Environment

- Heterogeneous Cluster of Workstations
  - POE and PVM
- 9076 System:
  - POE, PVM and PVMe
  - optionally configure LoadLeveler to use the Resource Manager
- Both:
  - POE and PVM
  - optionally configure LoadLeveler to use the Resource Manager
- NOTE using the Resource Manager will:
  - coordinate machine usage between interactive and batch jobs on the 9076
  - make different 9076 adapters available to LoadLeveler

Restrictions and Limitations

- Suspend/Flush/Checkpointing Unsupported
- PVMe/PVM Dynamic Add/Delete Nodes Unsupported
- LoadLeveler 1.2.0 Requires PSSP 1.2 for Parallel Jobs
  - PVMe 1.3.1
  - POE 1.2.1
  - PVM 3.3
- PVM ONLY over addr associated with name in machine stanza of the Admin File (will pass RM adapter addr's in the near future)
- PVMe ONLY US over switch (PTF2 IP over the switch may be supported in the July release).

Admin Setup

- Resource Manager Config File
  - Pool Members:
    - Batch: LoadLeveler only
    - General: LoadLeveler and Interactive
    - Interactive: Interactive only
  - Example:

```
POOL_ID = 0
ATTRIBUTES = Test_pool
MEMBERS_INTERACTIVE = k1bn01;
MEMBERS_BATCH       = k1bn02;
MEMBERS_GENERAL     = k1bn03;
```

- LoadLeveler Admin File

  - Machine Stanza Keywords:
    - rm_host: 9076 machine which will be communicating with the RM
    - pvm_root: location of the directory in which PVM is installed
  - Example:

```
k1bn02: type = machine  rm_host = true pvm_root = /u/admin/pvm3
k1bn03: type = machine  rm_host = true pvm_root = /u/admin/pvm3
```

- LoadLeveler Config File
  - Keywords:
    - Negotiator_Parallel_Defer: time job remains in the deferred queue
    - Negotiator_Parallel_Hold: time a job is given to accumulate processors
    - Class: limit PVMe and PVM to 1 class per node < policy >
    - Maxstarters: limit PVMe and PVM to 1 class per node < enforced >

Common Setup Problems

- Jobs that need RM adapters remain Idle
  - LoadL machines have rm_host=false
  - LoadL machines have rm_host=true:
    - LoadL has not been globally reconfiged since rm_host=true changed
    - RM is not running (use jm_status -P to check)
    - Machines are not specified as a general or batch pool member in the jmd_config (use jm_status -P to check)
- Jobs remain Pending
  - Nameserver is not consistent (compare results of host <machine_name> and host <IP_address>

Scheduling

- Negotiator selects parallel machines through temporary reservation:
  - Scans a list for machines with resources matching the job requirements
  - If sufficient machines are not available, reserves as many as it can
  - Reserves new machines meeting job requirements as they become available
  - Dispatches the job when required number of machines has been meet
  - Limits the time machines are held in reserve for "Negotiator_Parallel_Hold"
    - All reserved machines are released
    - Parallel job is placed on a deferred queue
    - Parallel job reenters the LoadL queue after "Negotiator_Parallel_Defer"
    - Parallel job maintains original system priority

RM/LoadL Interactions

- LoadLeveler machines configured to use the RM:
  - Startd will poll the RM for 9076 system specific info
    - Adapter
    - State (ie: dedicated/shared)
    - Pool
  - Startd will update the central manager with new info
  - If machine has been dedicated for interactive or batch use:
    - LoadLeveler will take the machine off line
    - Marks it's state as reserved
  - User will now be able to specify "Adapter" and "Pool" through the requirements keyword in the job command file
  - Starter will allocate machines selected by the Negotiator through the RM for both serial and parallel jobs
    - Machines are allocated as shared so that serial, parallel and interactively jobs may run simultaneously
    - Job will return to the queue if the allocation fails

