

Performance of P5+ at ECMWF (European Centre for Medium Range Weather Forecasting)

John Hague, IBM Consultant

14/7/06

ECMWF: Current and New (Phase4) System

- Current system: 2 clusters, each with
 - 68 32-way P4+ 1.9 GHz 32 GB p690s
 - Federation Switch
 - 4 adapters/node
 - File system for Research
 - GPFS on 30 raids with 8 (dual) controllers
- New Phase4 system: 2 clusters, each with
 - 142 16-way dual core P5+ 1.9 GHz 32 GB p575s
 - Federation Switch
 - 2 adapters/node
 - SMT enabled
 - File system for Benchmark
 - GPFS on 32 raids with 8 (dual) controllers

IFS (Integrated Forecast System)

- T799
 - Main 10 day Forecast
 - 25km resolution
- T399: EPS (Ensemble Prediction System)
 - 50Km resolution
 - 100 copies to give error probabilities
- 4D-Var: Data Assimilation
 - T799/T255/T95
 - Incorporates observations into latest forecast
- Cycle 28R3
 - RAPS8 - circa 2004 – for Phase4 benchmark
- Cycle 30R1
 - RAPS9 - circa 2006 – for next procurement (maybe)

Changes for New P5+ System

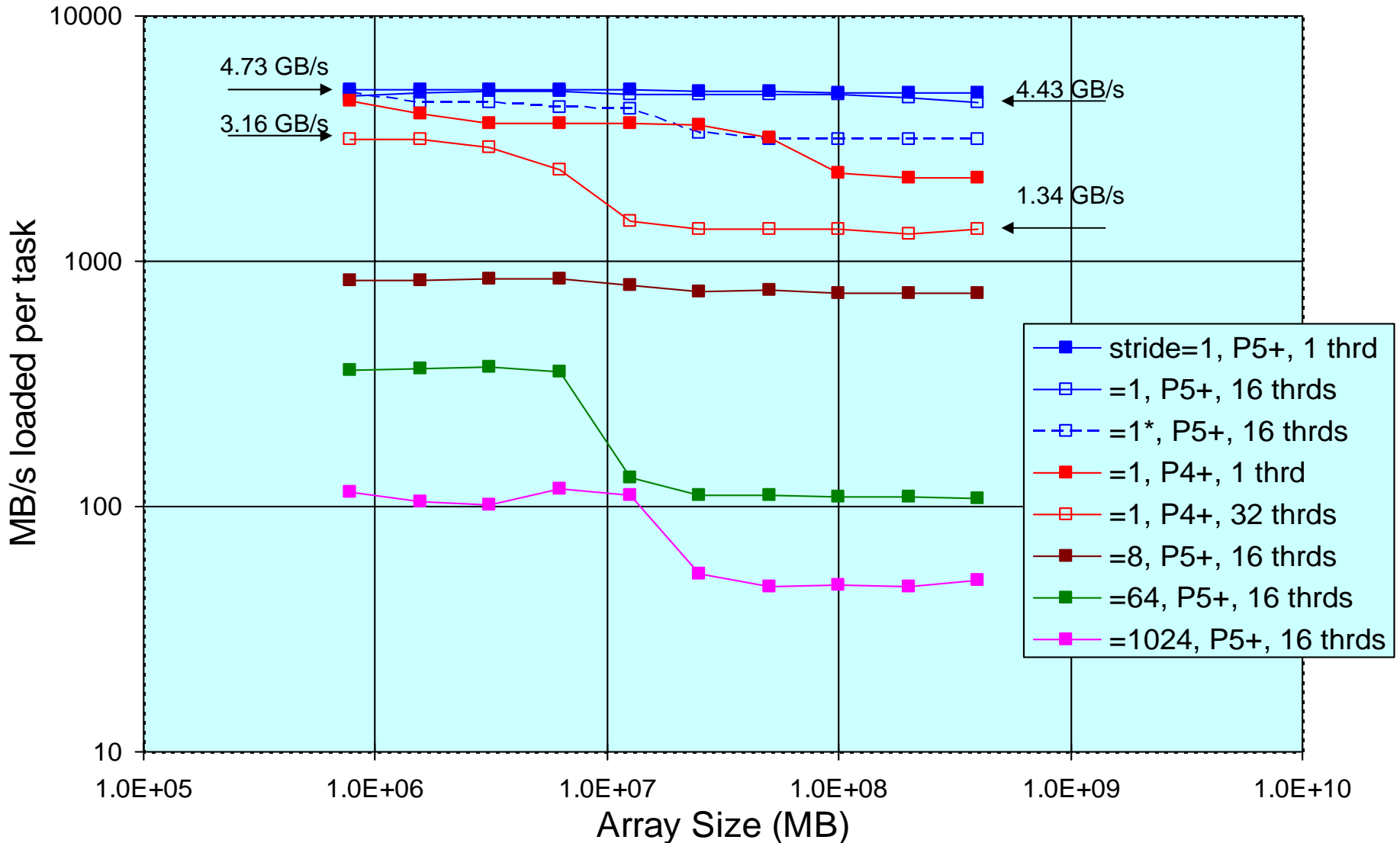
- xlf90_r –qarch=pwr5
 - 1 to 2% performance improvement for IFS
- Medium pages
 - 64k bytes (compared with 4k bytes)
 - 3% improvement for IFS
- Binding
 - Basic: 5 to 30% performance improvement
 - Special: 2% performance improvement for IFS
- Write Caching and Mirroring for GPFS
 - 3 to 5% performance improvement for IFS
- Monitoring
 - Synchronise sensors
 - 2 to 5% performance improvement for IFS
- “Throttling” in VSD layer
 - Reduced performance variability by 20%

