SASO 2008 – Workshop on Business Applications and Potential of Self-adaptive and Self-Organizing Systems

Predictable, Longevous Services

October 21, 2008

Jerry Rolia

Principal Scientist

Adaptive Infrastructure Lab, HP Labs

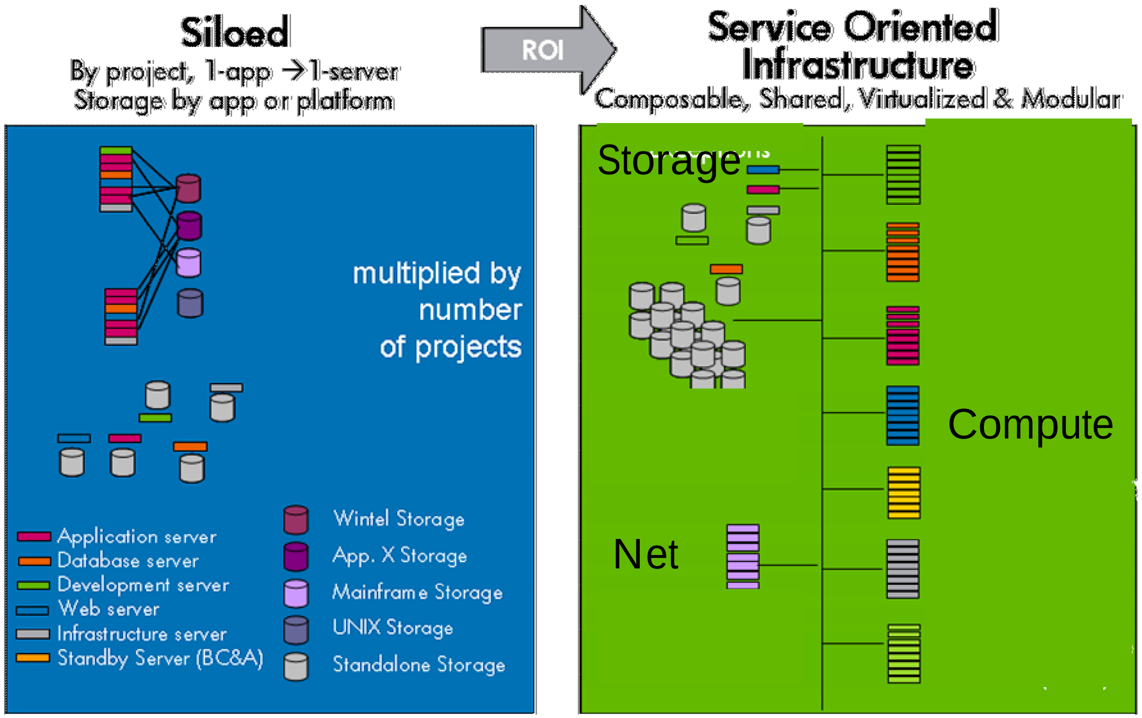
Jerry.Rolia@hp.com

Outline

- HP Labs self-management examples from next generation data centre
  - Power mgmt
  - Shared IO mgmt
  - Compute capacity mgmt
- Self-management and applications in cloud resource utilities
- Challenge: predictable, longevous services

Next Generation Data Center (NGDC)

