Run-time Energy Consumption Estimation Based on Workload in Server Systems

Adam Lewis, Soumik Ghosh, and N.-F. Tzeng
The Center For Advanced Computer Studies
The University of Louisiana

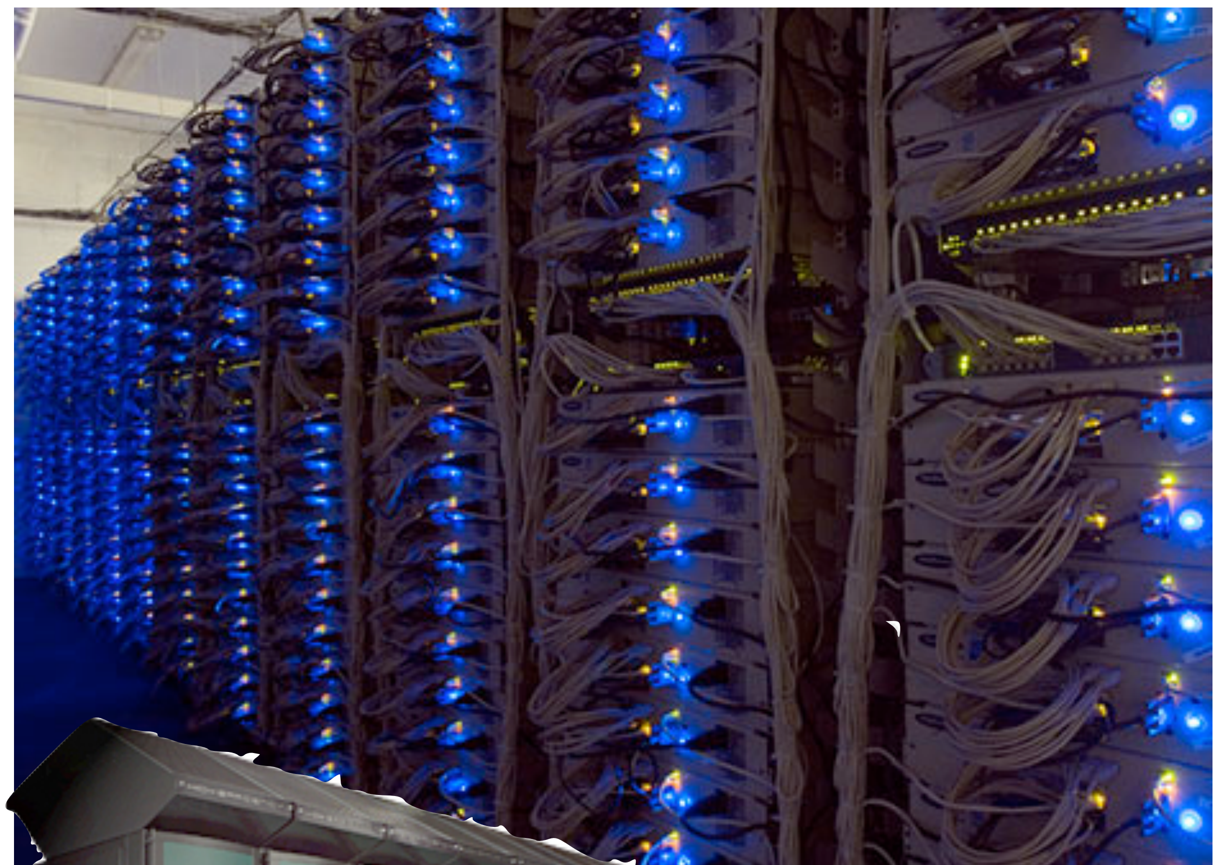
The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

What does this picture tell us?

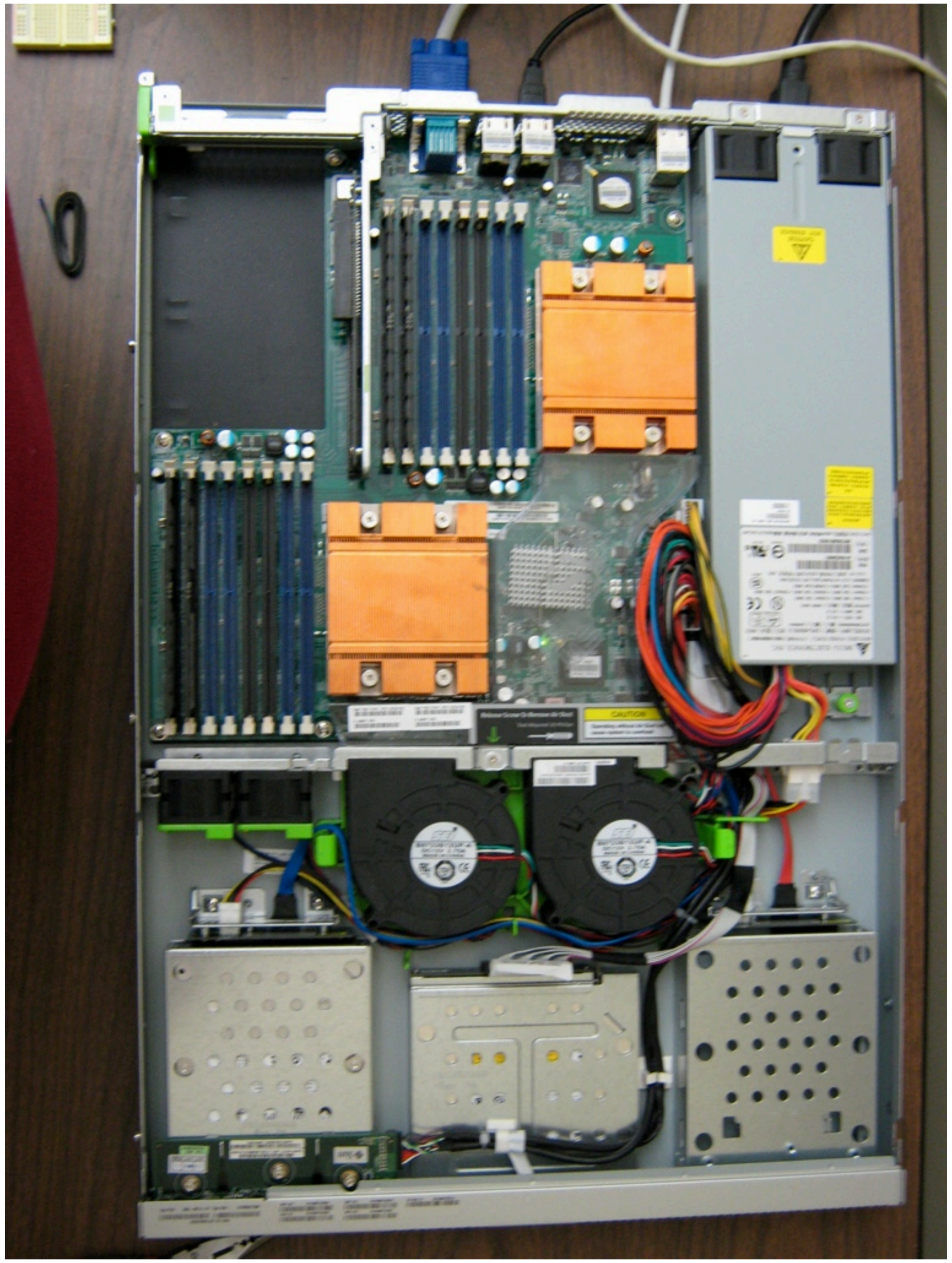
(c) The New York Times, June 14, 2006

Introduction



- Server workloads?
  - –Laptop/desktop techniques?
  - –Full system model
    - interactions between processor, memory, storage
  - –Linkage between power, temperature, and energy

