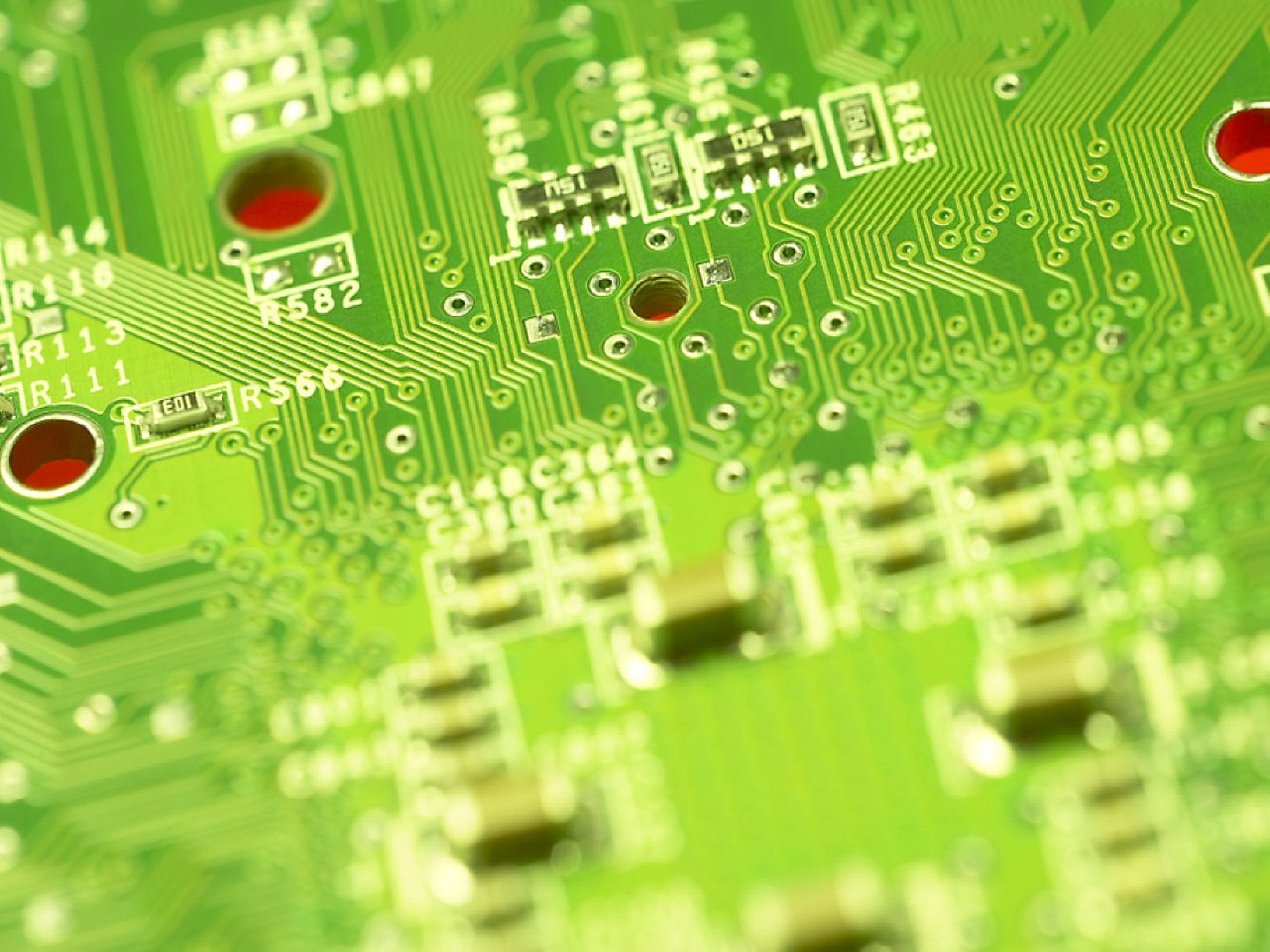

R114
R116
R113
R111
R566
R582
R463

# The Powers That Be (in HPC)

Kirk W. Cameron
Computer Science
Virginia Tech

# ENA-HPC Street Credit

- Over $6M related federal funding (since '04) (NSF, DOE, SBIR, IBM, Intel, and others)
- EPA Energy Star for servers (since '05)
- SPECPower Founding Member (since '05)
- Co-founder Green500 (since '06)
- Green IT Columnist (*IEEE Computer*)
- CEO and Founder, MiserWare Inc. (since '07)

spec

MiserWare
Saving Energy, Saving Money, Saving the World.

THE GREEN 500™

# The way we were (circa 2003)

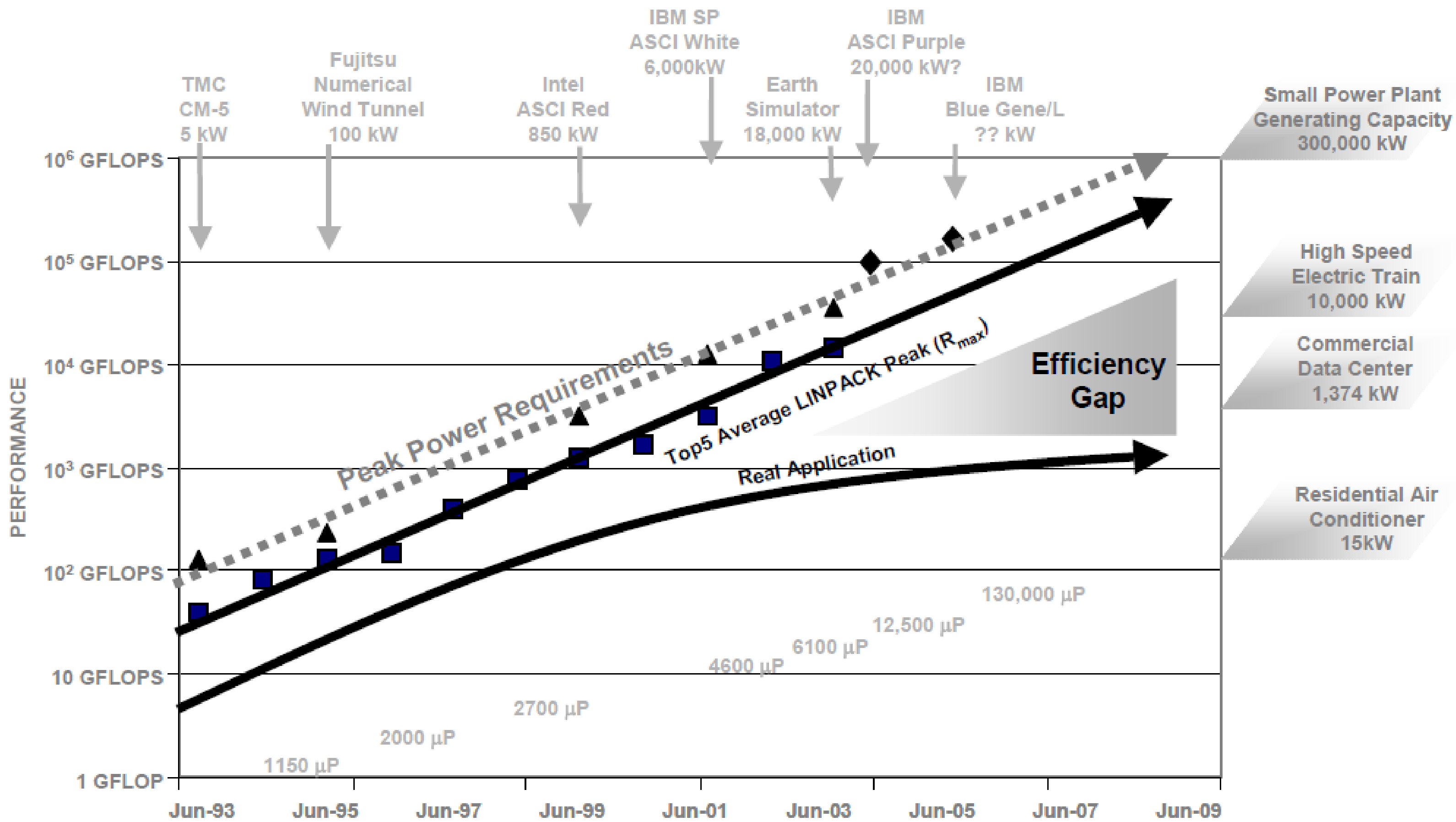
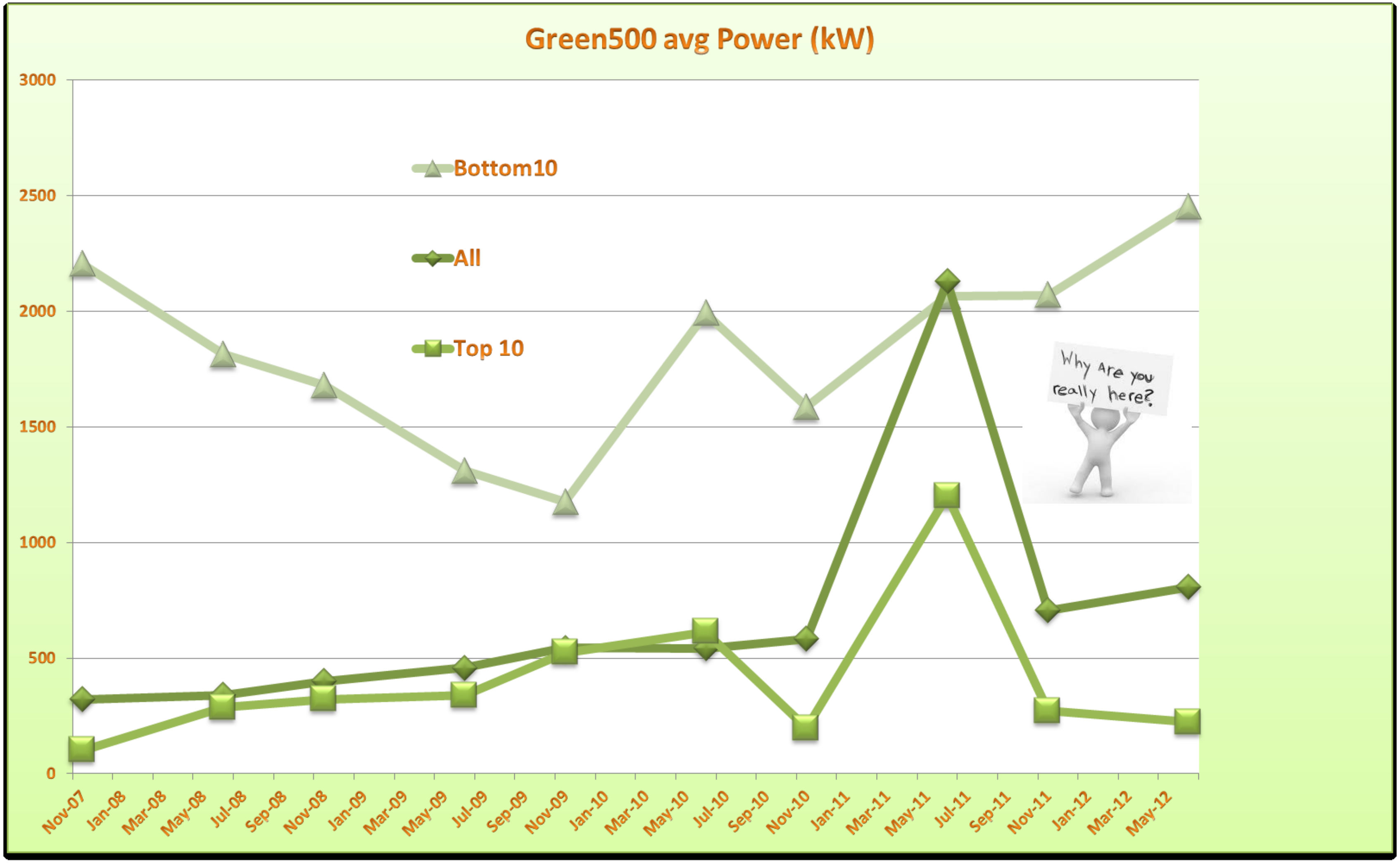



**Fig. 1 Power-performance trends in the supercomputer industry.** The computational demands of scientific applications have led to exponential increases in peak system performance (shown as average of peak LINPACK measurements), system power consumption (shown for several supercomputers), and

Source: CAREER: High-performance, Power-aware Computing
K. Cameron, NSF CCF-0347683, 3/1/04-2/28/09)

# You are here (September 2012)

# Or are you really here?



# Getting there…


<u>From 2007-2012…</u>

[6x ↑ Flops/watt]

[~2.5x ↑ power consumption]

[Commodity systems catch efficiency of top 10 in 18 mo.]

<u>Projections for 2012-2019…</u>

[2100 to ~15,000 MFlops/Watt]

[66 kW for 1 Petaflop System]

[66 MW for 1 Exaflop System}

[Need 50,000 Mflops/Watt for 1 Exaflop @ 20 MW by 2019!!!]

# Conclusion: We need help.

# How can we…help you…help us…

# What do we need…?

**Insight**

Where does energy go?

**Understanding**

Why does energy go?

**Action**

What can we do?

# SCAPE Research (circa 2002)

- My observations
  - Power will become disruptive to HPC
  - Laptops outselling PC's
  - Commercial power-aware not appropriate for HPC

**$800,000 per year per megawatt!**

$4,000/yr

TM CM-5
.005 Megawatts

$12,000/yr

Residential A/C
.015 Megawatts

$680,000/yr

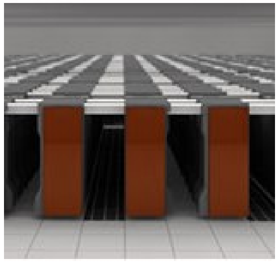
Conventional Power Plant
300 Megawatts

Intel ASCI Red
850 Megawatts

$8 million/yr

High-speed train
10 Megawatts

$9.6 million/yr



K Supercomputer
12 Megawatts

# SCAPE Launches HPPAC

2002

- High-performance, Power-aware Computing
  - Maintain Performance
  - Reduce energy waste
- Measurement tools
- No funding initially

# We were right! Whew.

2005 - Present

## IT confronts the datacenter power crisis

As energy costs escalate, conserving resources tops the list of challenges for today's IT managers

By Dan Goodin
October 06, 2006

E-mail Printer Friendly Reprints Slashdot It!

When David Young told his colocation provider late last year that his online applications startup, Joyent, planned to add 10 servers to its 150-system datacenter, he received a rude awakening. The local power utility in Southern California wouldn't be able to provide the additional electricity needed. Joyent's upgrade would have to wait.