thanks to Yuri Volobue, Gautam Shah et al for this

Look at

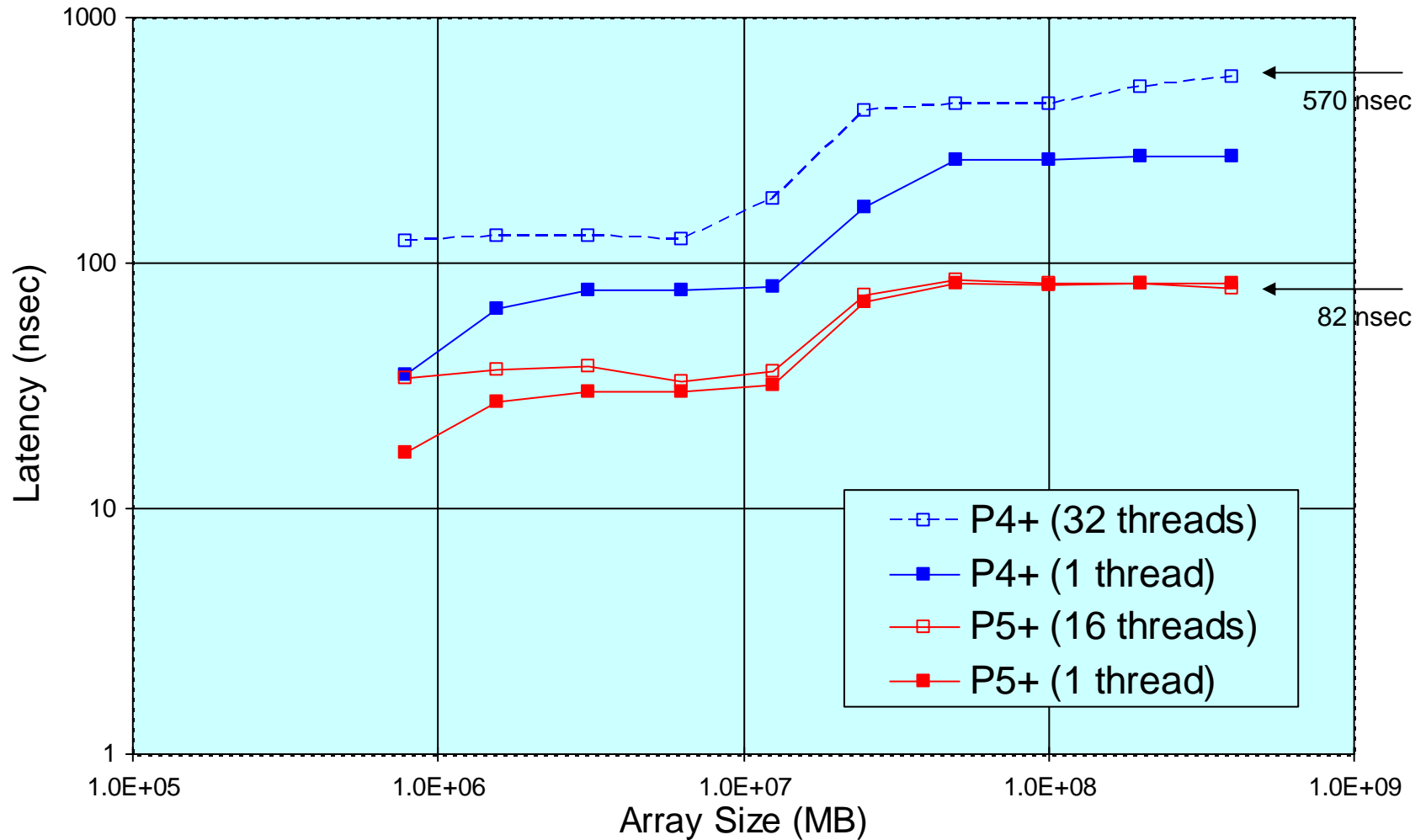
- Memory Bandwidth
- Switch Bandwidth
- SMT (Simultaneous Multi Threading)
- Binding
- Monitoring
- IFS on P5+ (and P4 comparison)
- IFS on Bluegene and JS21 Blades

Memory bandwidth/thread for P5+ (and P4+)



4 byte data items, (64KB data pages for P5+ except *=4K)

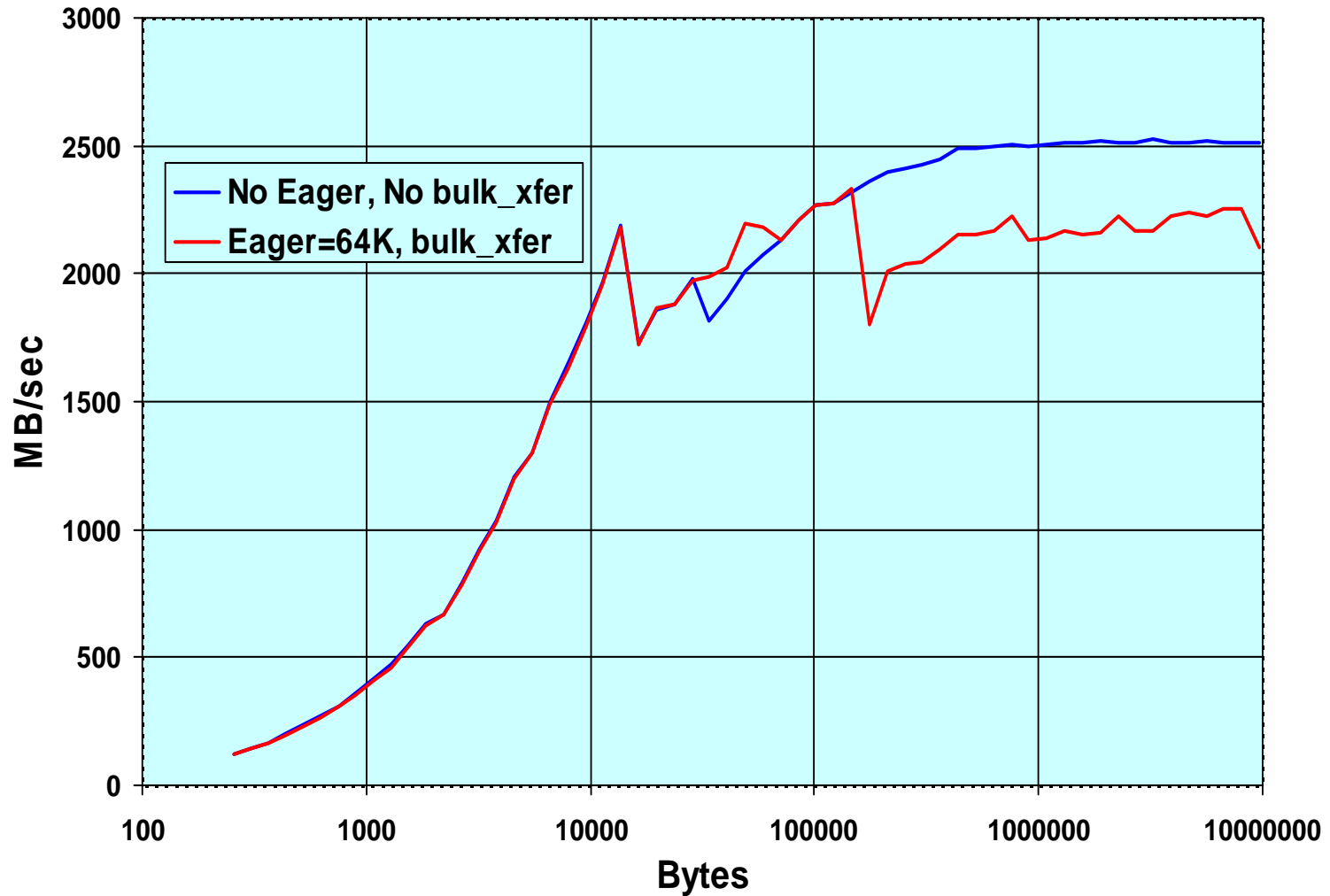
Memory latency: P5+ (and P4+)



