NSF workshop on Science of Power Management

Breakout session on software/middleware

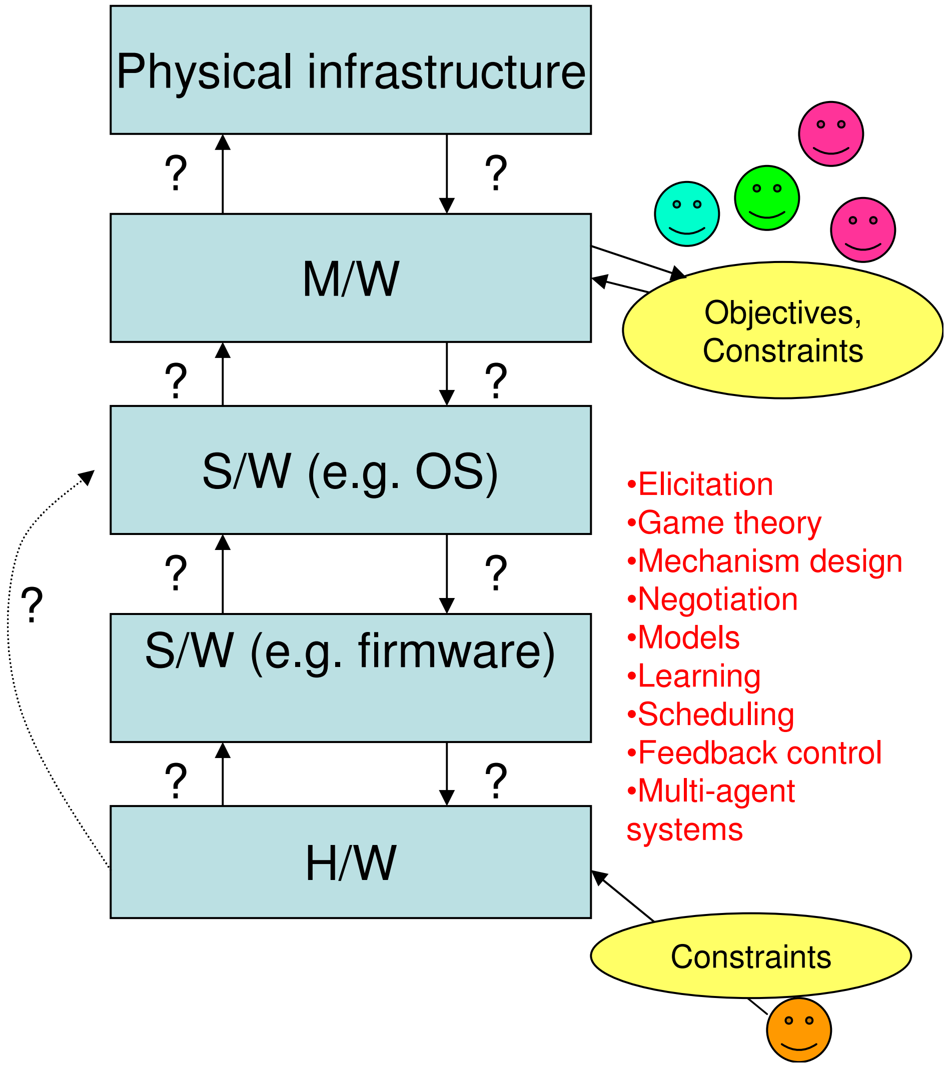
Jeff Kephart (IBM Research)
Maciej Brodowicz (LSU)
Varun Gupta (CMU)
Bala Kalyanasundaram (Georgetown)
Krishna Kant (NSF)
Rajiv Gandhi (Rutgers at Camden)

Overview

- The software/middleware stack can play a critical role in
  - *Eliciting* the objectives and constraints of the system/data center, including energy
  - *Satisfying* the objectives and constraints of the system/data center, including energy

