Job Scheduling Techniques for Large Origin Systems

Steve Caruso
HPC Systems Engineer, SGI
scc@sgi.com

SUMMIT 2001
43rd CUG Conference on High Performance Computing & Visualization

Agenda

- **Batch job scheduling with dynamic cpusets**
- **Preemptive, priority job scheduling for operational weather forecasting at FNMOC**

Batch Job Scheduling with Dynamic Cpusets

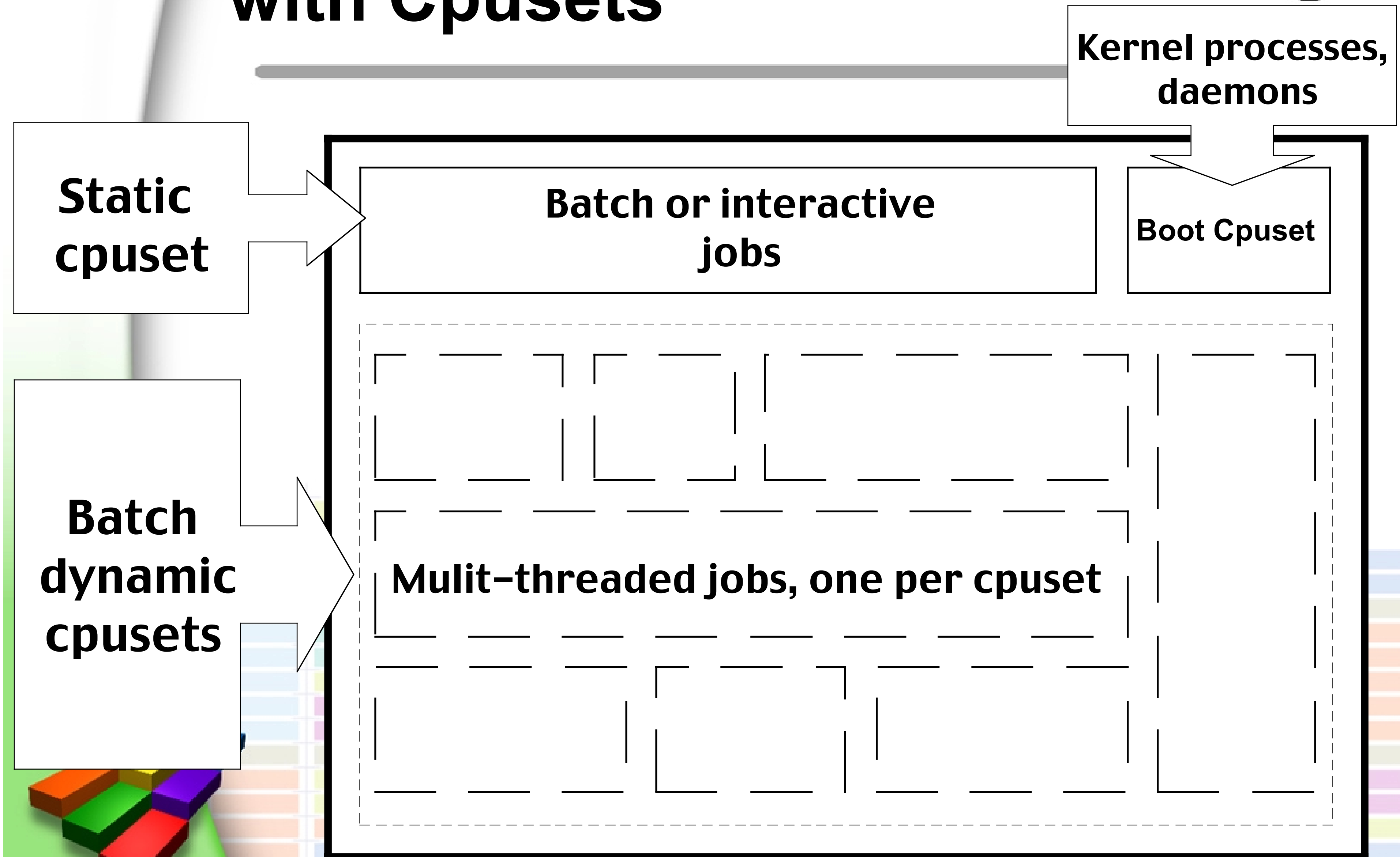
Motivation: Minimize runtime variation & job interference on a loaded system