## In the Data Center, the Heat Is On

Halamka John Today's Top Stories ▸ or Other Servers Stories ▸

October 23, 2006 (Computerworld) -- I recently began a project to consolidate two dat

## Data Center Budgets Face Radical Changes

Consortium head says facilities costs are surpassing the price of hardware

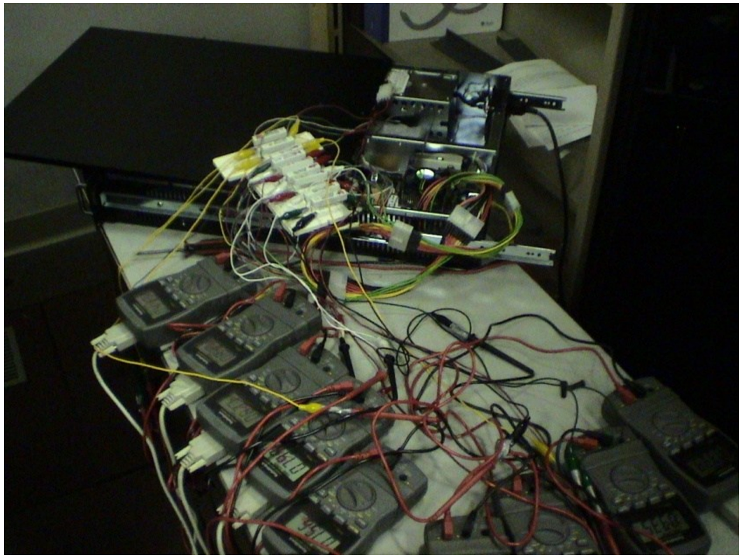
Patrick Thibodeau and Patrick Thibodeau Today's Top Stories ▸ or Other IT Management Stories ▸

October 30, 2006 (Computerworld) -- *The business value arising from Moore's Law, which says the number of*

**"You can only manage what you can measure."**

*Peter Drucker, writer*

# Measuring power is “tough”

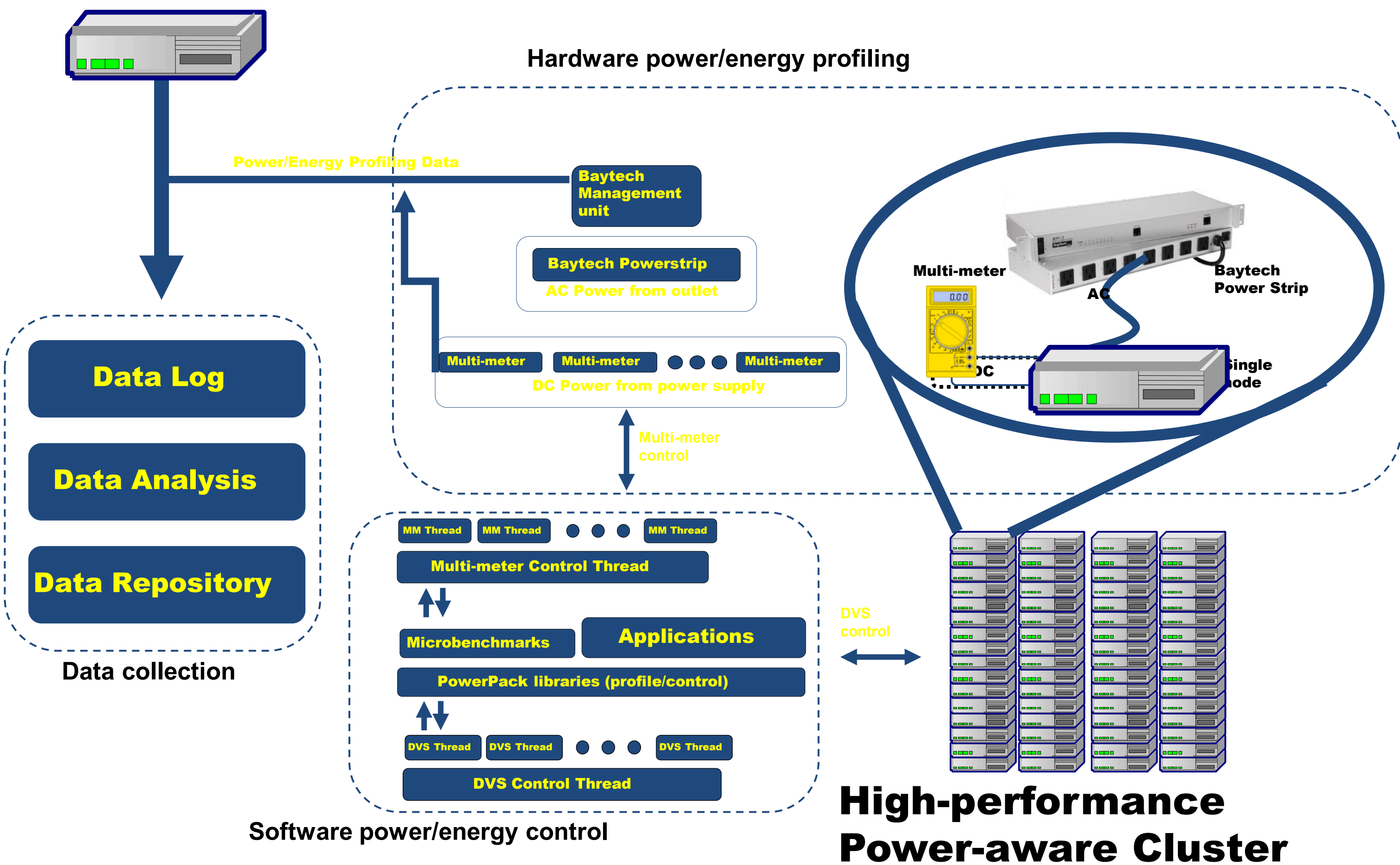


# HPPAC Tools

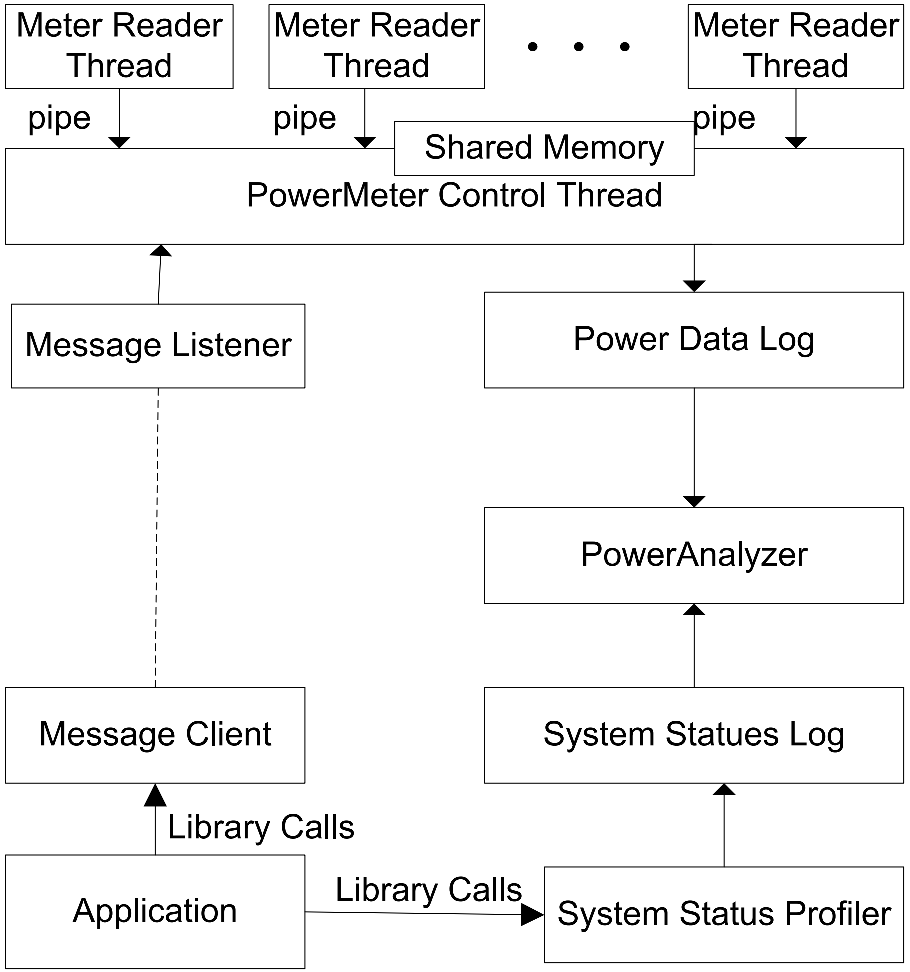
- PowerPack
  - Modularized software + HW sensors
  - Extended analytics for applicability
  - Extended to support thermals
- SysteMISER (evolves to MiserWare/Granola)
  - Improved analytics to weigh tradeoffs at runtime
  - Automated cluster-wide, DVS scheduling
  - Support for automated power-aware memory

# PowerPack

## Scalable, synchronized, and accurate.

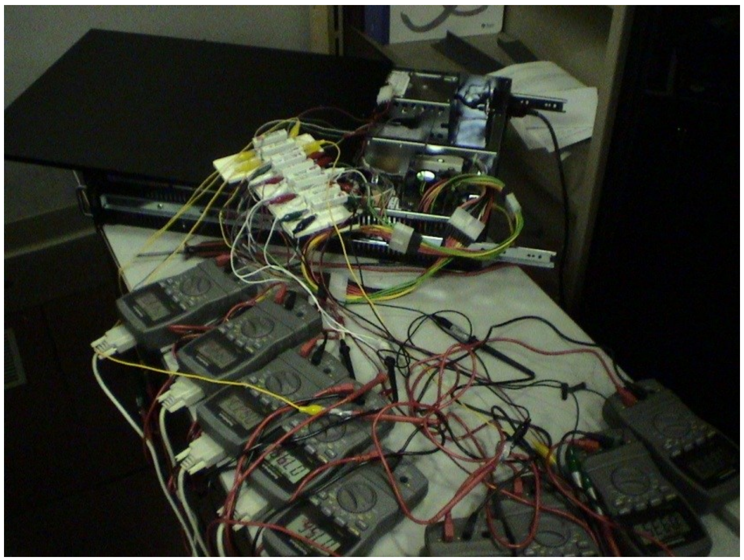


# PowerPack



Meter Reader Thread
Meter Reader Thread
Meter Reader Thread
pipe
pipe
pipe
Shared Memory
PowerMeter Control Thread
Message Listener
Power Data Log
PowerAnalyzer
Message Client
System Statues Log
Library Calls
Application
Library Calls
System Status Profiler

# DC Power Profiling



```
If node .eq. root then
       call pmeter_init (xmhost,xmport)
       call pmeter_log (pmlog,NEW_LOG)
endif

<CODE SEGMENT>

If node .eq. root then
       call pmeter_start_session(pm_label)
endif

<CODE SEGMENT>

If node .eq. root then
       call pmeter_pause()
       call pmeter_log(pmlog,CLOSE_LOG)
       call pmeter_finalize()
endif
```

**Multi-meters + 32-node Beowulf**

# Power Profiles – Single Node

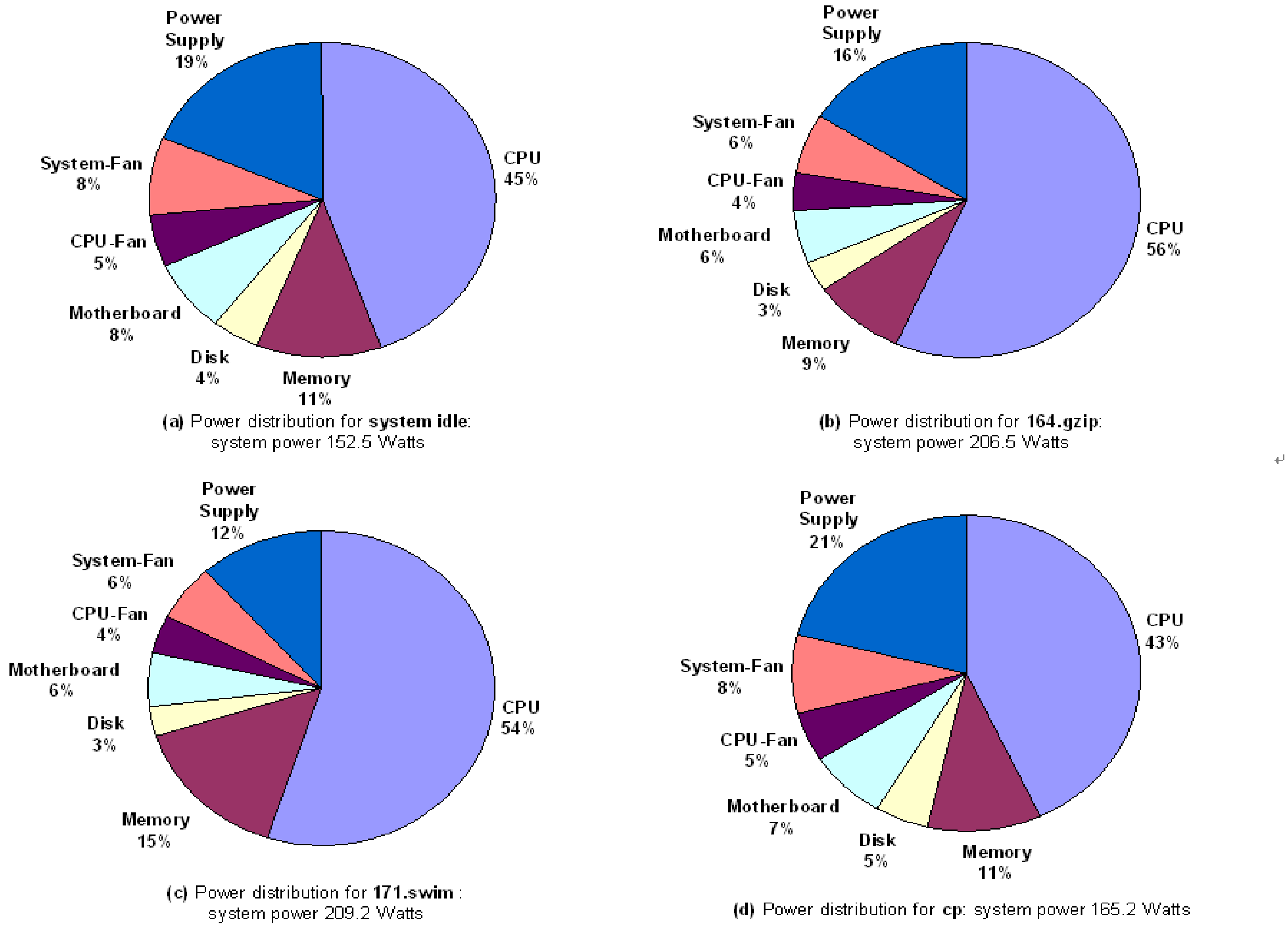
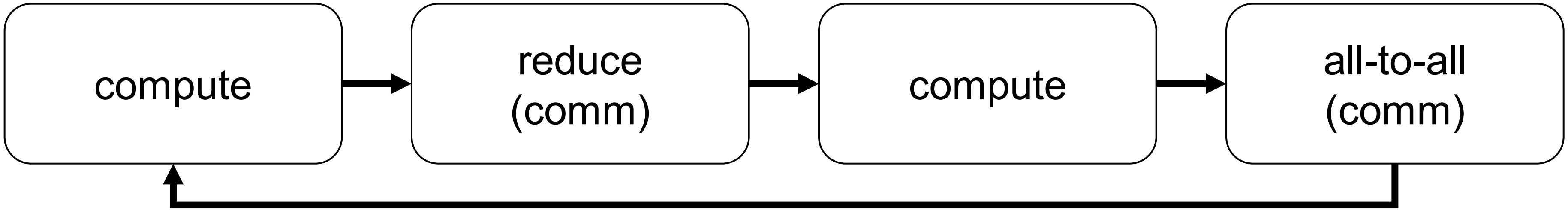



Fig. 5. Power distribution for a single node under different workloads: (a) zero workload (system is in idle state); (b) CPU bounded workload; (c) memory bounded workload; (d) disk bounded workload.

