

## PBS Professional® utilizes Intel® Cluster Checker to validate the health of an HPC Cluster

Scott Suchyta<sup>1</sup>, Andreas Hirstius<sup>2</sup>

<sup>1</sup> Altair Engineering Inc., 1820 Big Beaver Road, Troy, MI 48083, USA

<sup>2</sup> Intel Corporation, Dornacher Strasse 1, 85622 Feldkirchen/Muenchen, Deutschland

### Abstract

PBS Professional® is responsible for scheduling, executing, and tracking the user's jobs on the cluster. Being aware of changes in the environment, which will negatively impact the user's job, is very important. Currently, PBS Professional relies on a cluster provisioning system or a system administrator to provide accurate information about the status of the cluster. This information is not always correct. On clusters that are Intel® Cluster Ready (ICR) certified, PBS Professional can utilize Intel® Cluster Checker for gathering real-time information about the cluster at the very moment a job has been scheduled. Using this immediate feedback, PBS Professional can ensure that a job is only being executed on a cluster that has passed a check by Intel Cluster Checker.

**Keywords:** Altair, HPC, High-Performance Computing, Job scheduling, Intel, Intel Cluster Ready, Intel Cluster Checker, PBS, PBS Professional, workload management

### Introduction to Intel Cluster Ready

Intel Cluster Ready establishes an industry-standard cluster architecture that allows an end-customer specify, select, deploy and manage a high-performance computing (HPC) cluster almost as easily as a desktop PC. This architecture helps to ensure interoperability across the complete HPC solution stack. It supports small to very large clusters and provides considerable flexibility for hardware and software innovation.

Developed in collaboration with hardware and software vendors, the Intel Cluster Ready program provides the ability to match ICR certified HPC cluster systems with ICR registered applications, enabling the solutions provider or end-user to create a

configuration that best fits the application workload – without worrying if all components will work together. Pre-configured cluster solutions based on today's leading performance platforms and components are delivered ready to run, enabling the end-user to unleash more parallel performance faster.

## Introduction to Intel Cluster Checker

Intel Cluster Checker is a valuable tool for OEMs and their customers both before and after a cluster is deployed. An OEM uses Intel Cluster Checker as part of the manufacturing process for Intel Cluster Ready certified clusters. Intel Cluster Checker verifies that an HPC cluster follows the ICR specifications and is working properly before it is delivered to the customer. It also assists the OEM in finding potential problems before the system is delivered to the customer. The benefit for the customer is a much faster time to productivity and the fact that the status of the delivered system can be verified easily.

After the cluster is deployed, Intel Cluster Checker continues to deliver value by helping the user maintain productivity over the life of the cluster. Intel Cluster Checker helps identify and resolve problems quickly, improving cluster reliability, simplifying management, and reducing the need for specialized support. Using Intel Cluster Checker in this mode can significantly increase the overall uptime and productivity of the cluster and reduce the total cost of ownership (TCO).

Intel Cluster Checker runs as a sequence of individual tests called test modules. Each test module conducts a specific test over a single cluster node or over the whole cluster, depending on its type. Intel Cluster Checker can be customized very easily. The user can specify which test modules should be run and can even write custom test modules.

## Introduction to PBS Professional

PBS Professional is the commercial version of the Portable Batch System (PBS), a flexible workload management solution, originally developed to manage aerospace

computing resources at NASA Ames, PBS Professional has become the leader in supercomputing workload management and the de facto standard on Linux clusters.

Today, growing enterprises often support hundreds of users running thousands of jobs across many different types of machines across the world. In this distributed heterogeneous environment, it can be extremely difficult for administrators to collect detailed, accurate usage data, or to set system-wide resource priorities. As a result, many computing resources are under-utilized, while others are over-utilized. At the same time, users are confronted with an ever-expanding array of operating systems and platforms. Each year, scientists, engineers, designers and analysts must waste countless hours learning the nuances of different computing environments, rather than focusing on their primary goals. PBS Professional addresses these issues in computing-intensive industries such as science, engineering, finance and R&D. You can use the power of PBS Professional to better control your computing resources. This allows you to unlock the potential in the valuable assets you already have, while at the same time reducing the load on system administrators and operators, freeing them to focus on other activities. PBS Professional can help you effectively manage growth by tracking real usage levels across your systems and allowing you to target future purchases to your needs.

PBS Professional consists of two major component types: commands and daemons/services. A brief description of each is given here to help you understand how the pieces fit together.

**Commands** – PBS Professional supplies both a graphical interface and a set of POSIX 1003.2d-conforming command-line programs. These are used to submit, monitor, modify and delete jobs. These client commands can be installed on any system type supported by PBS and do not require local presence of any of the other components of PBS.

There are three command classifications: commands available to any authorized user, commands requiring special PBS privilege, and administrator commands. Administrator commands require root access or the equivalent to use.

**Server** – The server daemon/service, `pbs_server`, is the central focus of PBS Professional. All commands and other daemons/services communicate with the server via an Internet Protocol (IP) network. The server's main function is to provide the basic batch services such as receiving/creating a batch job, modifying the job, and passing the job to the execution node. One server manages the machines and jobs in a PBS complex; a secondary server may be configured to handle failover.

**Execution Node Manager (MOM)** – The MOM is the daemon/service, which actually places the job into execution. The `pbs_mom` daemon is informally called MOM, as it is the mother of all processes for jobs. (MOM is a reverse-engineered acronym that stands for Machine Oriented Mini-server). MOM places a job into execution when it receives a copy of the job from the Server. MOM creates a new session for each job and gathers information about the resource usage of that job. MOM also has the responsibility for communicating with all MOMs assigned to the job and returning the job's output to the user when directed to do so by the Server. One MOM runs on each execution host.

