Power Consumption of Clusters Control and Optimization

Ena-HPC, Sep 2-3, Dresden

Luigi Brochard (luigi.brochard@fr.ibm.com)
Raj Panda (panda@us.ibm.com)
Francois Thomas (ft@fr.ibm.com)

ethink High Performance Computing.
ata-intensive. Energy-efficient. Intuitive.

The Power Problem

A 1000 node cluster with
2 x86 sockets, 8 cores, 2.7 Ghz
consumes **340 kW** (Linpack)
not including cooling

In Europe (0.15€ per Kwh)
441K€ per year
In US (0.10$ per Kwh)
US$ 295K per year
In Asia (0.20$ per Kwh)
US$ 590K per year

Several ways to reduce power

Use better cooling (Direct Water Cooling)
Reduce power distribution losses
Choose processors with high Flops/Watt
Use power and energy aware tools
Tune the applications

Several ways to reduce power

Data center (PUE reduction)

- Use better cooling (Direct Water Cooling)
- Reduce power distribution losses

Hardware, microprocessor technologies

- Choose processors with high Flops/Watt

Software

- Use power and energy aware tools
- Tune the applications

Several ways to reduce power

Before your RFP starts

- Use better cooling (Direct Water Cooling)
- Reduce power distribution losses

Outcome of your RFP

- Choose processors with high Flops/Watt

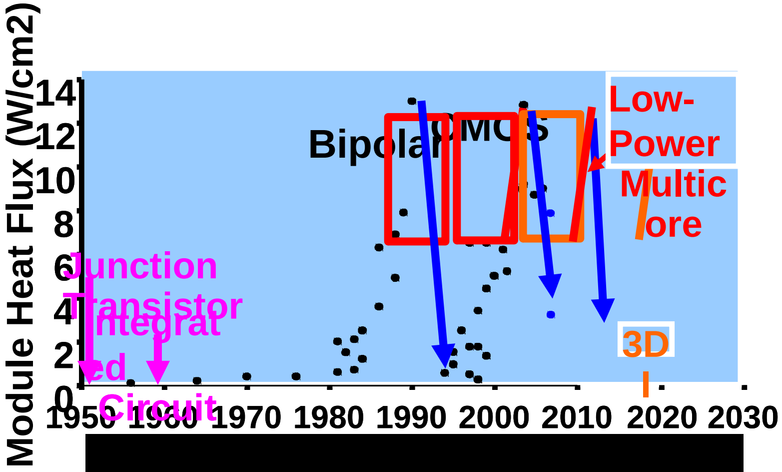
During the lifetime of you supercomputer

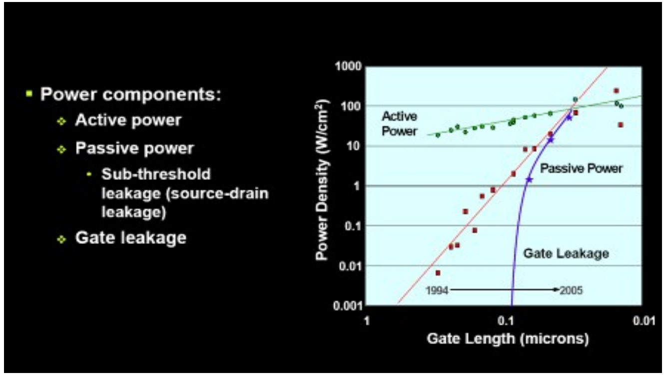
- Use power and energy aware tools
- Tune the applications

The Power Equation

Power=capacitance*voltage^2*frequency
Power~capacitance*voltage^3

- **Active power problem**
  - **Control frequency of active nodes**
- **Passive power problem**
  - **Minimize idle nodes power**





Power and Performance of JS22 and HS21

JS22 4.0 GHz						
Application	Average Power (watts)					
	Total	CPU	DIMM	Other	CPI	GBS
416.gamess	289	87	14	102	1,3	0,0
433.milc	306	76	51	103	6,8	16,3
435.gromacs	292	87	15	102	1,5	0,7
437.leslie3d	326	85	50	105	2,6	16,5
444.namd	296	89	14	104	1,4	0,3
454.calculix	301	91	18	103	1,0	1,9
459.GemsFDTD	315	80	49	106	5,1	15,8
481.wrf	311	84	39	103	1,5	12,7
Idle	212	48	14	102		

HS21 2.8 GHz						
Application	Average Power (watts)					
	Total	CPU	DIMM	Other	CPI	GBS
416.gamess	366	106	15	62	0,6	0,0
433.milc	321	64	30	66	9,8	6,2
435.gromacs	363	102	17	63	0,6	1,2
437.leslie3d	328	68	30	67	8,6	6,3
444.namd	356	100	15	64	0,7	0,2
454.calculix	379	106	20	64	0,6	2,2
459.GemsFDTD	323	66	29	66	9,5	6,1
481.wrf	329	69	29	66	5,2	6,1
idle	210	24	15	66		

Systems	Processors	Nominal Frequency	Memory
JS22 2 Sockets 2 cores	IBM Power6	4 GHz	4 x 4GB, 667 MHz DDR2
HS21 2 Sockets 4 cores	Intel Harpertown	2.86 GHz	8 x 2GB, 667 MHz DDR2

"CPU" includes N processor cores,L1 cache + NEST (memory, fabric, L2 and L3 controllers,..)

"Other" includes, L2 cache, Nova chip, IO chips, VRM losses, etc.