# NAS PB FT – Performance Profiling

Part of the timeline of FT.B.4 visualized by JUMPSHOT

# Power Profiles – Single Node

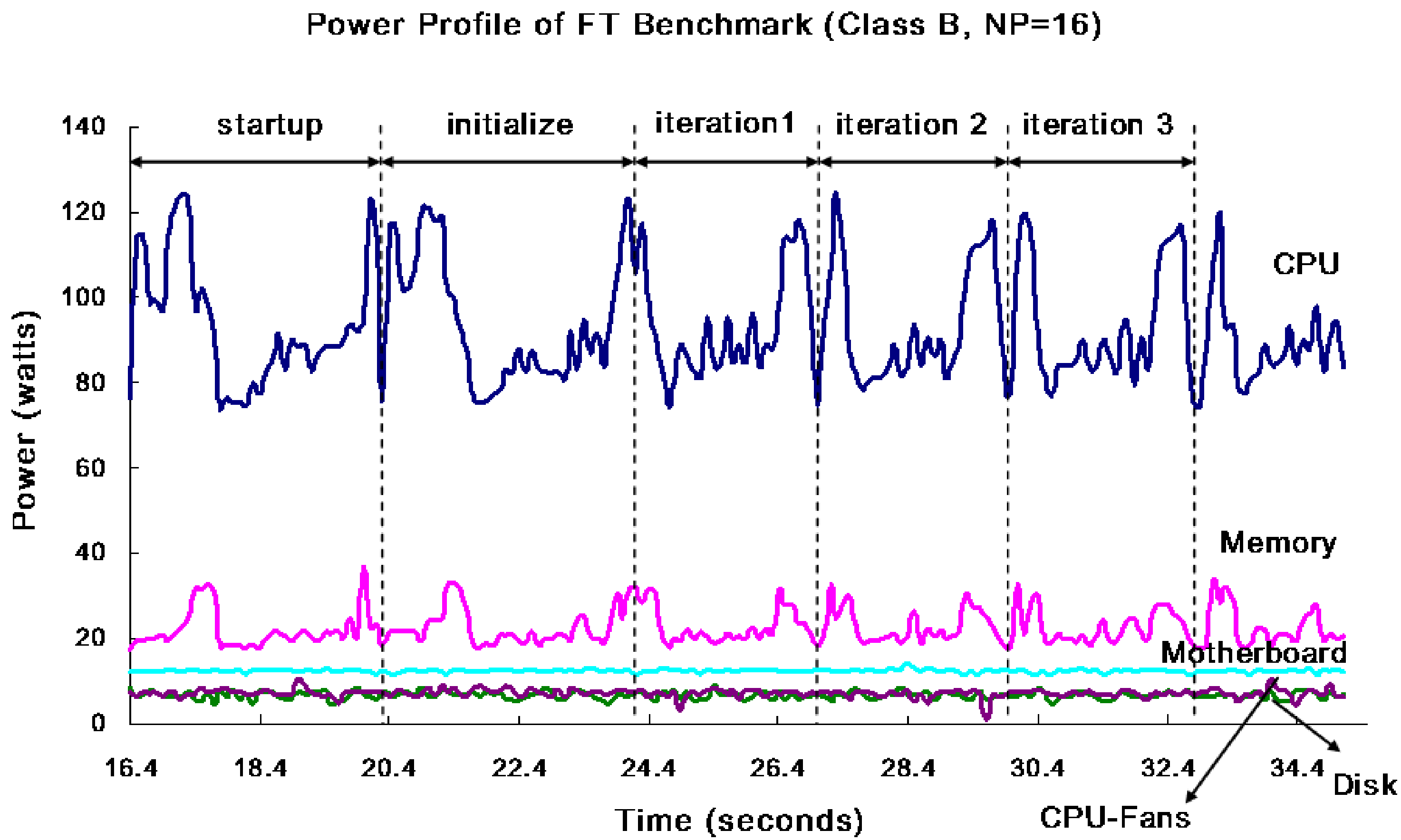
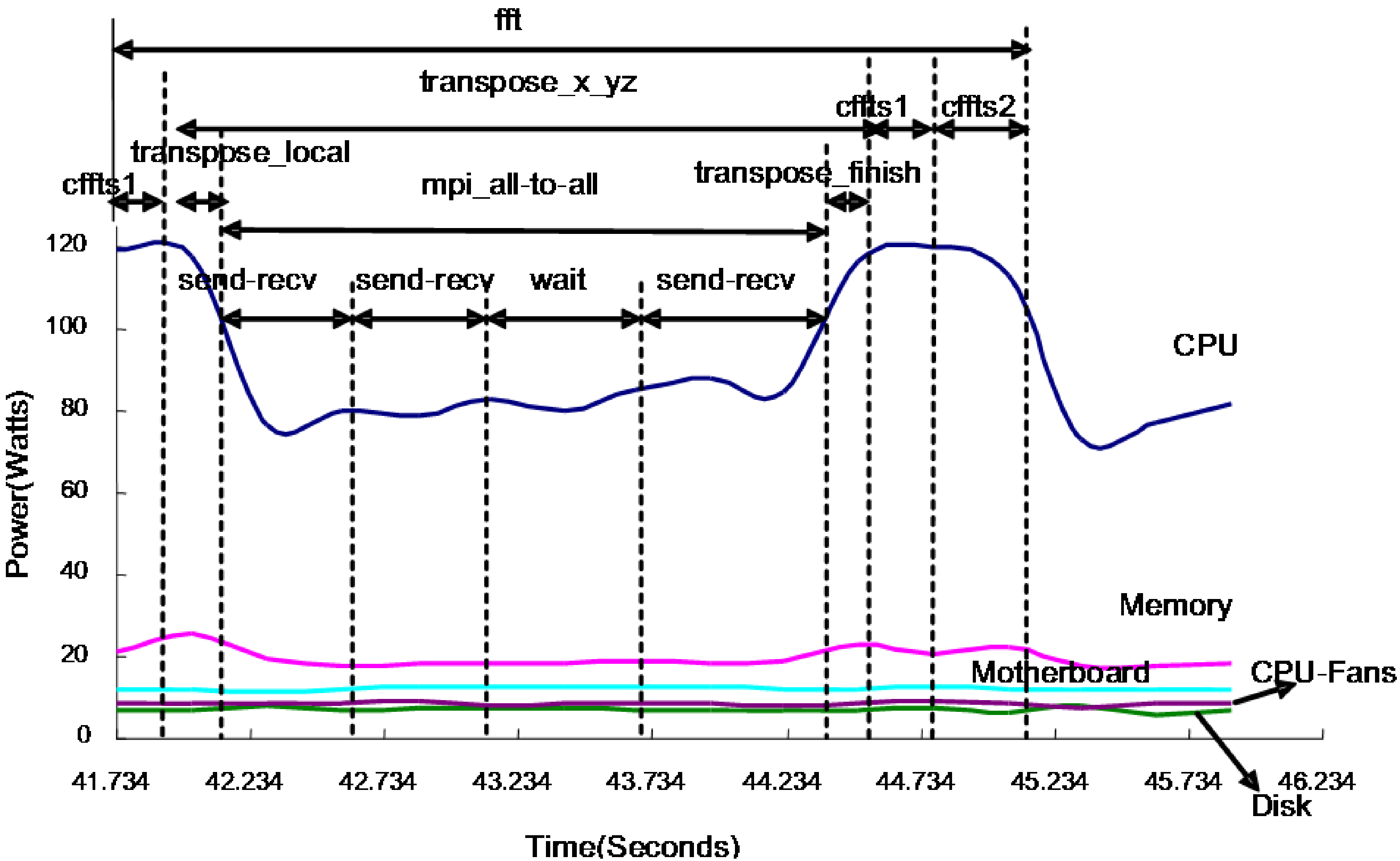


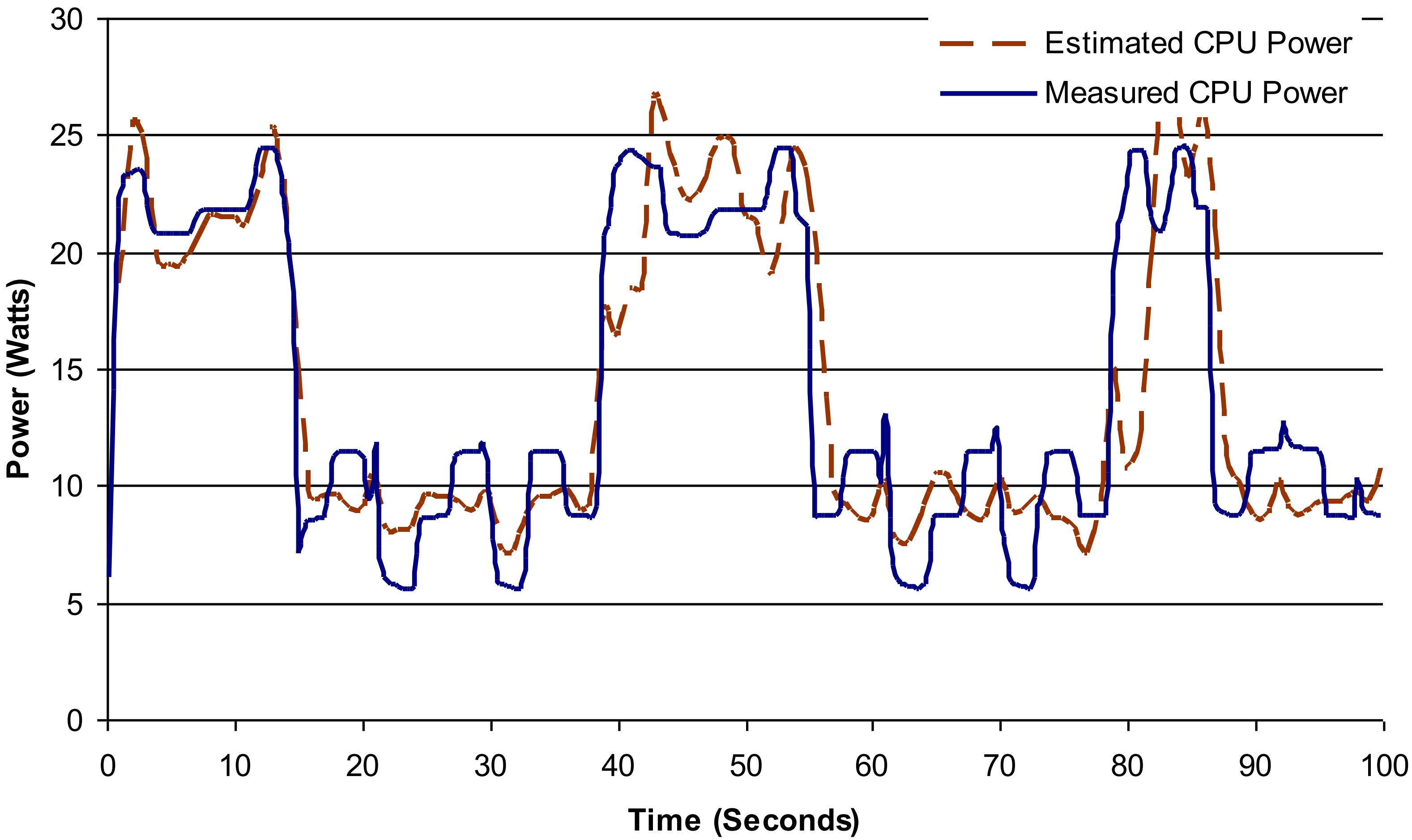

Fig. 6. shows the power use on one node of four for the FT benchmark, class B workload. Note: x-axis is overlaid for ease of presentation.

# PowerPack

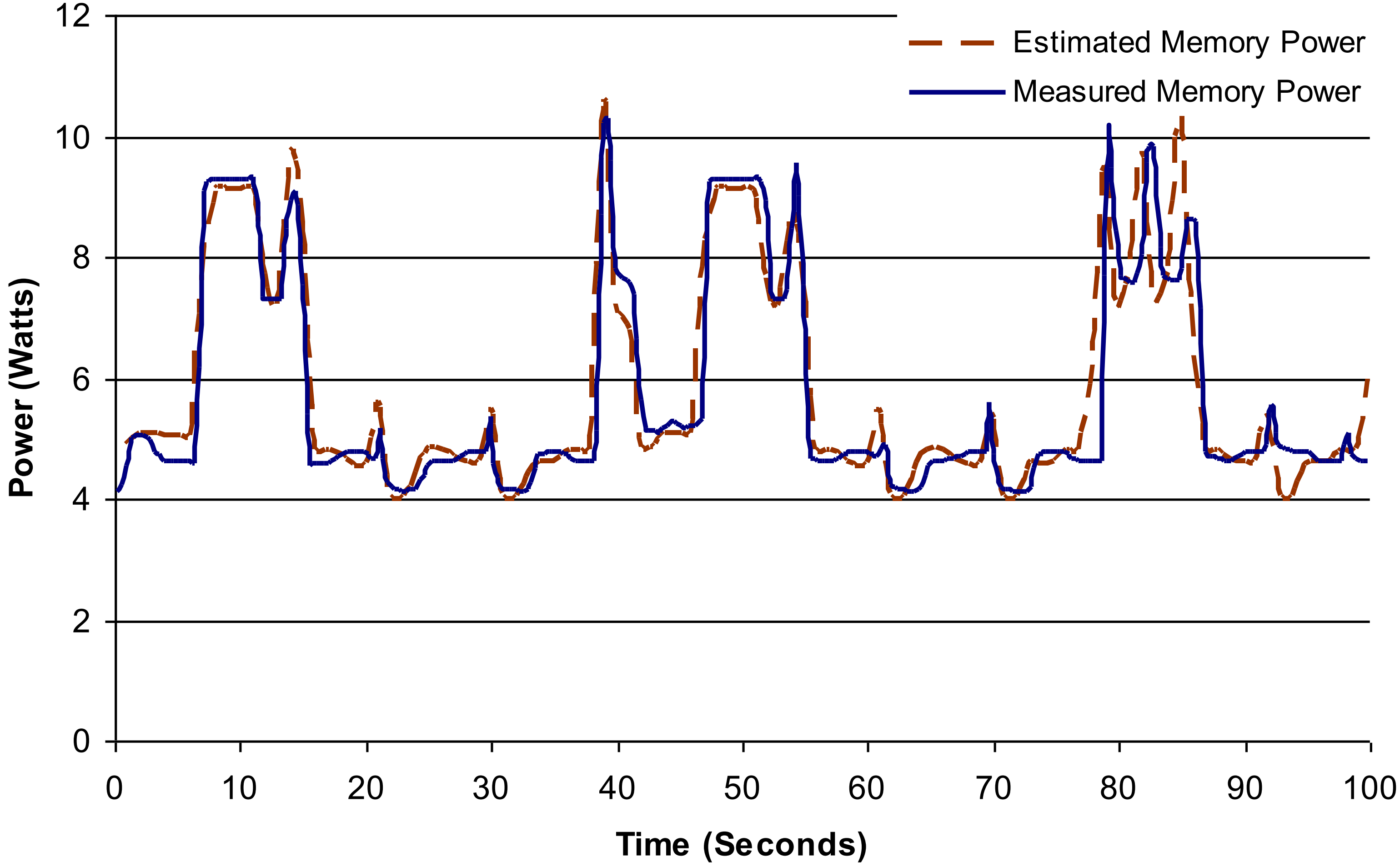


fft
transpose_x_yz
cffts1
cffts2
transppse_local
cffts1
mpi_all-to-all
transpose_finish
send-recv
send-recv
wait
send-recv
CPU
Memory
Motherboard
CPU-Fans
Disk
Power(Watts)
120
100
80
60
40
20
0
41.734
42.234
42.734
43.234
43.734
44.234
44.734
45.234
45.734
46.234
Time(Seconds)