Model: Characteristics

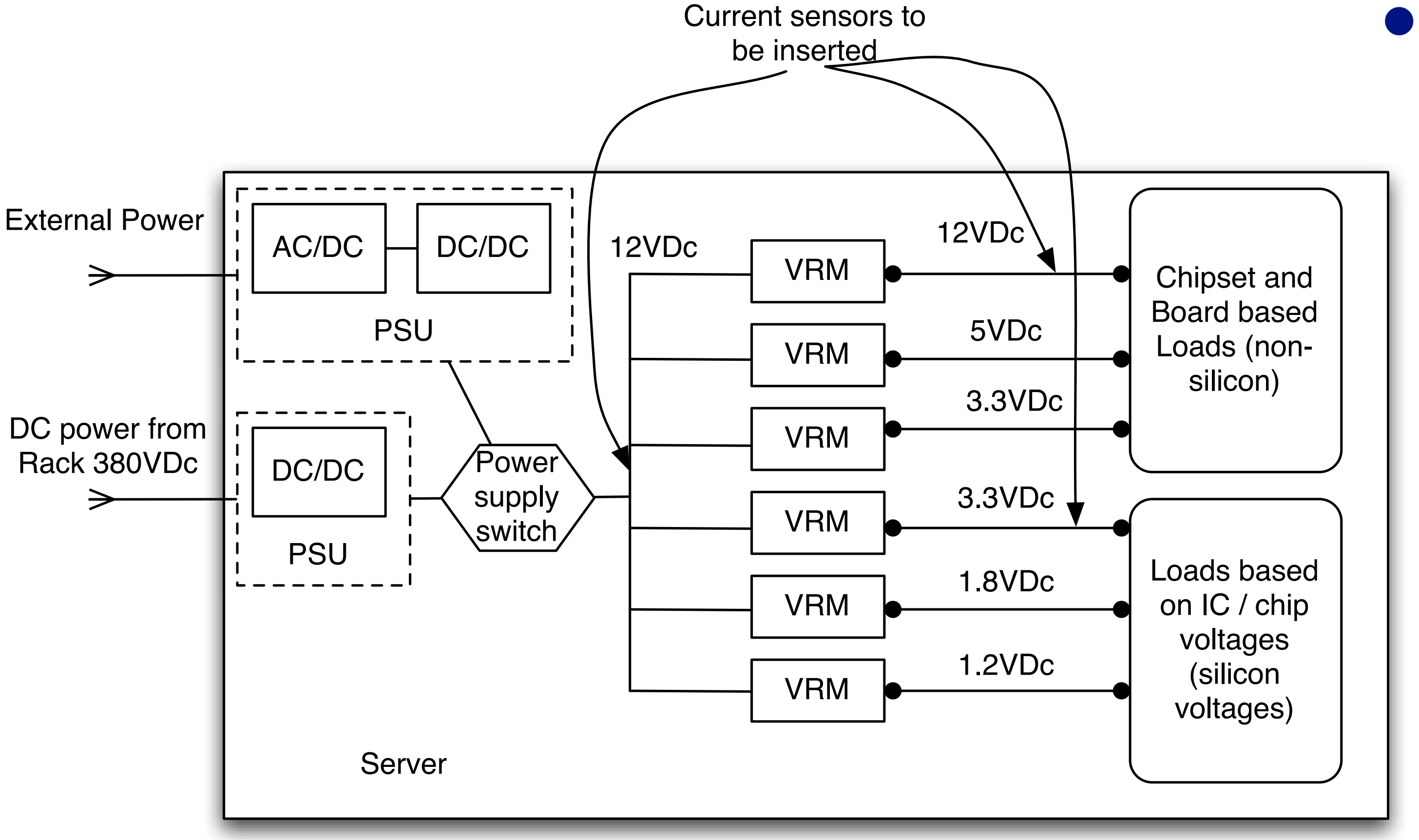


- Need a model that
  - Dynamically tracks DC power draw into system
  - We can use to control the power and thermal envelope of the system
    - based on the processing load