- IRIX Cpusets provide method for allocating and isolating CPUs and memory for individual jobs
- Effective partitioning of system resources
  - both static & dynamic partitions
- Batch systems support cpusets
  - LSF, PBS, GRD, …

IRIX Cpusets

- Cpusets:
  - Group physical CPUs (& memory) as unit
  - Jobs (parent, children) execute in cpusets
  - Can set process and memory policies
- Static - can exist across reboots
- Dynamic - create/destroy before/after job runs
- Integration with batch systems (via API & CLI)

Resource 'Partitioning' with Cpusets



Benchmark Example:LSF+Cpusets

- Customer throughput benchmark
- 50 jobs, 10 codes
  - same & different: # processors, input data
  - all MPI, one hybrid (MPI+OpenMP)
- No changes to number of processors or order of submission
- Included 5 min. sleeps between jobs
- No tricks allowed
- Used dedicated 128P/128GB O3000

  (all 128P used for compute)

Benchmark Example:LSF+Cpusets

- LSF ***without*** cpusets
  - Total elapsed time = 4:19:01
  - In particular, identical pgm_02 jobs:

```
038_pgm_02:real     39:34.36
039_pgm_02:real      8:17.02
```

- LSF ***with*** cpusets
  - Total elapsed time = 2:51:06
  - In particular, identical pgm_02 jobs:

```
038_pgm_02:real     8:08:39
039_pgm_02:real     8:03.69
```

  - Ideal time (no sleeps) = 2:21:22
  - Time without sleeps = 2:35:47

Benchmark Results: LSF with and w/o dynamic cpusets

code	Time dedicated	Time w/o cpusets	Ratio (w/o) / ded.	Time with cpusets	Ratio with / ded.
001_pgm_03	0:29:20	0:31:25	1.07	0:30:04	1.03
002_pgm_03	0:16:29	0:20:17	1.23	0:16:32	1.00
003_pgm_11	0:17:42	0:19:13	1.09	0:18:14	1.03
004_pgm_12	0:04:59	0:06:52	1.38	0:05:04	1.02
005_pgm_06	0:04:41	0:04:55	1.05	0:04:56	1.05
006_pgm_08	0:13:52	0:15:49	1.14	0:12:53	0.93
007_pgm_06	0:04:41	0:05:00	1.07	0:04:54	1.04
008_pgm_12	0:04:59	0:07:26	1.49	0:05:07	1.03
009_pgm_02	1:03:29	2:23:00	***2.25***	1:03:29	1.00
010_pgm_10	0:27:24	0:27:40	1.01	0:28:32	1.04
011_pgm_09	0:11:31	0:21:51	1.90	0:11:18	0.98
012_pgm_02	1:03:29	3:22:44	***3.19***	1:02:33	0.99
013_pgm_11	0:17:42	0:18:46	1.06	0:17:48	1.01
014_pgm_01	0:40:00	0:37:41	0.94	0:36:16	0.91
015_pgm_06	0:02:58	0:03:01	1.01	0:02:58	1.00
016_pgm_08	0:26:15	1:18:46	***3.00***	0:26:29	1.01
017_pgm_01	0:20:00	0:34:15	***1.71***	0:18:23	0.92
018_pgm_06	0:02:58	0:02:59	1.01	0:02:58	1.00
019_pgm_06	0:01:13	0:01:25	1.17	0:01:15	1.03
020_pgm_08	0:51:07	0:50:16	0.98	0:48:52	0.96
021_pgm_06	0:07:32	0:07:39	1.02	0:07:33	1.00
022_pgm_06	0:01:13	0:01:16	1.04	0:01:15	1.02
023_pgm_07	0:02:37	0:03:22	1.28	0:03:09	1.20
024_pgm_06	0:01:13	0:01:16	1.04	0:01:15	1.03
025_pgm_03	0:17:54	0:18:33	1.04	0:18:25	1.03

