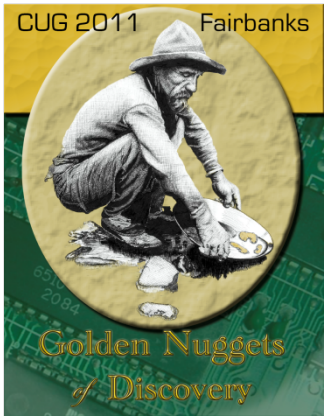
Platform Computing



Using Platform LSF with CLE

Mehdi Bozzo-Rey
HPC Solutions Architect
Platform computing

Overview

Platform Computing

- Platform Computing
- Platform LSF overview
- Integration with CLE: architecture overview
- Running LSF jobs on a CRAY system
- CCM and LSF: quick look at the POC
- Work in progress and future work

Platform Computing, Inc.

Platform Computing

Platform
Clusters, Grids, Clouds
Computing

The leader in cluster, grid and cloud management software:

- 18 years of profitable growth
- 2,000 of the world's most demanding client organizations
- 5,000,000 CPUs under management
- 500 professionals working across 13 global centers

Global Presence

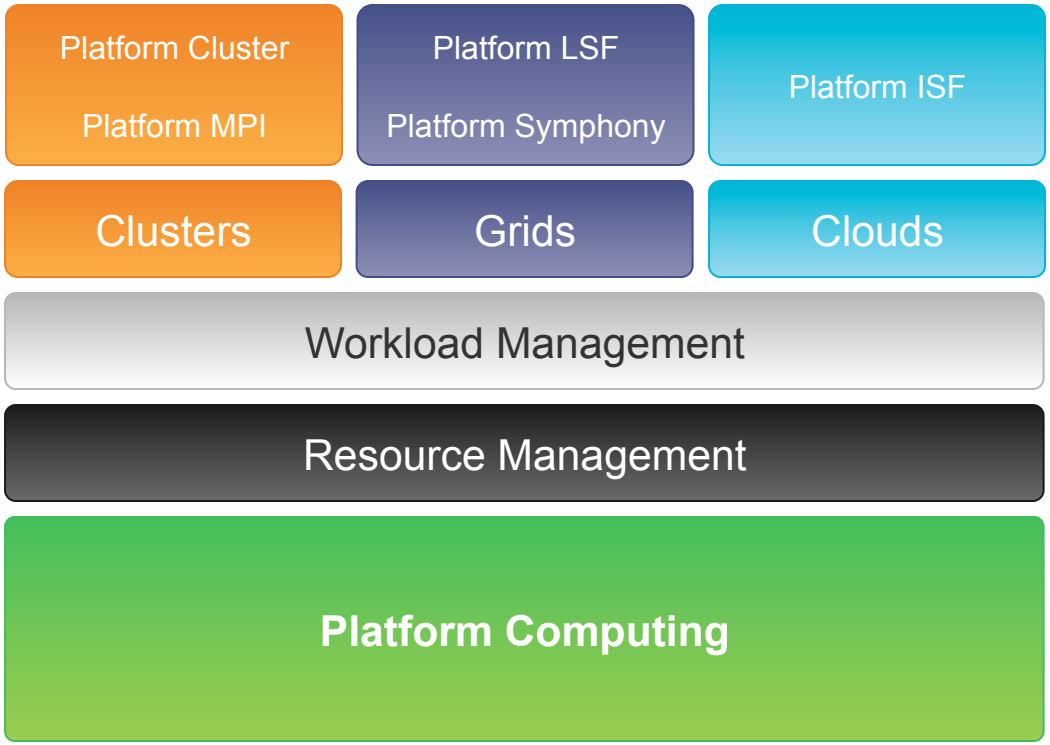
Platform Computing

North America
Toronto (HQ)
San Jose
Washington
Detroit
Los Angeles
Boston
New York