Display RM Machine Attrs

- llstatus -l (GUI equivalent: machine details):

```
name: "k1bn10.hpssl.kgn.ibm.com"
machine_context:
Running             = 0
ScheddAvail         = 1
StartdAvail         = 1
State               = Reserved                              <<<<<<<<<<
OpSys               = AIX32
Arch                = R6000
Machine             = k1bn10.hpssl.kgn.ibm.com
START               = T
SUSPEND             = F
CONTINUE            = F
VACATE              = F
KILL                = F
SYSPRIO             = (0 -  QDate)
MACHPRIO            = (0 -  LoadAvg)
ScheddRunning       = 0
Pending             = 0
Starting            = 0
Idle                = 0
Unexpanded          = 0
Held                = 0
Removed             = 0
RemovePending       = 0
Completed           = 0
DependantNotRun     = 0
TotalJobs           = 0
ScheddState         = 0
EnteredCurrentState = Mon Mar  6 10:30:05 1995
VirtualMemory       = 36100
Disk                = 23428
KeyboardIdle        = 7034
LoadAvg             = 0.021313
Class               = { "PVM3" "PVME" }
Pool                = 0                                     <<<<<<<<<<
Adapter             = { "ethernet" "hps_user" "hps_ip" } <<<<<<<<<<
Initiators          = { "PVM3" "PVME" }
Feature             = { }
Memory              = 64
Max_Starters        = 2
ConfigTimeStamp     = Mon Mar  6 10:29:59 1995
Cpus                = 1
Speed               = 1.000000
MasterMachPriority  = 0.000000
Subnet              = 129.40.131
CustomMetric        = 1
time_stamp: Mon Mar  6 10:34:59 1995
```

LoadL Log RM Error Msgs

- Master Startd Error Msgs:
  - JM_ERROR(-2) jm_nodequery: not able to process request at this time. Wait max 3 mins
  - JM_ERROR(-3) jm_nodequery: current socket connection is invalid
  - JM_ERROR(-6) jm_nodequery: error getting data in database

  - JM_ERROR(-7) jm_nodequery: hostname found via gethostname was not a known hostname
  - JM_WARNING jm_nodequery: Node is not part of any pool or is in a serial pool
  - JM_WARNING jm_nodequery: Node is not part of any subpool or is in an interactive subpool
- Master Starter Error Msgs:
  - JM_ERROR jm_connect_ub returned <number>
    - See RM functional spec/RM service transfer class for return codes
  - JM_ERROR jm_allocate returned <number>
    - See RM functional spec/RM service transfer class for return codes
  - JM_ERROR jm_allocate node n returned <number>
    - See RM functional spec/RM service transfer class for return codes

Control Flow Differences

- The POE PM, PVMe Daemon and the PVM Daemon are all Master processes started by the Master Starter.
- The POE PM and the PVMe Daemon interface with LoadL through LoadL API's:
  - ll_get_hostlist: to get the list of parallel nodes
  - ll_start_cmd: to start the remote tasks under the Starter's control. Only one task per node is allowed. Slave Starters will exit after task completes.
  - Reference control flow diagram p.11
- The PVM Daemon interfaces with LoadLeveler through PVM functions:
  - add_host: Adds a host to the PVM configuration
  - reg_hoster: Registers LoadL as a PVM slave pvmd starter so that LoadL may start the remote slave pvmd tasks under the Starter's control
  - Reference control flow diagram p.14

POE/PVMe API Error Msgs

- User's Job Error File:
  - ll_get_hostlist
    - POE
      - PEPM407 0031-407: LoadLeveler unable to allocate nodes to poe for batch job
    - PVMe
      - PVMD: LL reported error <number> during node allocation
    - Return numbers
      - -2 Cannot get LoadLeveler jobid from environment

        - -5 Cannot make socket
        - -6 Cannot connect to host
        - -8 Cannot get hostlist
    - ll_start_cmd
        - POE
            - PEPM409 0031-409: Unable to start Partition Manager daemon on node n rc = <number>
        - PVMe
            - PVMD: LL reported error <number> during start task
        - Return numbers
            - -1 Remote host cannot fork new process
            - -2 Cannot get LoadLeveler jobid from environment
            - -3 Cannot get my hostname
            - -4 Nameserver cannot resolve host
            - -5 Cannot make socket
            - -6 Cannot connect to host
            - -7 Cannot send PASS_OPEN_SOCKET command to remote host

Submit

- Job Command File
    - Keywords:
        - min_processors/max_processors: number of parallel processors job requires
        - parallel_path: path to use when starting a PVM 3.3 slave processes
        - job_type: type of job step to process
            - pvm3: for PVM 3.3 jobs
            - parallel: for POE and PVMe jobs
        - requirements(Pool/Adapter): adapter and parallel pool on the 9076 system to use
            - Adapter: hps_ip/hps_user/ethernet/fddi/tokenring/fcs

Submit POE

- POE PM MUST be specified as the executable or invoked from the JCF
- Parallel job MUST be passed as an argument (full path)
- When running in user space mode, MUST use POE flag -euilib us (ip default)
- MUST have a job_type of parallel (uses API's)