4 byte data items, 64KB data pages on P5+, stride = 1024

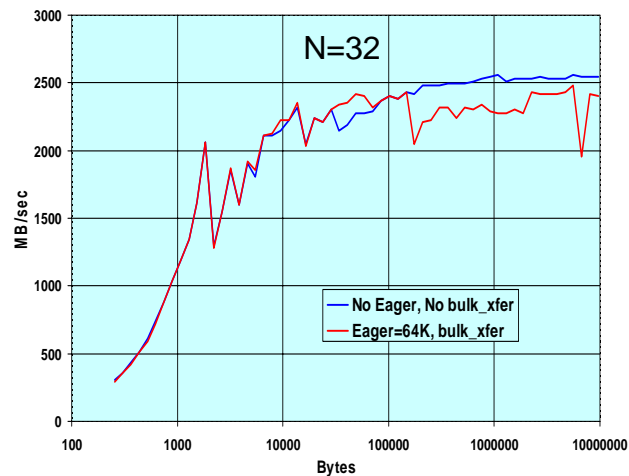
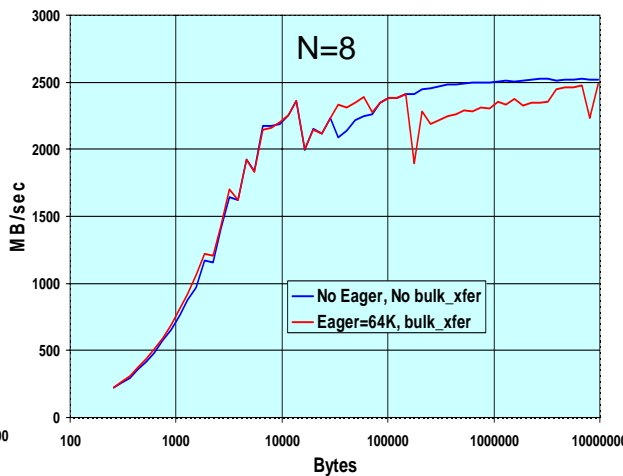
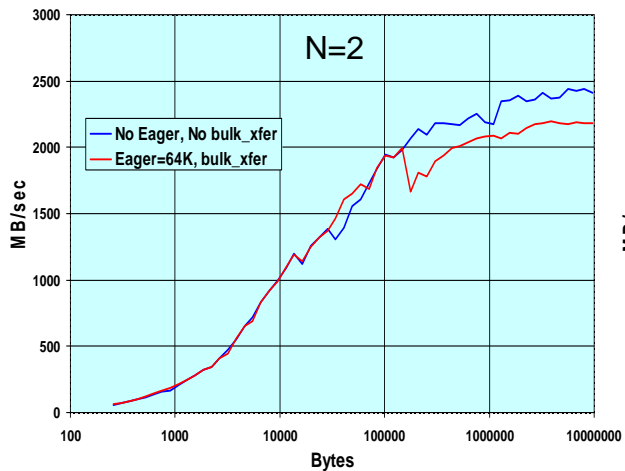
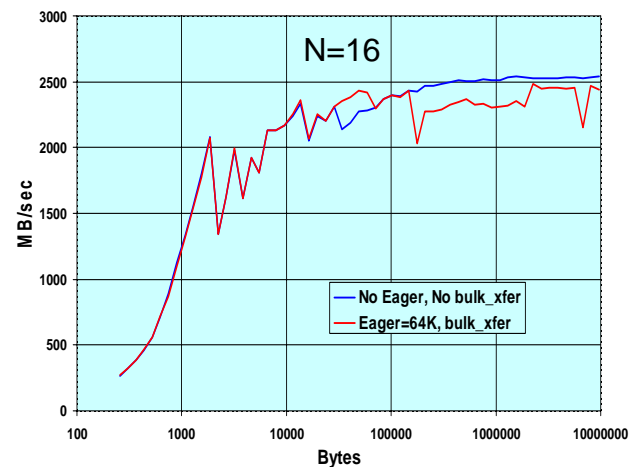
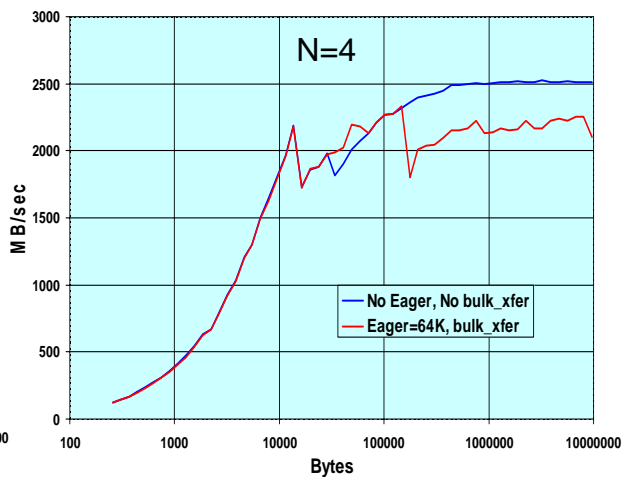
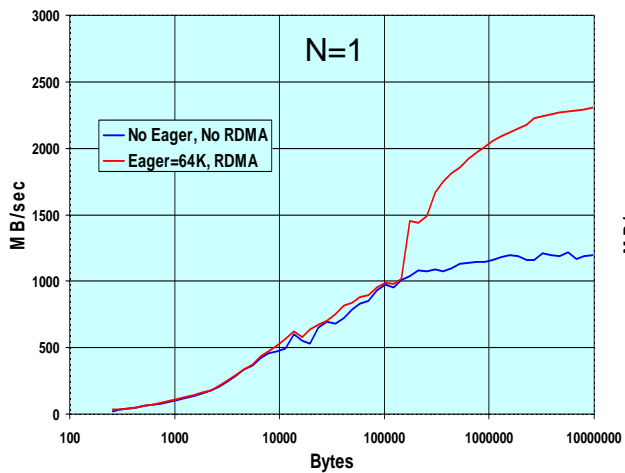
Exchange Bandwidth per Switch Link on P5+

(2 nodes, 4 tasks/node, 1 thread/task, 2 tasks/link)



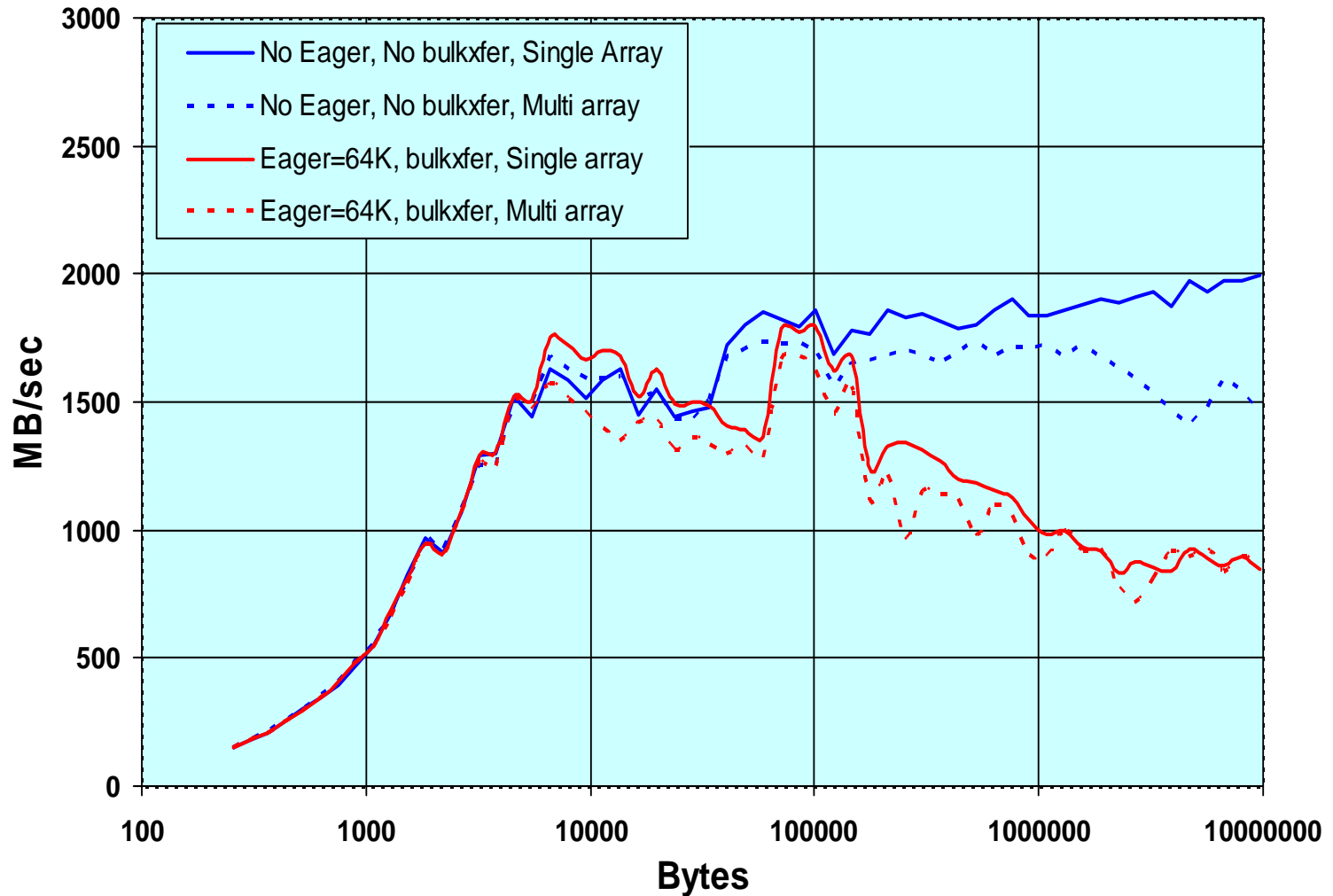
Exchange Bandwidth per Switch Link on P5+