code	dedicated	Time w/o cpusets	Ratio (w/o) / ded.	Time with cpusets	Ratio with / ded.
026_pgm_04	0:40:11	0:55:58	1.39	0:35:08	0.87
027_pgm_06	0:01:13	0:01:19	1.09	0:01:14	1.02
028_pgm_05	0:11:43	0:13:21	1.14	0:12:03	1.03
029_pgm_07	0:02:37	0:07:36	***2.91***	0:02:45	1.05
030_pgm_08	0:51:07	1:05:15	1.28	0:48:51	0.96
031_pgm_07	0:02:37	0:02:44	1.04	0:02:45	1.05
032_pgm_12	0:04:59	0:12:57	***2.60***	0:05:02	1.01
033_pgm_11	0:11:03	0:11:07	1.01	0:11:07	1.01
034_pgm_06	0:02:58	0:03:02	1.02	0:02:58	1.00
035_pgm_08	0:26:15	1:42:12	***3.89***	0:26:37	1.01
036_pgm_06	0:07:32	0:07:41	1.02	0:07:33	1.00
037_pgm_01	0:40:00	0:37:33	0.94	0:36:32	0.91
038_pgm_02	0:07:35	0:39:34	***5.22***	0:08:08	1.07
039_pgm_02	0:07:35	0:08:17	1.09	0:08:04	1.06
040_pgm_03	0:53:05	0:55:18	1.04	0:53:38	1.01
041_pgm_12	0:04:59	0:07:23	1.48	0:04:59	1.00
042_pgm_12	0:04:59	0:08:36	1.73	0:05:01	1.01
043_pgm_06	0:01:13	0:01:22	1.12	0:01:19	1.09
044_pgm_08	0:13:52	0:58:20	***4.21***	0:12:56	0.93
045_pgm_12	0:04:59	0:06:25	1.29	0:05:01	1.01
046_pgm_11	0:17:42	0:18:19	1.04	0:17:41	1.00
047_pgm_06	0:07:32	0:07:36	1.01	0:07:32	1.00
048_pgm_01	0:20:00	0:21:37	1.08	0:18:21	0.92
049_pgm_01	0:02:14	0:02:26	1.09	0:02:23	1.07

Batch Job Scheduling with Dynamic Cpusets - Conclusions

- Job runtime variations *dramatically* reduced
- Effective use of batch-only systems
  - Easy configuration
  - Increase system utilization *and* user satisfaction
- Can configure to work with batch+interactive systems
  - Using boot & static cpusets
  - Minimal sys admin once configured
- Supported in LSF, PBS, GRD

Preemptive, Priority Job Scheduler

- Developed for operational weather forecasting at Fleet Numerical Meterology & Oceanography Center (FNMOC)
- Replicate functionality on C90s

  (UNICOS, NQE, fair share scheduler)

  - High priority jobs preempt lower priority ones, obtain *required* resources fast
  - Low priority jobs resume after high priority jobs complete

Preemptive, Priority Job Scheduler

Implementation:

- LSF 4.1, IRIX cpusets, resource pool manager (rpmd)
- Normal-priority job process:
  - Jobs submitted to LSF, held in pending state
  - Rpmd manages available resources (cpu & memory)
    - If available, creates cpuset, then requests LSF to release and attach job to cpuset
    - Destroys cpuset when job exits

Preemptive, Priority Job Scheduler

- High-priority job process:
  - Jobs submitted to LSF, held in pending state
  - Rpmd considers lower priority jobs to preempt
    - Suspends lower priority job(s), destroys cpusets
    - Creates cpuset for priority job, requests LSF to release and attach job to cpuset
    - Destroys cpuset when jobs exits
    - Low priority job(s) resumed execution

Preemptive, Priority Job Scheduler

Status:

- Scheduler functionality complete; ready for operations
  - conversion to Trusted IRIX / MLS OS underway
- SGI retains rights to rpmd

For more info:

Preemptive, Priority Job Scheduling for Operational Weather Forecasting on the SGI Origin Platform

SC '01 Extended Technical Abstract

(send me email: scc@sgi.com)