VARs
U.S.
Italy
Israel
Germany
Spain
Korea
Taiwan
Singapore
Japan

International
China
France
Germany
Japan
Korea
Singapore
UK

Product Leadership



Industry Leadership

Platform Computing

Electronics	Financial	Manufacturing	Oil & Gas	Govt & Edu	Life Sciences
• AMD	• BNP	• Airbus	• Agip	• CERN	• Abbott
• ARM	• Citi	• Audi	• Anadarko	• US DoD, DoE	• AstraZeneca
• Broadcom	• Commerzbank	• BAE Systems	• BP	• ENEA	• DuPont
• Cadence	• Fortis	• Boeing	• BHP	• Georgia Tech	• Eli Lilly
• Ericsson	• HSBC	• Bombardier	• British Gas	• Harvard Med	• J&J
• Infineon	• JP Morgan Chase	• John Deere	• China Petro	• Japan Atomic	• Merck
• MediaTek	• Intl Monetary Fund	• Ford	• Chevron	• MaxPlanck	• NIH
• NEC	• LBBW	• GM	• ConocoPhillips	• MIT	• Novartis
• NVidia	• Mass Mutual	• Goodrich	• EMGS	• Singapore U.	• Partners Health
• Qualcomm	• Mitsubishi UFJ	• Honda	• Gaz de France	• Stanford Med	• Sanger Institute
• Samsung	• Nomura	• Nissan	• Hess	• U. Tokyo	
• Sony	• Prudential	• Northrop	• Kuwait Oil	• Washington U.	
• ST Micro	• Sal. Oppenheim	• Pratt & Whitney	• PetroBras		
• Synopsys	• Société Générale	• Proctor & Gamble	• Petro Canada	• Beijing Cloud Center	
• TI	• UBS	• Toyota	• Petro China	• Shanghai Supercomputing	
	• Unicredit	• Volkswagen	• Shell	• Texas Advanced Computing	
			• Schlumberger		
			• StatoilHydro		
			• Total		
			• Woodside		

Other Industries

AT&T Bell Canada Telecom Italia Telefonica Walmart GE Walt Disney

Platform LSF

Platform Computing

- Platform LSF in numbers: scaling
  - 6000 nodes for EDA (Electronic Design Automation)
  - 12000 nodes for typical HPC workload
  - Under implementation: 12000 nodes (EDA) / 24000 (HPC)
- A complete ecosystem
  - Platform Application Center
  - Platform RTM
  - Platform Session Scheduler
  - Platform Multicluster
  - Platform MPI
  - …

Platform LSF runs everywhere

Platform Computing

- **Sender: LSF System <mbozzore@mehdi.boznet.org>**
  Subject: Job 1: <cat /proc/cpuinfo> Done

  Job <cat /proc/cpuinfo> was submitted from host <mehdi.boznet.org> by user <mbozzore>.
  Job was executed on host(s) <mehdi.boznet.org>, in queue <normal>, as user <mbozzore>.
  </home/mbozzore> was used as the home directory.
  </home/mbozzore> was used as the working directory.
  Started at Thu Feb 1 07:38:36 2007
  Results reported at Thu Feb 1 07:38:36 2007

  Your job looked like:

  --
  # LSBATCH: User input
  cat /proc/cpuinfo
  --

  Successfully completed.

  Resource usage summary:

  CPU time : 0.04 sec.
  Max Memory : 2 MB
  Max Swap : 8 MB

  Max Processes : 1
  Max Threads : 1

  The output (if any) follows:

  processor : 0
  cpu : Cell Broadband Engine, altivec supported
  clock : 3192.000000MHz
  revision : 5.1 (pvr 0070 0501)

  processor : 1
  cpu : Cell Broadband Engine, altivec supported
  clock : 3192.000000MHz
  revision : 5.1 (pvr 0070 0501)

  timebase : 79800000
  machine : PS3PF

[root@mehdi RPMS]# lsid
Platform LSF 7.0, Jan 16 2007
Copyright 1992-2006 Platform Computing Corporation

My cluster name is ps3
My master name is mehdi.boznet.org

[root@mehdi RPMS]# **lshosts**
HOST_NAME type model cpuf ncpus maxmem maxswp server RESOURCES
mehdi.bozne LINUXPP DEFAULT 1.0 2 196M 415M Yes (mg)