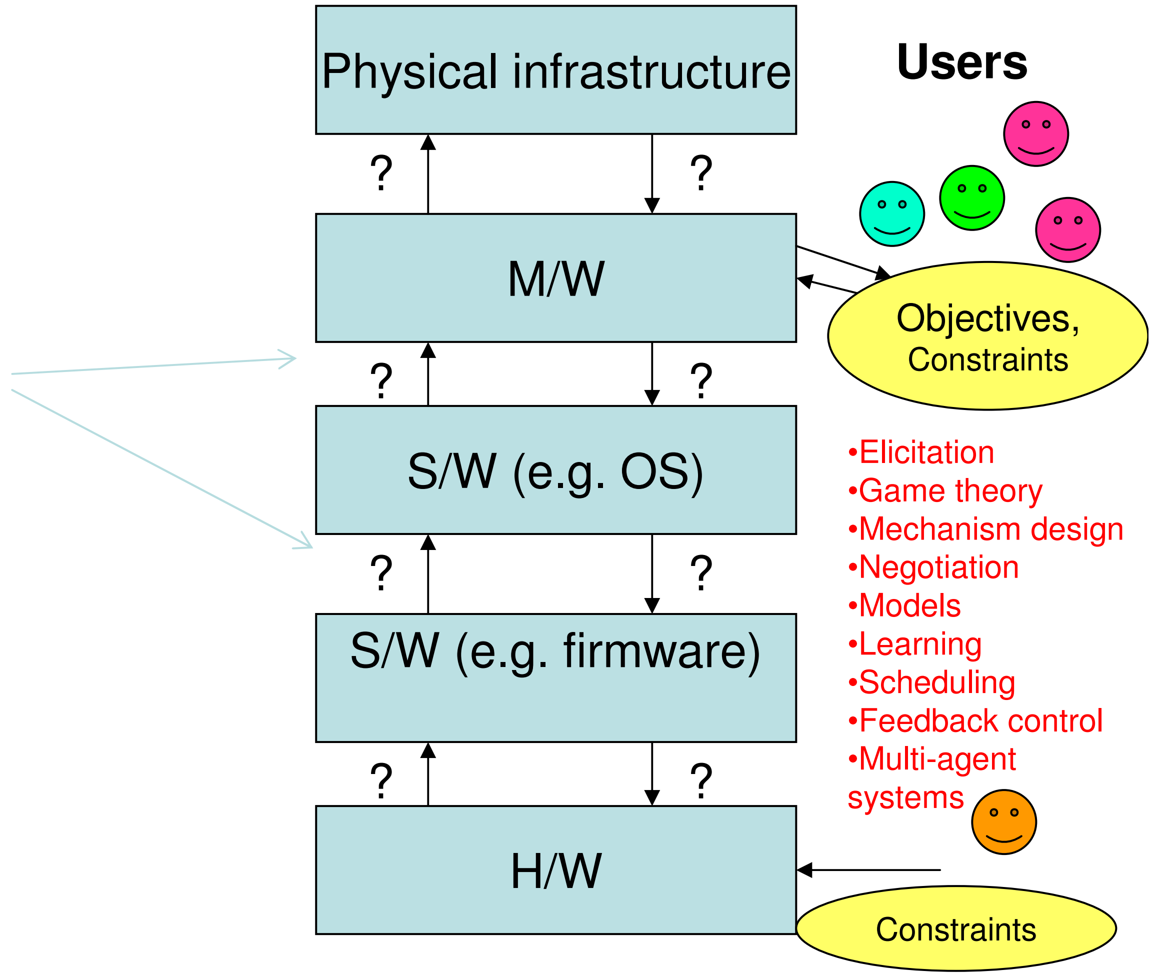
Software/Middleware Challenges

- The software/middleware stack can play a critical role in
  - *Establishing* the objectives and constraints of the system/data center, including energy
  - *Satisfying* these objectives and constraints
- Establishing objectives
  - Many concerns (customers, administrators, government)
    - How to elicit them? (Iteration might be needed)
    - Metrics must include performance, availability etc.; not just energy
    - How to combine and represent them?
    - Where are do they reside?
      - Possibly decentralized, leading to mechanism design questions
    - Also, constraints from hardware
- Managing to objectives
  - Middleware is ideally positioned to know the high-level objectives
    - But can't control everything directly
  - Key architectural questions
    - What does each level control directly?
      - Temporal and spatial scales largely determine this
    - How does mware influence behavior of other levels, e.g. by propagating objectives?
    - What essential information needs to be exchanged across layers?
  - Models of performance, energy consumption, latency, etc. are critically important
    - Needed to translate objectives across layers of stack
    - How to learn them?
  - Monitoring is critically important
    - Needed for control and accounting (for incentives)
    - More work on this needed in especially in virtualized environments