Power and Performance of iDataplex dx360 M4

Idataplex dx360 M4 – dual Sandy Bridge 2.7 Ghz (SSE42 binaries)

Application	Average Power (watts)				Perf metrics	
	Total	Core	DIMM	Other	CPI	GBS
416.gamess	275	100	5	71	0.9	0.3
433.milc	330	99	55	77	2.3	68.6
435.gromacs	260	95	5	65	1.2	5.0
437.leslie3d	332	99	57	78	3.1	65.0
444.namd	252	92	5	64	0.9	1.0
454.calculix	274	96	8	74	0.8	11.6
459.GemsFDTD	320	95	57	73	2.4	63.1
481.wrf	330	98	53	82	1.8	65.1
idle	85	6	5	68		

Idataplex dx360 M4 – dual Sandy Bridge 2.7 Ghz (AVX binaries)

Application	Average Power (watts)				Perf metrics	
	Total	Core	DIMM	Other	CPI	GBS
416.gamess	275	100	5	71	0.9	0.3
433.milc	327	97	55	78	2.4	68.5
435.gromacs	264	97	5	65	1.3	4.9
437.leslie3d	335	101	56	77	4.5	65.0
444.namd	253	90	5	68	1.0	1.0
454.calculix	281	100	8	73	0.9	12.5
459.GemsFDTD	320	95	57	73	2.4	62.5
481.wrf	332	101	53	77	2.2	65.2
idle	85	6	5	68		

Systems	Processors	Nominal Frequency	Memory
iDataplex dx360M4 2 Sockets 8 cores	Intel Sandy Bridge	2.7 GHz	8 x 16GB, 1600 MHz DDR3

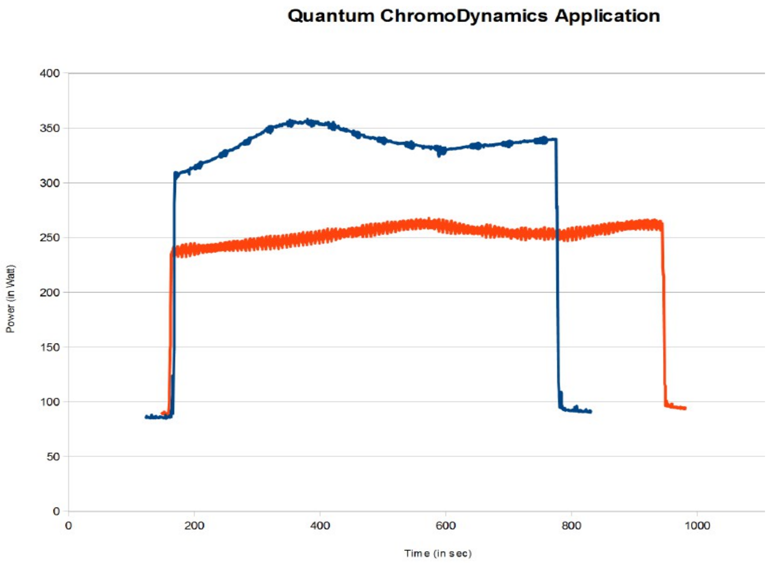
Power and Performance comparison of Nehalem and Sandy Bridge systems (3-4 years apart)

Application	Instances/hour		Energy/instance	
	NHM	SNB	NHM	SNB
416.gamess	35	83	24	12
433.milc	69	145	12	8
435.gromacs	91	242	9	4
437.leslie3d	51	100	17	12
444.namd	75	159	11	6
454.calculix	94	223	9	4
459.GemsFDTD	40	84	21	14
481.wrf	72	145	12	8

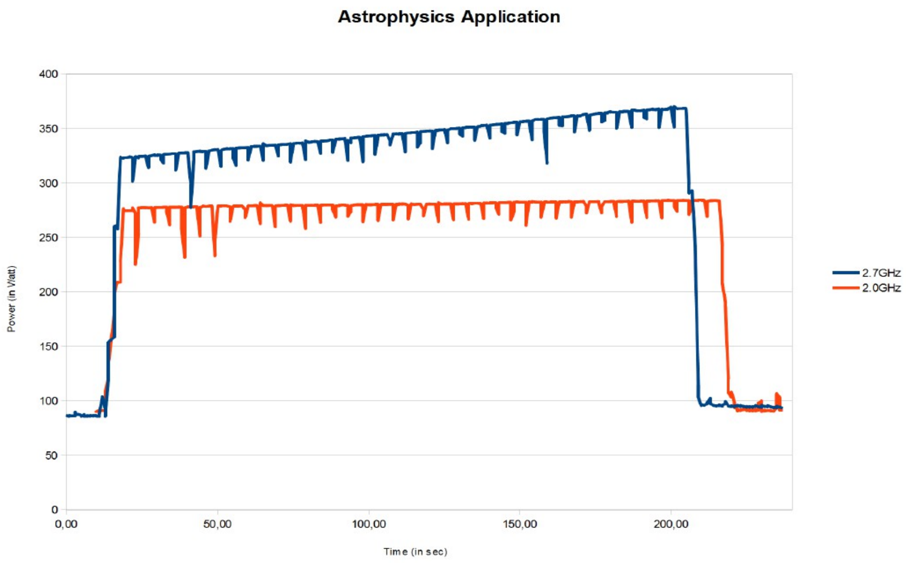
Throughput per core is **conserved**

Energy per job is **halved** (not exactly true for memory intensive jobs)

What happens when you just change frequency ?



Δf=-26%
ΔPower=-26%
ΔTime=+26%
ΔEnergy=~0%



Δf=-26%
ΔPower=-17%
ΔTime=+5%
ΔEnergy=-12%

How to find the performance/power trade-off ?