Model: Inputs

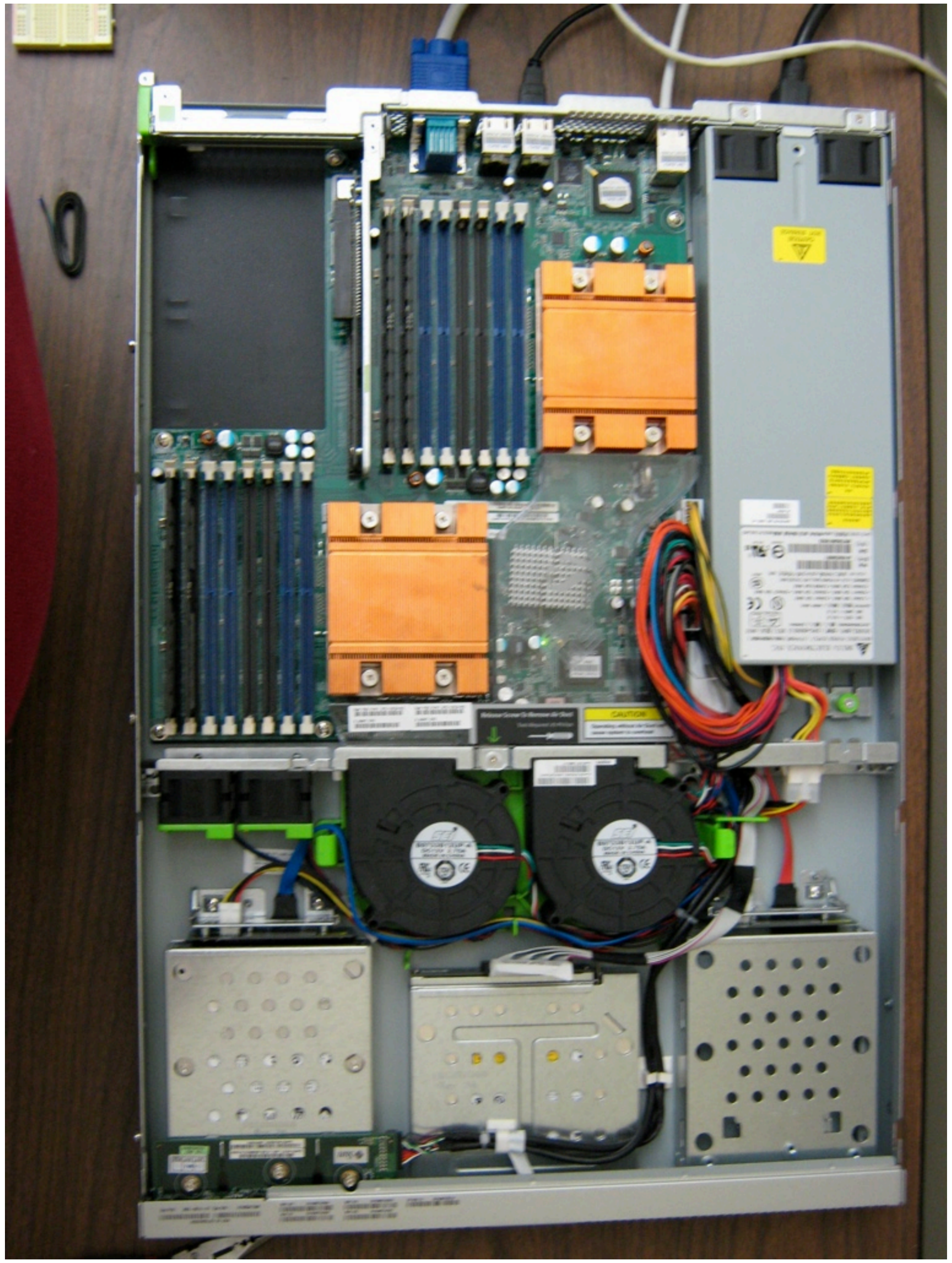
$$E_{dc} = E_{system}$$

$$E_{dc} = \int_{t_1}^{t_2} \sum_{j=0}^{N} \sum_{k=0}^{Mj} v_k(t) \cdot i_k(t) dt$$



- Three DC voltage domains
  - –12Vdc, 5.5Vdc, 3.3Vdc
- 5.5V and 3.3V domains limited to 20% of rated voltage

Model: Components

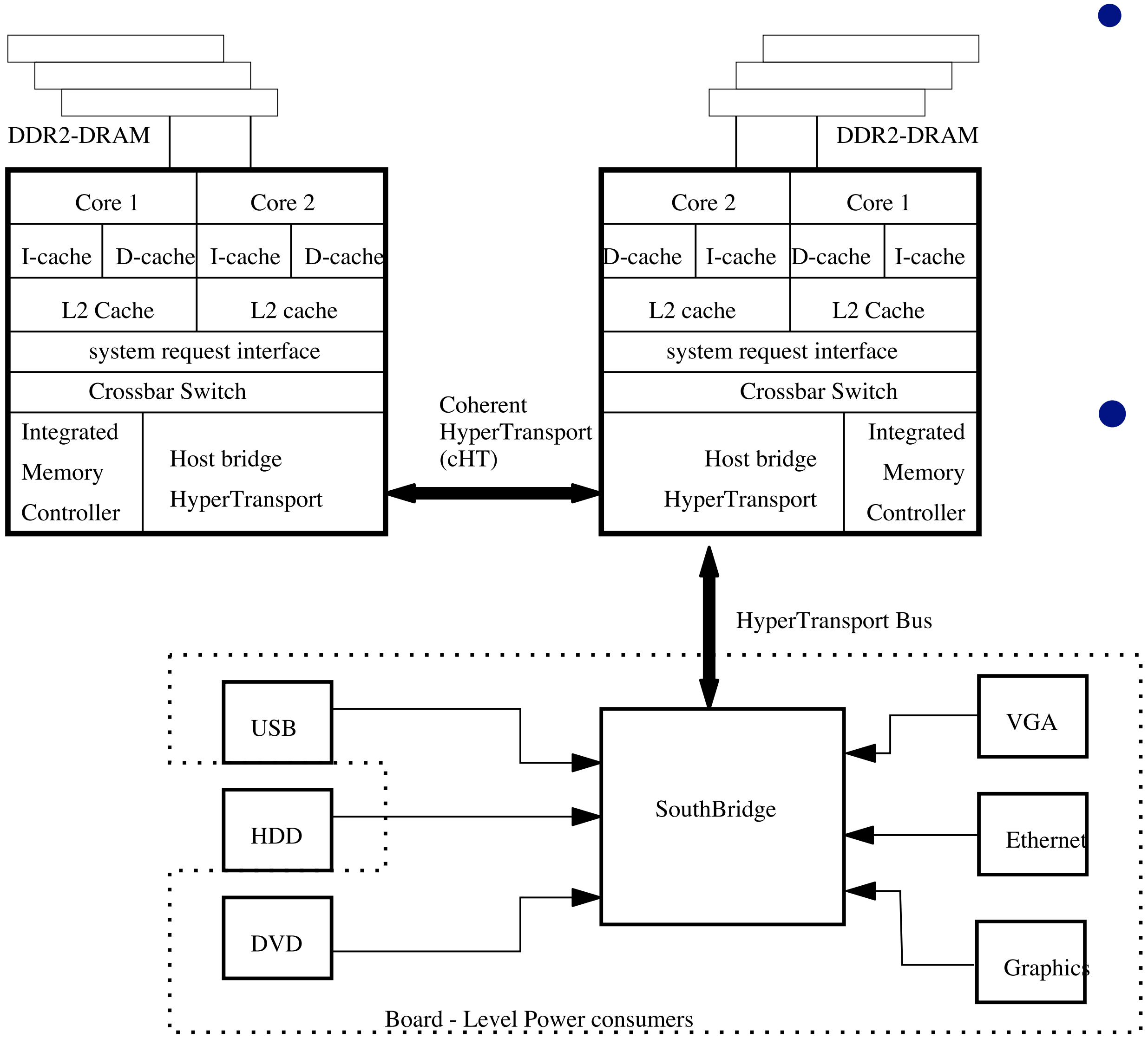


$$E_{dc} = E_{system}$$

$$\begin{aligned} E_{system} &= \alpha_0(E_{proc} + E_{mem}) \\ &+\alpha_1 E_{em} \\ &+\alpha_2 E_{board} \\ &+\alpha_3 E_{hdd} \end{aligned}$$