Platform LSF - Inside a LSF cluster

Platform Computing

- Definitions – LSF daemons
  - **mbatchd** : Master Batch Daemon
  - **mbschd** : Master Batch Scheduler Daemon
  - **lim** : Load Information Manager
  - **res** : Remote Execution Server
  - **pim** : Process Information Manager
  - **sbatchd** : Slave Batch Daemon
  - **elim**: external LIM
  - **Master lim**
  - **Rla**: platform topology adapter

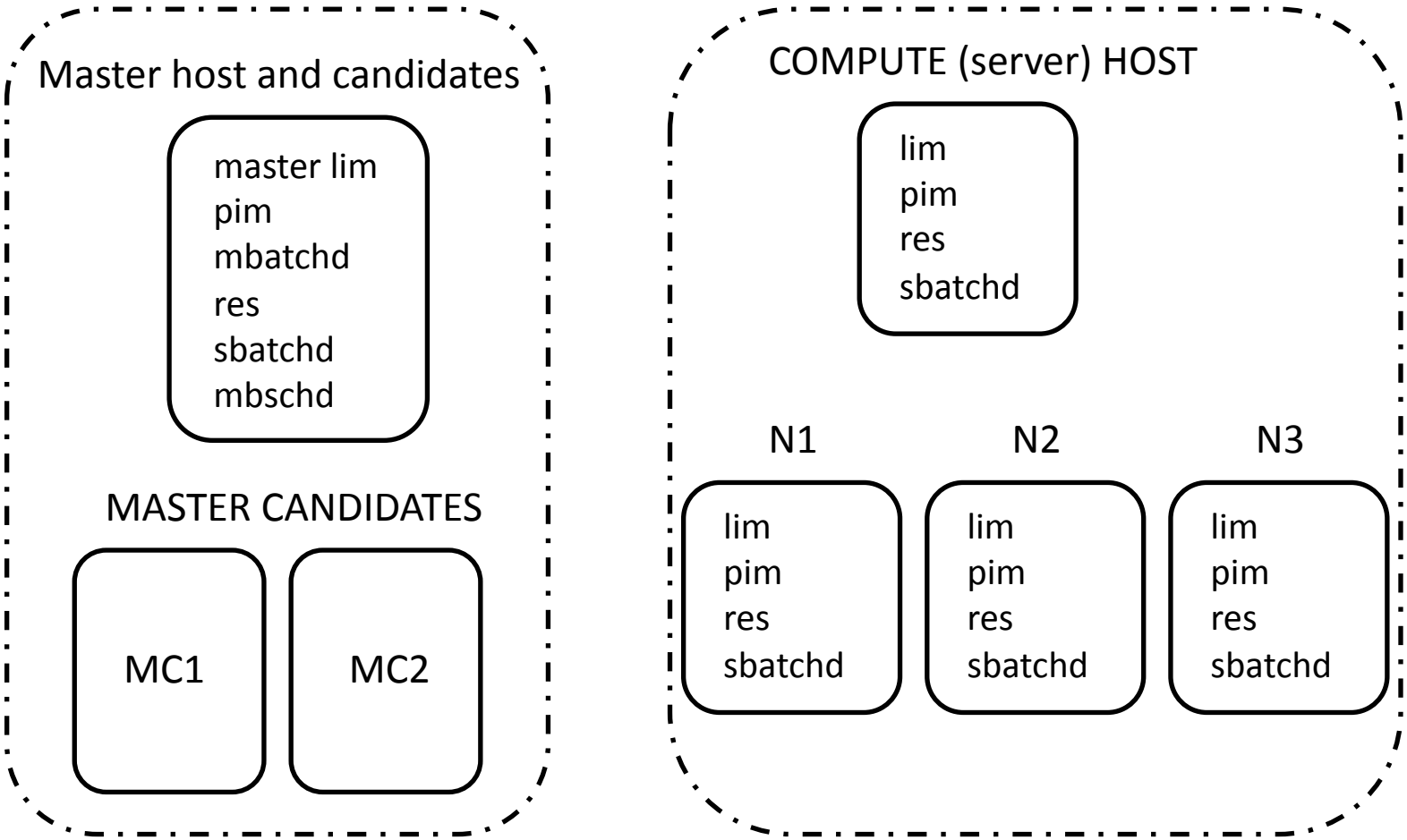
Platform LSF - Inside a LSF cluster

Platform Computing

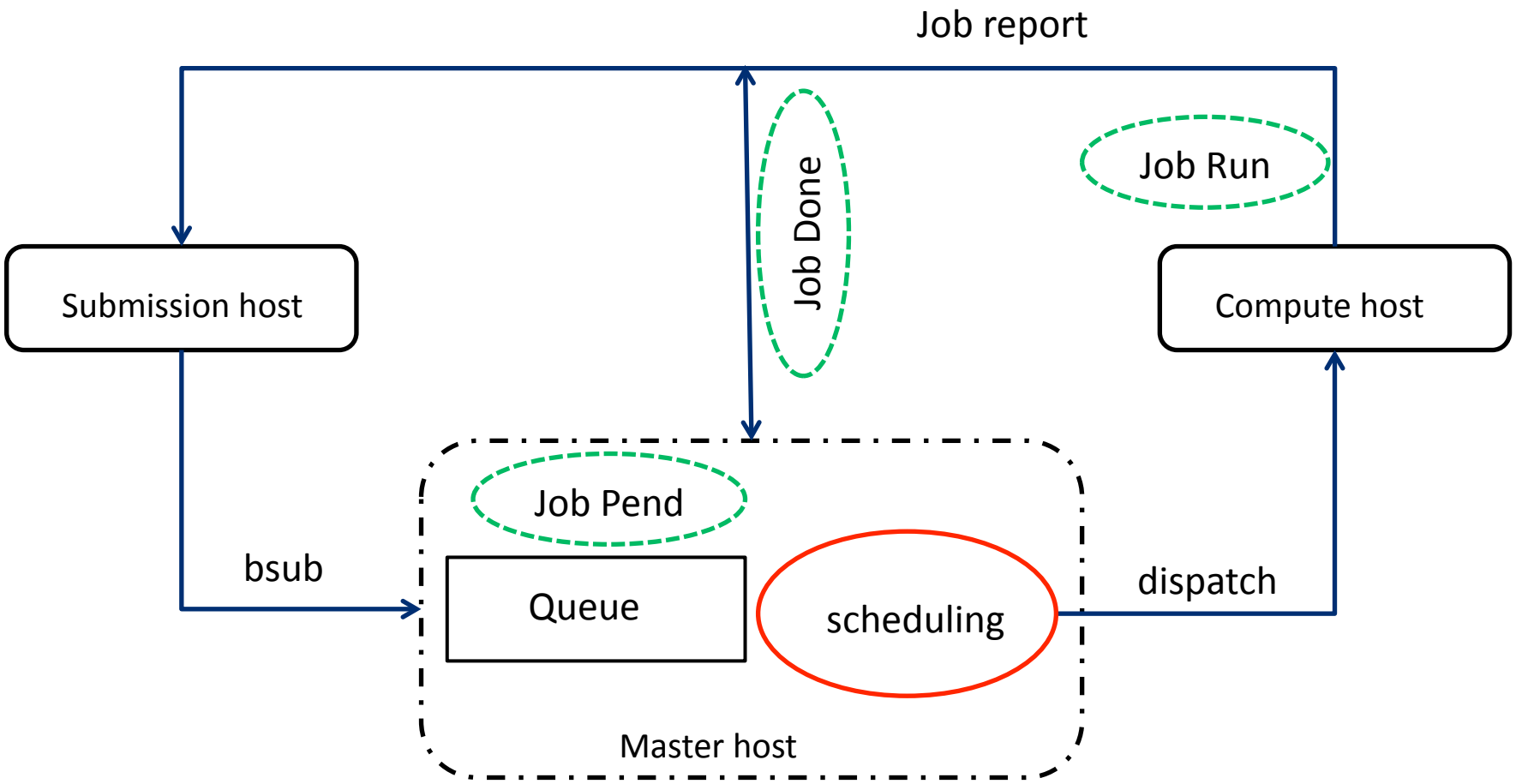
- Who does / handles what ?