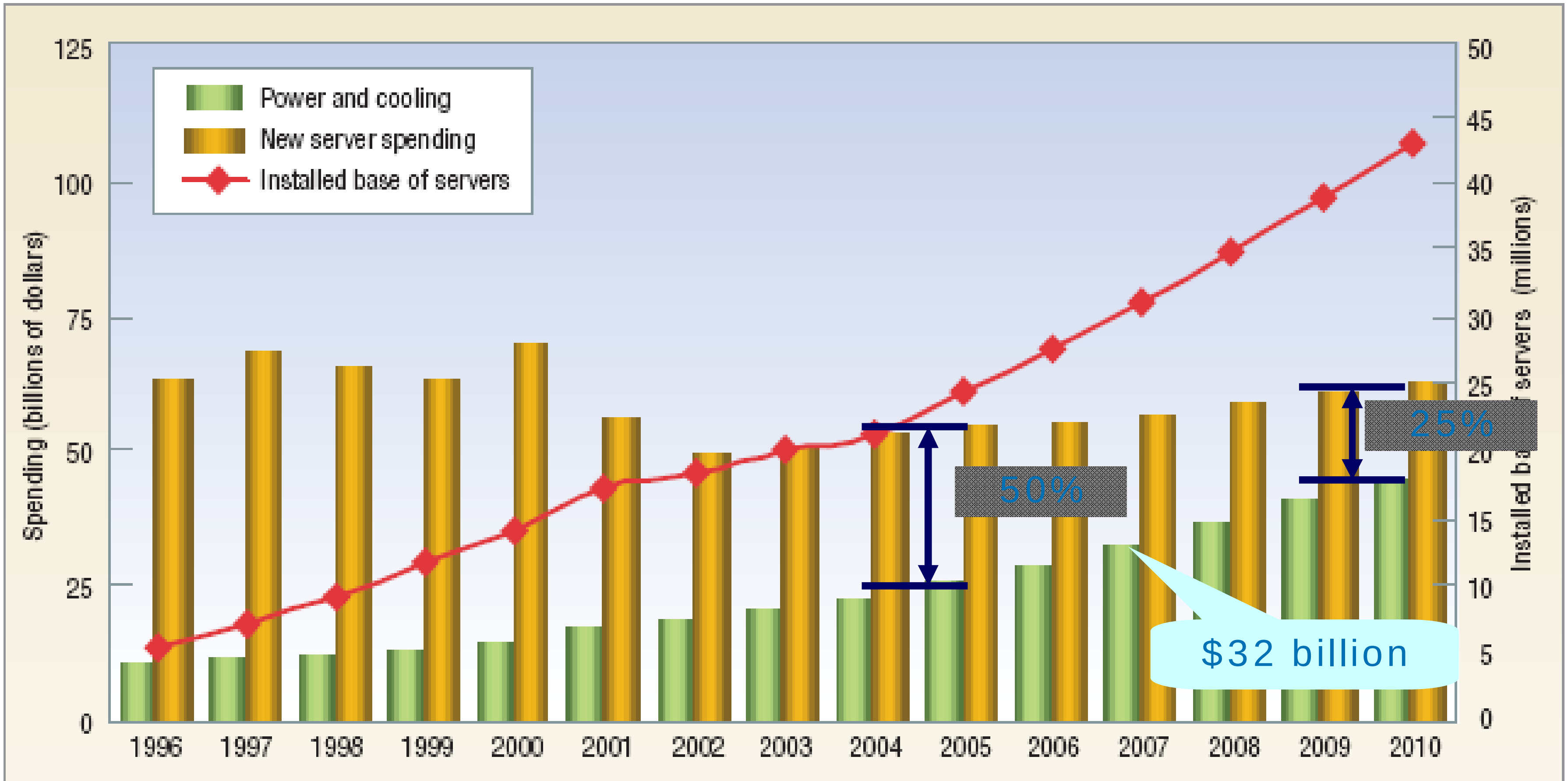
- Shared resource pools composed of physical and virtualized servers, networking, storage
- Flexible/programmable resource provisioning
- Consolidating multiple application workloads per server



What are examples of self-management issues for NGDC?

Power mgmt…

"No Power Struggles!"
Co-ordinated Multi-level Power Management for the Data Center
Ramya Raghavendra, Partha Ranganathan, Vanish Talwar, Zhikui Wang, Xiaoyun Zhu
ASPLOS 2008



World wide cost of purchasing and operating servers

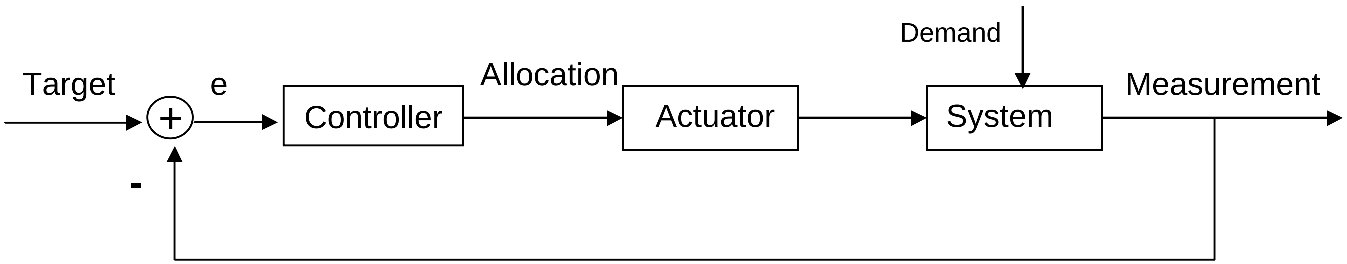
environmental impact, heat& density, reliability

Today…

- Data centres engineered to handle sum of peak power requirements of components
- Components rarely need peak power at the same time
  - Leads to higher costs, limits ability of data centres to grow

Challenge: bound peak power usage and minimize average usage while satisfying workload demand requirements.