Model: Processor

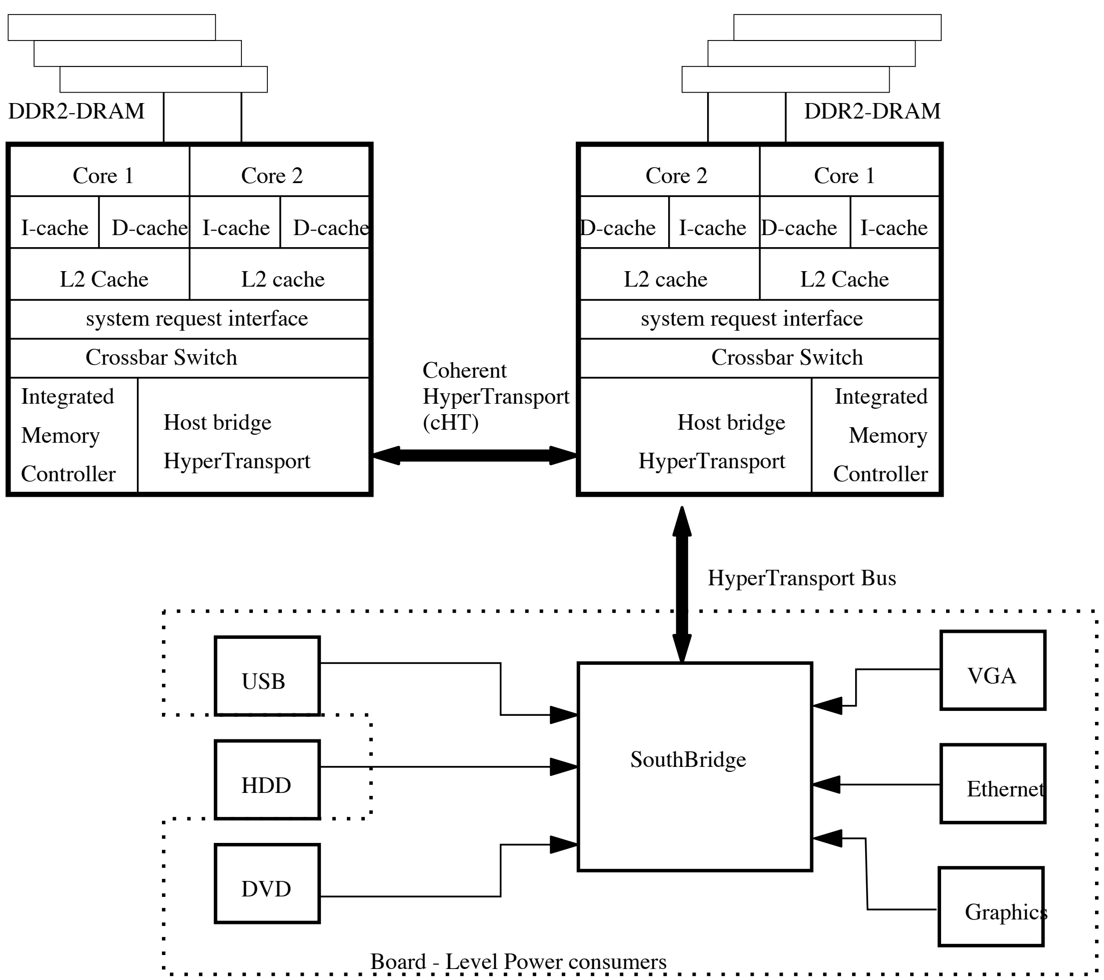
$$P_{proc} = \mathbf{H} \cdot \mathbf{X} = [\beta_0 \cdots \beta_{10}]^{\mathbf{T}} \cdot [\mathbf{Var_0} \cdots \mathbf{Var_{10}}]^{\mathbf{T}}$$



- Processor as black box
  - Energy = f(workload)
  - Manifests as heat
- HyperTransport
  - Data bus between cores and Southbridge
  - Reflects amount of data being processed by a processor or its cores

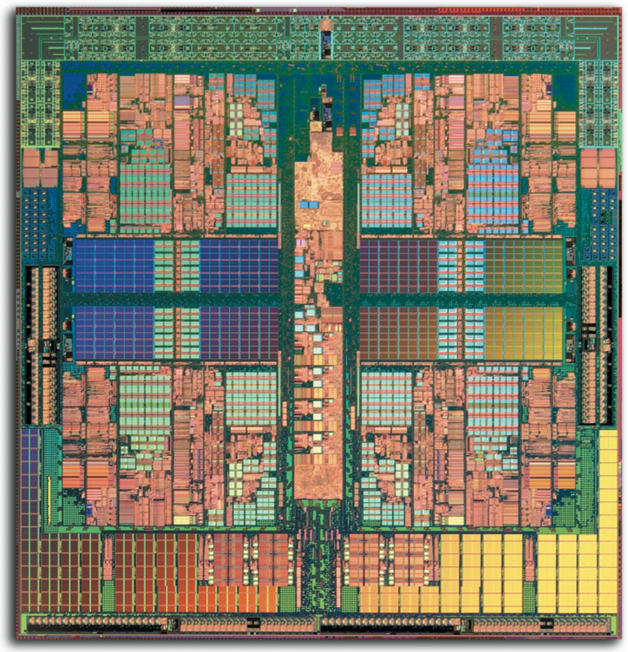
Model: Processor

$$P_{proc} = \mathbf{H} \cdot \mathbf{X} = [\beta_0 \cdots \beta_{10}]^{\mathbf{T}} \cdot [\mathbf{Var_0} \cdots \mathbf{Var_{10}}]^{\mathbf{T}}$$



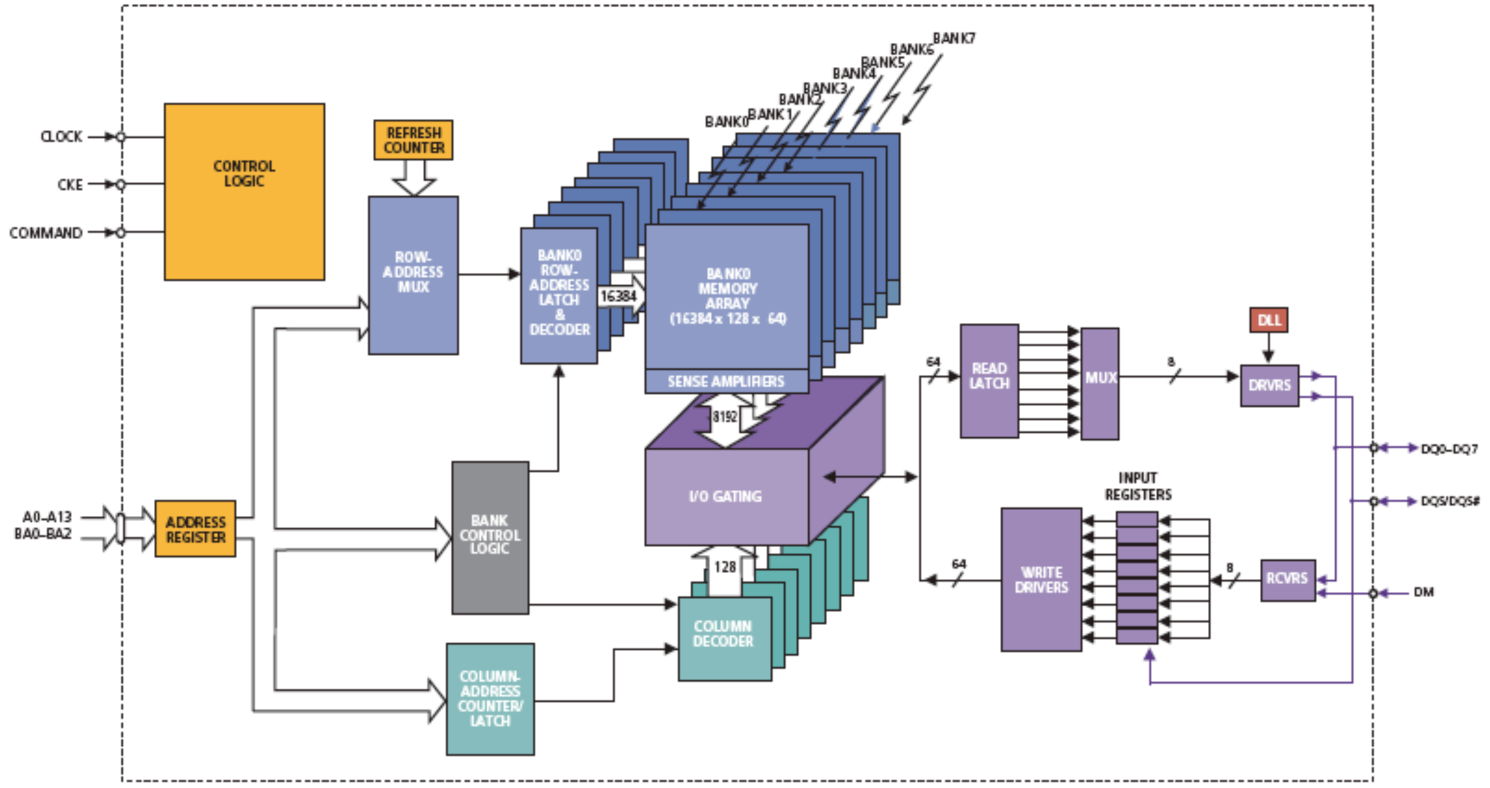
- The vector **X** contains variables for
  - Ambient and die temp.
  - HT1 and HT2 transactions
  - L2 cache misses per core