Software/Middleware

Power-aware abstract machine, Middleware interface



- Elicitation
- Game theory
- Mechanism design
- Negotiation
- Models
- Learning
- Scheduling
- Feedback control
- Multi-agent systems

Science of Software/Middleware

- Define power-aware abstract machine
  - Motivation: RAM, Log P, DAM, Cache-Oblivious
  - Algorithms/data structures => energy-efficient
  - Framework for metrics
  - Understanding of power/performance tradeoffs
  - Understanding of heterogeneous architectures
- Create interface for middleware
  - Combinatorial optimization to determine what information should be provided to different software levels
- Making users power efficient
  - Statistical characterization of user behavior
  - Economic incentives for green computing/smart defaults
- Optimally configuring large installations (HPC, data centers)
  - Queuing and control theory
  - Statistical methods given massive amounts of component sensor data

Backup for Software Group---Jeff

Objectives and Constraints (I)

- A first set of challenges has to do with establishing and representing the data center objectives and constraints
- One standard objective function imbedded by designers won't suffice
  - not just MIPS/watt !!
  - The objective function(s) and constraints must emerge somehow from the joint objectives and constraints introduced by several different human parties plus physical device constraints
- Several parties may be interested in establishing objectives and constraints
  - Administrator (IT) wants to maximize payments by satisfying SLAs for performance, availability, reliability …
  - Administrator (Facilities) wants to minimize infrastructure costs and operating costs including power
  - Customers want "good service", however they may want to define it
  - Government wants to reduce emissions, ensure safe infrastructure, … (and maybe wants to achieve this by introducing constraints, or influencing the objectives through incentives/taxes)
  - There may also be physical constraints and objectives pertaining to device characteristics
- What are the metrics/attributes of interest to these parties?

Objectives and Constraints (II)

- How to elicit objectives and constraints from each party?
  - The specified objectives will pertain to more than just energy – also performance, availability, reliability, security ….
  - It may not even be possible to do this up front
    - Humans may not want to, or be able to articulate them up front
    - Iteration may be required. People could experience service and provide feedback on the fly, and system may need to learn or infer preferences from this feedback
    - Common defaults will be important, or a small set of fixed choices can be offered (high performance, high power savings, something intermediate)
  - Various preference elicitation techniques from economics, etc. may be valuable
- How to combine these objectives and constraints across various parties?
- How to represent them (and where)?
  - Can they be combined into a single expression that gets propagated down the stack, or …
  - Do pieces get distributed throughout the system (e.g. in agents with internal utility functions that negotiate with one another)
    - Notions of game theory and mechanism design are likely to be of value in this context

Objectives and Constraints (III)

- A special property of middleware: it's in the middle!
- It is in a good position to be *informed* about the objectives of users, administrators, …
  - It can interact with user to elicit objectives (performance, availability, …)
  - It can interact with data center physical infrastructure to learn of physical constraints
- It is in a good position to *act upon* these objectives, and orchestrate how the data center satisfies them
- But it cannot directly control all of the energy saving mechanisms!

How can sw/mw control systems in accordance with objectives and constraints? (I)

- There are several key questions of architecture
  - Several levels of software stack, from driver up to workload management
  - Need to consider as well how these collaborate with the hardware and physical infrastructure
  - Is it a strict hierarchy of levels, each connected only to the one above and below? Or more of a tangled hierarchy, e.g. should BMC ever communicate directly with workload manager?
  - Multiple autonomic/ous systems interacting – what is the best architecture; how to attain objectives; how to avoid undesired emergent effects?

- For each level, we need to consider
  - What is it best qualified to control directly?
    - There are natural timescales and spatial scales associated with each
  - What can it influence control indirectly through interactions with other levels of the stack?
    - E.g. middleware requests application priority from OS, or … what? What language does it use
  - What is exchanged with the other levels above and below
    - Is this a command hierarchy? Probably not.
    - What are right interfaces – what can each level demand or request of the next one down, and what information or requests can be propagated from lower level to upper level?
    - What are the *essential* pieces of information that need to be exchanged – just enough, but not too much
      - Maybe we can learn this from statistical learning
    - Lower levels can say I need something, I'm having trouble, here's how it would be for me if only you could do this. Sorry - Or I can't do that. Meltdown scenario – got objectives from above, but some are built-in and we have to satisfy. So lower levels have to have a way to talk back. – sorry boss, no can do.
    - OS shouldn't tell hardware which p-state to go to. Just say what is the latency I can tolerate. Then something figures out the p-state. Lower levels need the flexibility to do things as it can. Also lets the lower level innovate.
    - What feedback is provided by lower level? Direct message to high level, or observation of performance, etc. experienced. Feedback can be used to generate models: when I ask for x, I experience y -> y(x).