Approach based on feedback control theory

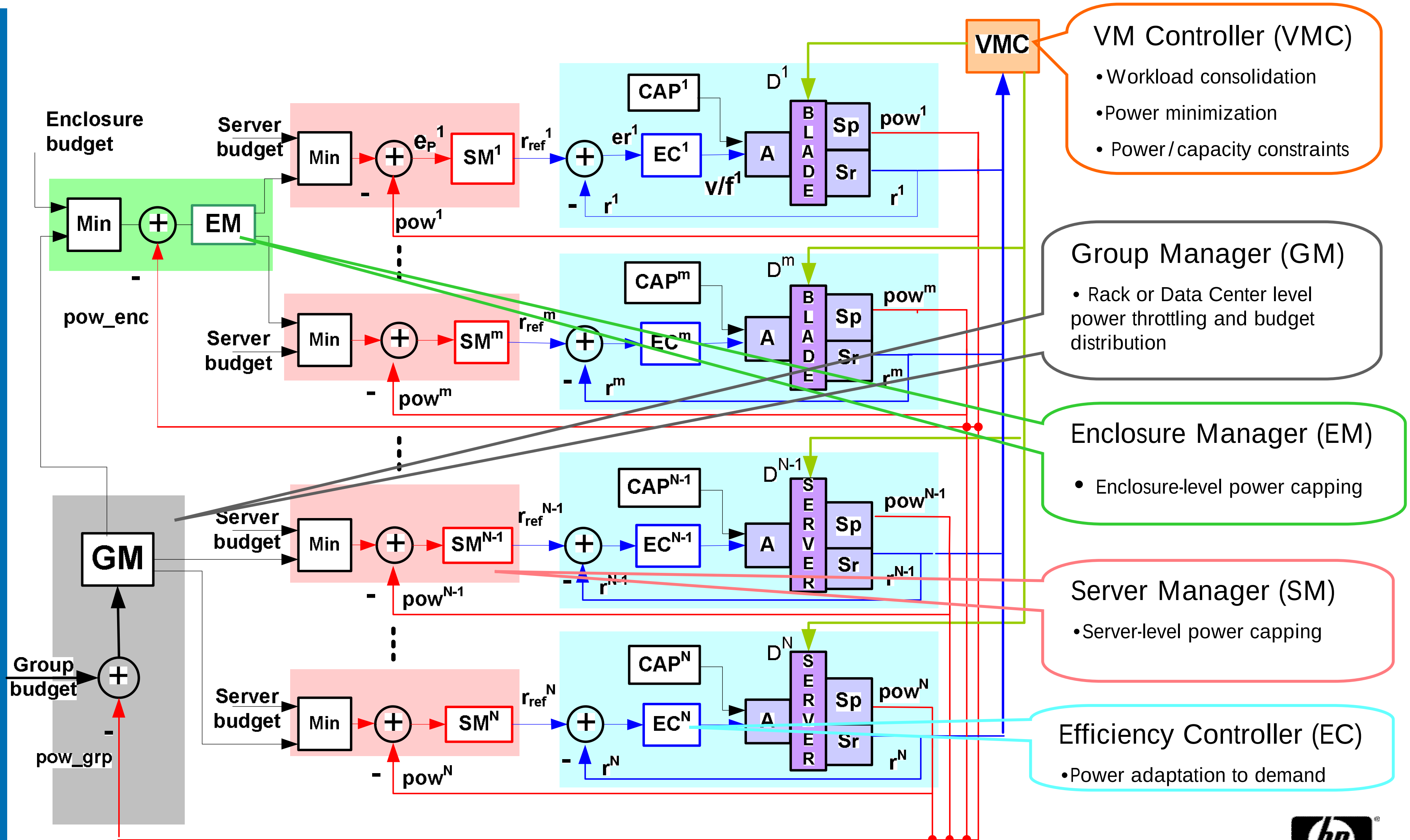


Standard feedback control loop

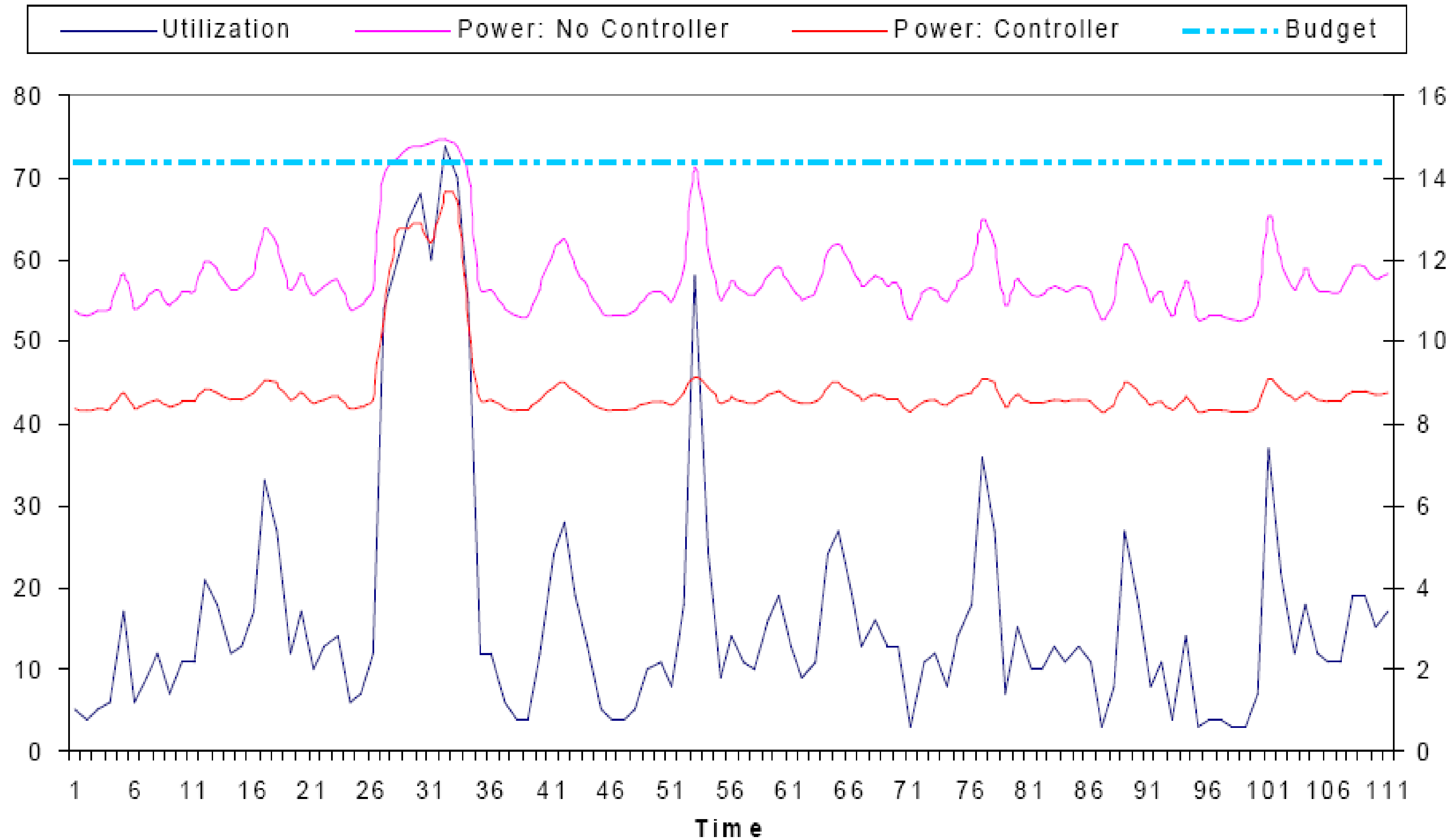
- First unified architecture for data center power management
  - Feedback control theory for mathematical rigor
    - Theoretical guarantees of stability and performance
  - Multiple controllers controlling different aspects of power
  - Adapts to workload on the fly

- Evaluation on real-world traces: correctness, stability, efficiency

Unified and Extensible Architecture



Sample of results from trace driven simulation



- Workload-aware adaptation & correctness

Shared IO mgmt….