Model: Processor



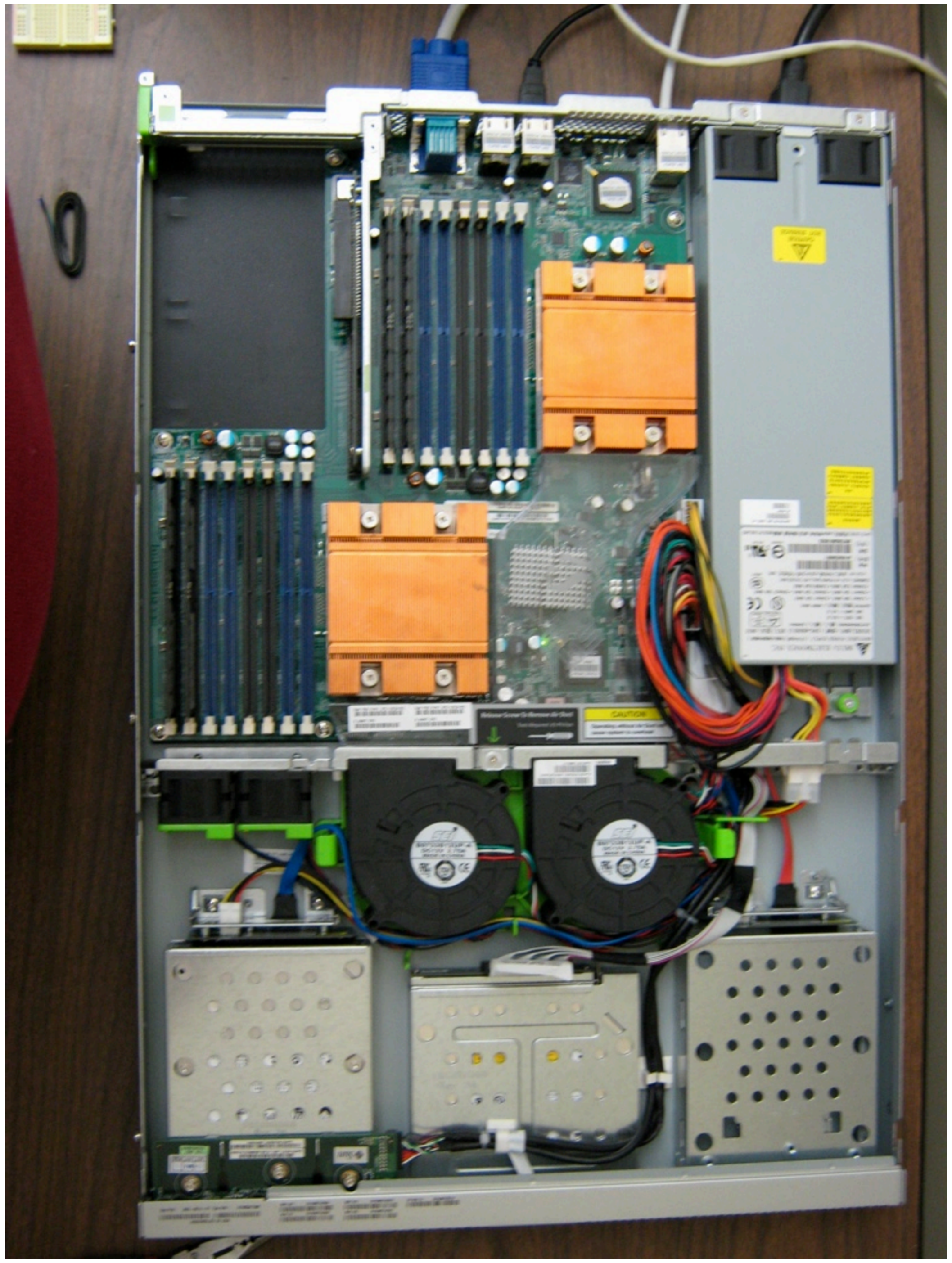
- Processor metrics
  - Performance counters
  - Avoid multiplexing as much as possible
- System metrics
  - Operating system metrics
  - Access through IPMI, sar, and other standard tools

Model: Memory



- DRAM Read/Write power + background power = known quantities
- Performance counters exist for measuring the count of highest level cache miss
- Combine these to compute the energy consumed