# Predicting CPU Power

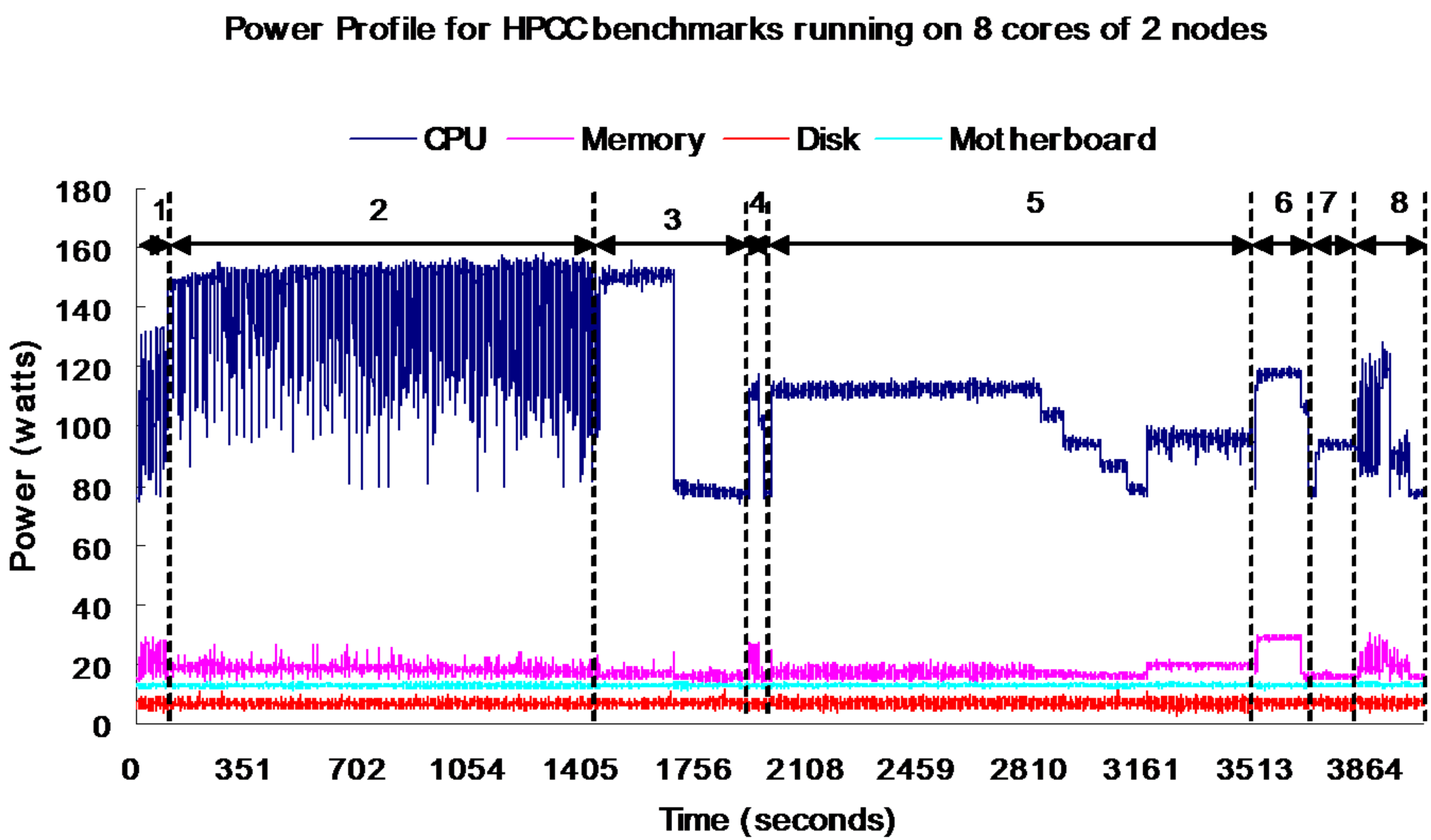


# Predicting Memory Power



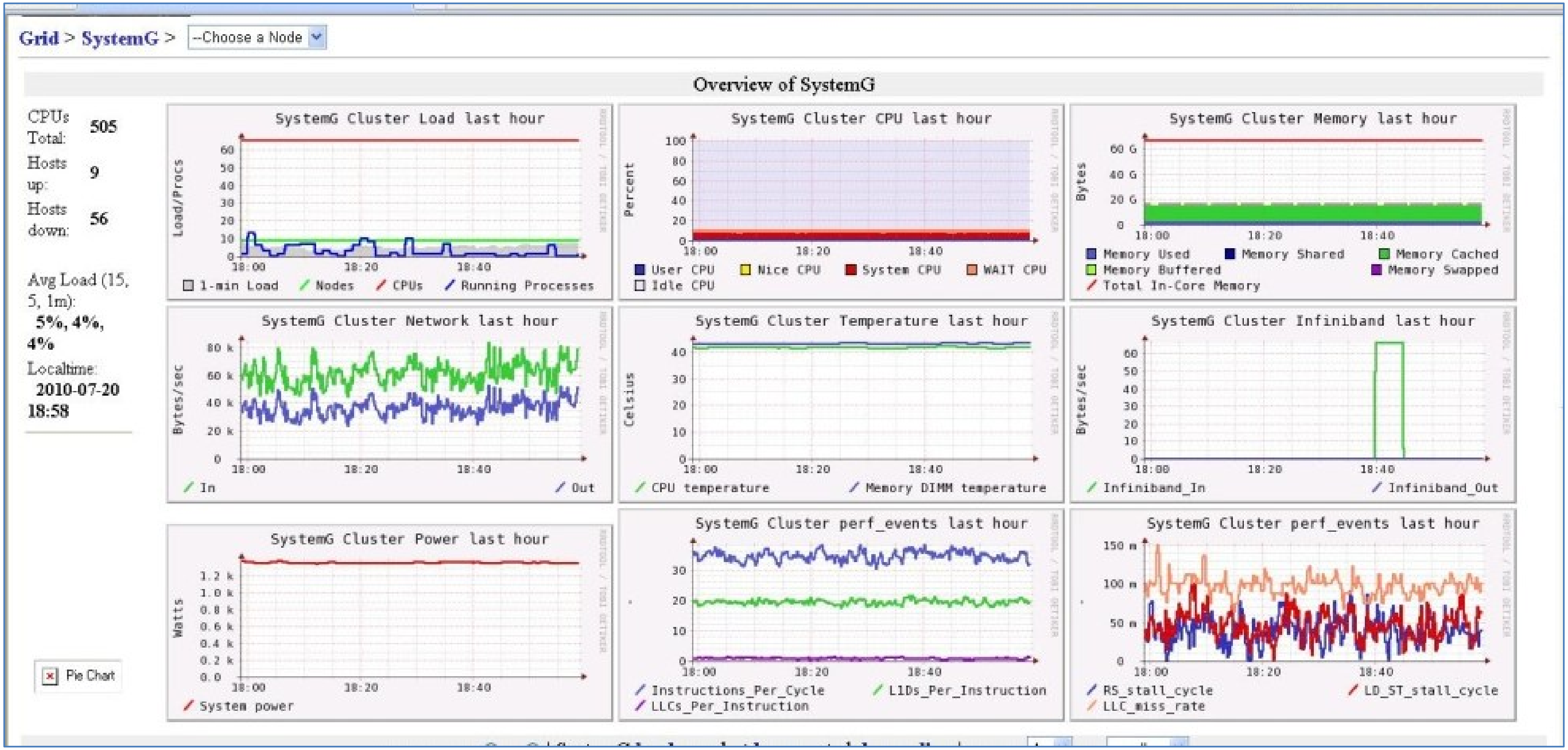
# SystemG Supercomputer

# PowerPack

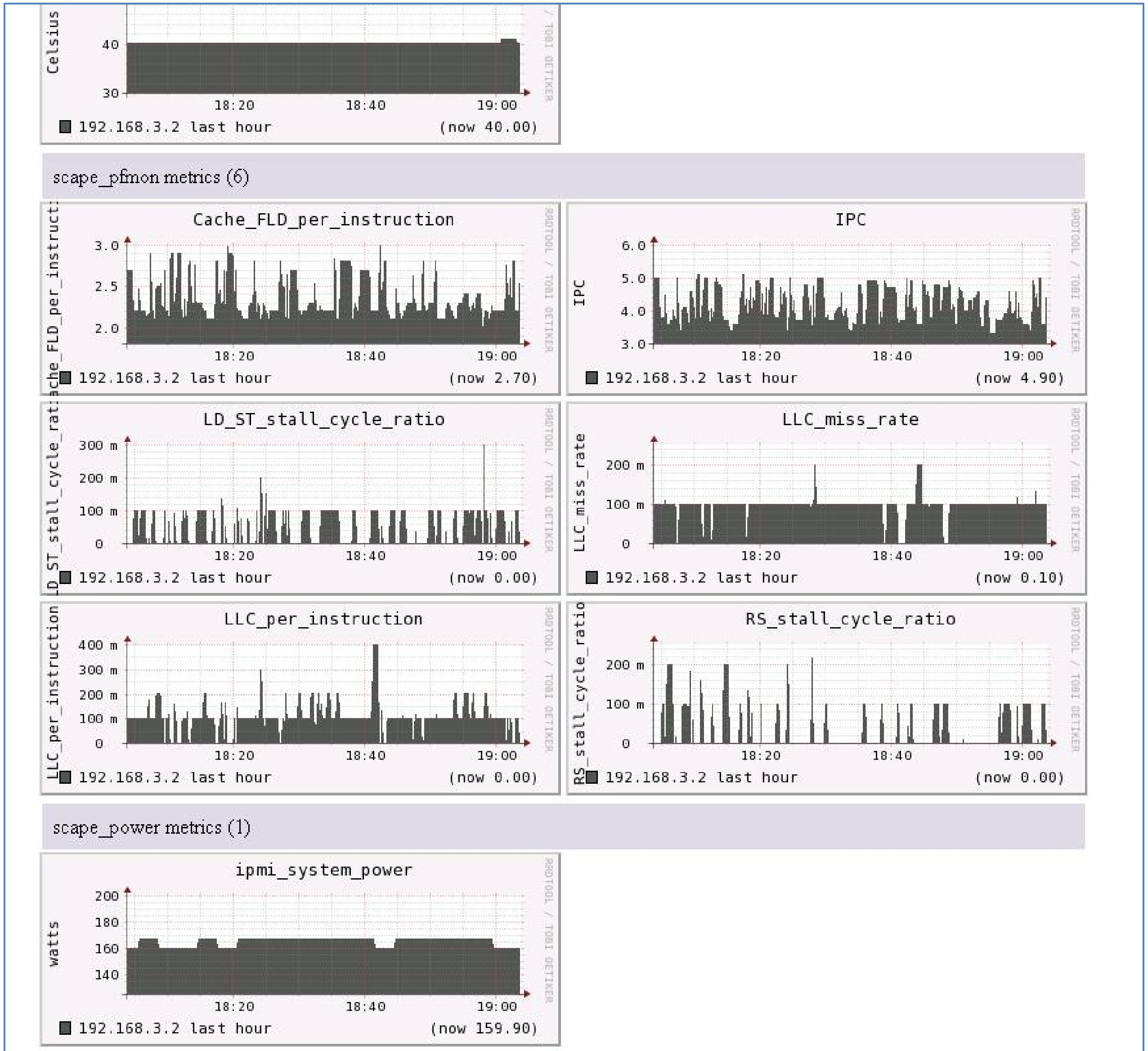


Power Profile for HPCC benchmarks running on 8 cores of 2 nodes
CPU
Memory
Disk
Motherboard
Power (watts)
180
160
140
120
100
80
60
40
20
0
1
2
3
4
5
6
7
8
0
351
702
1054
1405
1756
2108
2459
2810
3161
3513
3864
Time (seconds)