(2 nodes, N tasks/node, 1 thread/task, 2 links/node)



“Transposition” Bandwidth per Switch Link on P5+

(32 nodes, 4 tasks/node, 4 threads/task, 2 tasks/link)



SMT: Simultaneous Multi Threading

- Node has physical 16 CPUs = 8 dual-core chips
- 2 'logical CPUs' are allocated to each 'physical CPU'
- These CPUs can be used with MPI or OpenMP
 - best with appropriate binding
- Programs benefit from SMT if cpu pipes not fully utilised
- Some programs don't benefit from SMT if
 - they have a lot of memory traffic per Floating Point operation
 - they are 'FP bound' like SGEMM
 - the program doesn't scale well
 - needs twice as many mpi_tasks or OpenMP threads

SMT benefit (on P5+)

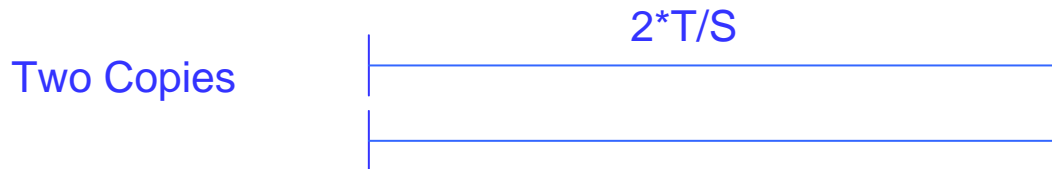
FP pipe use	Memory access	Code	GFLOP no SMT	GFLOP SMT
One	None	<code>s1=s1+1.d0</code>	5.0	10.0
Both	None	<code>s1=s1+1.d0</code> <code>s2=s2+1.d0</code>	10.1	19.9
Both	None	<code>s1=s1+1.d0</code> . . . <code>s10=s10+1.d0</code>	50.2	60.3
Both	Streaming	Stride 2: <code>s1=s1+1.1d0*a(i)</code> <code>s2=s2+1.1d0*a(i+1)</code>	11.3	16.2
Both	Skipping	Stride 10: <code>s1=s1+1.1d0*a(i)</code> <code>s2=s2+1.1d0*a(i+1)</code>	3.0	3.3

SMT for parallel jobs (on P5+)

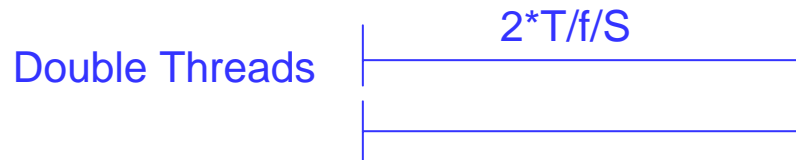
- If one copy of program takes T



- If SMT factor is S , 2 copies of program take $2*T/S$



- If scalability factor for doubling threads is f , program takes $2*T/f/S$

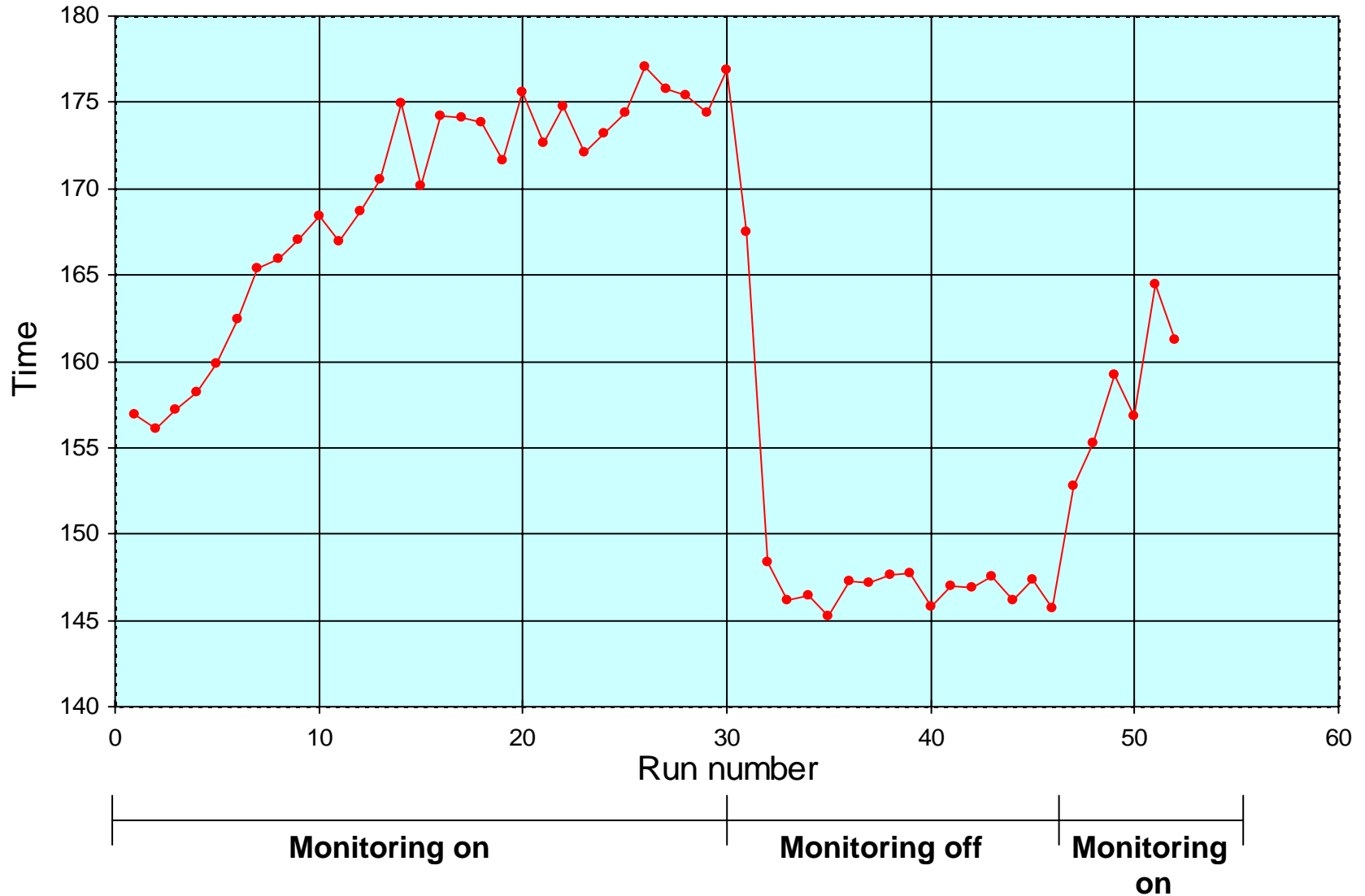


- For T799 on 1200 CPU, $S=1.3$, $f=1.8$
 - Speedup due to SMT is $S*f/2 = 1.35*1.8/2 = 1.22$