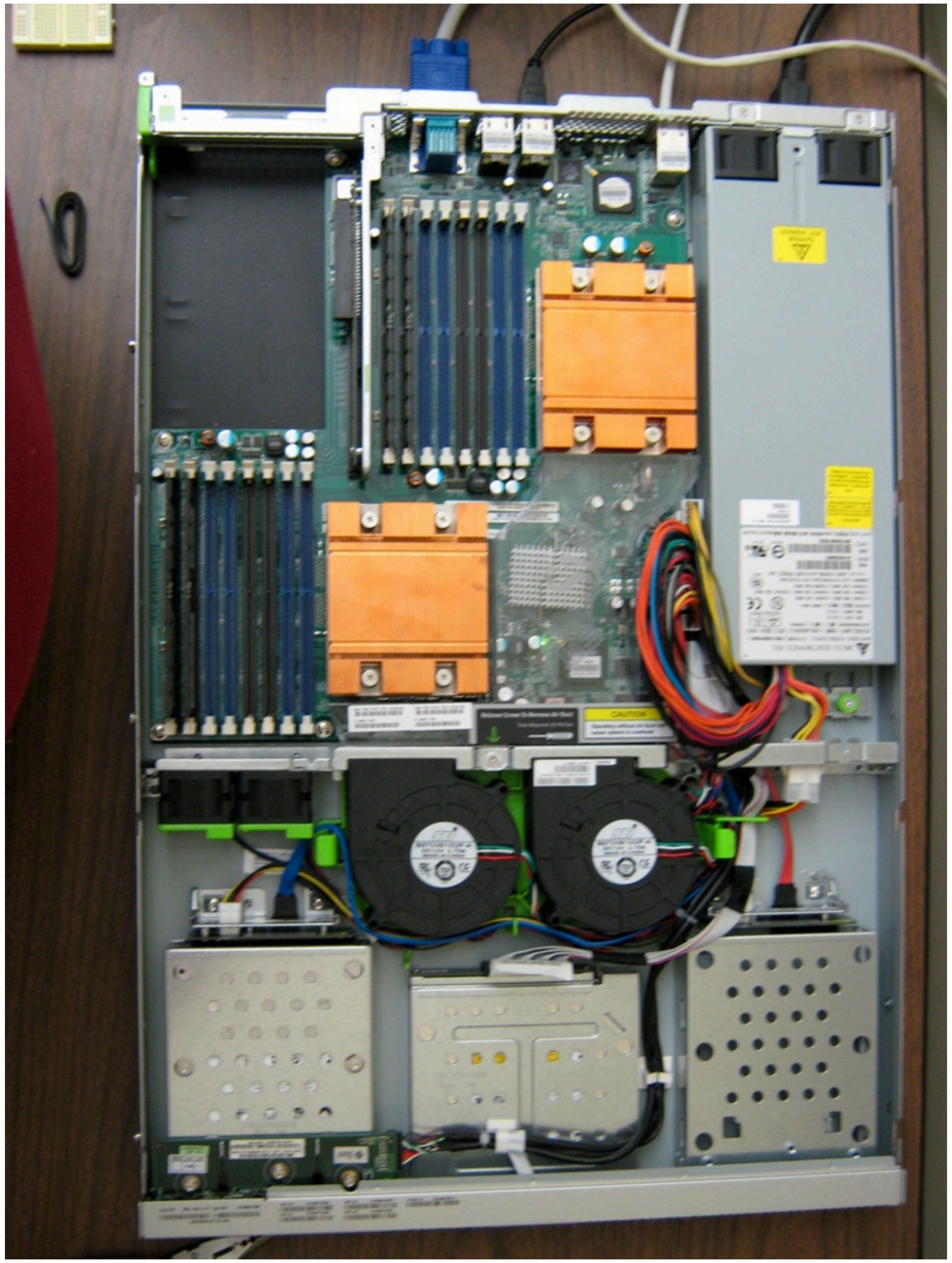
Model: Board



$$E_{board} = \left(\sum V_{power-line} \times I_{power-line}\right) \times t_{time-slice}$$

- System components that support the operation of the machine
  - –Typically in the 5.5Vdc and 3.3Vdc power domains
  - –Measured by current probe

Model: Electromechanical



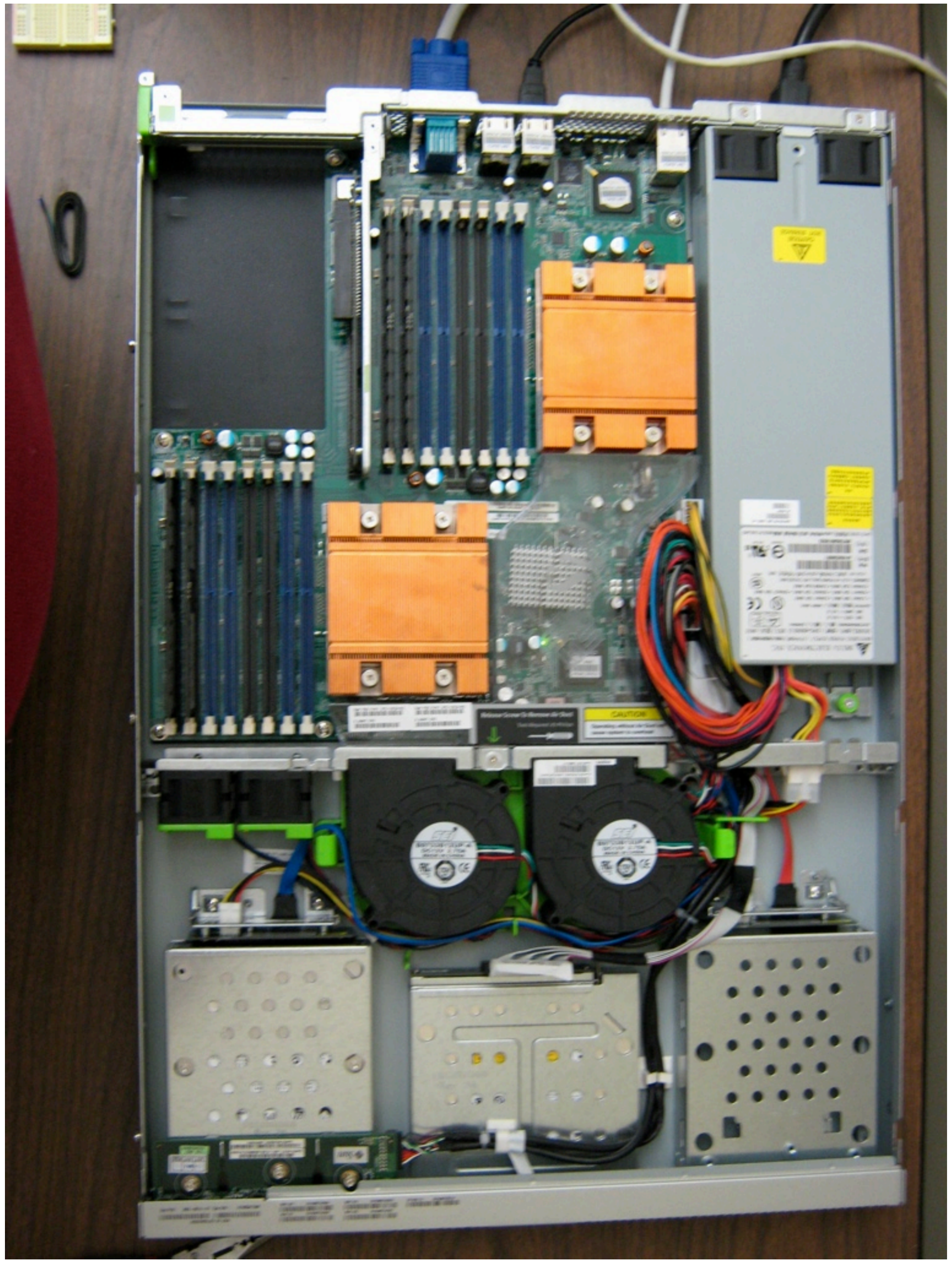
- Need to account for energy required to cool
  - –No performance counters
  - –Can measure power drawn by the fans
  - –System reports time slice
- Energy consumption by optical devices

Model: Storage

$$
\begin{aligned}
E_{hdd} \quad &= P_{spin-up} \times t_{su} + P_{read} \sum N_r \times t_r \\
&+ P_{write} \sum N_w \times t_w + \sum P_{idle} \times t_{idle}
\end{aligned}
$$