# PowerPack 3.0



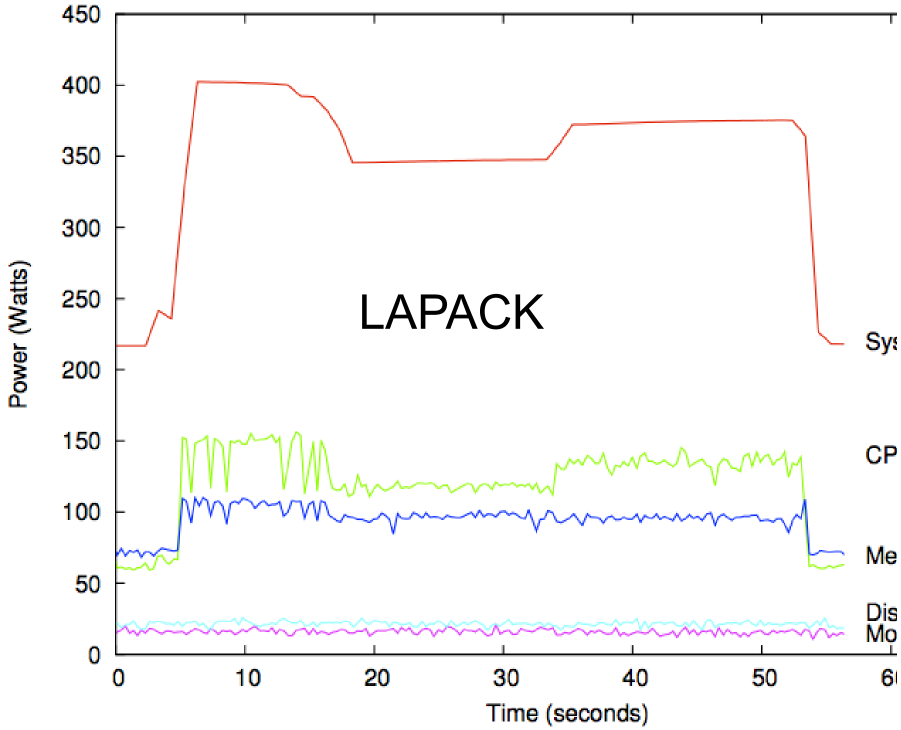
# PowerPack 3.0

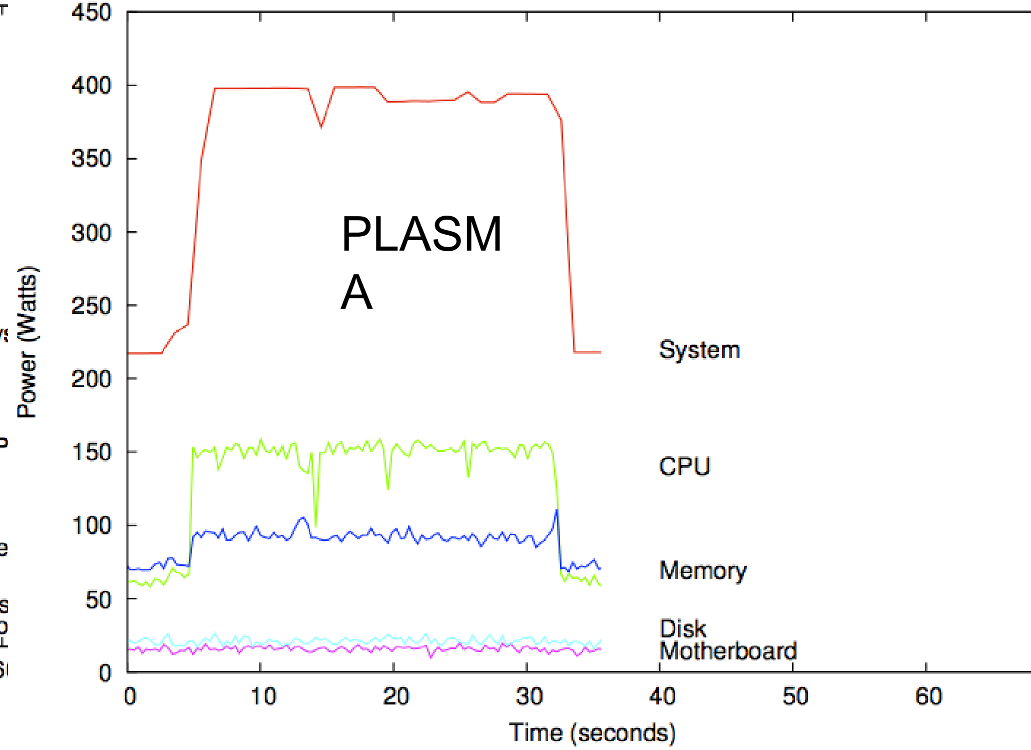


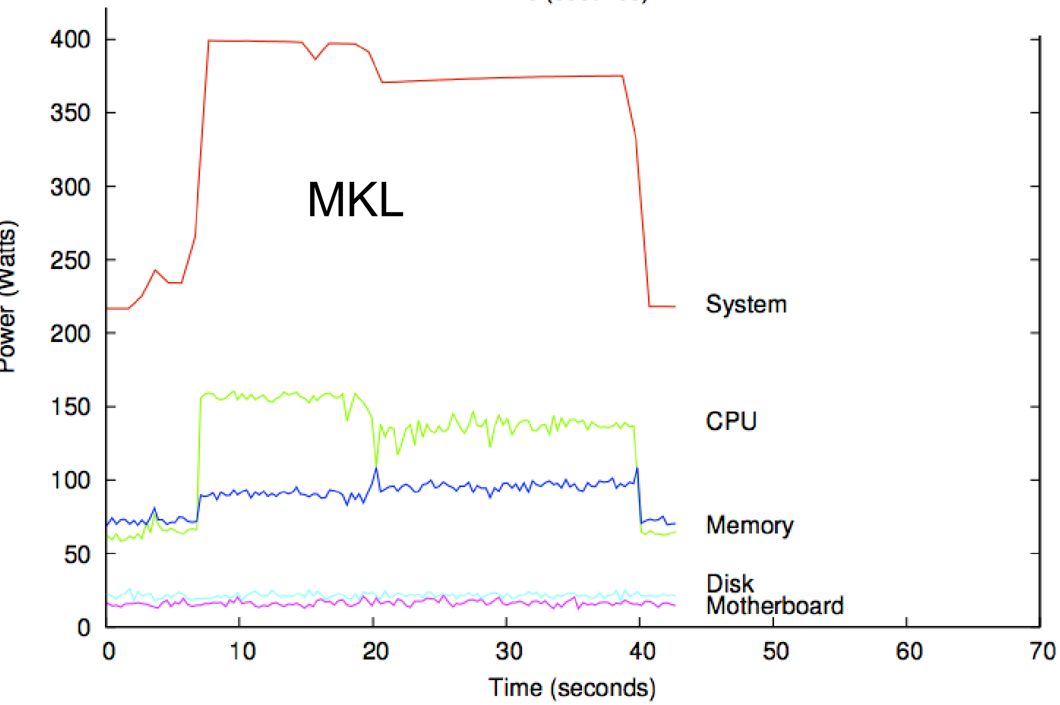
# Who uses PowerPack? SystemG?

- Texas A&M (Taylor et al)
- UTenn-Knoxville (Moore, Dongarra, et al)
- Oxford University
- Lawrence Livermore National Lab
- Pacific Northwest National Lab
- Oak Ridge National Lab
- University of Florida
- KAUST (Saudi Arabia)
- University of Madrid (Spain)

…and many others



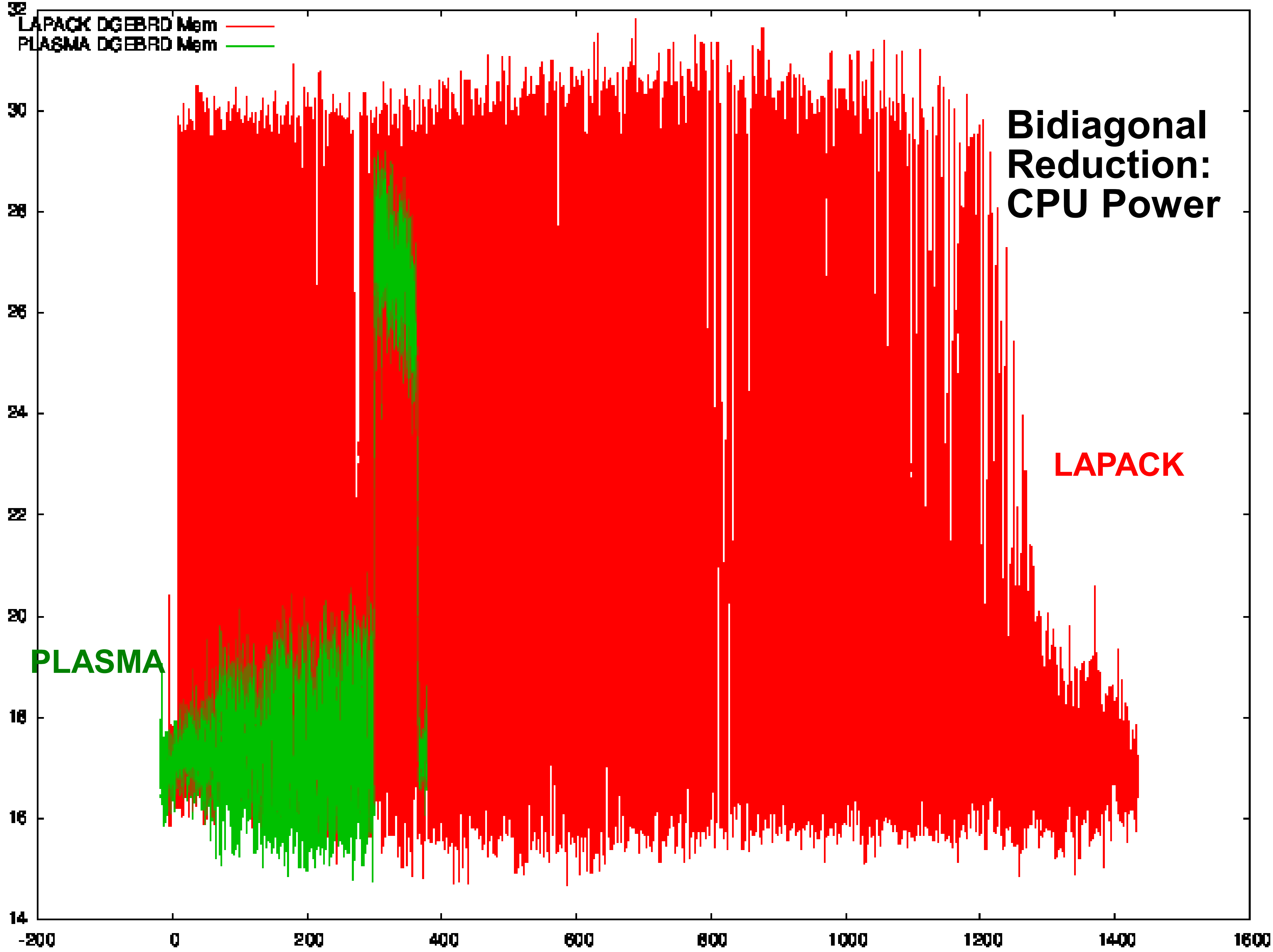




## Power consumption over time

## Matrix inverse

Sources:

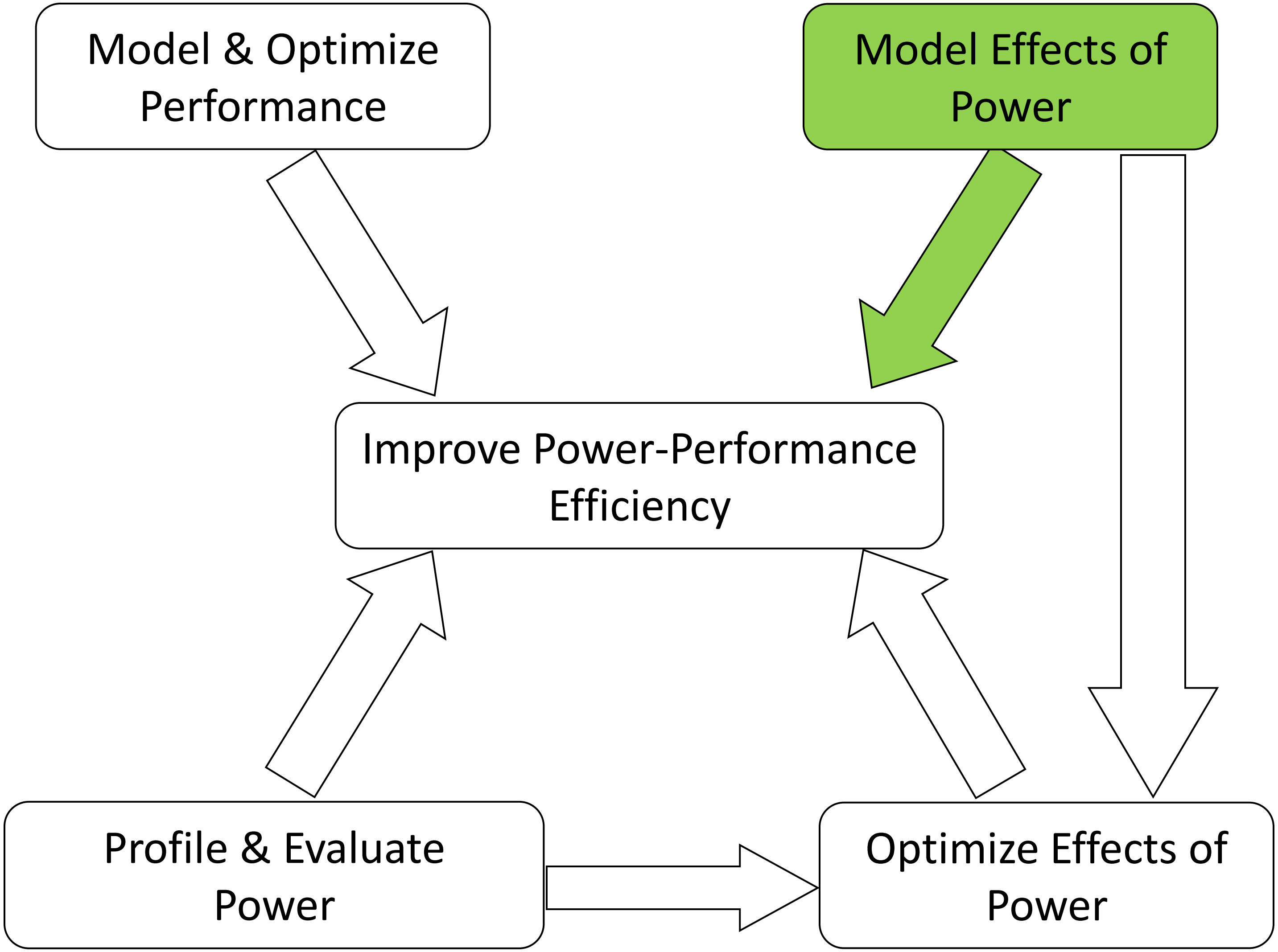
Piotr Luszczek    Hatem Ltaief


LAPACK DGEBRD Mem
PLASMA DGEBRD Mem
Bidiagonal
Reduction:
CPU Power
LAPACK
PLASMA
32
30
28
26
24
22
20
18
16
14
-200
0
200
400
600
800
1000
1200
1400
1600

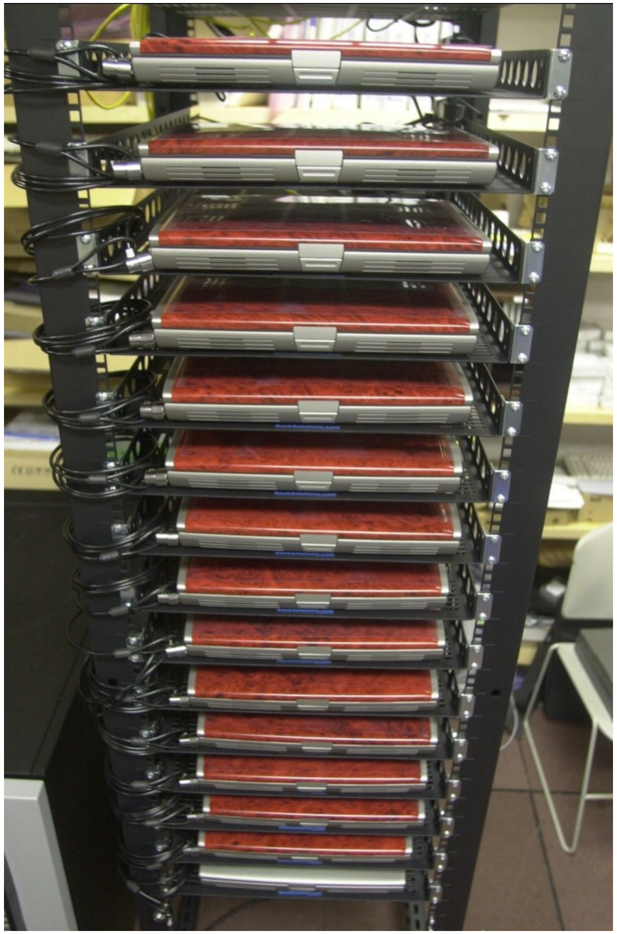
“To know is to understand.”