Examples:

```
# @ executable   = /usr/bin/poe
# @ arguments    = /home/userid/my_parallel_job -euilib us
# @ job_type     = parallel
# @ output = my_parallel_job.$(cluster).$(process).out
# @ error  = my_parallel_job.$(cluster).$(process).err
# @ min_processors = 8
# @ max_processors = 10
# @ requirements = (Adapter == "hps_user")
# @ class        = POE
# @ queue

# @ job_type     = parallel
# @ output = my_parallel_job.$(cluster).$(process).out
# @ error  = my_parallel_job.$(cluster).$(process).err
# @ min_processors = 3
# @ max_processors = 3
# @ requirements = (Adapter == "hps_ip" && Pool == 4)
# @ class        = POE
# @ queue
/usr/bin/poe /home/userid/my_parallel_job -infolevel 2
```

Submit PVMe

- PVMe daemon MUST be invoked from the JCF
- Parallel job MUST be passed as an -exec argument (full path)
- User space Adapter must be specified
- MUST have a job_type of parallel (uses API's)

Example:

```
# @ job_type     = parallel
# @ output = my_parallel_job.$(cluster).$(process).out
# @ error  = my_parallel_job.$(cluster).$(process).err
# @ min_processors = 2
# @ max_processors = 2
# @ requirements = (Adapter == "hps_user")
# @ class        = PVMe
# @ queue
/usr/lpp/pvm3/pvmd3e <daemon opts> -exec my_parallel_job <prog args>
```

Submit PVM3

- The parallel executable MUST be specified as the executable
- Machine with specified adapter is selected, but adapter is not used (addr not passed on from RM)
- MUST have a job_type of pvm3

Example:

```
# @ executable    = my_parallel_job
# @ job_type      = pvm3
# @ output = my_parallel_job.$(cluster).$(process).out
# @ error  = my_parallel_job.$(cluster).$(process).err
# @ min_processors = 8
# @ max_processors = 10
# @ parallel_path = /home/userid/cmds/pvm/$PVM_ARCH
# @ class         = PVM3
# @ requirements  = (Pool == 4)
# @ queue
```

Display Parallel Job Attrs

- llq -l (GUI equivalent: job details):

```
=============== Job Step k1bn09.hpssl.kgn.ibm.com.1.0 ===============
        Job Step Id: k1bn09.hpssl.kgn.ibm.com 1.0
           Job Name: k1bn09.hpssl.kgn.ibm.com.1
          Step Name: 0
  Structure Version: 8
              Owner: lluser
         Queue Date: Mon Mar  6 11:11:22 1995
             Status: Running
    Completion Date: (not completed)
    Completion Code: (not completed)
      User Priority: 50
       user_sysprio: 0
      class_sysprio: 0
      group_sysprio: 0
    System Priority: -2481
          q_sysprio: -2481
      Notifications: Always
        User Number: 1
 Virtual Image Size: 2 kilobytes
         Checkpoint: no
     Hold Job Until: (no hold)
                Cmd: test_pvme.cmd
               Args:
                 In: /dev/null
                Out: my_parallel_job.1.0.out
                Err: my_parallel_job.1.0.err
Initial Working Directory: /u/lluser/test_parallel
         Dependency:
       Requirements: (Arch == "R6000") && (Adapter == "hps_user") <<<<<<<<<<<
        Preferences:
           Job Type: General Parallel                              <<<<<<<<<<<
     Min Processors: 2                                             <<<<<<<<<<<
     Max Processors: 2                                             <<<<<<<<<<<
Allocated host count: 2                                            <<<<<<<<<<<
Allocated host names: k1bn09.hpssl.kgn.ibm.com                     <<<<<<<<<<<
                      k1bn12.hpssl.kgn.ibm.com
    Submitting host: k1bn09.hpssl.kgn.ibm.com
        Notify User: lluser@k1bn09.hpssl.kgn.ibm.com
              Shell: /bin/ksh
           LL_group: No_Group
              Class: PVME
            Account: NONE
         Unix_group: staff
```

Submit Problem Summary

- POE:
  - Parallel job instead of POE PM specified as the executable
  - Parallel job is not passed to POE as an argument
  - Parallel job does not exist or does not have a full path name
  - User space job does not specify flag -euilib us
  - User specifies the wrong or no adapter when needed
  - User specifies POE job twice from the JCF

  - User compiled with unsupported level of POE
  - User specifies a job_type other then parallel
- PVMe:
  - PVMe daemon invoked from the executable keyword
  - Parallel job is not passed to PVMe as an -exec argument
  - Parallel job does not exist or does not have a full path name
  - User did not specify user space adapter
  - User specifies adapter hps_ip
  - User specifies PVMe job twice from the JCF
  - User compiled with unsupported level of PVMe
  - User specifies a job_type other then parallel
- PVM:
  - PVM daemon versus parallel job specified as the executable
  - User specifies RM adapter
  - User compiled with unsupported level of PVM
  - User specifies a job_type other then pvm3