Monitor the application (hpm counters, power)

Build a performance and power model for prediction

- Which depends on the processor/node and the application

Is it worth tuning applications ?

IBM System x iDataPlex dx360 M4

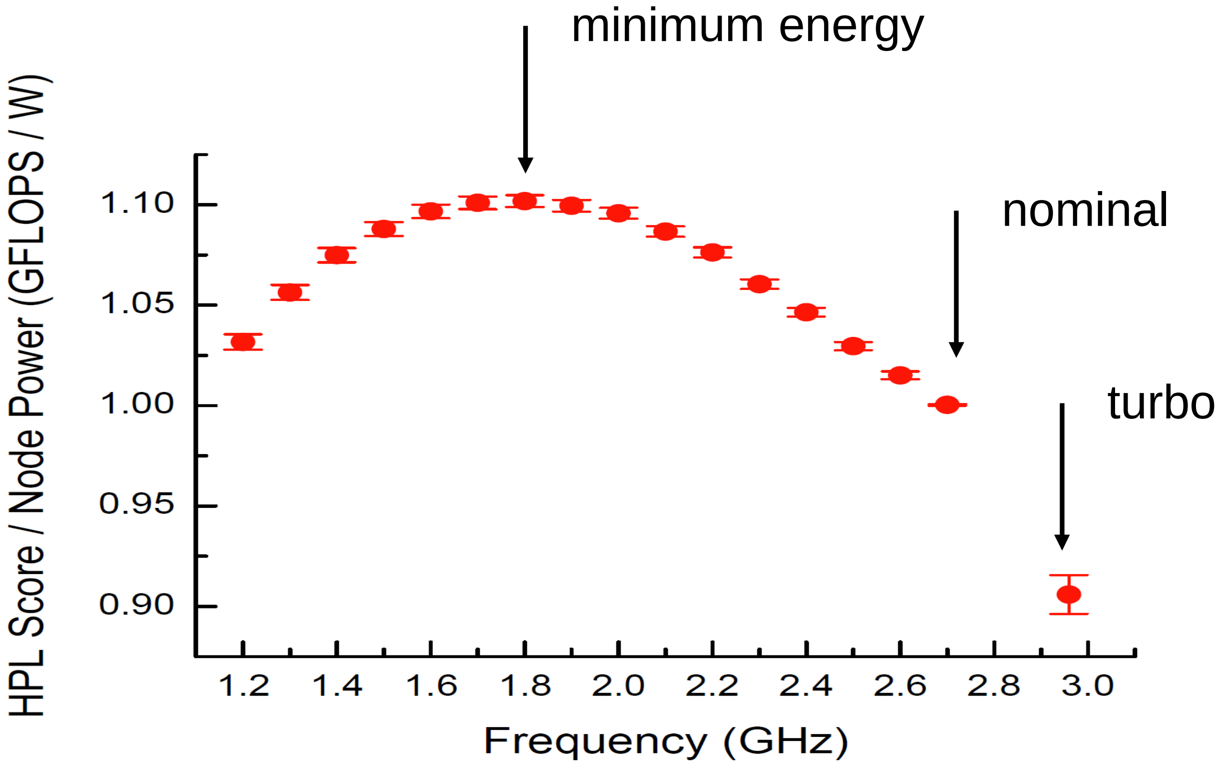
2x Intel SB-EP 2.7 GHz 130 W. 8x 4 GB.

Code version	Compiler options	Time (s)	Energy (J)	DC Power (W)	IPC
base	-O	45.4	12846	282	2.45
base	-O3 -xAVX	32.5	8874	272	2.43
base	-O2 -xSSE2	27.8	7495	269	2.68
SIMD intrinsics	-O3 –xAVX	7.6	2047	270	2.87

DC Power = cpu + dimms + static ~ (150w -180w) + (70w – 30w) + 60w

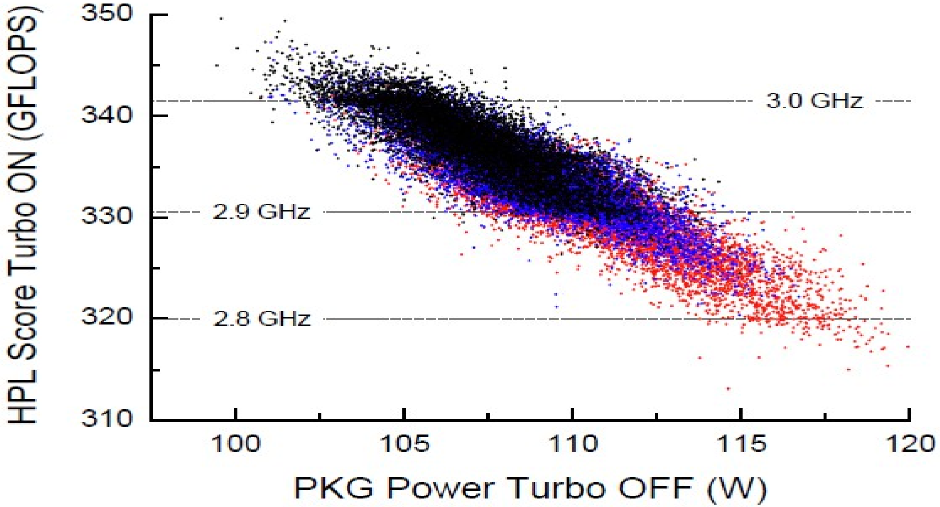
Is it worth using Turbo ?