Binding (for P5+)

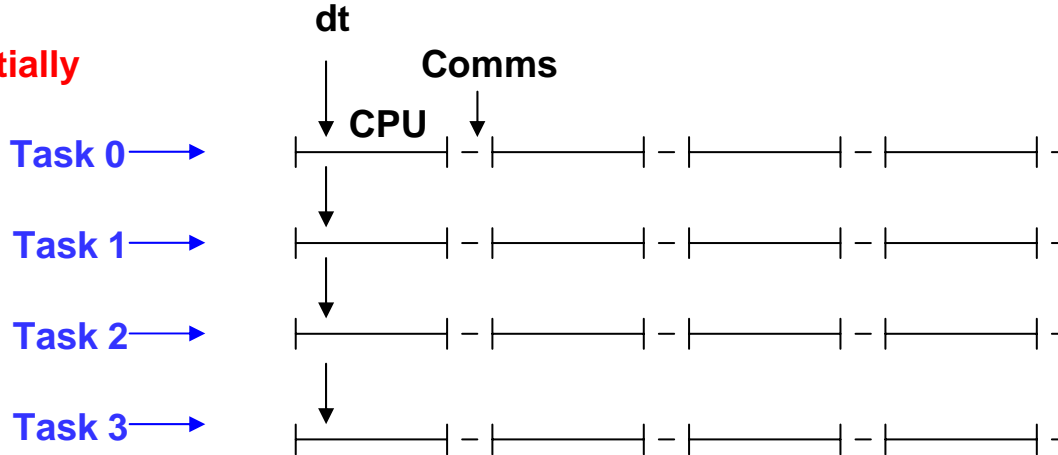
- **MP_TASK_AFFINITY**
 - Keeps task (and all it's threads) on same resource (i.e chip or 2 cpu's on dual core p575)
 - Not good if more than 2 threads without SMT, or more than 4 threads with SMP
- Binding keeps all threads on specified CPU
 - Use special binding code
 - Uses file created at boot time relating physical cpus to bind cpus
- Without SMT, specify
 - **0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30**
- With SMT, specify
 - **0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31**
- With SMT and “front loaded” threads, specify (for 16 threads per task)
 - **0 2 4 6 8 10 12 14 1 3 5 7 9 11 13 15 16 18 20 22 24 26 28 30 17 19 21 23 25 27 29 31**
- Also use **MEMORY_AFFINITY=MCM**

Monitoring: 4D-Var min1 comms increase with time (RAPS8 on P5+)



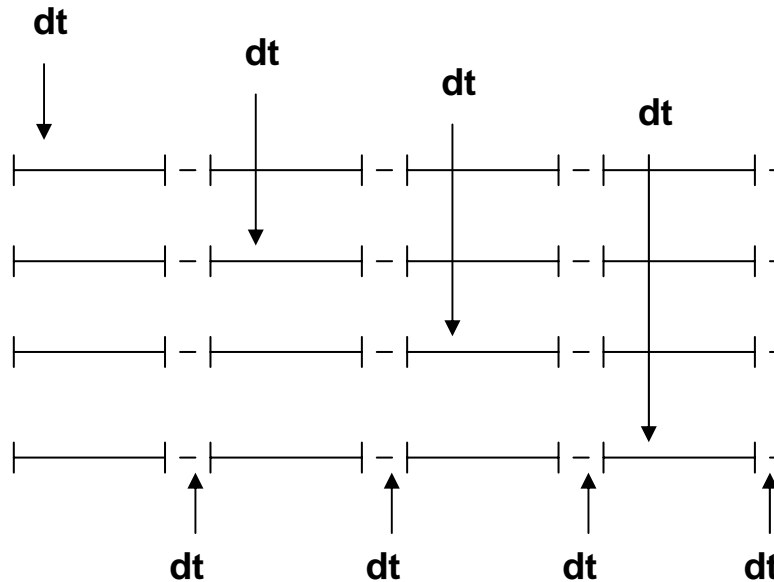
Effect of monitoring

Initially



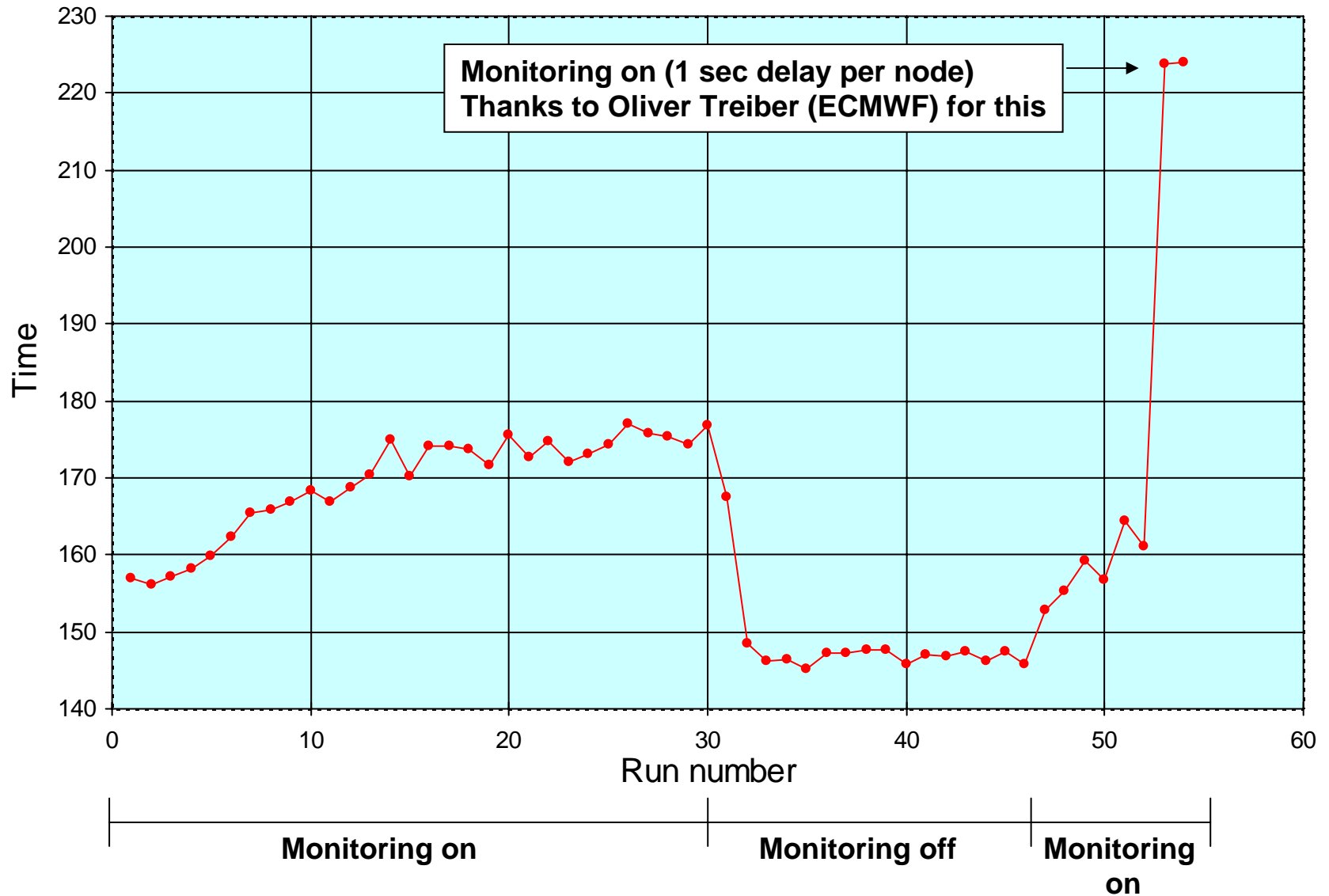
Every minute:
delay dt in CPU

After several hours



Every minute (for N tasks):
delay dt in CPU
delay $N \cdot dt$ in Comms

4D-Var min1 Comms increase with time

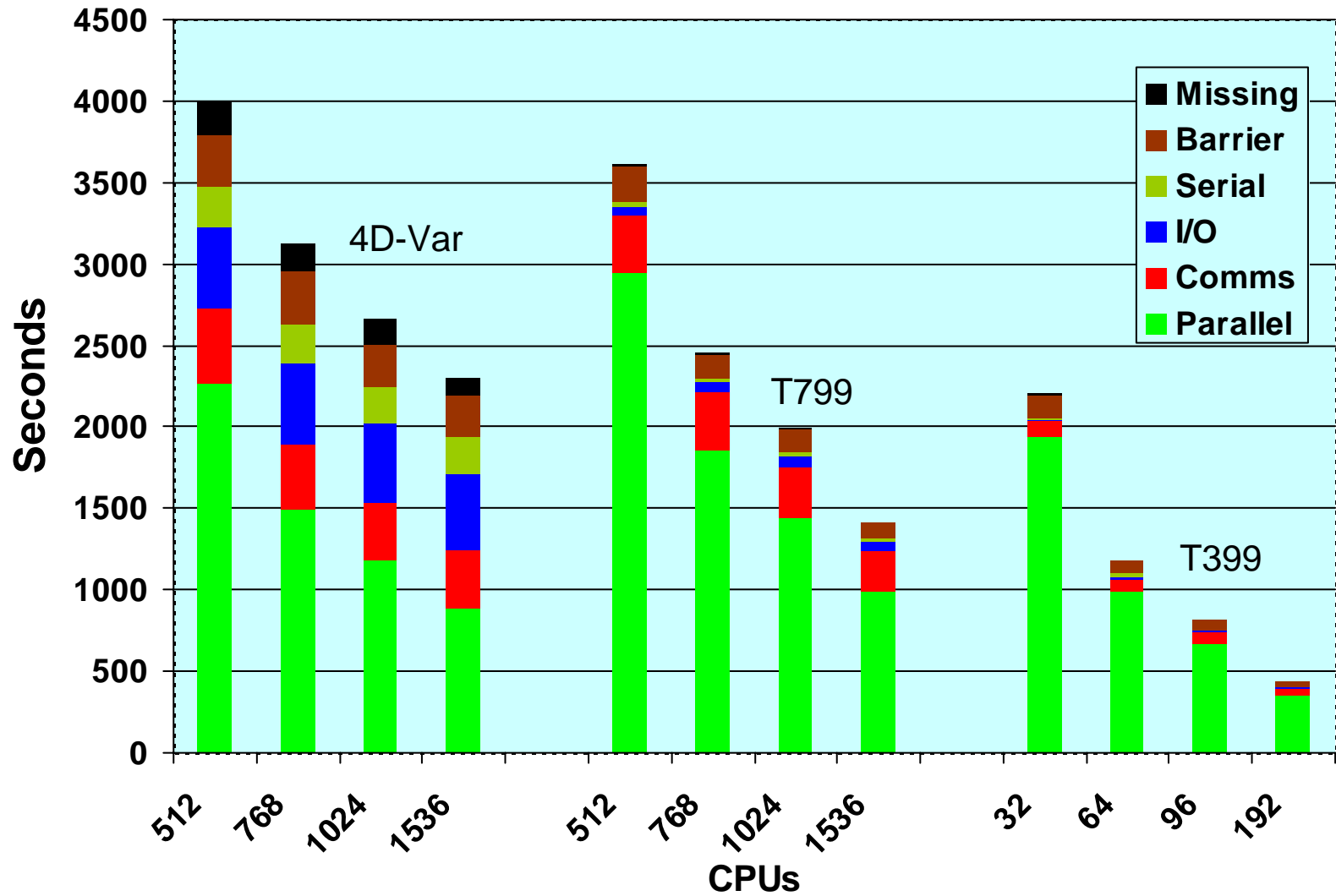


Times for IFS on P5+ (RAPS9)

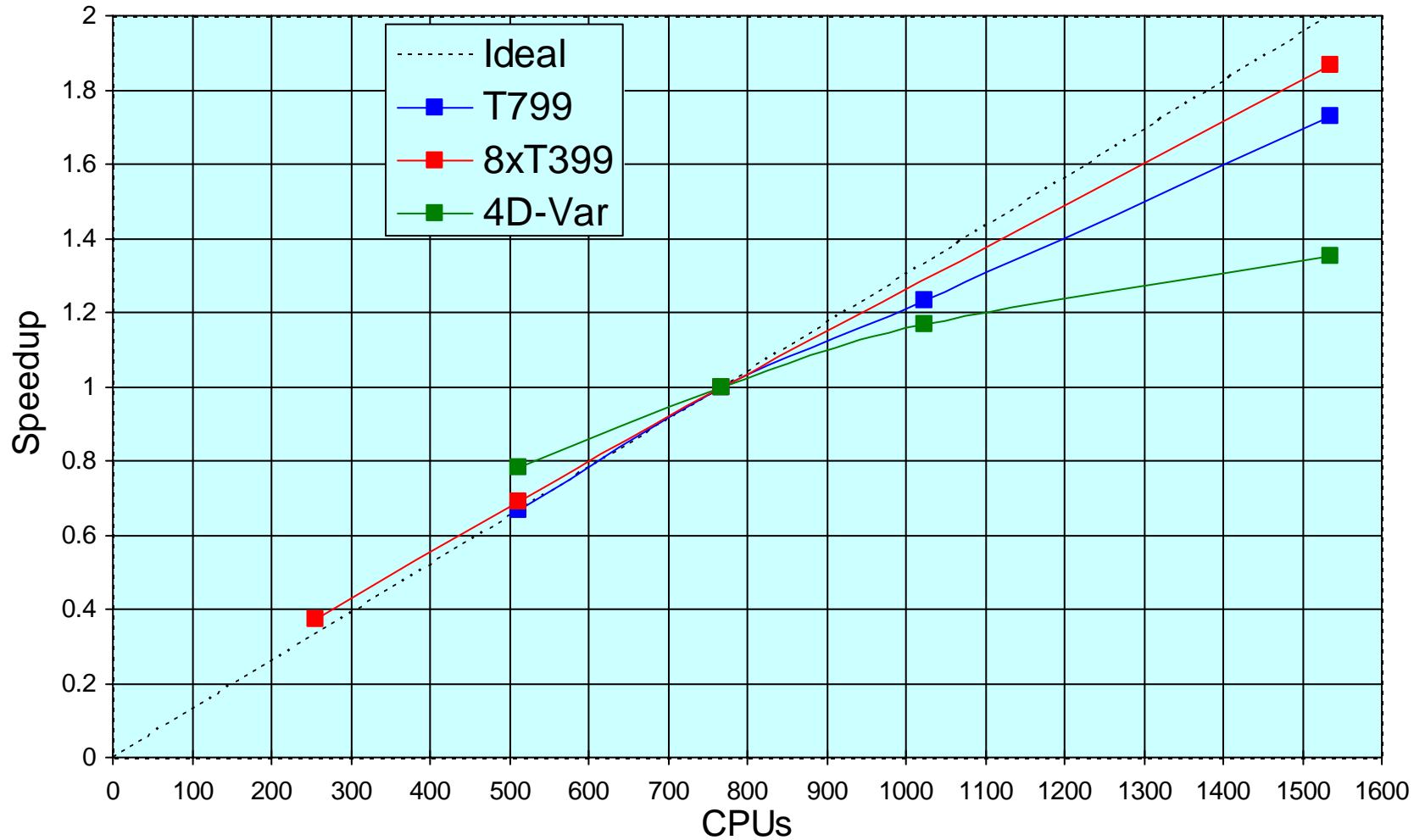
	CPU	Tsks_Thrds	WALL 1	Comms	Parallel	I/O	Serial	Barrier	Missing	GFLOP	% Peak
T399 EPS	96	48_4	815	69 9%	669 82%	9 1%	4 1%	62 8%	2 0%	69444	11.7
T799 10day forecast	768	192-8	2457	358 14%	1856 76%	57 2%	26 1%	151 6%	9 0%	1696200	11.8
4D-Var	768	96_8*	3121	397 13%	1492 48%	505 16%	235 8%	330 11%	162 5%	1034445	5.68

* 16 threads for Min1

IFS on P5+ (RAPS9)



IFS Scalability on P5+ (RAPS9)



Normalised to 1 for operational number of CPUs

T399 on P5+: SMT and Binding (RAPS8)

1 Day Forecast, 48 steps, 3 16-CPU dual_core nodes, SMT Enabled

Tasks_ Thrds	Use SMT	Bind	MEM AFF	Time
12_4	N	N	N	202
12_4	N	N	Y	195
12_4	N	Y	N	193
12_4	N	Y	Y	182
24_4	Y	N	N	159
24_4	Y	N	Y	155
24_4	Y	Y	N	155
24_4	Y	Y	Y	152

- **Conclusions**

- MEMORY_AFFINITY is worth a percent or two
- SMT is worth about 20%
- Binding is worth a few %, particularly without SMT
- MP_TASK_AFFINITY=MCM has no noticeable effect

T799L91 10-day forecast (RAPS9)

P4+ to P5+ comparison

	CPU MPI x OMP	WALL (secs)	%Comms	Gflops	% of peak
Power4++ 1.9GHz p690+ hpcd	768 192 x 4	3848	12.6%	444	7.6%
Power5+ 1.9GHz p575+ hpce	768 SMT 192 x 8	2457	14.0%	696	11.8%

Total floating-point ops = $1,710,000 \times 10^9$

Speed-up: Power4++ \rightarrow Power5+ = 1.56

MFLOPS per routine for T399 on P4+ (RAPS9)

drhook statistics

%Self Time	Self Sec	Calls	MIPS	MFLOPS	%Div	Routine
8.94	41.61	20318	1738	589	4.3	CLOUDSC
8.66	40.32	6763206	1948	1122	0.5	CUADJTQ
2.93	13.63	39932	1921	927	0.0	LAITQM
2.93	13.63	5658	3167	3162	0.0	MXMAOP
2.73	12.72	120647	2992	2745	0.0	VERINT
2.68	12.48	10480	1449	311	4.3	CUBASEN
2.47	11.49	170933	1197	662	0.0	LAITLI
2.36	10.96	20960	1382	66	4.1	CUASCN
2.23	10.38	29985	2116	1058	1.3	VDFEXCU
2.20	10.26	30147	1615	954	0.0	LAITQMH
2.15	10.01	30186	1505	226	0.3	LASCAW
2.08	9.67	240456	1420	871	0.8	VDFEXCS
1.94	9.00	48	414	0	0.0	>OMP-CPG
1.73	8.04	29985	1304	326	0.5	VDFMAIN
1.63	7.59	30210	2502	1675	0.0	LARCHE
1.47	6.84	51	803	410	0.0	>OMP-FTINV_CTL
1.45	6.76	10060	1142	296	0.3	SLTEND
1.32	6.16	80	532	77	7.2	>OMP-RADINTG-INPUT
1.31	6.08	10054	1562	250	0.0	CALLPAR
1.09	5.07	1352707	1286	455	0.0	LAIDDIOBS

Thanks to
Sami Saarinen
for drhook

MFLOPS on P5+

Counter	Name	Groups
1	PM_FPU_1FLOP: FPU executed one flop	82,142,145,150
2	PM_FPU_FMA: FPU executed multiply-add	81,142,145,150
3	PM_FPU_STF: FPU executed store	84,143,145,147
4	PM_FPU_FIN: FPU produced a result	82,120,144,145,146
4	PM_FPU_FEST: FPU executed move or estimate	81

Two methods of calculating MFLOPS

$FLOPS = 2 * FMA + 1FLOP - FEST$ needs groups 81,145

$FLOPS = FMA + FIN - STF - FEST$ needs groups 81,145

Requires application to be run twice

Would like a new Group to include FMA, 1FLOP and FEST

CPI (Cycles Per Instruction) Analysis on P5 (RAPS9)

MAIN				
T	Cycles		1,049,275,053,531	
A		Groups	371,422,635,466	0.354
B		GCT	16,540,437,851	0.016
C		Stalls	661,311,980,214	0.630
A	Groups		371,422,635,466	
A1		Base	370,476,829,425	0.353
A2		Cracking	945,806,041	0.001
A1	Base		370,476,829,425	
A1A		Inst	245,031,876,787	0.234
A1B		Grouping	125,444,952,638	0.120
B	GCT		16,540,437,851	
B1		IC_Miss	7,365,359,574	0.007
B2		BR_MPred	6,124,862,149	0.006
B3		SRQ	2,905	0.000
B4		Other	3,050,213,223	0.003
C	Stalls		661,311,980,214	
C1		LSU	194,682,702,732	0.186
C2		FXU	65,549,096,175	0.062
C3		FPU	340,351,526,592	0.324
C4		Other	60,728,654,715	0.058
C1	LSU		194,682,702,732	
C1A		Reject	54,305,104,136	0.052
C1B		Dcache	77,594,158,087	0.074
C1C		Other	62,783,440,509	0.060
C2	FXU		65,549,096,175	
C2A		DIV	1,251,638,534	0.001
C2B		Other	64,297,457,641	0.061
C3	FPU		340,351,526,592	
C3A		FDIV	69,100,870,268	0.066
C3B		Other	271,250,656,324	0.259

Thanks to Lawrence Hannon,
IBM Houston, for assistance

CPI Analysis on P5+ (RAPS9)

LAIITLI

T	Cycles	18,544,481,257		
A	Groups	4,812,590,956	0.260	
B	GCT	77,943,390	0.004	
C	Stalls	13,653,946,911	0.736	
A	Groups	4,812,590,956		
A1	Base	4,811,679,026	0.259	
A2	Cracking	911,930	0.000	
A1	Base	4,811,679,026		
A1A	Inst	3,626,900,544	0.196	
A1B	Grouping	1,184,778,482	0.064	
B	GCT	77,943,390		
B1	IC_Miss	29,743,361	0.002	
B2	BR_MPred	35,234,272	0.002	
B3	SRQ	0	0.000	
B4	Other	12,965,757	0.001	
C	Stalls	13,653,946,911		
C1	LSU	11,515,115,570	0.621	
C2	FXU	322,778,945	0.017	
C3	FPU	1,562,957,716	0.084	
C4	Other	253,094,680	0.014	
C1	LSU	11,515,115,570		
C1A	Reject	3,959,632,336	0.214	
C1B	Dcache	7,113,000,180	0.384	
C1C	Other	442,483,054	0.024	
C2	FXU	322,778,945		
C2A	DIV	6,199,763	0.000	
C2B	Other	316,579,182	0.017	
C3	FPU	1,562,957,716		
C3A	FDIV	3,188,380	0.000	
C3B	Other	1,559,769,336	0.084	

CLOUDSC3

T	Cycles	18,536,291,356		
A	Groups	7,638,718,046	0.412	
B	GCT	639,681,301	0.035	
C	Stalls	10,257,892,009	0.553	
A	Groups	7,638,718,046		
A1	Base	7,622,129,292	0.411	
A2	Cracking	16,588,754	0.001	
A1	Base	7,622,129,292		
A1A	Inst	4,002,027,296	0.216	
A1B	Grouping	3,620,101,996	0.195	
B	GCT	639,681,301		
B1	IC_Miss	255,040,299	0.014	
B2	BR_MPred	289,763,875	0.016	
B3	SRQ	0	0.000	
B4	Other	94,877,127	0.005	
C	Stalls	10,257,892,009		
C1	LSU	1,241,869,614	0.067	
C2	FXU	3,988,830,685	0.215	
C3	FPU	2,713,645,467	0.146	
C4	Other	2,313,546,243	0.125	
C1	LSU	1,241,869,614		
C1A	Reject	87,962,372	0.005	
C1B	Dcache	550,216,030	0.030	
C1C	Other	603,691,212	0.033	
C2	FXU	3,988,830,685		
C2A	DIV	53,629,176	0.003	
C2B	Other	3,935,201,509	0.212	
C3	FPU	2,713,645,467		
C3A	FDIV	698,417,902	0.038	
C3B	Other	2,015,227,565	0.109	

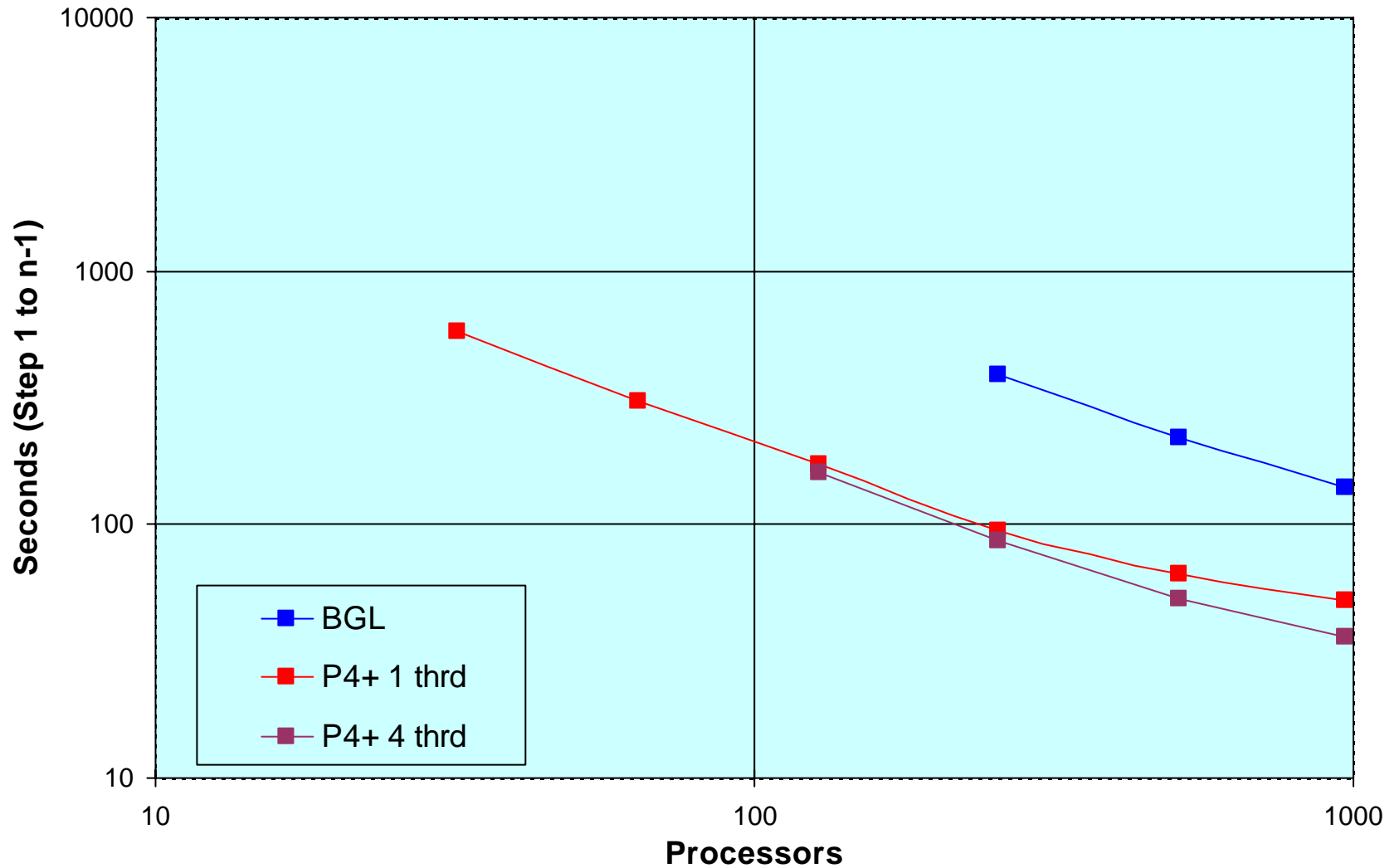
T799 L91 10 day forecast on P5+ (RAPS8)

- Run on P5+ at Poughkeepsie on 2112 CPUs =132 nodes
 - 528 MP5 tasks, and 8 OpenMP threads/task
 - 4224 total threads
- Run time for 10-day forecast was 804 seconds
 - 2.08 Tflops
 - 13% of peak
- For more on information about IFS
 - See ECMWF's "Twelfth Workshop on Use of High Performance Computing in Meteorology Oct30-Nov3
 - e-mail to hpcworkshop@ecmwf.int

Other Systems

- BlueGene L
- JS21 2.5 GHz Blade
- Power v Performance

T399 36hr Forecast: Pwr4+ v BGL times



VN Mode, 2 computational cpus/node

Comparison of JS21 Blade with Pwr4+ for RAPS9 IFS T399

- 2.5GHz JS21 Blade
 - 12 nodes
 - 4 cpus/node, 8GB/node
 - Myrinet switch
- 1.9GHz Pwr4+
 - 1 node
 - 32 cpus/node, 32GB/node
 - Federation switch
- IFS T399
 - RAPS9, 48 Steps, 1 day Forecast with Wave Model
 - ESSL and MASS libraries used
 - No additional tuning

Comparison of JS21 Blade with Pwr4+ for IFS T399

		2.5GHz JS21 Blade, 4 cpu nodes				1.9 GHz P4+, 32 cpu node				
CPUs	Time	Percent				Time	Percent			
		Comms	Barrier	Serial	I/O		Comms	Barrier	Serial	I/O
32	709s	11%	5%	4%	2%	549	3%	6%	3%	2%
48	510s	16%	5%	3%	2%					

- On 32 cpus, excluding Comms, Pwr4+ is $(631/532) = 1.18$ times faster than Blades
- So Pwr5+ is about $1.55 \times 1.18 = 1.83$ times faster than Blades

Power v Performance

System	Frequency	Power
Power5+	1.9 GHz	2.4KW for 16 proc node
JS21 Blade	2.5 GHz	5.4KW for Rack of 14 4-way Blades
BlueGene L	0.7 GHz	28KW for 2048 CPU Frame

System	KW/CPU	Power for equal IFS performance	
		CPUs	Power
Power5+	0.15	1	0.15
JS21 Blade	0.096	2	0.19
BlueGene L	0.0136	10	0.136

Approximate data: 10% error probable