How can sw/mw control systems in accordance with objectives and constraints? (II)

- *Models* play a key role
  - How do we create these models?
    - Learning, translating expert advice, standard models that apply across wide
  - How are they used to connect one layer's understanding of the world with the next layer's understanding of the world?
    - E.g. use them to transform utilities from one level to another

Virtualized environments

- In virtual environment – important to be able to measure and control VM power consumption? (generally true for applications)
  - Important for maintaining proper incentives
  - Important to detect that something is going awry

MW/SW to physical infrastructure

- Instrumentation of physical infrastructure – is it adequate?
  - What information needs to be exchanged
  - Who is in charge?
- Temperature/power-aware scheduling
- Do we need additional theoretical improvements?
  - Timescales are slow – do feedback control and other techniques work at the slow time scales, or do we need something different?
  - Is diversity of timescales a good thing (if enough separation), or does it make it hard
  - Power supply efficiency is low if loaded lightly; efficiency goes up as loaded more. Different phases that can be switched for tens to hundreds milliseconds. Also phase shedding voltage regulators. So exchange between infrastructure and mware can be useful.

Miscellaneous

- What about failures of one level of the hierarchy – fault tolerance?

Backup

Software/Middleware

- New science to address intelligent power/thermal management from driver to distributed data center level
- Areas might include
  - Power/perf tradeoffs (including complexity and bounds)
  - Cross-domain and multi-level controls
  - Metrics, measurements, and control in virtual environments
  - Hw-Mw interactions and Mw-physicals interactions and smart power/cooling/load control
  - Interactions between consumption and supply sides

Outcomes

- Generate list of major outstanding issues and potential approaches
- Multiple subgroups within a group may come up with different and sometimes conflicting items
- Group level meetings expected to
  - Resolve conflicts (to extent possible)
  - Bubble up most significant issues to top

Discussion

- Are power control mechanisms opening a new avenue for hackers to cause mayhem in systems?
  - No one talks about this aspect
- Models of how controls+environmental variables -> service-level, …
- Models of hardware capabilities are needed by various levels

Backup for Software Group 2---Dave

Problems/Solutions

- How to manage power in server farms?
  – Queueing theory/control theory
- How to use external storage devices in a power-efficient manner?
  – Algorithms and Data Structures
- How to determine performance degradation when using power-saving mechanisms, and how to develop more detailed and accurate models?
  – Statistical Methods, Computational Models

Problems/Solutions

- How to manage power and temperature in extreme scale HPC installations?
  - Statistical methods based on sensors
- How to define structure and requirements of middleware in terms of power and thermal constraints?
  - Combinatorial optimization to determine what information is pertinent

Problems/Solutions

- Scheduling for energy efficiency
  - Scheduling and queueing theory
- Metrics for green computing
  - Scientific backing
- Power-efficient routing

- Develop theory for cooling/minimizing heat
  - Integration between CFD models and computational techniques

Problems/Solutions

- How can software recognize and adapt to heterogeneity in new architectures?
  - Model to predict performance and power on a heterogeneous system given a workload
- Develop a theory of power complexity
  - Make power a “first class object” in education
  - Take into account in algorithm design/complexity
  - Take into account in data structure design
  - Understand power implications of HPC algorithms
  - Find a model for power efficiency that is as simple to use as possible and as accurate as possible
  - Economic models for incentivizing energy efficiency
- Software should understand efficiency of power supply

Problems/Solutions

- How do we “save the PCs”?
  - Statistical methods for characterizing user behavior, smart ways for studying default behavior, economic models for incentivizing energy efficiency

Problems/Solutions

- How do we “save the PCs”?
  - Statistical methods for characterizing user behavior, smart ways for studying default behavior, economic models for incentivizing energy efficiency