Tuning Applications for the Origin

Ruud van der Pas

SiliconGraphics

European High Performance Computing Team

Programming Models

Status of the SGI compilers:

- Interprocedural Analysis based optimizations
- Multi–level unrolling
- Instruction scheduling well under control
- Tuning for cache usage (auto–blocking)
- Increased focus on auto–parallelisation

Programming Models

Areas where the user has an advantage

- High–level parallelisation
- Exploiting the memory hierarchy

However, be aware:

- Compilers are getting better
- Think first, before spending time:

Hardware Latency Hierarchy

Access to processor registers	0 ns
Access to 1st level cache (3 cycles –> 1 cycle hw overlap) (5 ns/cycle for 200 MHz cpu)	5 ns
Access to 2nd level cache (10 cycles; also hw&sw overlap) (133 MHz or 7.5 ns cycle time) (Proper scheduling of the code allows pipelining references to the 2nd level cache, reducing the given latency)	75 ns
Access to Local Memory (61 cycles) • 100 ns in R10000 (L2), 213 ns in Hub (with prefetch these latencies can be further reduced)	313 ns
Back–to–back Hubs Memory Access (100 cycles) • 250 ns for local miss • 235 ns remote hub	497 ns
Remote memory Access through Routers: • 485 ns for hub pair management • 100 ns for each additional router layer (e.g. 1 router 117 cycles, 2 routers 137 cycles, 3 routers 157 cycles)	497+n•103 ns

Where to put effort

King Cache: Get it into L2

- **Differences in nonlocal cache refill latencies**
  - –important to know about
  - –can be critical in a minority of cases (18% !)

- **Approximate latencies on Origin** (in cycles; 32 cpus)

Registers	0
L1	1
L2	10
Local memory	61
1 Router away	117
2 Routers	137
3 Routers	157
PChall memory	>200

Reduced by HW assisted overlap

Reduced by SW assisted overlap

- **Move memory –> L2**
  remote –> local

> 6X win
< 3X win

Where to put effort

The Hypercube topology: A lot of memory is "nearby"

CPUs (max)	Memories	Distance (max)	Memory Fraction
32 cpu system (= 16 nodes, 16 memories)			
2	1	local	6.3%
4	2	1 Router	12.5%
16	8	2 Routers	50.0%
32	16	3 Routers	100.0%
64 cpu system (=32 nodes, 32 memories)			
2	1	local	3.1%
4	2	1 Router	6.3%
16	8	2 Routers	25.0%
32	16	3 Routers	50.0%
64	32	4 Routers	100.0%

On a large machine the memory fraction at distance <= d routers grows ~exponentially until significant fraction of total memory is included

Almost never need go more than 1–2 hops, even under load