*Aristotle*

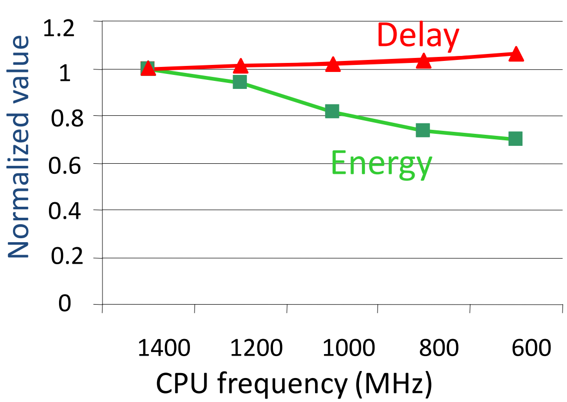
# Power-Performance Efficiency

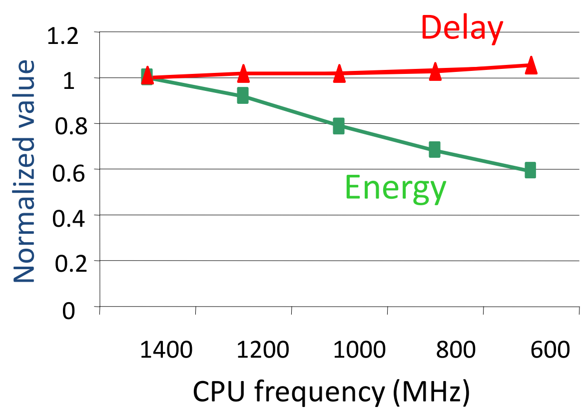


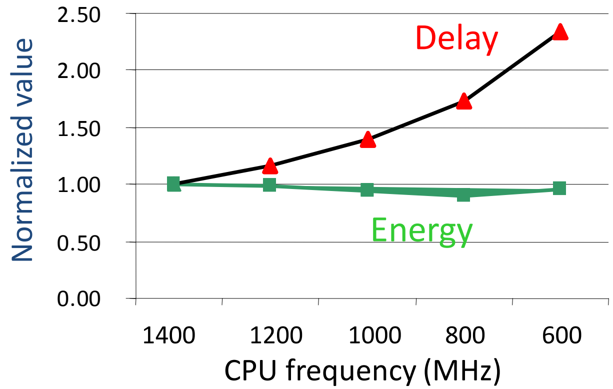
# First power-aware “HPC” cluster



# How DVFS affects HPC efficiency




Communication bound

Memory bound

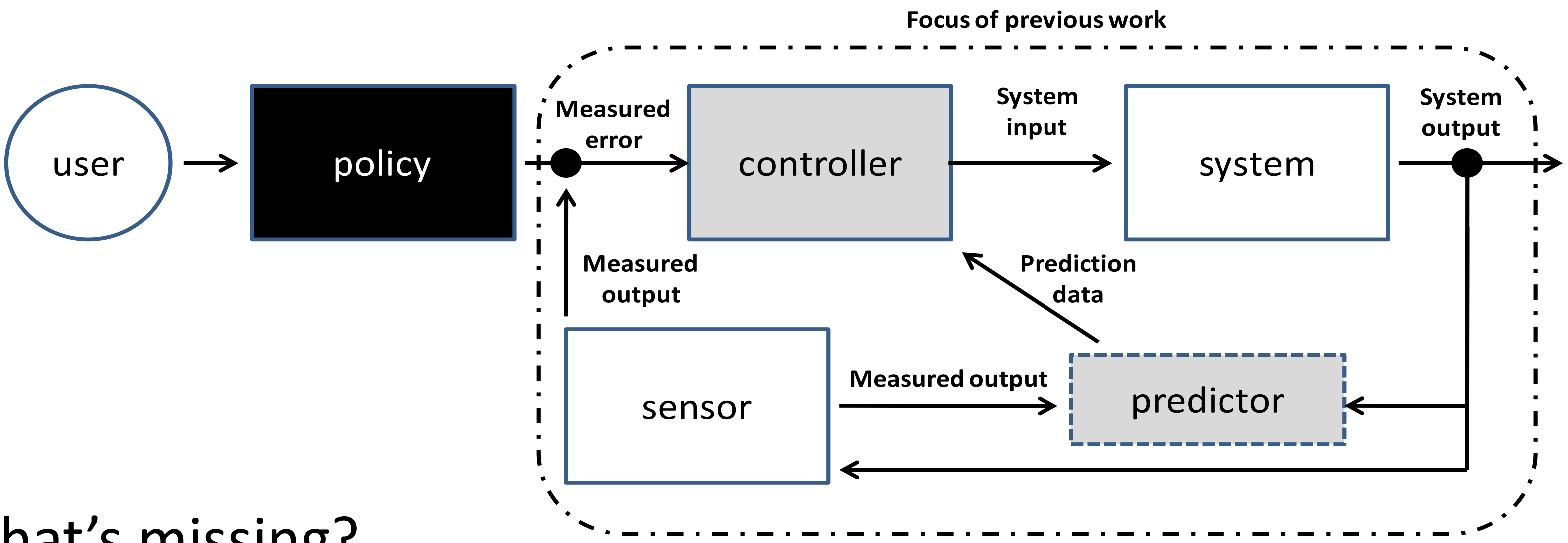
CPU bound

Lower *f* for energy savings with minimal perf. loss

Higher *f* for better perf. and less energy

# Understanding power-performance

Early system level approaches focus on power mode **predictor and controller** design: This is great for *reacting* to change.




What's missing?

What are the bounds on efficiency? In HPC?

How does power-performance quantitatively affect efficiency?

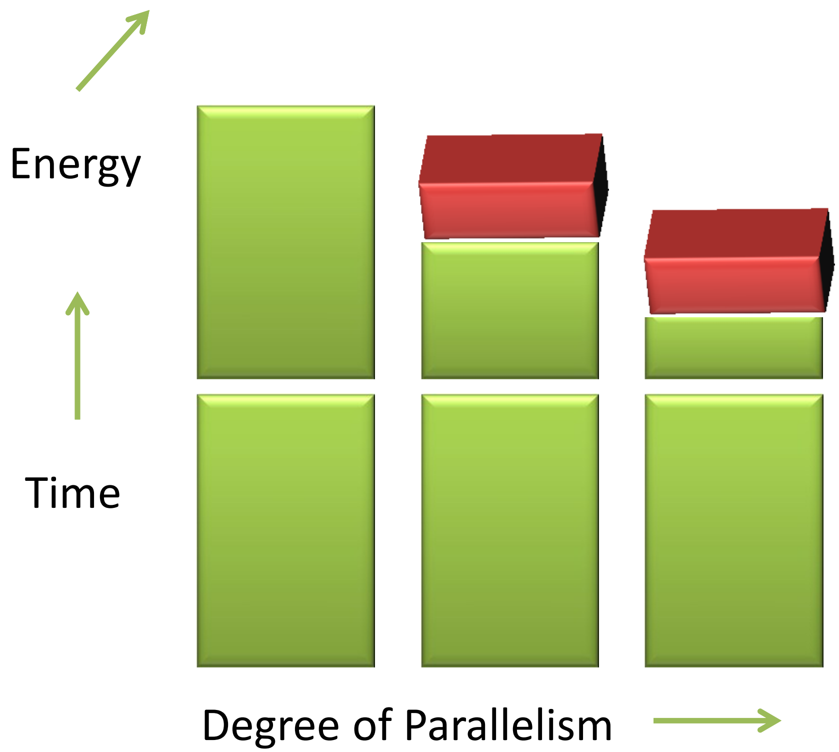
How do we create policies to guarantee power-performance?

Strong need to improve <u>understanding</u> of power-performance.

# Amdahl's Law

- Classical speedup
  - Amdahl's law for 1 enhancement (parallelism)

$$S_N(w) = \frac{T_1(w)}{T_N(w)} = \left[ (1 - FE) + \frac{FE}{SE} \right]^{-1}$$




## Time ~ energy. Right?

So we only get energy savings by reducing time. Right?

Then why does PM (e.g. DVFS) save energy? And sometimes without affecting time?

## Amdahl = no overhead

But, overhead is the key to savings energy without loss!

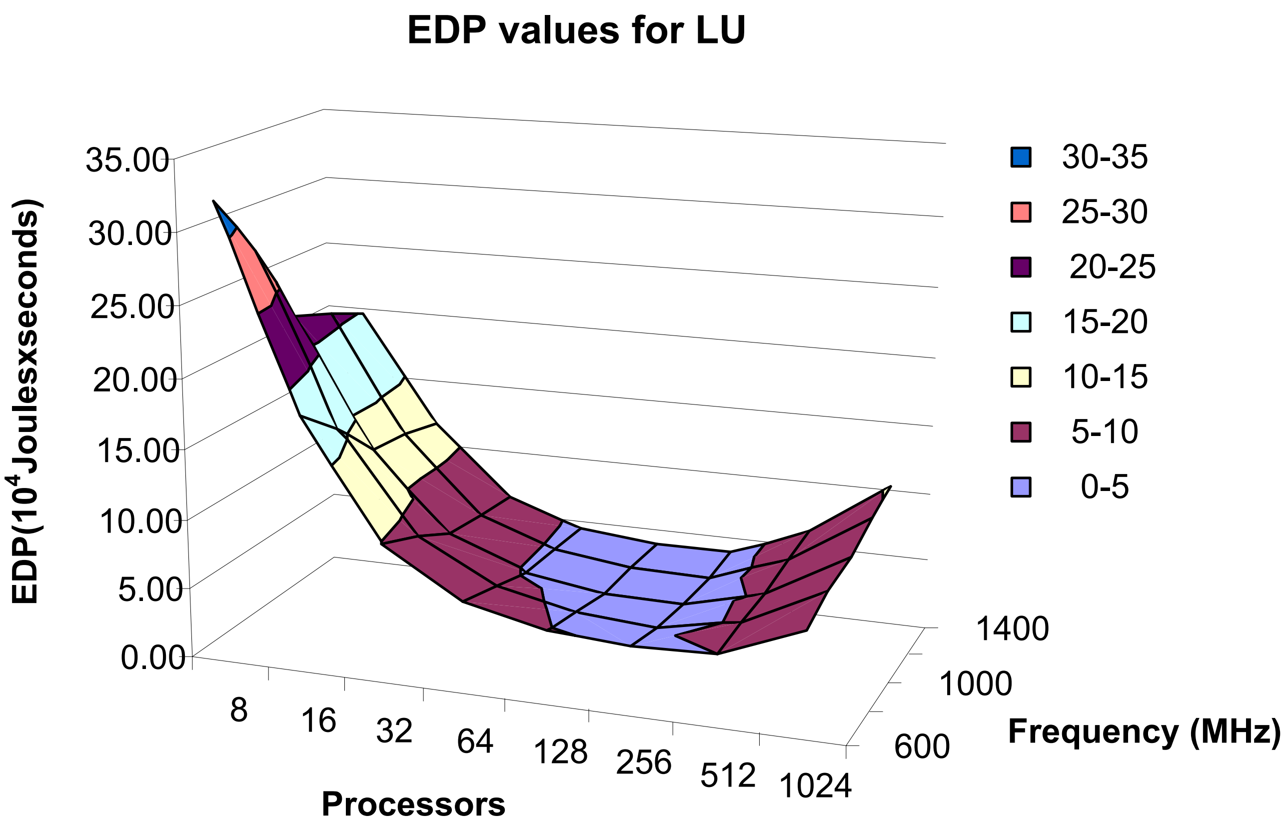
# Power-Aware Speedup

- **Definition**
  - Speedup

$$S_N(w, f) = \frac{T_1(w, f_0)}{T_N(w, f) + O(w, f)}$$

  - *w:* workload
  - *N:* number of nodes
  - *f*: the clock frequency and $f_0$ is the base value
  - $T_1(w, f_0)$: sequential execution time at base frequency $f_0$
  - $T_N(w, f)$: parallel execution time at *N* processors at frequency *f*

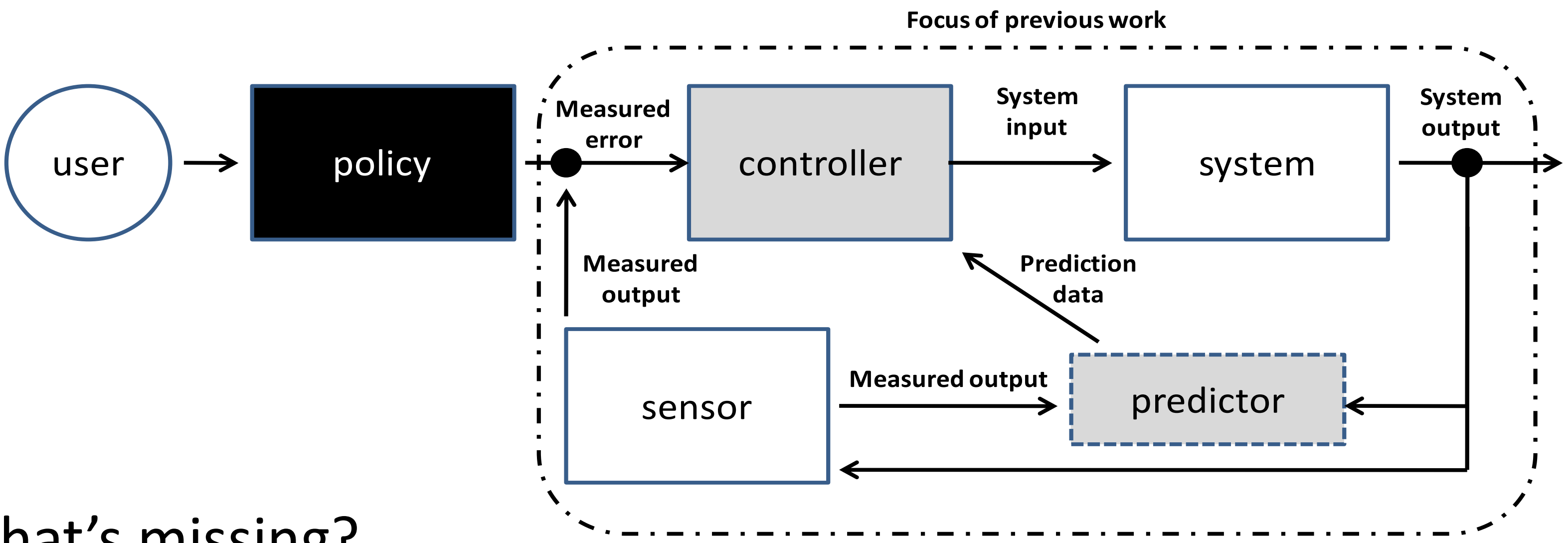
# Bounding Efficiency at Scale




- Optimal system configuration
  - # processors: 256
  - CPU frequency: 1200MHz

# Understanding power-performance

Early system level approaches focus on power mode **predictor and controller** design: This is great for *reacting* to change.




What's missing?

What are the bounds on efficiency? In HPC?

How does power-performance quantitatively affect efficiency?

How do we create policies to guarantee power-performance?

Strong need to improve <u>understanding</u> of power-performance.

# Iso-energy-efficiency

Grama et al: performance efficiency can be held constant if we increase both number of processors and problem size simultaneously.

Algorithm + Scale → fixed performance

Iso-energy-efficiency

Algorithm + Scale + Power Modes → (power, performance)

- Requires accurate performance model
- Requires accurate power model
- Must be accurate, useful, usable

# Iso-energy-efficiency Derivation

General form of our Iso-energy-efficiency model:

$$EE = \frac{E_1}{E_p} = \frac{E_1}{E_1 + E_o} = \frac{1}{1 + {E_o}/{E_1}}$$

$EE$ *: system-wide energy efficiency*

$E_1$ *(baseline):* total energy consumption of sequential execution on one processor

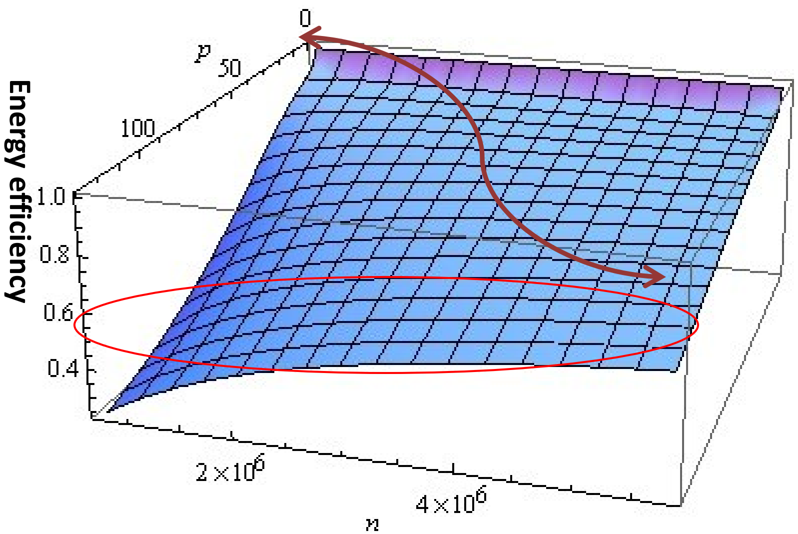
$E_p$ : the total energy consumption of parallel execution for a given application on *p* parallel processors

$E_o$ : the additional energy overhead required for parallel execution and running extra system components

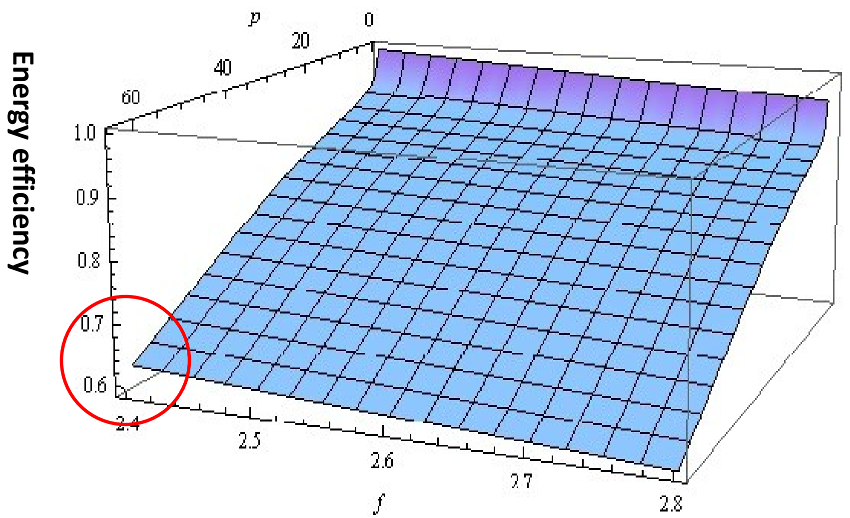
# Maintaining Efficiency in 3-D FFT

$$EE_{FFT} = \frac{1}{1 + \frac{6.87 log_2 p - 1.75 f log_2 p + p(p-1) f (\frac{11500}{n} + \frac{0.376}{4^{log_2 p - 2}})}{163 + 22.7 f}}$$