Energy Efficiency IBM iDataPlex DWC dx360 M4



IBM System x iDataPlex Direct Water Cooled dx360 M4

2x Intel SB-EP 2.7 GHz 130 W. 8x 4 GB.



Ingmar Meijer, 2012

What can we do from a software perspective ?

Reduce power of inactive nodes

- by C- or S-states

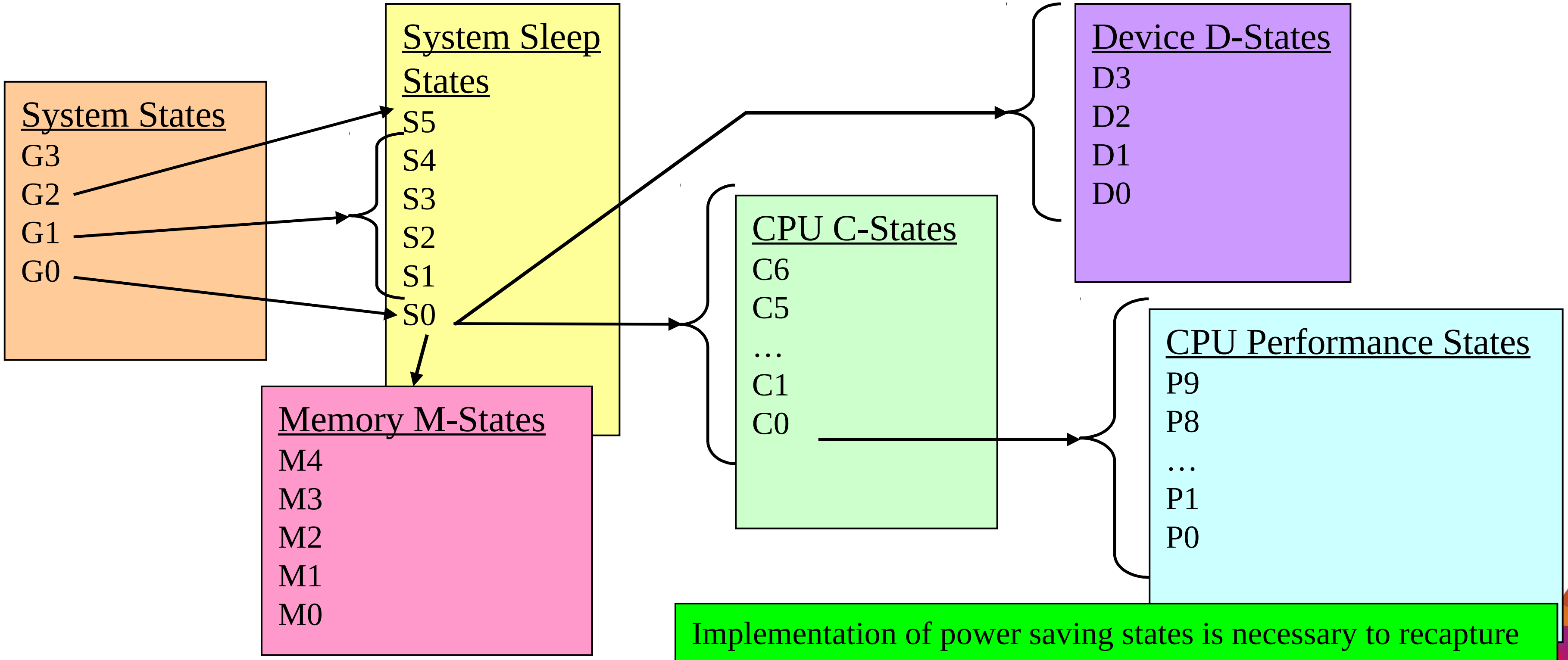
Reduce power of active nodes

- by P-state / CPUfreq
- by memory throttling

ACPI State Hierarchy

ACPI =Advanced Configuration and Power Interface (http://www.acpi.info/)

The ACPI specification defines several system and component states designed to save power.

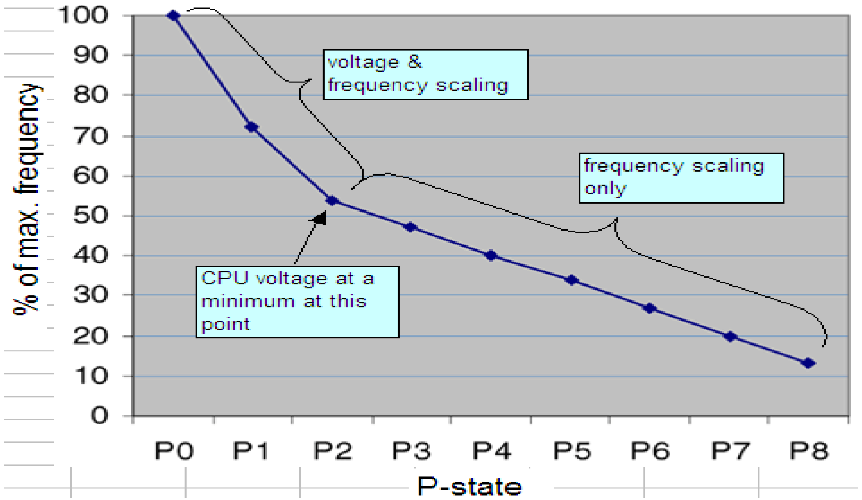


Implementation of power saving states is necessary to recapture lost power when a server or components in a server are idle.