Parameter	Value
Interface	Serial ATA
Capacity	250 GB
Rotational speed	7200 rpm
Power (spin up)	5.25 W (max)
Power (Random read, write)	9.4 W (typical)
Power (Silent read, write)	7 W (typical)
Power (idle)	5 W (typical)
Power (low RPM idle)	2.3 W (typical for 4500 RPM)
Power (standby)	0.8 W (typical)
Power (sleep)	0.6 W (typical)

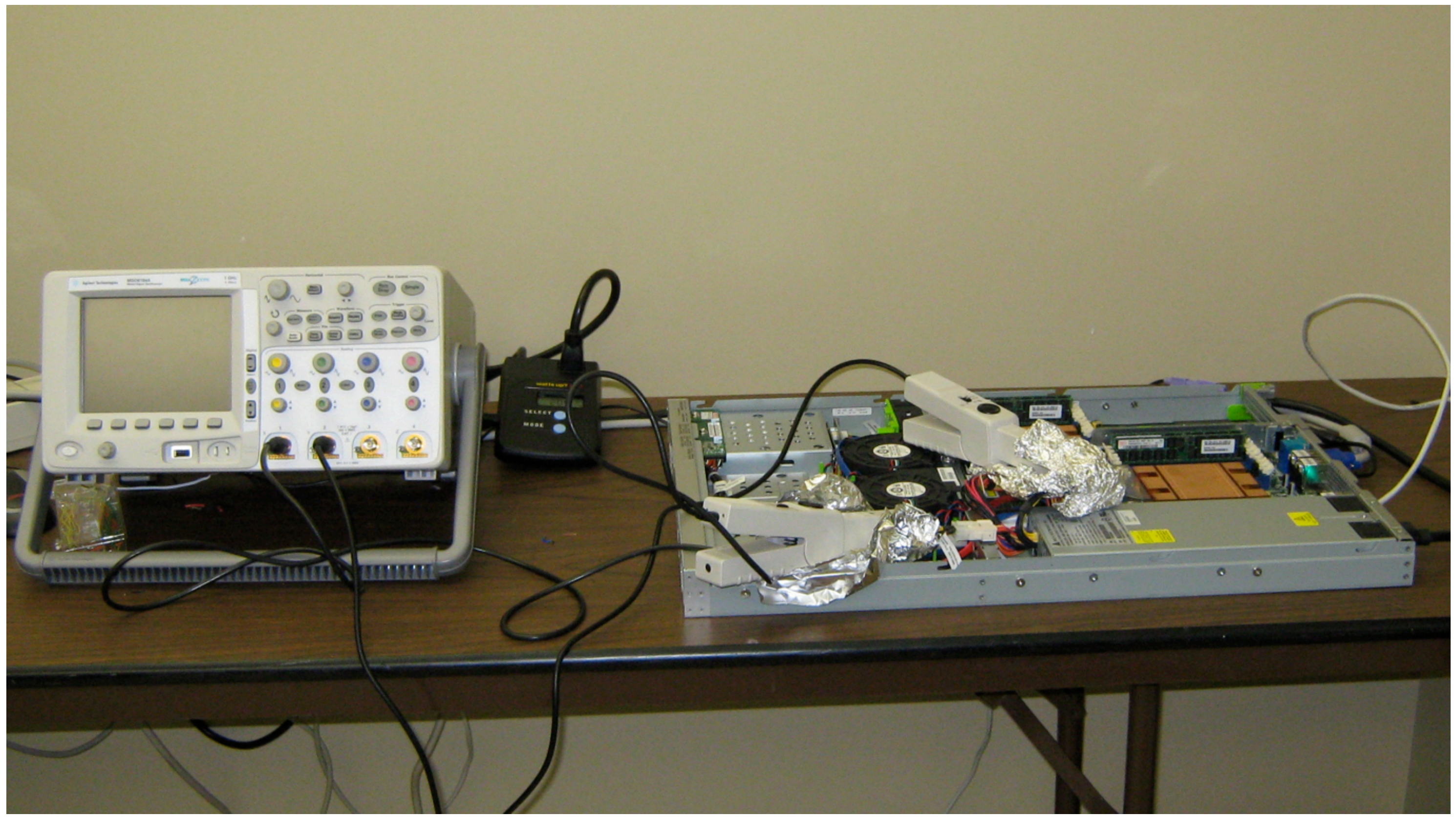
Model: Components



$$E_{dc} = E_{system}$$

$$\begin{aligned} E_{system} &= \alpha_0 (E_{proc} + E_{mem}) \\ &+ \alpha_1 E_{em} \\ &+ \alpha_2 E_{board} \\ &+ \alpha_3 E_{hdd} \end{aligned}$$