FT's system-wide energy efficiency with p and n as variables



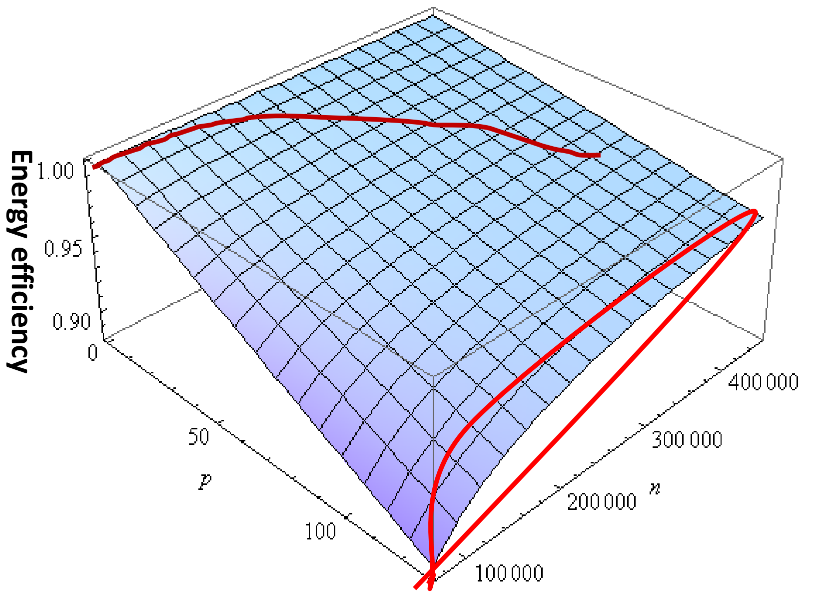

FT's system-wide energy efficiency with p and f as variables



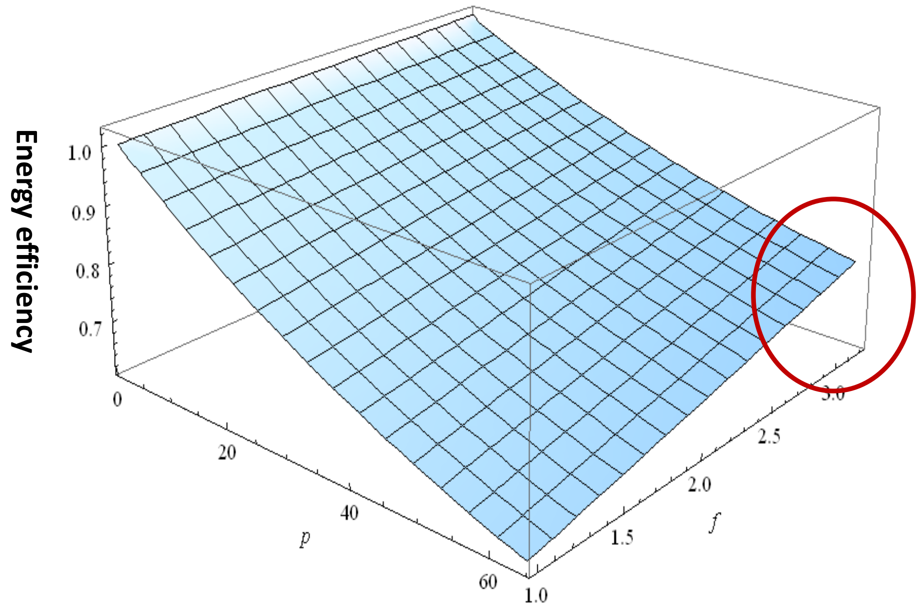

- ***Problem size scaling effective in maintaining overall system energy***
- ***CPU frequency scaling: only slightly improves EE***
- ***But, the effects of CPU clock frequency on on-chip workload diminish while scaling up system size.***

# Maintaining Efficiency in CG

CG's system-wide energy efficiency with p and n as variables




CG's system-wide energy efficiency with p and f as variables




- *Overall EE decreases with system size*
- *EE can be maintained or improved by scaling up problem size N.*
- *Applying higher frequency will improve system-wide EE while system size scales up.*
- *In contrast to FT, effects of frequency on on-chip workload diminish at a slower rate.*

“Those that can, do.
Those that can’t, complain.”

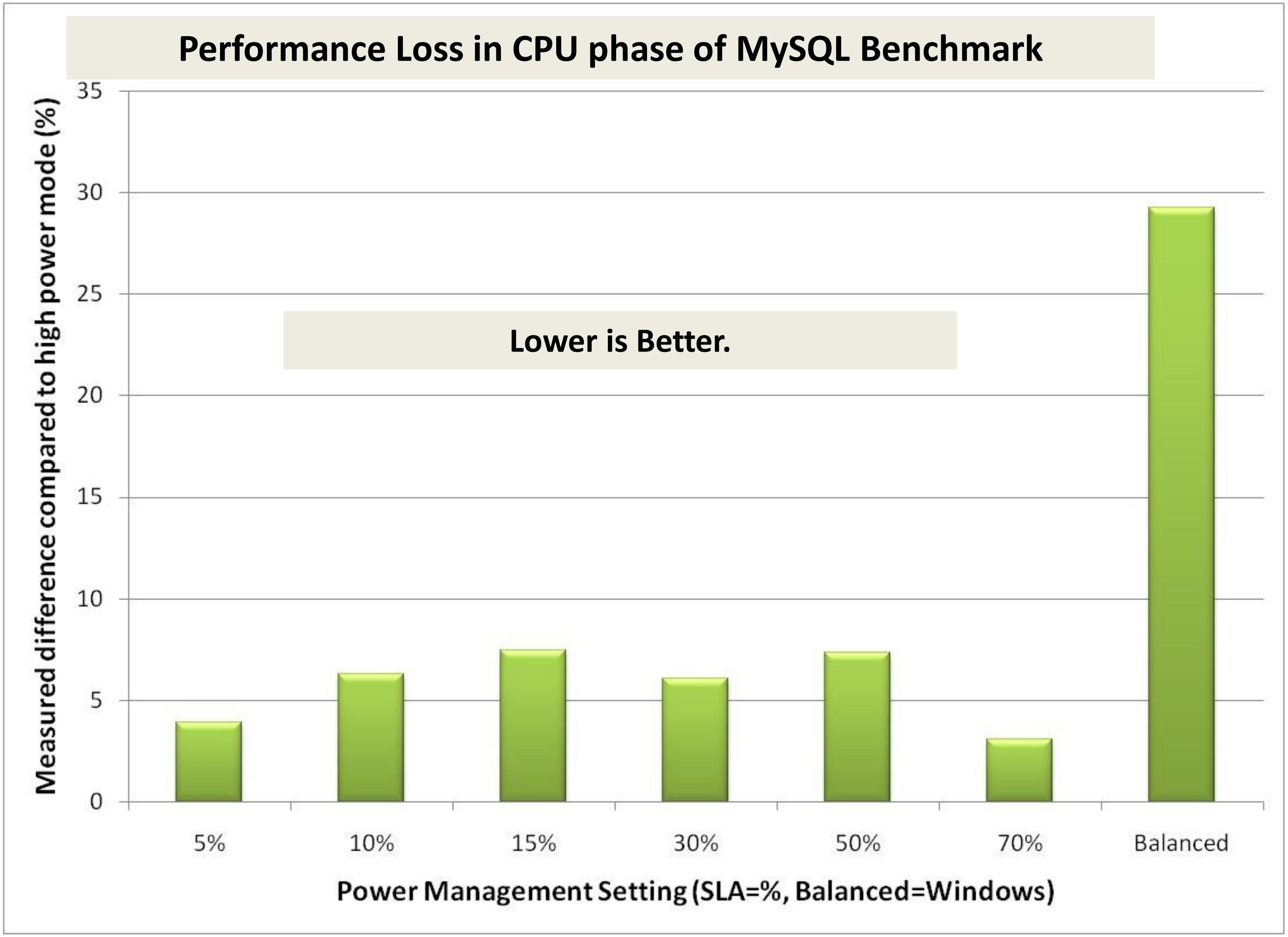
*Linus Torvalds*

# State of the art PM

**Amount and cost of power continues to increase.**

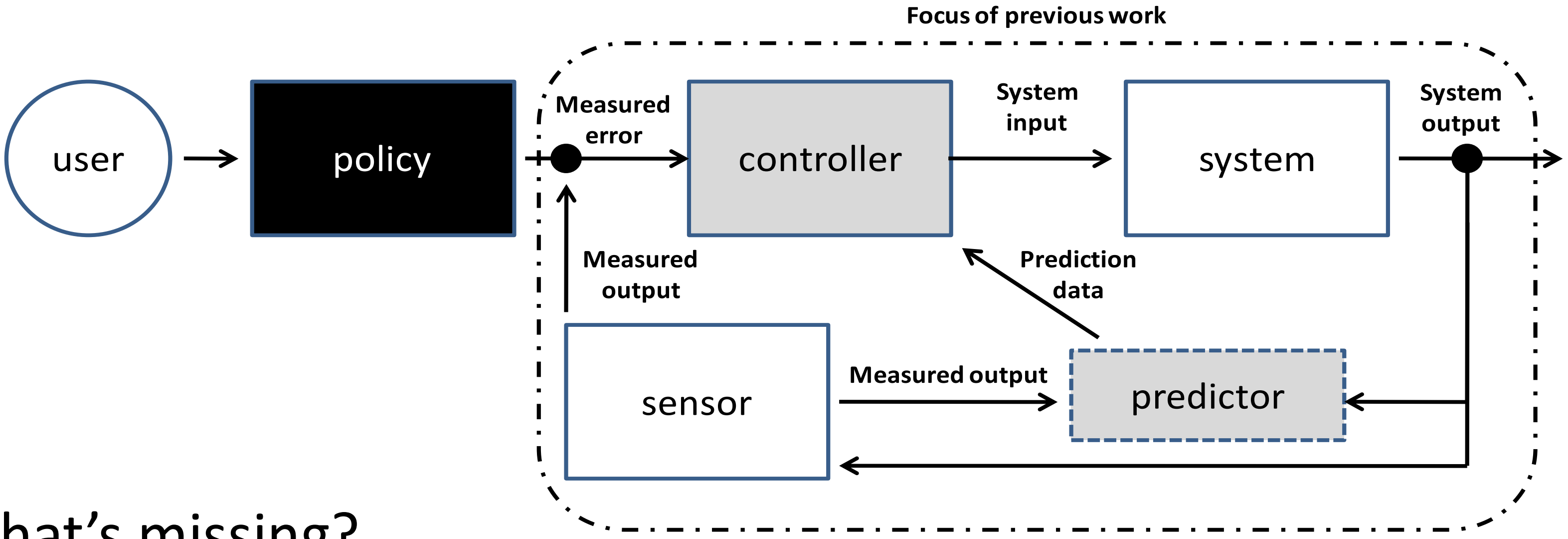

**Power management features disabled by default.**

# Why is PM turned off?



# Understanding power-performance

Early system level approaches focus on power mode **predictor and controller** design:  This is great for *reacting* to change.




What's missing?

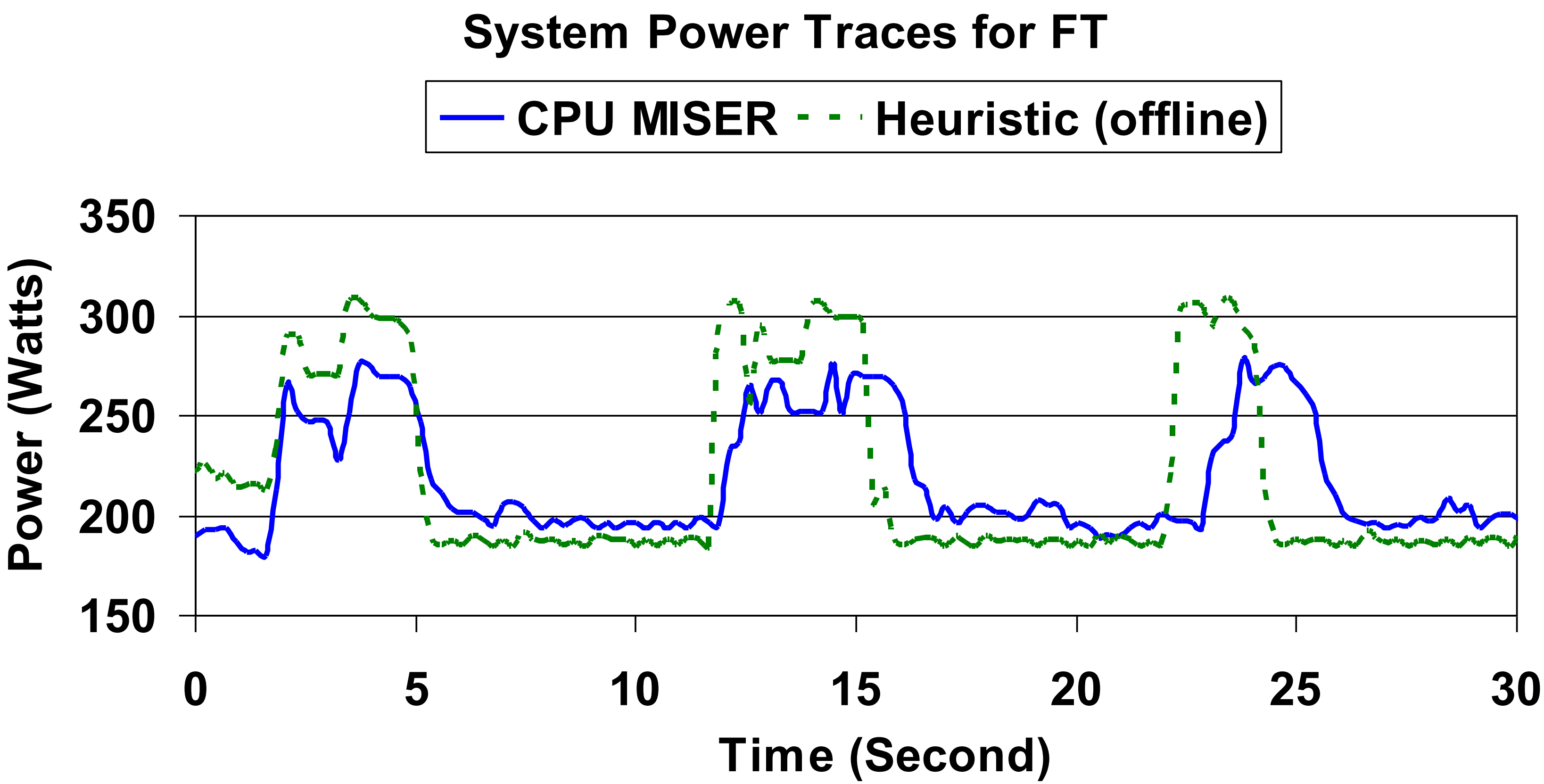
What are the bounds on efficiency? In HPC?

How does power-performance quantitatively affect efficiency?

How do we create policies to guarantee power-performance?

Strong need to improve <u>understanding</u> of power-performance.