Sharing disk arrays today

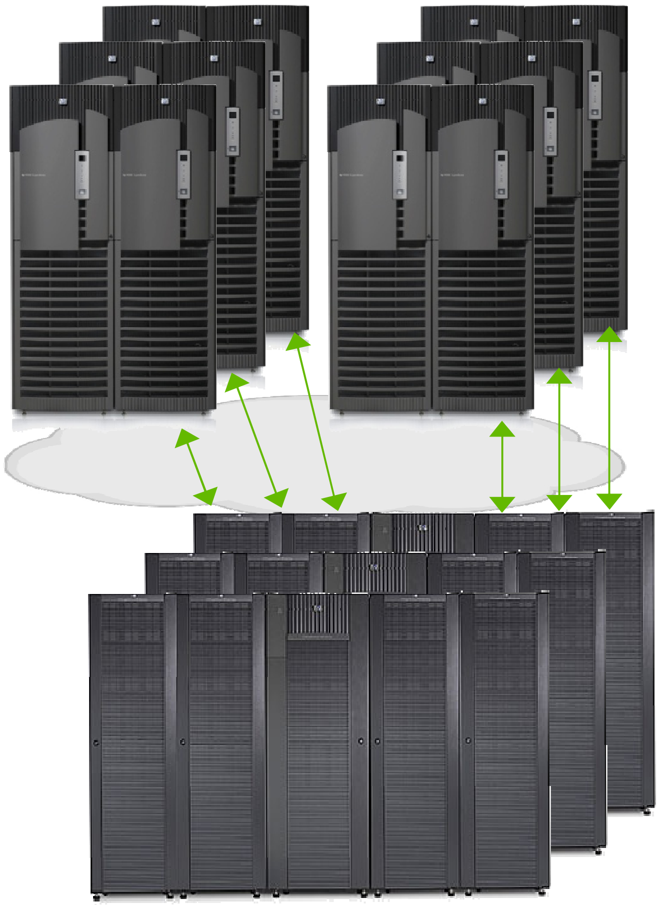
Ajay Gulati, Arif Merchant, Mustafa Uysal, Peter J. Varman, HPL-2007-186

Disk arrays are shared by many clients

- Due to server & storage consolidation, virtualization
- Storage clients can have very different requirements:
  - Transaction – random io
  - Batch – sequential io

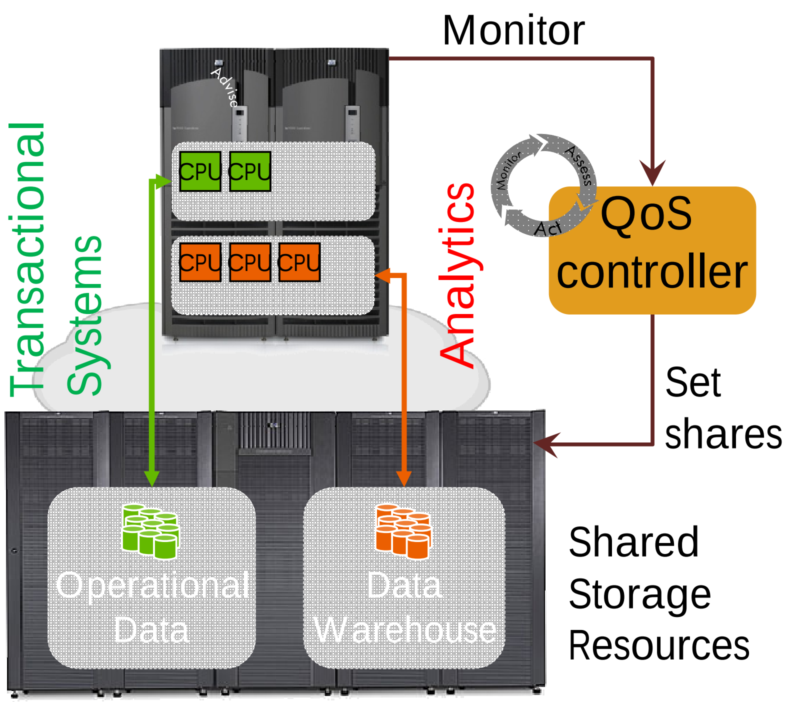
Inadequate support for sharing arrays

- Arrays ignore app. requirements when scheduling IOs
- Challenges:
  - Ensuring adequate performance for apps
  - Protecting against misbehaving / runaway applications
- QoS is typically achieved by static partitioning and over-provisioning.