LSF daemon	Role
mbatchd	Job requests and dispatch
mbschd	Job scheduling
sbatchd	Job execution
res	Job execution
lim	Host information
pim	Job process information
elim	Dynamic load indices

Platform LSF - Inside a LSF cluster

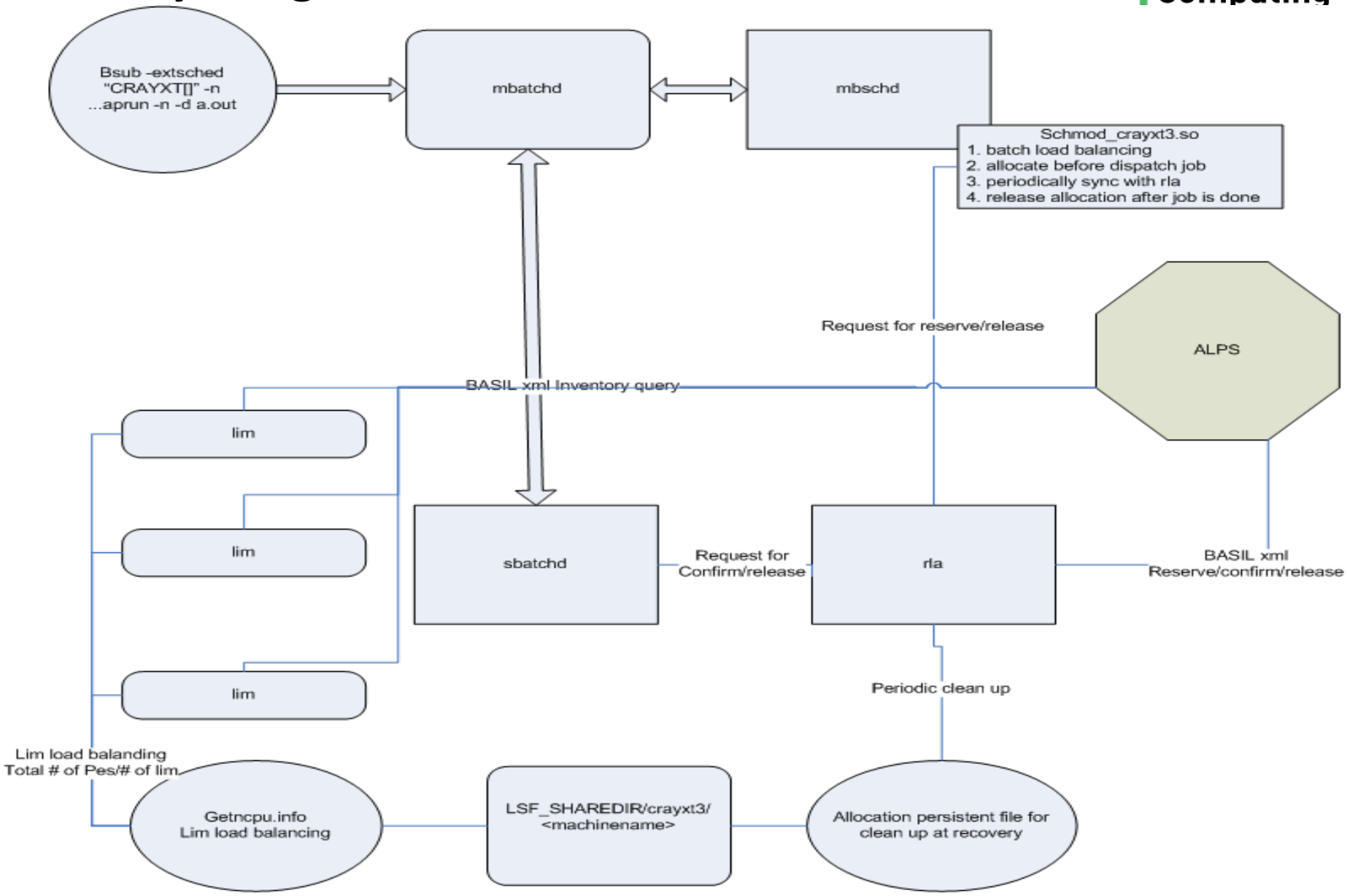


Platform LSF - job lifecycle



Cray integration architecture

Platform Computing



Cray integration: Features & limitations

Platform Computing

- Load balancing when running multiple lims on multiple login nodes
- Batch load balancing
- Large memory node support (*)
- Multicore (multiple PEs) support (*)