# Model-directed Scheduling



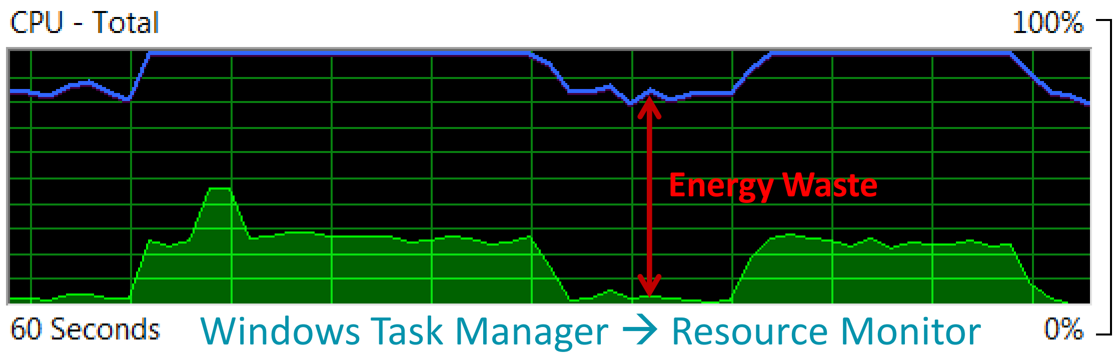

- Automatically and transparently schedule CPU frequency to reduce power

# Better SW for the masses…

## Before Granola:

Power

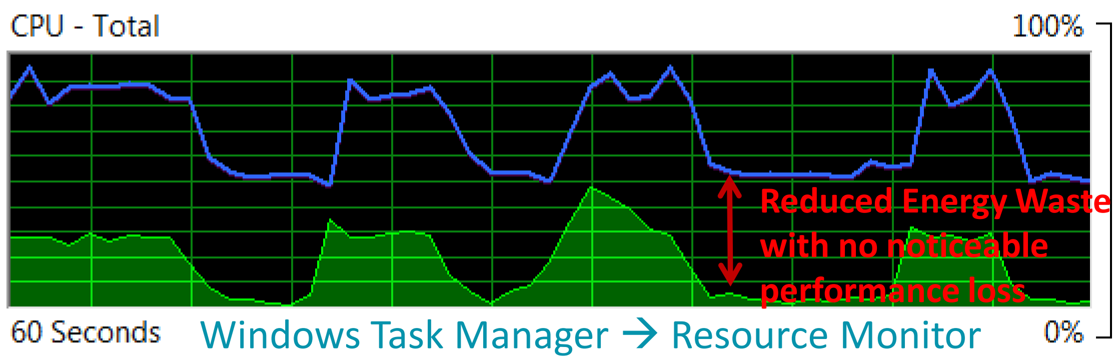
CPU Utilization




## After Granola:

Power

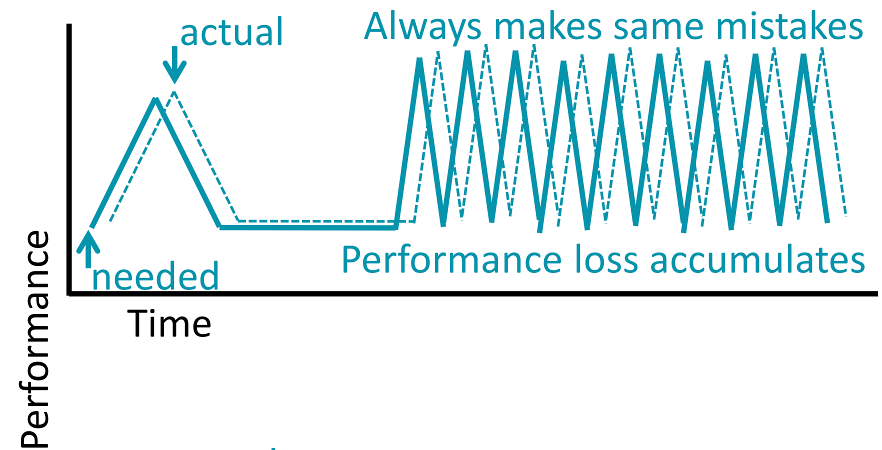
CPU Utilization



# Reducing IT Costs *Without* Performance Loss

EVERYONE ELSE[1]
Intel, HP,
Windows,
VMWare, …

current info →
past info →
adjust power modes

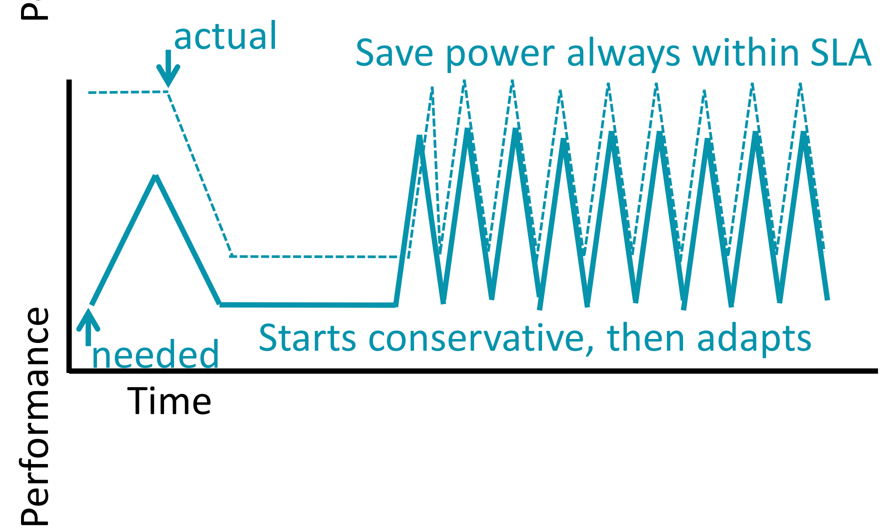



GRANOLA
POWER TUNING

Performance Guarantee Technology

current info →
past info →
user-defined SLA →
current level of service →
adjust power modes

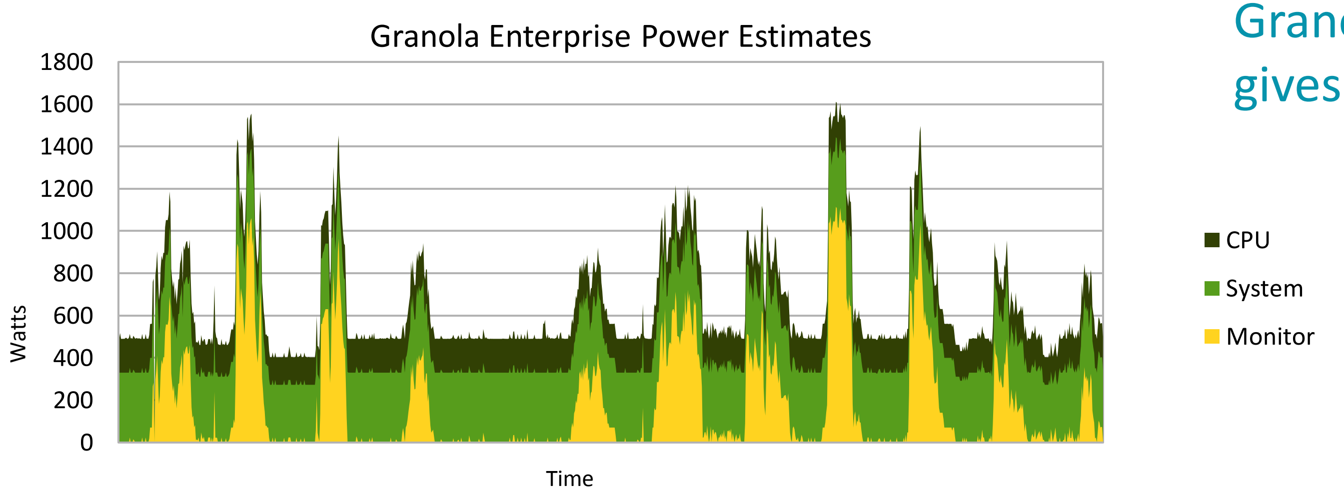
algorithm metric feedback



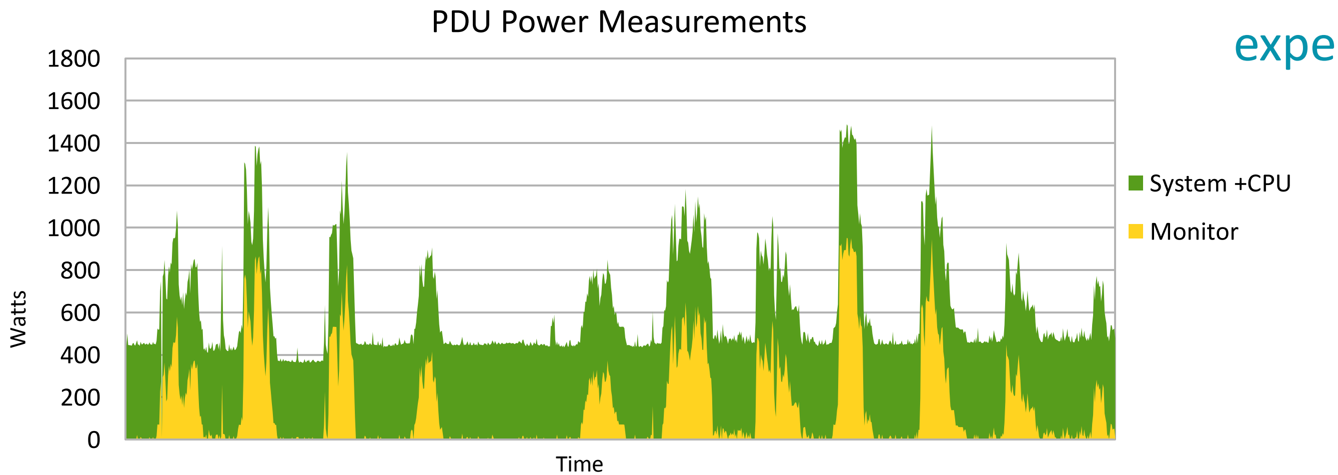



[1]Note: Verdiem, 1E, and others *only* turn systems off when not in use. We offer that too as needed.

# Commercial grade measurement…



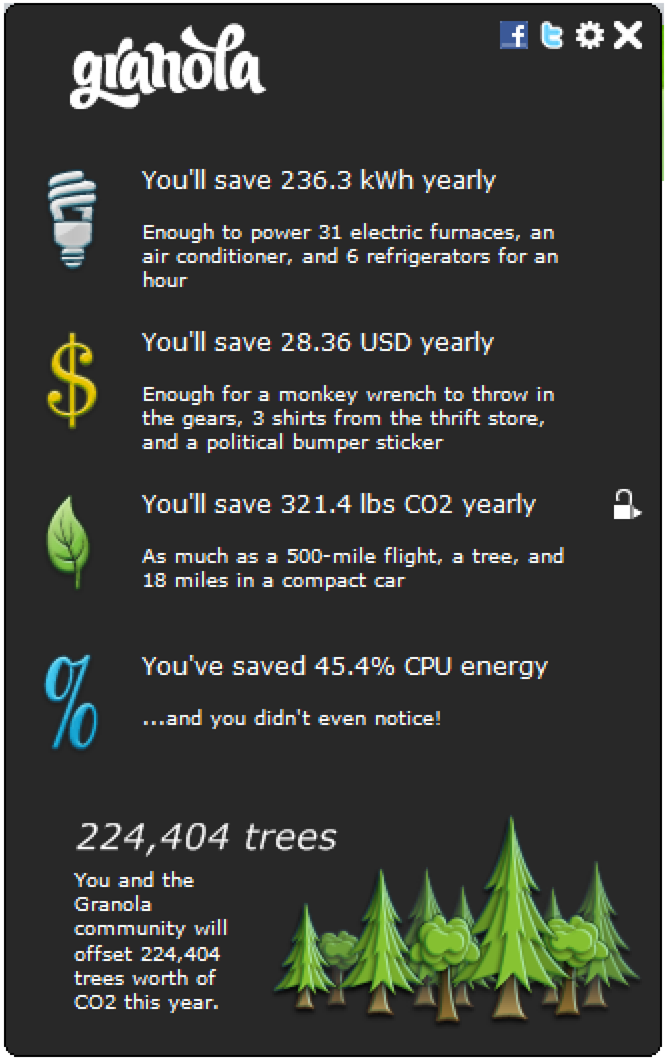

Granola software gives more detail…




…same accuracy as expensive hardware

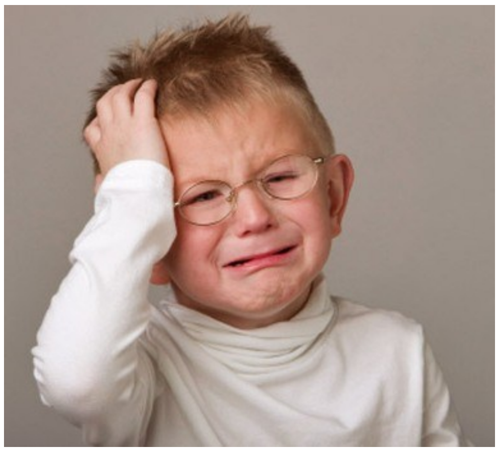
# Granola (http://grano.la)

- Launched Earth Day 2010
- Free home version
- 300K+ Downloads so far…
- 160+ Countries
- Uses: laptops, PCs, servers
- *Performance Guarantees*



# The hard truth about the future

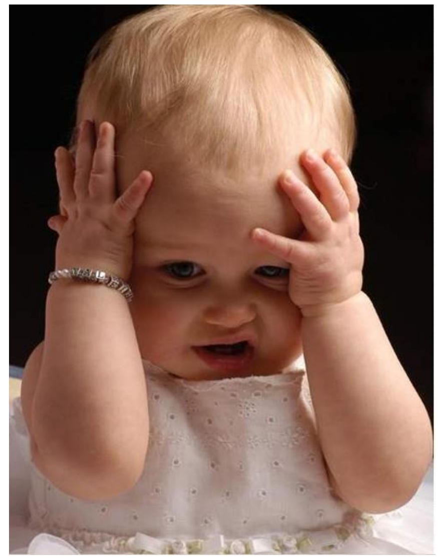
## Measurement



DEFINITIONS

- Experts needed
- Easy to get a wrong answer/conclusion
- Scalability questionable

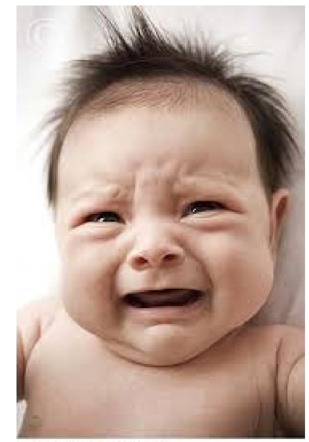
## Analysis



CHAOTIC

- Power-performance relationship not well understood
- How can we help?
- Who are we helping?

## Optimization



CONTENTIOUS

- Many point solutions
- Reactive
- Making something no one wants

# Where do we go from here?


We need lots of help.

Disruptive vs. Incremental.

Silver bullet is unlikely.

Commodity matters.

Markets matter.

Tools matter.

Wanted: Major catastrophe.

Custom system is likely the only answer by 2019. Energy wall?

"Victory" is inevitable when you change the game.

# Thank you.

# Fine-Grain Parameterization

- Assumptions
  - Workload perfectly parallelizable: $T_s^{on}=T_s^{off}=0$
- Methodology
  - Measure system prior to application execution
    - CPI/f for on-chip workload for all frequencies
    - $t^{off}$ for off-chip workload
    - Empirically estimate $T_{PO}$
  - Profile workload at base frequency
    - Accesses for on-chip workload
    - Accesses for off-chip workload
  - Predict perf of node and frequency combinations