Challenge: How to balance QoS with IO efficiency?

Storage QoS Control



- Solution: QoS control
  - Administrator specifies app. Requirements
  - Feedback controller monitors app. performance and sets its resource share dynamically
  - Scheduler at disk array enforces app. resource allocations

Control knobs: io request batch size and concurrency

Vary to achieve higher efficiency while controlling short-term fairness

Sample results

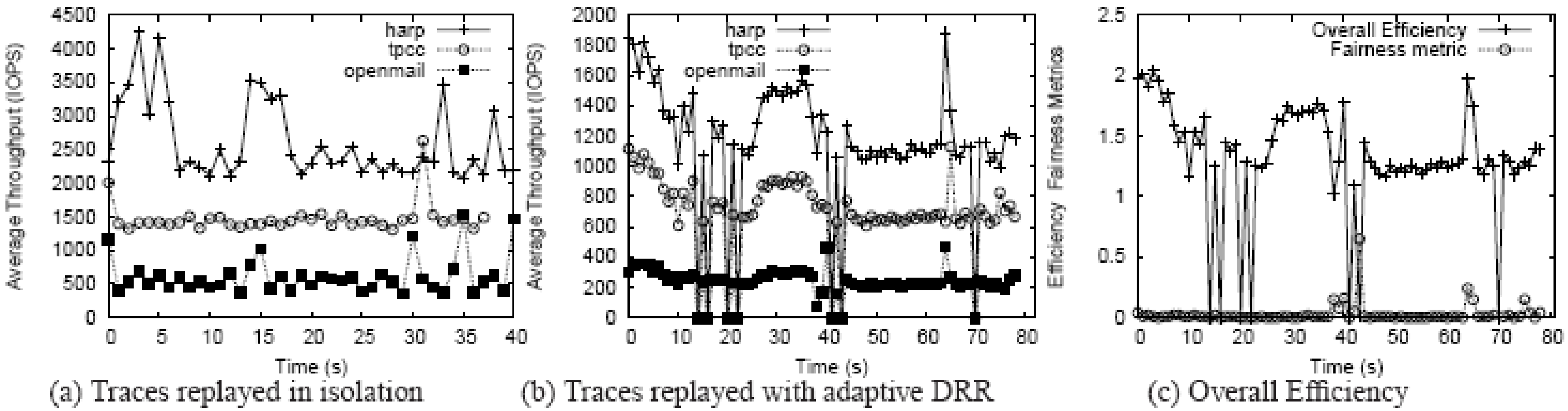


Figure 12: Running three different traces (openmail, tpcc and harp) using adaptive DRR.

a) shows IOPS for a file system, db, and email io trace run in isolation

b) shows IOPS when run together, as expected individual IOPS decrease

c) shows an average efficiency of 1.4 (improvement of 40% wrt a))

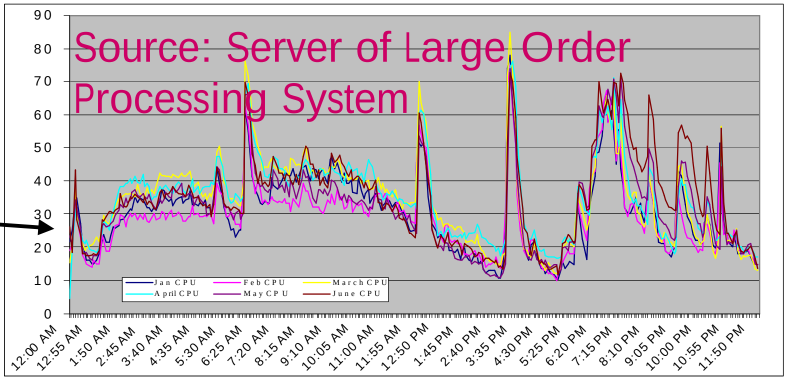
Compute capacity mgmt….

