the system make conflicting decisions.

Today's systems do not manage power and cooling in a coordinated manner either, losing opportunities for potential energy reductions and reliability improvements. For example, air flaps (in a server or blade enclosure) can be redirected or floor tiles (in a data center) reconfigured, according to predicted workload/power behaviors and their effect on enclosure temperature.

Clearly, we need to develop a better understanding of the interactions between these areas, domains, and tiers. Composable models and frameworks may be one way to achieve this understanding. Moreover, we need to develop formalisms and methodologies for management roles and interfaces, as well as cooling designs that can produce a better integration and/or coordination of all managers.

3. Formal optimization techniques
Current approaches to power management either rely on simple, single-variable feedback control or on (even simpler) heuristics. Formalizing behaviors and interactions should facilitate the use of more sophisticated formal optimization techniques that are being utilized extensively in other areas. Two particularly important directions here are the optimization of multiple variables at the same time and the coordination of multiple optimizing agents/controllers.
Mastering and exploiting certain formal optimization techniques, such as game theory, can be highly beneficial in many broader ways. For example, these techniques can be used to optimize the electricity grid as a whole, by optimizing the supply/demand of electricity across the grid. They can also be used to create a new incentive structure for the main actors in electricity consumption (power plants, utilities, and consumers). Finally, formal optimization techniques can be used to mathematically optimize the distribution of load across data centers, according to energy sources, ambient temperatures, electricity costs,

time zones, etc. Ideally, we would like to enable the integrated computer-aided design and optimization of entire computer systems, their workloads, and their supply of power.

4. New technologies
A number of new technologies are being created in many areas, such as materials (e.g., high-temperature materials), cooling (e.g., free and liquid cooling), and power supplies and storage (e.g., smart and reconfigurable power supplies, low-loss and green power storage). We need to study these technologies and assess their ability to reduce power consumption and/or improve power management. We also need to develop models and approaches for incorporating these technologies into power management frameworks.

5. Methodologies for scale-down and repeatability
The ability to repeat experiments is one of the cornerstones of any scientific endeavor. However, computer science is still in the Stone Age in this regard. Today, system-wide experiments are very difficult (if not impossible) to repeat exactly, since many aspects of the experiments are hard to control. For example, thermal management experiments with clusters of servers can be disturbed by the simple act of opening the door to the machine room. Furthermore, software behaviors can change across experiments due to intrinsic non-determinism, e.g. the asynchronous execution of operating system daemons and interrupt handlers, slightly different I/O device timings, or multithreading. We desperately need to design frameworks, test beds, and methodologies for experimental repeatability.

Furthermore, we may want to experiment with systems (e.g., warehouse-scale data centers) that are too large to reproduce completely. For these cases, we will need to develop methodologies, frameworks, and actual software for scale-down and result extrapolation.

Recommendations for NSF

From the discussion above, it is clear that many scientific advances are required in modeling, formalisms, methodologies, frameworks, and new technologies for power management. NSF can drive investigation towards these issues by targeting one or more of them with specific calls for proposals. The most obvious would be a general call on the science of power management. In this context, NSF should probably explicitly require a focus on cross-* integration/coordination, as well as on concerns such as experimental repeatability. Another possibility would be an infrastructure call seeking to create realistic, scaled-down test beds (and the required software for sharing them with groups across the country) for data center research