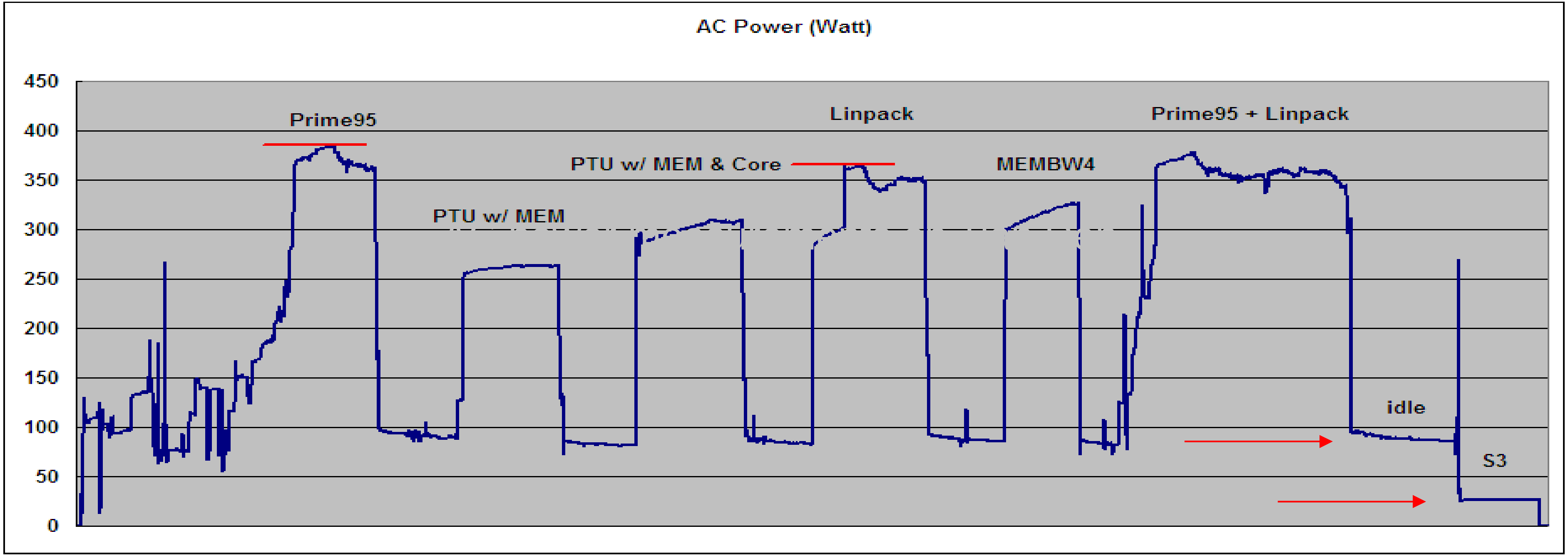
Effect of P-states

Between Vmax and Vmin, frequency is changed with voltage
Lower frequency reduces power reduction
- But not like f3 since there are other components than processor in the node

Lower frequency reduces performance
- Can be as much as ~ f, but could be less depending on the application/use case profile



Active and Idle power measurements on dx360m4



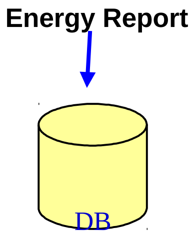
IBM Energy Aware Scheduling

Report

- temperature and power consumption per node/rack/cluster
- power consumption, performance (CPI, GBS, GFLOPs) and energy per job

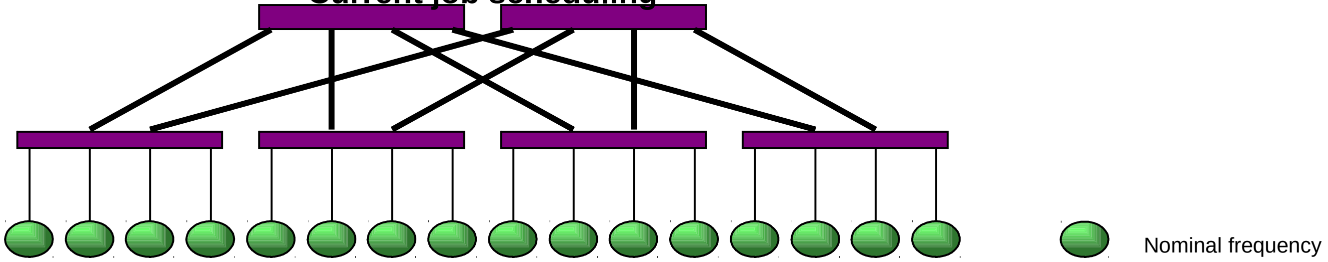
Optimize

- Reduce power of inactive nodes
- Optimize energy of active nodes

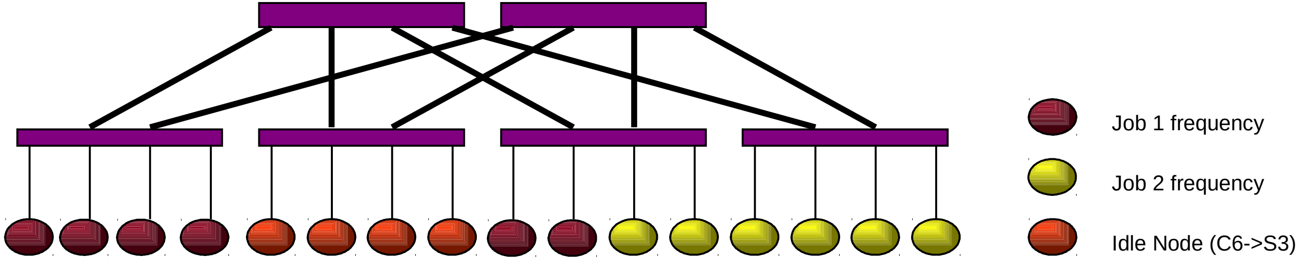


Energy Aware Scheduling

Current job scheduling



Energy Aware job scheduling



Before each job is submitted, change the state/frequency of the corresponding set of nodes to match a given energy policy defined by the Sys Admin