- Preemption
- Reservation
- Advance Reservation
- Backfill

- CCM (**)
- CR (***)

Install and key configuration parameters

Platform Computing

- Compute resources need to be in batch mode:
  - xtprocadmin -k m batch
- Standard LSF install through xtopview
  - Install.config file:
    - LSF_MASTER_LIST, LSF_ADD_SERVERS =<login or service nodes only>
    - EGO_DAEMON_CONTROL="N“ ; ENABLE_HPC_CONFIG="Y“
  - /ufs must be shared among all login/service nodes
  - lsf.cluster.<cluster_name> must contain
    - All login nodes
    - Boolean resource crayxt3 assigned to the nodes
  - lsf.conf:
    - LSB_SHAREDIR=/ufs/lsfhpc/work
    - LSF_LOGDIR=/ufs/lsfhpc/log
    - LSF_CRAY_XT_PES_PER_NODE=n

Running LSF jobs

Platform Computing

- Standard job:
  - bsub -n x -ext "CRAYXT[...]" aprun -n y /full_path/myjob
- If the large memory feature is enabled:
  - bsub -n 2 -q lowregular -ext "CRAXT[]" aprun -n 2 myjob
    - Will be submitted to the regular memory queue
  - bsub -n 2 -q highlargemem -ext "CRAXT[LARGEMEM]" aprun -n 2 myjob
    - Will be submitted to the large memory queue
- With LSF_CRAY_XT_PES_PER_NODE=2
  - bsub -n 2 -ext"CRAYXT[]" aprun -n 4 -d 1 -N 2 myjob
    - LSF creates a reservation that includes 2 nodes,the job spawns 4 tasks in total, with 2 tasks running on each node.

CCM and LSF : quick look at the POC

Platform Computing

- Integration done through LSF 8 pre and post exec scripts, at queue level
- Key parameters / files:
  - /etc/lsf.sudoers
    - LSB_PRE_POST_EXEC_USER=root
  - lsb.queues
    - Location of the scripts
    - LOCAL_MAX_PREEXEC_RETRY=1
  - lsb.params
    - JOB_INCLUDE_POSTPROC=Y

CCM and LSF: CCM in action

Platform Computing

crayadm@nid00060:~/mehdi/bin> **bsub -n 6 -ext"CRAYXT[]" -q test -I aprun -b -a xt -cc none -n 1 bash**

Job <1900> is submitted to queue <test>.

<<Waiting for dispatch ...>>

<<Starting on nid00060>>

cat runlin.sh

export PATH=$PATH:/home/crayadm/mehdi/openmpi-1.4.3/bin/

export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/home/crayadm/mehdi/openmpi-1.4.3/lib

export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/gcc/4.5.2/snos/lib64:/opt/gcc/gmp/4.3.2/lib:/opt/gcc/mpfr/2.4.2/lib:/opt/gcc/mpc/0.8.1/lib:/opt/acml/4.4.0/gfortran64

mpirun -np 12 --mca btl_tcp_if_exclude lo,rsip -hostfile /home/crayadm/.crayccm/ccm_nodelist.$LSB_JOBID --prefix /home/crayadm/mehdi/openmpi-1.4.3 /home/crayadm/mehdi/bin/xhpl

for i in `cat /home/crayadm/.crayccm/ccm_nodelist.$LSB_JOBID`; do ssh $i hostname; done

nid00038

nid00039

nid00040

nid00041

nid00054

nid00055

./runlin.sh

==========

HPLinpack 1.0a -- High-Performance Linpack benchmark -- January 20, 2004

Written by A. Petitet and R. Clint Whaley, Innovative Computing Labs., UTK

==========

Work in progress and future work

Platform Computing

- CR
- CLE 4 certification
- New / enhanced integration

Special thanks

Platform Computing

- Jason Coverston (CRAY)
- Tara Fly (CRAY)
- Blaine Ebeling (CRAY)

Questions

Platform Computing

- Questions ? → mbozzore@platform.com