Application of Model



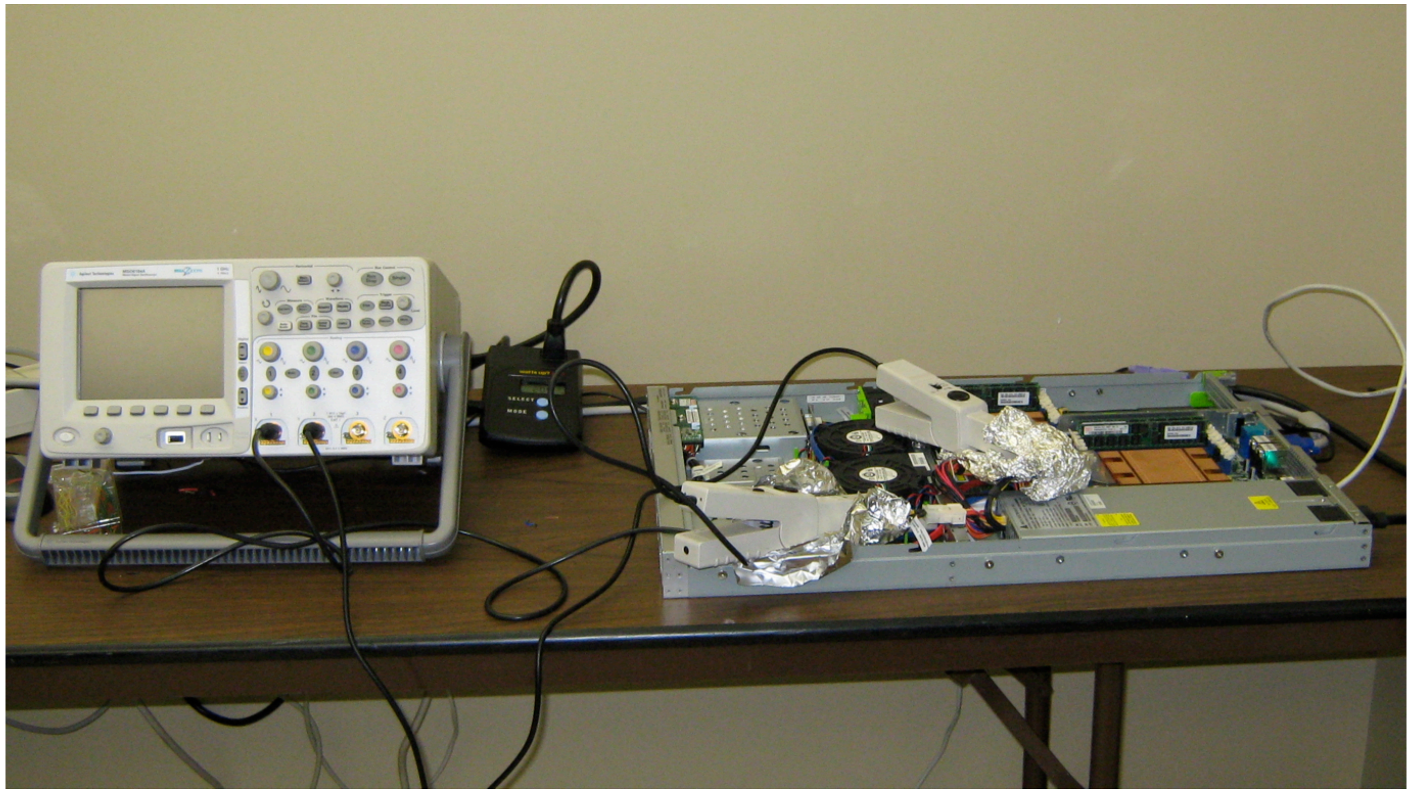
System Under Test

	Sun Fire 2200
CPU	2 AMD Opteron
CPU L2 cache	2x2MB
Memory	8GB
disk	2060GB
Network	2x1000Mbps
Video	On-board
Height	1 rack unit

SPEC CPU2006 Benchmarks

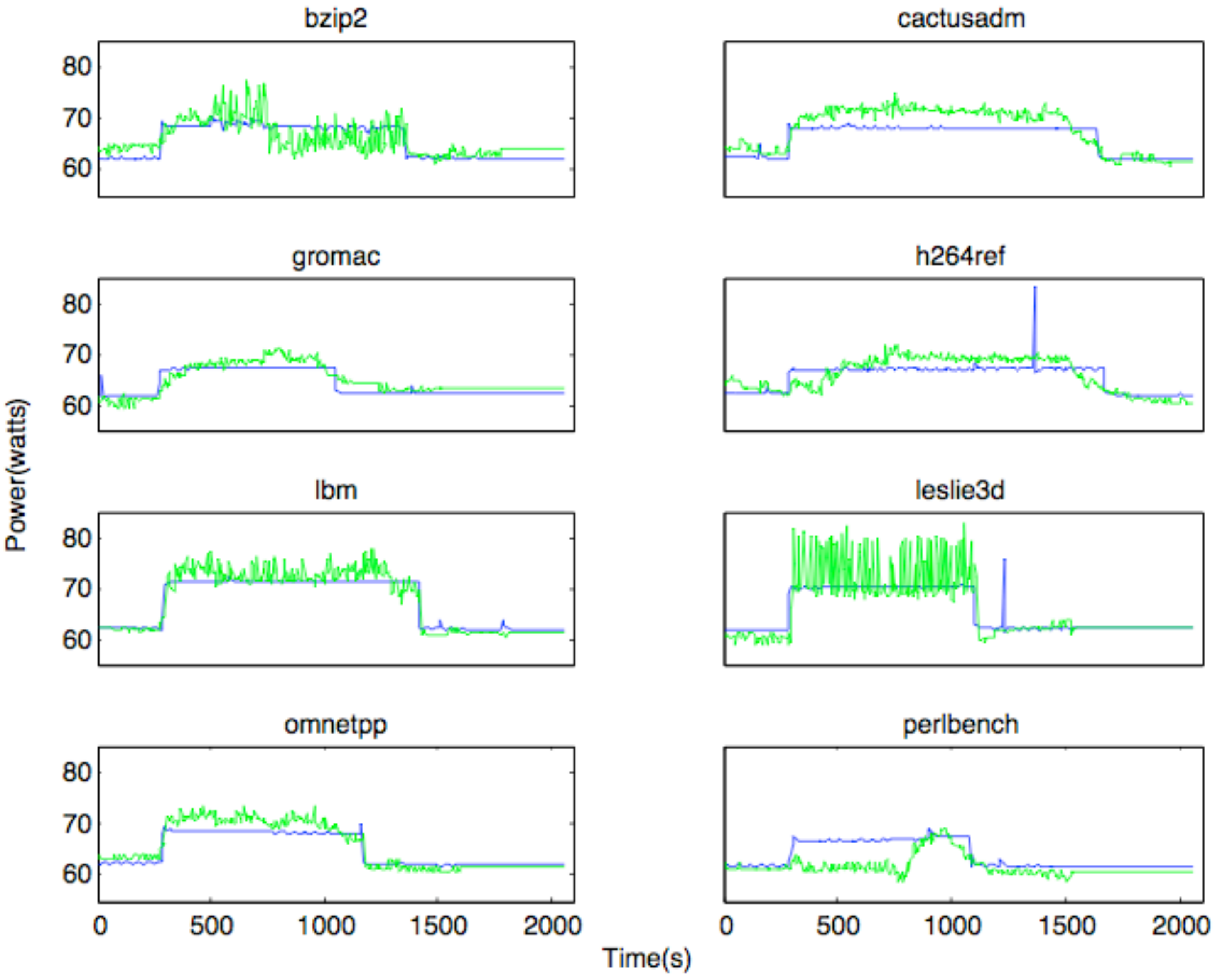
Benchmark		***Type***	***Use***
perlbench	C	Integer	PERL Programming Language
bzip2	C	Integer	Compression
mcf	C	Integer	Combinatorial Optimization
omnetpp	C++	Integer	Discrete Event Simulation
gromacs	C/Fortran	Float	Biochemistry/Molecular Dynamics
cacstusADM	C/Fortran	Float	Physics/General Relativity
leslie3d	Fortran	Float	Fluid Dynamics
lbm	C	Float	Fluid Dynamics

Application of Model



- Sample for each benchmark
- Use geometric mean of samples to create a predictive model for this configuration
- Create linear regression model for each benchmark and geometric mean
- Use ANOVA techniques to compare
- Regression model built from geometric mean
  - Adjusted R-square value of 0.965
  - Projected error of 3.5%
  - 95% confidence level

Evaluation



- Use model constructed from geometric mean to predict for each benchmark used to calibrate model
- Error range from 2.0% to 3.5% for each benchmark
- Noise in prediction results from use of geometric mean

Observations

- One model to rule them all?
- Inputs
  - How many PeC at one time?
  - Lack of standardization for environmental sensors
  - Placement of sensors
- Linear or non-linear?
- Calibrate/evaluate using other HPC benchmarks
- Other architectures?
  - Intel Xeon
  - Cell BE
- How well does the model scale to larger number of cores?

Next steps

- Calibrate with larger set of benchmarks
  - Disk and memory benchmarks
  - Other HPC benchmarks
- Evaluate and extend model for other architectures
  - Intel Xeon
  - Cell BE
- Incorporate model into a energy conserving process scheduler

Acknowledgements

This work was supported in part by the U.S. Department of Energy and by the Louisiana Board of Regents

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette