



Performance analysis & tuning case studies

Brian Wylie & Markus Geimer Jülich Supercomputing Centre scalasca@fz-juelich.de October 2010



Additional Live-DVD example experiments



- jugene_sweep3d
 - ► 294,912 & 65,536 MPI processes on BG/P (trace)
- jump_zeusmp2
 - ► 512 MPI processes on p690 cluster (summary & trace)
- marenostrum_wrf-nmm
 - 1600 MPI processes on JS21 blade cluster, solver extract
 - summary analysis with 8 PowerPC hardware counters
 - trace analysis showing NxN completion problem on some blades
- neptun_jacobi
 - 12 MPI processes, or 12 OpenMP threads, or 4x3 hybrid parallelizations implemented in C, C++ & Fortran on SGI Altix
- ranger_smg2000
 - 12,288 MPI processes on Sun Constellation cluster, solve extract



- Comparison of NPB-BT class A in various configurations run on a single dedicated 16-core cluster compute node
 - 16 MPI processes
 - ► optionally built using MPI File I/O (e.g., SUBTYPE=full)
 - optionally including PAPI counter metrics in measurement (e.g., EPK_METRICS=PAPI_FP_OPS:DISPATCH_STALLS)
 - 16 OpenMP threads
 - 4 MPI processes each with 4 OpenMP threads (MZ-MPI)
- NPB-BT-MZ class B on Cray XT5 (8-core compute nodes)
 - 32 MPI processes with OMP_NUM_THREADS=8
 - More threads created on some processes (and fewer on others) as application attempts to balance work distribution
- NPB-MPI-BT on BlueGene/P with 144k processes
 - 1536x1536x1536 gridpoints distributed on 384x384 processes

Concurrency & Computation: Practice & Experience 22(6):702-719 (2010)

16-process trace analysis



16-process summary analysis with HWC metrics VI-HPS



16-process summary analysis: MPI File I/O time VI-HPS



16-process summary analysis: MPI File I/O time VI-HPS



16-thread summary analysis: Execution time

Cube 3.0 QT: epik_bt_A_0x16_sum/summary.cube.gz 9 _ □ × File Display Topology Help Peer percent Absolute Absolute Ŧ Ŧ Ŧ Metric tree Call tree Flat view System tree Topology 0 🕂 🗌 - Linux MVAPICH2 Intel 中 □ 0.00 Time 占 🔲 0.00 adi * 由 □ - i163-107 🗄 📕 178.74 Execution 占 🗖 0.01 compute rhs ⊡ 0.00 OMP 占 🗌 - Process 0 占 🔲 0.11 !\$omp parallel @rhs.f:17 - 🗌 0.00 Flush – 📕 8.11 !\$omp do @rhs.f:23 88.83 Thread 0 🗄 🔲 0.11 Management 🕂 🗖 5.55 !\$omp do @rhs.f:48 97.00 Thread 1 L 🔲 0.75 Fork 🔲 0.00 !\$omp master @rhs.f:61 71.25 Thread 2 🗄 🗌 0.00 Synchronization 📕 10.11 !\$omp do @rhs.f:67 72.67 Thread 3 占 🗌 0.00 Barrier 0.00 !\$omp master @rhs.f:170 88.96 Thread 4 – 🗌 0.00 Explicit 📥 🗖 9.86 !\$omp do @rhs.f:178 100.00 Thread 5 🗆 🗖 26.75 Implicit └ 🗌 0.00 !\$omp ibarrier @rhs.f:279 77.64 Thread 6 – 🔲 0.00 Critical 0.00 !\$omp master @rhs.f:280 93.17 Thread 7 0.00 Lock API 🕁 📕 16.32 !\$omp do @rhs.f:288 95.60 Thread 8 0.00 Overhead 📃 0.36 !\$omp do @rhs.f:346 85.67 Thread 9 占 🔲 0.55 Idle threads 0.12 !\$omp do @rhs.f:359 95.43 Thread 10 └ 🔲 0.00 Limited parallelism 8.01 !\$omp do @rhs.f:371 85.49 Thread 11 9.64e4 Visits 0.17 !\$omp do @rhs.f:387 93.50 Thread 12 🗄 📕 0.76 Computational imbalance 🗄 🔲 0.08 !\$omp do @rhs.f:400 94.93 Thread 13 📕 87.92 Thread 14 🔲 0.00 !\$omp master @rhs.f:411 3.35 !\$omp do @rhs.f:415 └ 📕 47.96 Thread 15 – 🗌 0.00 !\$omp ibarrier @rhs.f:426 🗄 🛄 31.73 x solve -• • F • [4 | F] 178.74 (86.39%) 9.86 (5.52%) 178.7 0.00 206.91 0.00 47.96 100.00 0.34 (3.49%) 9.86 Thread 15 finishes its work fastest ...

16-thread summary analysis: Implicit barrier time 16-thread summary analysis: Implicit barrier t

🗙 Cube 3.0 QT: epik_bt_A_Ox16_sum/summary.cube.gz 🥯 📃 🗌 🗙								
<u>F</u> ile <u>D</u> isplay <u>T</u> opology <u>H</u> elp								
Absolute 👻	Absolute 🗸		Peer percent -					
Metric tree	Call tree	Flat view		System tree Topology 0				
[中 🗋 0.00 Time 📃	_ ¢ □ ().00 adi		🗄 🗌 - Linux MVAPICH2 Inte				
📄 📄 📮 178.74 Execution	中[0.00 compute_rhs_		📄 📥 - i163-107				
白口 0.00 OMP		🗄 🔲 0.00 !\$omp parallel @rhs.f:17		🔄 📥 - Process 0				
– 🗌 0.00 Flush		– 🗌 0.00 !\$omp do @rhs.f:23		– 📃 37.42 Thread	0			
🗗 📮 0.11 Management		🕁 🛄 5.27 !\$omp do @rhs.f:48		– 💶 6.97 Thread 1				
🗌 🗆 🗖 0.75 Fork		– 🗌 0.00 !\$omp master @rhs.f:61		– 💻 74.73 Thread 🛛	2			
📥 🗌 0.00 Synchronization	*	– 🗌 0.00 !\$omp do @rhs.f:67		– 📕 69.32 Thread 🛛	3			
🕂 🕂 🗘 0.00 Barrier	*	– 🗌 0.00 !\$omp master @rhs.f:170		– 🗖 36.12 Thread	4			
– 🗌 0.00 Explicit		💠 🔲 0.00 !\$omp do @rhs.f:178		– 💶 14.87 Thread	5			
🗆 🗆 🖵 26.75 Implicit		🛛 📙 📕 4.53 !\$omp ibarrier @rhs.f:279		– 📃 59.15 Thread	6			
– 🗖 0.00 Critical 🎬		– 🗌 0.00 !\$omp master @rhs.f:280		– 💶 2.27 Thread 7				
L 🗌 0.00 Lock API		🕁 🗖 4.35 !\$omp do @rhs.f:288		– 💶 18.80 Thread	8			
– 🗌 0.00 Overhead		– 🗌 0.00 !\$omp do @rhs.f:346		– 📃 30.25 Thread	9			
🔄 📥 0.55 Idle threads	*	– 🗌 0.00 !\$omp do @rhs.f:359		– 💶 20.11 Thread	10			
🗆 🗆 🗆 0.00 Limited parallelism	r 	– 🗌 0.00 !\$omp do @rhs.f:371		– 🗖 35.88 Thread	11			
– 📕 9.64e4 Visits		– 🗌 0.00 !\$omp do @rhs.f:387		– 💶 1.87 Thread 1	2			
🗄 🖬 0.76 Computational imbalance 👘		🕁 🗖 2.78 !\$omp do @rhs.f:400		– 💶 20.47 Thread	13			
		– 🗌 0.00 !\$omp master @rhs.f:411		– 🗖 37.77 Thread	14			
		– 🗌 0.00 !\$omp do @rhs.f:415		🚽 🖵 🗖 100.00 Thread	15			
		🗕 📃 0.78 !\$omp ibarrier @rhs.f:426						
_		_ 1.61 x_solve_	.		-			
	•		•					
0.00 26.75 (12.93%) 206.91	0.00	4.53 (16.95%)	-	0 100.00	100.00			
but must then wait 0.80 (17.67%) 4.53								
		longest at end	OT I	oop				

16-thread summary analysis: Thread fork time VI-HPS

Absolute	Absolute 👻	Peer percent	
Metric tree	Call tree Flat view	System tree Topology 0	
O.00 Time O.00 Time O.00 OMP O.00 OMP O.00 Flush O.00 Flush O.00 Synchronization O.00 Synchronization O.00 Barrier O.00 Explicit O.00 Critical O.00 Critical O.00 Critical O.00 Critical O.00 Lock API O.00 Limited parallelism O.55 Idle threads O.55 Idle threads O.76 Computational imbalance ess than 1% overhead cr thread management	 □ 0.00 MAIN □ 0.00 set_constants □ 0.01 initialize □ 0.02 exact_rhs □ 0.00 adi □ 0.00 compute_rhs □ 0.00 compute_rhs □ 0.00 x_solve □ 0.00 x_solve □ 0.00 y_solve □ 0.00 y_solve □ 0.00 y_solve □ 0.14 !\$omp parallel @y_solve.f:42 □ 0.00 z_solve □ 0.14 !\$omp parallel @z_solve.f:42 □ 0.00 add □ 0.00 elapsed_time □ 0.00 verify □ 0.00 print_results 	- Linux MVAPICH2 Intel - i163-107 - Process 0 - 63.54 Thread 0 - 20.83 Thread 1 - 19.95 Thread 2 - 19.78 Thread 3 - 20.48 Thread 4 - 61.36 Thread 5 - 23.17 Thread 6 - 100.00 Thread 7 - 62.18 Thread 8 - 94.31 Thread 9 - 61.63 Thread 10 - 64.18 Thread 11 - 63.72 Thread 12 - 24.56 Thread 13 - 64.06 Thread 15	
0.00 0.75 (0.36%) 206.9	0.00 0.22 (28.75%) 0.75	0.00 64.06 100.00	

16-thread summary analysis: Idle threads time VI-HPS

Cube 3.0 QT: epik_bt_A_0x16_sum/summary.cube.gz 9 File Display Topology Help Absolute Peer percent Absolute Ŧ Ŧ Ŧ Metric tree Call tree Flat view System tree Topology 0 🕂 🗌 - Linux MVAPICH2 Intel 🕂 🗌 0.00 Time 占 🗖 0.02 MAIN ٠ 占 🗌 - i163-107 占 📕 178.74 Execution – 🔲 0.00 set constants ⊡ 0.00 OMP 占 🗌 - Process 0 🕂 🛄 0.00 initialize - 🗌 0.00 Flush 🗄 🛄 0.00 exact rhs – 🗌 0.00 Thread 0 🗄 🔲 0.11 Management 占 📃 0.02 adi 100.00 Thread 1 L 🔲 0.75 Fork 古 🗖 0.11 compute rhs 100.00 Thread 2 🗄 🗌 0.00 !\$omp parallel @rhs.f:17 🗄 🗌 0.00 Synchronization 100.00 Thread 3 占 🛄 0.10 x solve 古 🗌 0.00 Barrier 📕 100.00 Thread 4 – 🗌 0.00 Explicit 🗄 🗌 0.00 !\$omp parallel @x_solve.f:44 100.00 Thread 5 ₲ 🗌 0.00 🗆 🗖 26.75 Implicit Online description ⁽⁹⁾ 2 🗆 – 🔲 0.00 Critical 0.00 Lock API 🗄 📃 0.10 y solv OpenMP Idle threads Time 由 🗌 0.00 !\$o 0.00 Overhead 🕂 🔲 0.55 Idle threads Description: 古 🗖 0.10 z solv Idle time on CPUs that may be reserved for teams of threads when the process is 🗆 🗌 0.00 Limited parallelism 中口 0.00 !\$o executing sequentially before and after OpenMP parallel regions, or with less than 9.64e4 Visits 🕂 🔲 0.09 add the full team within OpenMP parallel regions. 🗄 📕 0.76 Computational imbalance 🕁 🗔 0.00 !\$o 🗌 0.00 elapsed 🕁 🛄 0.00 verify 99.74% of execution time parallel region body serial serial 🗄 🔲 0.00 print_res found in parallel regions ocation parallel region body parallel region body 0.00 0.55 (0.26%) 206.91 0.00 time

4x4 summary analysis: Execution time



4x4 summary analysis: OpenMP time

9% OpenMP time mostly Cube 3.0 QT: epik bt-mz-A 4x4 sum/summary.cube.gz found at implicit barriers File Display Topology Help Peer percent Absolute Ŧ Absolute -Ŧ Metric tree Call tree Flat view System tree Topology 0 🕂 🗌 0.00 Time ☆ □ - i127-211 🗄 🛄 0.02 exact rhs * Topology 0 System tree 占 📕 116.30 Execution 占 🗌 0.00 exch 🛛 dbc ⊢ □ - Process 0 占 🔲 0.01 MPI 🗄 🗖 0.12 copy 🗴 face – 🔲 0.18 Thread 0 🗄 🔲 0.02 Synchronization 🗄 🛄 0.10 copy y face 1.49 Thread 1 古 🗌 0.00 Communication – 🗌 0.00 MPI Isend 1.96 Thread 2 – 📃 3.37 Point-to-point – 🗌 0.00 MPI lrecv 100.00 Thread 3 0.03 Collective 🗆 🗌 0.00 MPI Waitall □ Process 1 由 □ 0.00 File I/O 🕂 🗌 0.00 adi – 🔲 0.11 Thread 0 - 📃 0.64 Init/Exit 🗄 🗖 2.47 compute rhs 1.41 Thread 1 中□0.00 OMP 🗄 📕 4.09 x solve 2.23 Thread 2 - 🗌 0.00 Flush 占 🗌 0.00 y solve 85.15 Thread 3 🗄 🔲 0.94 Management 占 🗌 0.00 !\$omp parallel @y_solve.f:43 🕂 🗌 - Process 2 - 🔲 0.14 Thread 0 🕂 🗖 12.82 Barrier └ 🔲 4.35 !\$omp ibarrier @y_solve.f:406 - 📕 0.82 Thread 1 – 🔲 0.00 Critical 🗄 🛄 1.33 z solve 0.67 Thread 2 🗆 🗔 0.00 Lock API 🕁 🗖 0.33 add 84.46 Thread 3 🔲 0.07 Overhead 0.00 MPI Barrier 🕂 🗌 - Process 3 🗄 🗖 13.18 Idle threads 🕁 🛄 0.01 verify - 🗖 0.77 Thread 0 6.34e6 Visits 0.00 MPI Reduce 0.58 Thread 1 🗄 📕 8 Synchronizations – 🗌 0.00 print results 0.49 Thread 2 . ▲ ▼ 4872 Communications 🕂 🗌 0.00 MPI Finalize 82.97 Thread 3 -₹Þ • **4 F** • 4 F • 147.39 4.35 (33.93%) 0.00 12.82 (8.70%) 0.00 12.82 0.00 100.00 0.14 0.00 (0.04%) 0 0 0 4 35

4x4 summary analysis: Idle threads time



4x4 summary analysis: MPI time



4x4 combined summary & trace analysis

Cube 3.0 QT: epik bt-mz-A 4x4 trace/combo.cube.gz × File Display Topology Help • Absolute Absolute Ŧ Peer percent Metric tree Call tree Flat view System tree Topology 0 ۲ 中 □ 0.00 Time 🗄 🗌 0.00 exact rhs ☆ 🗌 - i166-106 ٠ System tree Topology 0 🗄 📕 114.36 Execution 占 🗌 0.00 exch 🛛 dbc 中□-Process 0 占 🔲 0.00 MPI 🕁 🗌 0.00 copy 🗴 face 2.04 Thread 0 🗄 🔲 0.04 Synchronization 🗄 🗌 0.00 copy y face 0.00 Thread 1 – 🗌 0.00 MPI Isend – 🗌 0.00 Thread 2 🗄 🔲 0.44 Point-to-point – 🗌 0.00 MPI Irecv 0.00 Thread 3 🗄 🗖 2.70 Late Sender 🗆 🗖 2.70 MPI Waitall 🕂 🗌 - Process 1 └ □ 0.00 Late Receiver r □ 0.00 adi – 📕 100.00 Thread 0 🗄 🔲 0.00 Collective 🗄 🗌 0.00 compute rhs - 🗌 0.00 Thread 1 中口 0.00 File I/O 🗄 🗌 0.00 x solve 0.00 Thread 2 L 🚺 0.72 Init/Exit 🗄 🗌 0.00 y solve 🗆 🗔 0.00 Thread 3 ⊡ □ 0.00 OMP 占 🗌 0.00 !\$omp parallel @y_solve.f:43 □ Process 2 - 🗌 0.00 Flush ☐ 0.00 !\$omp do @y solve.f:52 61.27 Thread 0 🕂 🔲 1.03 Management └ 🗌 0.00 !\$omp ibarrier @y_solve.f:406 – 🗌 0.00 Thread 1 🗄 🗌 0.00 Synchronization 🗄 🗌 0.00 z solve - 🗌 0.00 Thread 2 ሱ 🔲 14.13 Barrier 🕁 🗌 0.00 add 0.00 Thread 3 – 🔲 0.00 Critical 0.00 MPI Barrier 🕂 🗌 - Process 3 🗆 🗔 0.00 Lock API 🕁 🗌 0.00 verify 85.03 Thread 0 0.03 Overhead - 🗌 0.00 MPI Reduce 0.00 Thread 1 💠 🗖 12.66 Idle threads 🗌 0.00 print results 0.00 Thread 2 7.39e5 Visits 0.00 Thread 3 • -•• •• **4 F** • • 2.70 (100.00%) 0.00 2.70 (1.85%) 146.12 0.00 2.70 0.00 100.00 100.00 1.09 (40.27%) 0 0 0 2.70

32x8 summary analysis: Excl. execution time

Generally good process Cube 3.0 QT: epik_bt-mz_1p32x8_sum/summary.cube.gz and thread load balance File Display Topology Help Peer percent Absolute Absolute -Topology 2 Metric tree Call tree Flat view Topology 1 占 🗌 0.00 Time 0.00 zone starts * 古 🗖 1581.71 Execution 0.00 set constants 🗗 🔲 0.04 MPI ሱ 🔲 11.83 initialize 🗄 🛄 39.28 Synchronization 🗄 🛄 2.51 exact rhs ☆ □ 0.00 Communication 🗄 🔲 0.09 exch qbc 🗕 🗔 1456.00 Point-to-point 🗄 🛄 1.64 copy x face 0.43 Collective 🗄 🛄 1.74 copy y face 由 □ 0.00 File I/O - 🗌 0.00 MPI Isend 2.33 Init/Exit - 🗌 0.00 MPI Irecv 中□0.00 OMP 🗌 0.00 MPI Waitall - 🗌 0.00 Flush 占 🔲 0.03 adi 🗄 🗖 827.66 Management 🗄 🗖 72.41 compute rhs 🗄 📃 478.64 x solve 古 🗌 0.00 Barrier 🗄 📃 488.57 v solve – 🗌 0.00 Explicit 🕂 🗖 512.77 z solve 🗆 🗖 1275.00 Implicit 🕁 🔲 10.07 add – 🔲 0.01 Critical 0.00 MPI Barrier 0.00 Lock API 🕁 🛄 1.28 verify 0.37 Overhead 0.00 MPI Reduce 🕂 📕 10499.94 Idle threads 0.01 print results └ 📃 103.91 Limited parallelism 🕂 🗌 0.00 MPI Finalize --•• **4** | F ۰. (**I)** • • 0.00 1581.71 (10.02%) 0.00 15786.70 0.00 512.77 (32.42%) 1581.71 100.00 100.00 512.77 (16477.68%) 0 0 0 3.11

32x8 summary analysis: Limited parallelism



32x8 summary analysis: Idle threads time



32x8 summary analysis: MPI communication tim



32x8 summary analysis: Implicit barrier time

Thread imbalance also Cube 3.0 QT: epik_bt-mz_1p32x8_sum/summary.cube.gz results in substantial loss File Display Topology Help Peer percent Absolute • Absolute -Topology 2 Metric tree Call tree Flat view Topology 1 中口 0.00 Time 0.00 zone starts * 🗄 🗖 1581.71 Execution 0.00 set constants 占 🔲 0.04 MPI 💠 🔲 3.26 initialize 🗄 📃 39.28 Synchronization 🗄 🛄 1.67 exact rhs 古 🗌 0.00 Communication 🗄 🗌 0.00 exch gbc 🗕 🗖 1456.00 Point-to-point 🗄 📃 245.33 copy x face 0.43 Collective 🕀 🗖 216.09 copy y face 由 □ 0.00 File I/O - 🗌 0.00 MPI Isend - 🔲 2.33 Init/Exit - 🗌 0.00 MPI Irecv 中□0.00 OMP 🗌 0.00 MPI Waitall - 🗌 0.00 Flush 占 🗌 0.00 adi 🗄 🗖 827.66 Management 🗄 🗖 236.79 compute rhs 🗄 🗌 0.00 Synchronization 🗄 📃 165.04 x solve 古 🗌 0.00 Barrier 🗄 🗖 166.29 y solve – 🗌 0.00 Explicit 🕂 🔄 181.88 z solve └ 🛄 1275.00 Implicit 🕁 🗖 56.77 add – 🔲 0.01 Critical 0.00 MPI Barrier 🗆 🗌 0.00 Lock API 🕁 🛄 1.88 verify 0.37 Overhead 0.00 MPI Reduce 🕂 📕 1.05e4 Idle threads 0.00 print results └ 🔲 103.91 Limited parallelism 🕂 🗌 0.00 MPI Finalize --Ŧ **4** | F ۰. (**4** | **F**] • • 0.00 1275.00 (8.08%) 0.00 0.00 181.88 (14.26%) 1275.00 100.00 100.00 1.58e4 0.00 181.88 (3896.57%) 4 67

32x8 summary analysis: Thread management

Thread management cost Cube 3.0 QT: epik_bt-mz_1p32x8_sum/summary.cube.gz high with over-subscription File Display Topology Help Peer percent Absolute • Absolute • Topology 2 Metric tree Call tree Flat view Topology 1 🕂 🗌 0.00 Time 0.00 zone starts 占 🗖 1581.71 Execution 0.00 set constants 占 🔲 0.04 MPI 🗄 🛄 0.63 initialize 🗄 📃 39.28 Synchronization 🗄 🔲 0.27 exact rhs ☆ □ 0.00 Communication 🗄 🗌 0.00 exch qbc 🗕 🗖 1456.00 Point-to-point 🗄 📃 246.02 copy x face 0.43 Collective 🕀 📃 251.42 copy y face 由 □ 0.00 File I/O - 🗌 0.00 MPI Isend 2.33 Init/Exit - 🗌 0.00 MPI Irecv 中□0.00 OMP 🗌 0.00 MPI Waitall – 🗌 0.00 Flush 🕂 🗌 0.00 adi 🕂 🗖 827.66 Management 🗄 🗖 63.23 compute rhs 🗄 🗌 0.00 Synchronization 🗄 🗖 68.47 x solve 古 🗌 0.00 Barrier 🗄 🗖 62.99 y solve - 🗌 0.00 Explicit 🗄 🗖 63.69 z solve - 📃 1275.00 Implicit 🕂 🗖 69.95 add - 🔲 0.01 Critical 0.00 MPI Barrier 0.00 Lock API 🕁 🛄 0.97 verify 0.37 Overhead 0.00 MPI Reduce 🕂 📕 10499.94 Idle threads 0.00 print results └ 📃 103.91 Limited parallelism 🕂 🗌 0.00 MPI Finalize --Ŧ **4** | F ۰. (**4** | **F**] • • 0.00 827.66 (5.24%) 0.00 15786.70 0.00 63.69 (7.70%) 827.66 100.00 100.00 0 0 0 63.69 (3405.86%) 1.87

32x8 summary analysis: Critical section time

Atomic statements during Cube 3.0 QT: epik_bt-mz_1p32x8_sum/summary.cube.gz 9 verification are efficient File Display Topology Help Peer percent Absolute • Absolute -Topology 2 Metric tree Call tree Flat view Topology 1 4 中 □ 0.00 Time 🗆 🗌 0.00 MPI Waitall 🗄 🗖 1581.71 Execution i ⊡ 0.00 adi 占 🔲 0.04 MPI 🗄 🗌 0.00 compute rhs 🗄 🔲 39.28 Synchronization 古 🗌 0.00 Communication 🕁 🗌 0.00 y solve 🗕 📃 1456.00 Point-to-point 🕁 🗌 0.00 z solve 0.43 Collective 中 □ 0.00 add 由 □ 0.00 File I/O 0.00 MPI Barrier 2.33 Init/Exit 占 🗌 0.00 verify 中□0.00 OMP 🗄 🗌 0.00 error norm - 🗌 0.00 Flush 占 🗌 0.00 !\$omp parallel @error.f:27 🗄 🗖 827.66 Management 0.01 !\$omp atomic @error.f:51 古 🗌 0.00 Barrier └ 🕅 0.00 !\$omp ibarrier @error.f:54 – 🗌 0.00 Explicit 🗄 🗌 0.00 compute rhs 🗆 📃 1275.00 Implicit 🗄 🗌 0.00 rhs norm 0.01 Critical 占 🗌 0.00 !\$omp parallel @error.f:86 └ 🕅 0.00 Lock API - 🗌 0.00 !\$omp do @error.f:91 0.37 Overhead - 🗖 0.01 !\$omp atomic @error.f:104 🕂 📕 10499.94 Idle threads 🗆 🔲 0.00 !\$omp ibarrier @error.f:107 🔒 └ 🔲 103.91 Limited parallelism 0.00 MPI Reduce -Ŧ • F (F • F ۰. • • 0.00 0.01 (0.00%) 0.00 15786.70 0.00 0.01 (37.96%) 0.01 100.00 100.00 0 0 0 0.01 (9609.64%) 0 00

NPB-MPI-BT on **BlueGene/P** case study



- 3D solution of unsteady, compressible Navier-Stokes eqs
 - NASA NAS parallel benchmark suite Block-Tridiagonal solver
 - series of ADI solve steps in X, Y & Z dimensions
 - ~9,500 lines (20 source modules), mostly Fortran77
- Run on IBM BlueGene/P in VN mode with 144k processes
 - Good scaling when problem size matched to architecture
 - 1536x1536x1536 gridpoints mapped onto 384x384 processes
 - Measurement collection took 53 minutes
 - 38% dilation for summarization measurement compared to uninstrumented execution (using 10 function filter)
 - MPI trace size would be 18.6TB
 - 25% of time in ADI is point-to-point communication time
 - ► 13% copy_faces, 23% x_solve, 33% y_solve, 31% z_solve
 - 128s for a single MPI_Comm_split during setup!

NPB-MPI-BT on jugene@144k summary analysis



NPB-MPI-BT on jugene@144k summary analysis





- Molecular mechanics simulation
 - original version developed by Robert W. Harrison
- SPEC OMP benchmark parallel version
 - ~14,000 lines (in 28 source modules): 100% C
- Run with 32 threads on SGI Altix 4700 at TUD-ZIH
 - Built with Intel compilers
 - 333 simulation timesteps for 9,582 atoms
- Scalasca summary measurement
 - Minimal measurement dilation
 - 60% of total time lost in synchronization with lock API
 - 12% thread management overhead

ammp on jupiter@32 OpenMP lock analysis





ammp on jupiter@32 OpenMP fork analysis





WRF/MareNostrum case study



- Numerical weather prediction
 - public domain code developed by US NOAA
 - flexible, state-of-the-art atmospheric simulation
 - Non-hydrostatic Mesoscale Model (NMM)
- MPI parallel version 2.1.2 (Jan-2006)
 - >315,000 lines (in 480 source modules): 75% Fortran, 25% C
- Eur-12km dataset configuration
 - 3-hour forecast (360 timesteps) with checkpointing disabled
- Run with 1600 processes on MareNostrum
 - IBM BladeCenter cluster at BSC
- Scalasca summary and trace measurements
 - 15% measurement dilation with 8 hardware counters
 - 23GB trace analysis in 5 mins

Journal of Scientific Programming 16(2-3):167-181 (2008)

WRF on MareNostrum@1600 with HWC metrics VI-HPS



WRF on MareNostrum@1600 trace analysis





WRF on MareNostrum@1600 time-line extract





- Limited system I/O requires careful management
 - Selective instrumentation and measurement filtering
- PowerPC hardware counter metrics included in summary
 - Memory/cache data access hierarchy constructed
- Automated trace analysis quantified impact of imbalanced exit from MPI_Allreduce in "NxN completion time" metric
 - Intermittent but serious MPI library/system problem, that restricts application scalability
 - Only a few processes directly impacted, however, communication partners also quickly blocked
- Presentation using logical and physical topologies
 - MPI Cartesian topology provides application insight
 - Hardware topology helps localize system problems

Journal of Scientific Programming 16(2-3):167-181 (2008)

XNS on BlueGene/L case study



- CFD simulation of unsteady flows
 - developed by RWTH CATS group of Marek Behr
 - exploits finite-element techniques, unstructured 3D meshes, iterative solution strategies
- MPI parallel version (Dec-2006)
 - >40,000 lines of Fortran & C
 - DeBakey blood-pump dataset (3,714,611 elements)



XNS-DeBakey on jubl@4096 summary analysis VI-HPS



XNS-DeBakey scalability on BlueGene/L







- Globally synchronized high-resolution clock facilitates efficient measurement & analysis
- Restricted compute node memory limits trace buffer size and analyzable trace size
- Summarization identified bottleneck due to unintended P2P synchronizations (messages with zero-sized payload)
- 4x solver speedup after replacing MPI_Sendrecv operations with size-dependent separate MPI_Send and MPI_Recv
- Significant communication imbalance remains due to mesh partitioning and mapping onto processors
- MPI_Scan implementation found to contain implicit barrier
 - responsible for 6% of total time with 4096 processes
 - decimated when substituted with simultaneous binomial tree

Proc. 14th EuroPVM/MPI, LNCS 4757 (2007)



- Coulomb solver used for laser-plasma simulations
 - Developed by Paul Gibbon (JSC)
 - Tree-based particle storage with dynamic load-balancing
- MPI version
 - PRACE benchmark configuration, including file I/O
- Run on BlueGene/P in dual mode with 1024 processes
 - 2 processes per quad-core PowerPC node, 1100 seconds
 - IBM XL compilers, MPI library and torus/tree interconnect
- Run on Cray XT in VN (4p) mode with 1024 processes
 - 4 processes per quad-core Opteron node, 360 seconds
 - PGI compilers and Cray MPI, CNL, SeaStar interconnect

PEPC@1024 on BlueGene/P: Wait at NxN time

Time waiting for last rank Cube 3.0 QT: epik_PEPC_dual1024_trace/trace.cube.gz 🤗 to enter MPI Allgather File Display Topology Help Own root percent Absolute Peer percent Ŧ Ŧ Metric tree Call tree Flat view System tree Topology 0 🕂 🗌 0.00 Time 🗄 🗌 0.00 jube kernel run 占 🗌 - IBM BG/P (JUGENE) 占 📕 67.65 Execution 0.00 laser 🕁 📕 99.64 R12-M0-N0 ⊡ 0.00 MPI 🗗 🗌 0.00 pepc fields p 🕂 📕 99.73 R12-M0-N4 🗄 🗌 0.00 Synchronization 0.00 utils NMOD cput 由 📕 100.00 R12-M0-N8 ⊨ □ 0.00 Collective 🕂 🗖 152.36 tree domains 99.71 R12-M0-Nc 🛛 📕 29.28 Wait at Barrier – 🗌 0.00 tree build 98.24 R12-M0-N1 🗆 🛄 0.00 Barrier Completion 🗄 🔲 0.21 tree branches 🕁 📕 98.12 R12-M0-N5 占 🗌 0.00 Communication 🗄 🗌 0.00 tree fill 🕂 📕 98.33 R12-M0-N9 🗄 🛄 0.14 Point-to-point 🗄 🔲 0.04 tree properties 98.17 R12-M0-Nd 🗄 🔲 0.02 Late Sender ☆ □ 0.00 tree walk 🕂 📕 97.80 R12-M0-N2 🗄 🔲 0.00 Messages in Wrong Order – 🗌 0.00 utils NMOD cpu ሱ 📕 98.24 R12-M0-N6 0.00 Late Receiver - 🗌 0.00 MPI Barrier ሱ 📕 97.93 R12-M0-Na 占 🔲 0.39 Collective 9735.67 MPI Allgather 🕂 📕 97.99 R12-M0-Ne – 🗌 0.00 Early Reduce – 🔲 7.65 MPI Alltoall 🕂 📕 97.20 R12-M0-N3 🗌 0.00 Early Scan – 🗌 0.00 MPI Irecv ሱ 📕 97.84 R12-M0-N7 0.00 Late Broadcast – 🗌 0.00 tree utils NMOD ሱ 📕 97.18 R12-M0-Nb – 📃 1.20 Wait at N 🗙 N - 🗌 0.00 MPI Isend 🕁 📕 96.77 R12-M0-Nf 0.02 N x N Completion – 🗌 0.00 MPI Request free 0.00 File I/O 🗆 🗌 0.00 MPI Waitany L 🔲 0.18 Init/Exit 0.00 sum force ▲ ▼ 1.10 Overhead 371.96 MPI Allreduce -**()** • ٩Þ **4** | F • • 0.00 100.00 0.00 9735.67 (72.33%) 13460.87 0.00 100.00 1.20 0.00 13460.87 (1.20%) 1118863.88

PEPC@1024 on Cray XT4: Wait at NxN time

Time waiting for last rank Cube 3.0 QT: epik_PEPC_4p1024_trace/trace.cube.gz 9 to enter MPI Allgather File Display Topology Help Own root percent Absolute Peer percent Ŧ Ŧ Metric tree Call tree Flat view System tree Topology 0 🗄 🗌 - Cray XT (Jaguar) 🕂 🗌 0.00 Time 🗄 🗌 0.00 jube kernel run 占 📕 62.84 Execution 🗄 📕 96.11 c29-3c2s4n3 0.00 laser ⊡ 0.00 MPI 🕂 📕 96.79 c29-3c2s5n0 占 🗌 0.00 pepc fields p 🗄 🗌 0.00 Synchronization - 🗌 0.00 cput 🕂 📕 96.55 c29-3c2s5n1 🕂 🔲 0.04 Collective 🕂 📕 95.34 c29-3c2s5n2 🗄 🛄 125.19 tree domains - 📕 33.37 Wait at Barrier ሱ 📕 97.77 c29-3c2s5n3 🗌 0.00 tree build – 🔲 0.04 Barrier Completion 由 📕 94.84 c29-3c2s6n0 🕀 🗖 104.67 tree branches 占 🗌 0.00 Communication 🕁 🗌 0.00 tree fill 由 📕 97.33 c29-3c2s6n2 🗄 🛄 0.15 Point-to-point 🕂 📕 97.28 c29 3c2s6n3 🕂 🔲 2.62 tree properties 🗄 🔲 0.04 Late Sender 中 📕 97.85 c29-3c2s7n0 🗄 🗌 0.00 tree walk 🗄 🔲 0.02 Messages in Wrong Order ሱ 📕 94.63 c29-3c2s7n1 – 🗌 0.00 cput 🗆 🗔 0.00 Late Receiver 由 📕 96.41 c29-3c2s7n2 0.00 MPI Barrier 占 🗖 1.16 Collective 4248.05 MPI Allgather 由 📕 95.73 c29-3c2s7n3 – 🗌 0.00 Early Reduce 📕 349.33 MPI Alltoall 由 📕 94.78 c27 0c0s0n0 🗌 0.00 Early Scan ሱ 📕 93.84 c27-0c0s0n1 0.00 MPI lrecv 0.00 Late Broadcast - 🗌 0.00 indsort i4 ሱ 📕 98.06 c27-0c0s0n2 – 📃 1.80 Wait at N 🗙 N 🕁 📕 94.99 c27-0c0s0n3 - 🗌 0.00 MPI Isend 0.36 N x N Completion 由 📕 96.73 c27 0c0s1n0 – 🗌 0.00 MPI Request free 0.00 File I/O 由 📕 94.20 c27 0c0s1n1 – 🗌 0.00 MPI Waitany 🗆 🛄 0.05 Init/Exit – 🗌 0.00 make hashentry 中 📕 96.36 c27-0c0s1n2 0.14 Overhead 井 📕 93.67 c27-0c0s1n3 T 0.00 next node --**()** • • [4] F.] 4 | F 0.00 4248.05 (65.12%) 100.00 0.00 6523.90 0.00 100.00 1.80 0.00 6523.90 (1.80%) 363345.55



- Despite very different processor and network performance, measurements and analyses can be easily compared
 - different compilers affect function naming & in-lining
- Both spend roughly two-thirds of time in computation
 - tree_walk has expensive computation & communication
- Both waste 30% of time waiting to enter MPI_Barrier
 - not localized to particular processes, since particles are regularly redistributed
- Most of collective communication time is also time waiting for last ranks to enter MPI_Allgather & MPI_Alltoall
 - imbalance for MPI_Allgather twice as severe on BlueGene/P, however, almost 50x less for MPI_Alltoall
 - collective completion times also notably longer on Cray XT



- 3D reservoir simulator combining alternating
 - PFLOW non-isothermal, multiphase groundwater flow
 - PTRAN reactive, multi-component contaminant transport
 - developed by LANL/ORNL/PNNL
- MPI with PETSc, LAPACK, BLAS & HDF5 I/O libraries
 - ~80,000 lines (97 source files) Fortran9X
 - PFLOTRAN & PETSc fully instrumented by IBM XL compilers
 - Filter produced listing 856 USR routines (leaving 291 COM)
 - ► 1732 unique callpaths (399 in FLOW, 375 in TRAN)
 - ► 633 MPI callpaths (121 in FLOW, 114 in TRAN)
 - 29 distinct MPI routines recorded (excludes 15 misc. routines)
- Run on IBM BlueGene/P with '2B' input dataset (10 steps)
 - Scalasca summary & trace measurements (some with PAPI)
- 22% dilation of FLOW, 10% dilation of TRAN [8k summary] Dagstuhl seminar 10181 (2010)











Geology of DOE Hanford 300 area (WA)







VI-HPS



Selected "13106.965 MPI_Comm_dup"



- Initialization phase dominates at larger scales
 - 10% of total execution time spent duplicating communicators with 128k processes on Cray XT5
 - otherwise collective MPI File I/O relatively efficient
 - typically amortized in long simulation runs
- Solver scaled well to 64k processes before degrading
 - similar computation/communication patterns in FLOW & TRAN
 - callpath profiles distinguish costs
 - MPI_Allreduce collective communication becomes a bottleneck
 - communication overhead explodes for smaller FLOW problem
 - TRAN problem is 15x larger due to 15 chemical species
 - inactive processes induce clear computational imbalance
 - ► and are associated with large amounts of MPI waiting time
 - however, they constitute a relatively small minority

Dagstuhl seminar 10181 (2010)



- ASC benchmark
- direct order solve uses diagonal sweeps through grid cells

VI-HPS

- 'fixups' applied to correct unphysical (negative) fluxes
- MPI parallel version 2.2b using 2D domain decomposition
 - ~2,000 lines (12 source modules), mostly Fortran77
- Run on IBM BlueGene/P in VN mode with 288k processes
 - 7.6TB trace written in 17 minutes, analyzed in 10 minutes
 - of which 10 minutes for SIONlib open/create of 576 physical files
 - (compared to 86 minutes just to open/create a file per MPI rank)
 - Mapping of metrics onto application's 576x512 process grid reveals regular pattern of performance artifacts
 - computational imbalance originates from 'fixup' calculations
 - combined with diagonal wavefront sweeps amplifies waiting times

Proc. IPDPS Workshop on Large-Scale Parallel Processing (2010)

sweep3d on jugene@288k trace analysis



sweep3d on jugene@288k trace analysis



sweep3d on jugene@288k trace (wait) analysis VI-HPS





- The application and benchmark developers who generously provided their codes and/or measurement archives
- The facilities who made their HPC resources available and associated support staff who helped us use them effectively
 - ALCF, BSC, CSC, CSCS, EPCC, JSC, HLRN, HLRS, ICL, KSL, LRZ, NCAR, NCCS, NICS, RWTH, RZG, SARA, TACC, ZIH
 - Access & usage supported by European Union, German and other national funding organizations
- The Scalasca development team





Scalable performance analysis of large-scale parallel applications

- toolset for scalable performance measurement & analysis of MPI, OpenMP & hybrid parallel applications
- supporting most popular HPC computer systems
- available under New BSD open-source license
- sources, documentation & publications:
 - http://www.scalasca.org
 - mailto: scalasca@fz-juelich.de