Exploiting patterns for improved quality / capacity

- Enterprise workloads often have patterns, e.g.,

Monthly average cpu util for 5 minute intervals within a 24 hour day

Each curve represents a month



5 minute intervals over a 24 hour day

- Can we exploit historical time varying capacity requirements as predictor for future requirements?
- Can this help to improve capacity management?
- How can this be integrated into a framework for control?

Proactive and Reactive Controllers

1000 Islands: An Integrated Approach to Resource Management for Virtualized Data Centers
Xiaoyun Zhu, Donald Young, Brian J. Watson, Zhikui Wang, Jerry Rolia, Sharad Singhal, Bret McKee, Chris Hyser, Daniel Gmach†, Robert Gardner, Tom Christian, Lucy Cherkasova (To appear in Cluster Computing Journal)

- Proactive workload placement controller (global workload optimizer)
  - Proactive controller uses historical information to pack VMs to nodes
  - Periodically initiates migration of VMs to reduce “likelihood” of violations
  - Keeps number of nodes in proportion to demand
- Reactive VM migration controller (local workload optimizer)
  - Reactive controller
  - Initiates migration of VM to “alleviate” violations
    - May cause a node to be added
  - Initiates migration of VMs to free up nodes & reduce power consumption
  - Will add nodes if necessary

Integration of controllers

- Parallel integration of controllers
  - Run workload placement controller periodically
  - VM migration controller initiates migrations whenever there is a violation
- Tight integration
  - Migration controller invokes workload placement controller on demand, may cause many migrations when there is a violation

Case Study:

- 138 historical workload demand traces
  - SAP enterprise applications
  - CPU and memory demands
  - Measurements every 5 minutes
  - Used 12 weeks of data for case study
- Workloads required
  - CPU: between 2 and 8 virtual CPUs
  - Memory: between 6 GB and 32 GB
- Simulated resource pool
  - 20 servers
  - 16 cores/server
  - 256GB/server
  - 10 Gbps networking infrastructure

Some results (to appear in cluster computing journal)