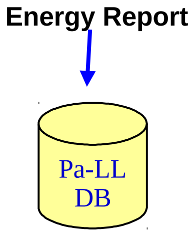
Features available to reduce and control power

xCAT

- Manage power consumption on an ad hoc basis
  - Query: Power saving mode, power consumed info, CPU usage, fan speed, environment temperature
  - Set: Power saving mode , Power capping value, Deep Sleep (S3 state)

LL (and later this year LSF)

- Report power and energy consumption per job
  - Energy report is created and stored in the DB
- Optimize power and energy consumption per job
  - Optimize power of idle nodes:
    - **set nodes at lowest power consumption when no workload is scheduled on this set of nodes**
  - Optimize power of active nodes:
    - **set nodes at optimal processor frequency according to an energy policy for a given parallel workload (i.e minimize energy with maximum performance degradation)**



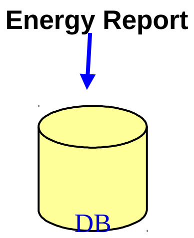
IBM software to monitor and reduce power

Report

- Temperature, fan speed and power consumption per node
- power consumption, energy and performance per job

Optimize

- Reduce power of inactive nodes
- Reduce power of active nodes



How LL-EAS manages idle nodes

When a job has completed on a set of nodes, LL set those nodes in a state which does let the OS to turn them into lowest C-state (C6)

When nodes are idle and no jobs are in queue, LL will ask xCAT to put them into S3 state according to the idle power policy parameters.

- Idle power policy parameters are determined by the system admin

When new jobs are submitted which require nodes to be awaken , LL asks xCAT to resume the desired nodes from S3 before it submits the job

LL-EAS energy policies available

Predefined policy

- Minimize Energy within max performance degradation bound of X%
  - LL will determine the frequency (lower than default) to match the X% performance degradation while energy savings is still positive
- MinimizeTime to Solution
  - LL will determine a frequency (higher than default) to match a table of expected performance improvement provided by sysadmin
  - This policy is only available when default frequency < nominal frequency
- Set Frequency
  - User provides the frequency he wants hos jobs to run
  - This policy is available for authorized user only

- Policy thresholds are dynamic, i.e values can be changed any time and will be taken into account when next job is submitted

Site provided policy

- Sysadmin provides an executable to set frequency based on the information stored in DB

LL-EAS phases to set optimal frequency for jobs

Learning phase

- LL evaluates the power profile of all nodes and store it in the xCAT/LL DB

System admin defines a default frequency for the cluster

- Can be nominal frequency or a lower frequency

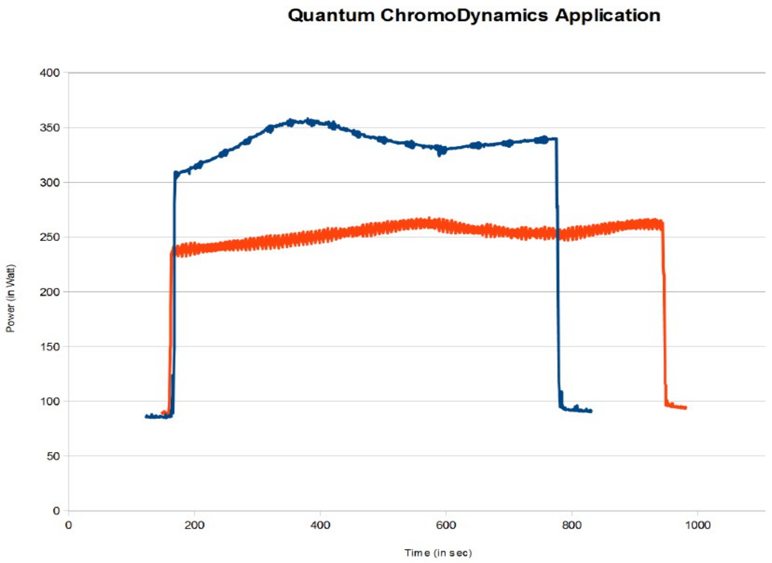
User submits a job

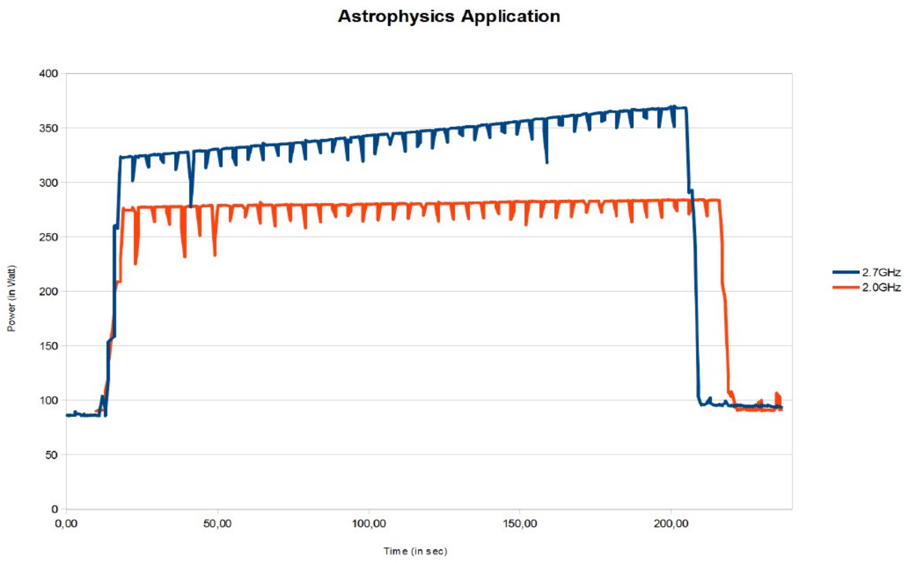
- User submits his/her job with a tag
- Job is run at default frequency
- In the background:
  - LL measures power, energy, time and hpm counters for the job
  - LL predicts power(i), energy(i), time (i) if job was run a different frequency i
- LL writes Energy report for the job in the xCAT/LL DB