Workload placement controller alone with Perfect knowledge

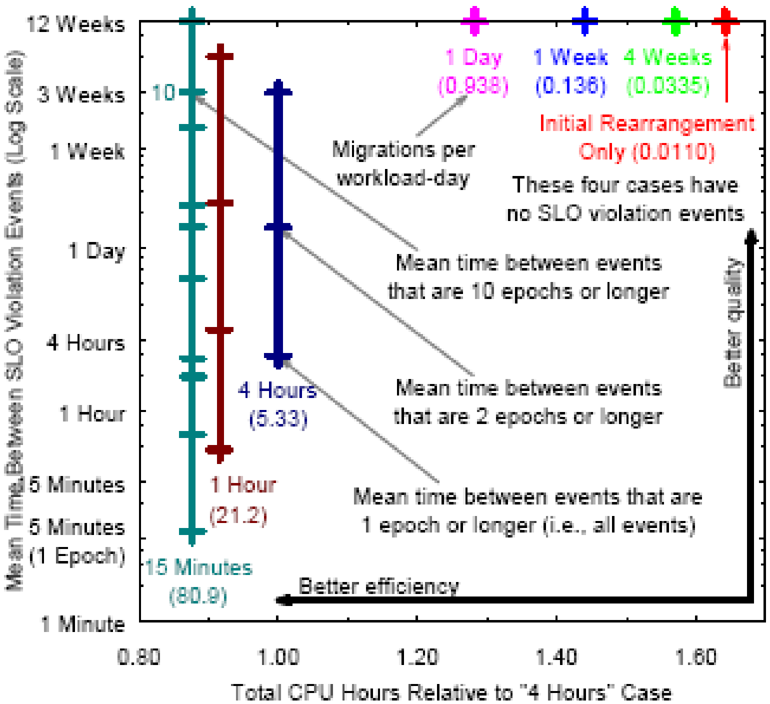


Figure 4. CPU quality vs. rearrangement periods for pod set controller only

Integrated Controllers

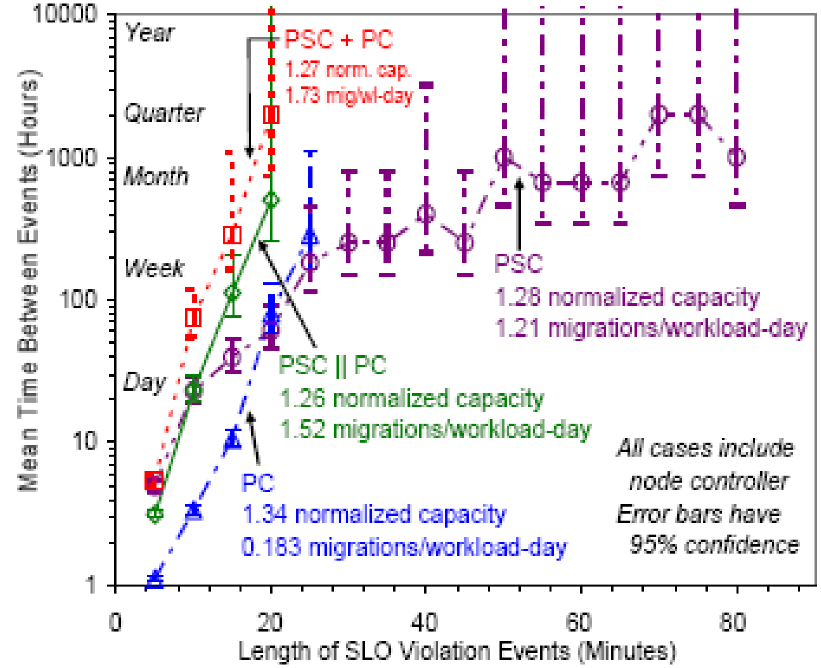


Figure 5. Emulation results for four different combinations of controller policies

Use of historical information helps

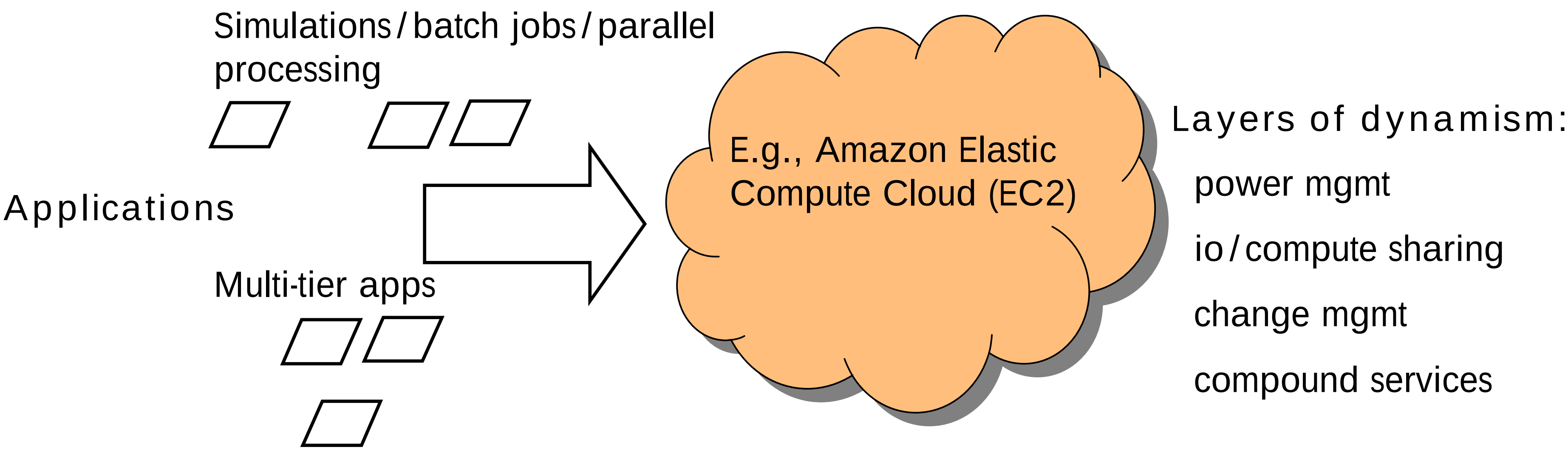
Tighter integration of controllers leads to higher quality while reducing use of capacity

.. But causes more migrations/workload-day

Applications mgmt….

Cloud computing with resource utilities

Deploy, configure, change mgmt, run-time management



Challenges:

How to predict/guide their behaviour?

How to operate for a long time (longevous) without human intervention?

Summary

- Self-management examples for next generation data centre infrastructure
  - Power mgmt
  - Shared IO mgmt
  - Compute capacity mgmt
- New challenges
  - Applications exploiting cloud resource utilities
  - How can self-management help?