User submits another job with the same tag

- Given the energy policy and the tag, LL determines optimal frequency j
- LL sets nodes for the job at frequency j and run the job
  - LL measures power, energy, time and hpm counters for the job
- LL adds information in DB and creates a new energy report

Example: what happens when you just change frequency





Δf=-26%
ΔPower=-26%
ΔTime=+26%
ΔEnergy=~0%

Δf=-26%
ΔPower=-17%
ΔTime=+5%
ΔEnergy=-12%

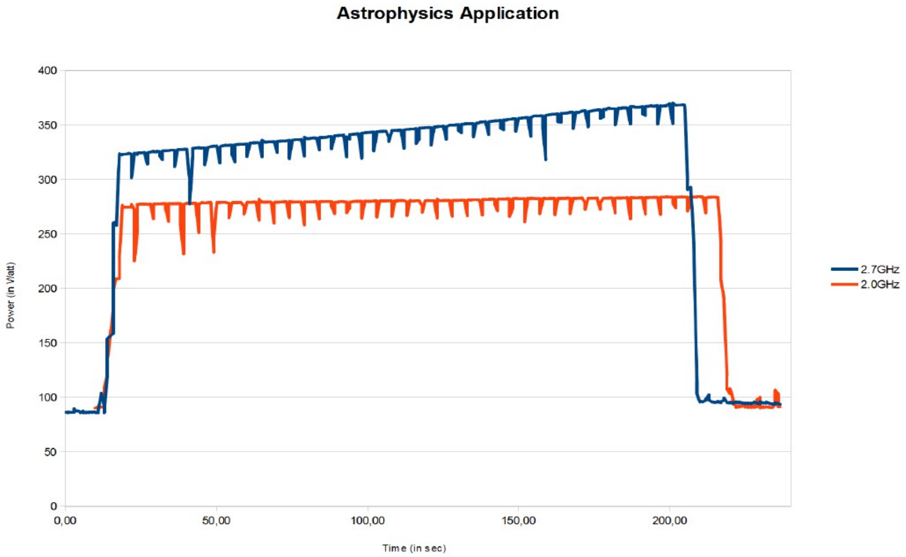
Example: how to submit a job first time

```
#!/bin/bash
# @ job_name = test
# @ account_no = 99999
# @ class = parallel
# @ job_type = MPICH
# @ network.MPI = sn_all,,US
# @ total_tasks = 128
# @ node = 8
# @ output = $(jobid)_output
# @ error  = $(jobid)_error
# @ initialdir = /bench/gpfs/fs1/users/fthomas/lleas/Astrophysics
# @ node_usage = not_shared
# @ energy_policy_tag = Astro
# @ energy_output = energy.dat
# @ queue

. ~/.bashrc
```

Example: how to submit a job with a policy

```
#!/bin/bash
# @ job_name = test
# @ account_no = 99999
# @ class = parallel
# @ job_type = MPICH
# @ network.MPI = sn_all,,US
# @ total_tasks = 128
# @ node = 8
# @ output = $(jobid)_output
# @ error  = $(jobid)_error
# @ initialdir = /bench/gpfs/fs1/users/fthomas/lleas/Astrophysics
# @ node_usage = not_shared
# @ energy_policy_tag = Astro
# @ energy_output = energy.dat
# @ max_perf_decrease_allowed = 5
# @ queue

. ~/.bashrc
```

Example: what happens with max perf degrad policy=5%



f= 2.6 GHz
ΔPower=-5%
ΔTime=+2%
ΔEnergy=-3%

f=2.0 GHz
ΔPower=-17%
ΔTime=+5%
ΔEnergy=-12%

UM: Energy Report

perf., power

Clock (MHz)	CPI	Time (s)	Power (Watt)	Energy (KW/h)	PerfVar (%)	PowerVar(%)	EnergyVar (%)
2700	0,986	158	274	0,0120	0	0	0
2600	0,977	163	259	0,0117	-2,9%	5,3%	2,6%
2500	0,970	168	249	0,0116	-6,2%	9,1%	3,4%
2400	0,956	172	243	0,0116	-9,1%	11,3%	3,2%
2300	0,946	178	232	0,0114	-12,6%	15,4%	4,7%
2200	0,938	184	224	0,0115	-16,8%	18,2%	4,4%
2000	0,915	198	210	0,0115	-25,2%	23,4%	4,0%
1900	0,905	206	202	0,0116	-30,5%	26,3%	3,8%
1800	0,897	216	195	0,0116	-36,5%	28,9%	3,0%
1700	0,891	227	188	0,0119	-43,6%	31,3%	1,3%
1600	0,880	238	183	0,0121	-50,6%	33,2%	-0,6%
1500	0,873	252	175	0,0123	-59,4%	36,0%	-2,1%
1400	0,867	268	166	0,0123	-69,6%	39,5%	-2,6%

Ramses: Energy Report:

perf., power

Clock (MHz)	CPI	Time (s)	Power (Watt)	Energy (KW/h)	PerfVar (%)	PowerVar(%)	EnergyVar (%)	Clock (MHz)
2700	3,639	189	288	0,0151	0	0	0	2700
2600	3,619	189	275	0,0144	0,0%	4,7%	4,7%	2600
2500	3,525	190	269	0,0142	-0,5%	6,7%	6,2%	2500
→ 2400	3,442	191	263	0,0140	-1,1%	8,7%	7,7%	2400
2300	3,370	193	256	0,0137	-2,1%	11,4%	9,5%	2300
2200	3,274	195	248	0,0134	-3,2%	14,0%	11,3%	2200
2000	3,164	200	239	0,0133	-5,8%	17,0%	12,2%	2000
1900	3,058	203	232	0,0131	-7,4%	19,7%	13,8%	1900
1800	3,023	206	224	0,0128	-9,0%	22,5%	15,5%	1800
1700	2,948	211	217	0,0127	-11,4%	24,8%	16,3%	1700

BQCD : Energy report for 1K and 8K tasks , perf., power

Clock	CPI	Time	Power	Energy	PerfVa	PwrVa	EnyVar
2700	1,075	509	308	0,0435	0	0	0
2600	1,062	522	290	0,0420	-2,6%	5,8%	3,3%
2500	1,038	531	280	0,0413	-4,3%	8,8%	4,9%
2400	1,015	540	275	0,0413	-6,2%	10,6%	5,0%
2300	0,994	552	261	0,0400	-8,5%	15,3%	8,0%
2200	0,972	565	255	0,0399	-10,9%	17,2%	8,1%
2000	0,932	596	237	0,0393	-17,1%	22,8%	9,6%
1900	0,908	611	228	0,0386	-20,0%	25,9%	11,1%
1800	0,894	635	220	0,0388	-24,7%	28,4%	10,8%
1700	0,877	659	212	0,0388	-29,6%	31,1%	10,7%
1600	0,848	677	207	0,0390	-33,0%	32,6%	10,4%
1500	0,831	708	199	0,0392	-39,2%	35,2%	9,8%
1400	0,821	750	188	0,0391	-47,3%	38,9%	10,0%
1300	0,807	794	179	0,0394	-55,9%	41,9%	9,4%

Clock	CPI	Time	Power	Energy	PerfVa	PwrVar	EnyVar
2700	0,661	304	290	0,0244	0	0	0
2600	0,651	311	273	0,0236	-3,2%	5,7%	2,6%
2500	0,645	320	263	0,0234	-5,3%	9,2%	4,4%
2400	0,634	328	257	0,0235	-7,9%	11,1%	4,1%
2300	0,626	338	244	0,0229	-11,1%	15,6%	6,2%
2200	0,620	350	237	0,0231	-15,2%	18,1%	5,6%
2000	0,598	372	222	0,0229	-22,2%	23,3%	6,3%
1900	0,593	387	213	0,0229	-27,4%	26,4%	6,2%
1800	0,584	403	206	0,0230	-32,5%	29,0%	5,9%
1700	0,581	424	199	0,0234	-39,6%	31,4%	4,2%
1600	0,575	446	194	0,0240	-46,7%	33,2%	1,9%
1500	0,571	473	186	0,0244	-55,5%	35,8%	0,1%
1400	0,566	502	175	0,0244	-65,1%	39,5%	0,1%

Savings example

1000 node cluster, 0.15€ per KWh

Linpack power consumption per year = 442K€

Inactive nodes

With 80% workload activity and nodes in S3 half of the idle time (10% of overall time)

Savings per year = 24.5 K€

Active nodes

With a 3% performance degradation threshold, about 8% power saved (cf examples)

Savings per year = 20.4 K€

Total savings: 45K€, ~10%

EAS functions in LSF

Energy Aware Scheduling features in LSF

- **First features available in July 2013**
  - **Energy report (with no prediction)**
  - **Idle node power management**
  - **Set frequency policy**
- **Full features available November 2013 (announced October 2013)**
  - **Full energy report including prediction**
  - **Minimize Energy and Minimize Time to Solution Energy Policies**

New features to be developed in the future :

- **Support new Intel processor (IVB and HSW)**
  - **Use of Lock-in Turbo to Extend Minimize Time to Solution with Turbo**
  - **Control power and performance per core vs per node**
- **Support ManyCore processors like Xeon Phi and NVIDIA**
  - **Inactive and active nodes**
- **New energy policy like Intelligent Power Capping at cluster level**
- **Reporting of power and energy in Analytics**

3 PFlops SuperMUC system at LRZ

Fastest Computer in Europe on Top 500 June 2012

- 9324 Nodes with 2 Intel Sandy Bridge EP CPUs
- 3 PetaFLOP/s Peak Performance
- Infiniband FDR10 Interconnect
- Large File Space for multiple purpose
- 10 PetaByte File Space based on IBM GPFS
  - with 200GigaByte/s aggregated I/O Bandwidth
  - 2 PetaByte NAS Storage with 10GigaByte/s aggregated I/O Bandwidth

Innovative Technology for Energy Effective Computing

- Hot Water Cooling
- Energy Aware Scheduling

Most Energy Efficient high End HPC System

- PUE 1.1
- Total Power consumption over 5 years to be reduced by ~ 37% from 27.6 M€to 17.4 M€

Thank you !

High Performance Computing
For a Smarter Planet

Rethink High Performance Computing.