**Scheduler** – The job scheduler daemon/service, `pbs_sched`, implements the site's scheduling policy, controlling when each job is run and on which resources. The Scheduler may communicate with the various MOMs to query the state of system resources and with the Server for availability of jobs to execute.

## Intel Cluster Checker and Test Modules

adds Beyond testing for compliance with the ICR specification, Intel Cluster Checker is a very versatile tool for system maintenance and it is highly customizable. The test modules that should be run can be specified in an XML file or on the command line.

Creating custom XML files that are being used to regularly check the health of an HPC cluster is done very easily.

Even though test modules can be run on their own, certain test modules depend on the successful running of other test modules. For example, test modules that test the cluster interconnect using MPI depend on the test module that checks basic network connectivity. Intel Cluster Checker can also be configured to not check dependencies. If

test modules that are provided with Intel Cluster Checker are not sufficient for the needs of an OEM or end-user, an API is provided for writing custom test modules.

These powerful capabilities of Intel Cluster Checker provide the end-user the peace of mind that the cluster is fully operational. In case of problems, Intel Cluster Checker can be used to pinpoint the issue quickly and thus solving the problem faster.

The Intel Cluster Checker configuration file (config.xml)[3] used in the demonstration is below:

```
> cat /opt/intel/clck/1.7/etc/config.xml
<!-- Intel Cluster Checker Config: /opt/intel/clck/1.7/etc/config.xml -->
<cluster>
  <!-- global configuration for setting paths and fabric/devices -->
  <global_configuration>
    <cc-path>/opt/intel/cce/12.0.191</cc-path>
    <fc-path>/opt/intel/fce/12.0.191</fc-path>
    <mpi-path>/opt/intel/impi/4.0.2.003</mpi-path>
    <network>
      <fabric name="ETH" enabled="on">
        <device options="-env I_MPI_DEBUG 5">ssm</device>
      </fabric>
    </network>
  </global_configuration>

  <!-- nodefile is overridden via commandline via prologue w/
  PBS_NODEFILE. This ensures the tests are executed on the nodes
  identified by PBS Professional -->
  <nodefile>/home/icr/nodesfile</nodefile>

  <exclude_module> uid_sync </exclude_module>
  <exclude_module> stray_uids </exclude_module>
  <exclude_module> file_tree </exclude_module>

  <include_module> X11_clients </include_module>

  <test>
    <process_check>
      <exclude>postgres</exclude>
      <exclude>sshd</exclude>
      <exclude>cmd</exclude>
      <exclude>pbs_mom</exclude>
    </process_check>
    <clean_ipc>
      <sem_arrays>1</sem_arrays>
    </clean_ipc>
    <disk_bandwidth>
      <options> -i0 -i1 -r 16m -s 64m -+n -+r -I</options>
    </disk_bandwidth>
```

```

<intel_mpi_rt_internode>
  <process-number>1</process-number>
  <device>sock</device>
</intel_mpi_rt_internode>
<imb_collective_intel_mpi>
  <fabric>
    <device>sock</device>
  </fabric>
</imb_collective_intel_mpi>
<imb_message_integrity_intel_mpi>
  <fabric>
    <device>sock</device>
  </fabric>
</imb_message_integrity_intel_mpi>
<imb_pingpong_intel_mpi>
  <fabric>
    <bandwidth>110</bandwidth>
    <bandwidth-deviation>3</bandwidth-deviation>
    <device>sock</device>
    <latency>35</latency>
    <latency-deviation>3</latency-deviation>
  </fabric>
  <fabric>
    <bandwidth>620</bandwidth>
    <bandwidth-deviation>3</bandwidth-deviation>
    <latency>10</latency>
    <latency-deviation>3</latency-deviation>
  </fabric>
  <iterations>100,20</iterations>
</imb_pingpong_intel_mpi>
</test>
</cluster>

```

## PBS Professional Prologue

PBS Professional provides a facility for the administrator to run a site-supplied script or program before and/or after each job run. This allows initialization or cleanup of resources, such as temporary directories or scratch files. The script or program that runs before the job is the prologue[4]; the one that runs after the job is the epilogue. When multiple nodes are allocated to a job, the script is executed on the primary execution node (aka head node).

The primary purpose of the prologue is to provide a site with some means of performing checks prior to starting a job. Thus, the integration with Intel Cluster Checker utilizes this capability.

Depending on the type and/or number of tests the Intel Cluster Checker is configured to execute, it might be necessary to increase the prologue alarm (\$prologalarm) value from 30 seconds (default) to 60 second or more. Increasing the value of the prologue alarm is done by adding the \$prologalarm attribute to the \$PBS\_HOME/mom\_priv/config, found on each execution node (MOM), and then kill -HUP the pbs\_mom pid.

```
$prologalarm <value>
```

where <value> is in seconds

```
kill -HUP `cat $PBS_HOME/mom_priv/mom.lock`
```

The prologue (and epilogue) are located in the \$PBS\_HOME/mom\_priv directory on each execution node (MOM). The prologue must be owned by root and is executed as root.

```
> ls -l $PBS_HOME/mom_priv/prologue
-rwx----- 1 root root 2150 Aug  2 10:37 mom_priv/prologue
```

The prologue provides the ability to requeue the job if the script exits with a negative value.

The prologue used in the demonstration has been adapted to support a module framework, which allows a site to incorporate additional scripts and languages instead of porting this plug-in to their environment. The prologue will check the existence of the Intel Cluster Checker (clck.py) plug-in and execute it. Otherwise the prologue will exit the prologue and the job will begin execution.

```
> cat $PBS_HOME/mom_priv/prologue
#!/bin/sh
```

```
#####
```

```
# Source /etc/pbs.conf

#####

# Source in the /etc/pbs.conf; it is 'good practice' to
# incorporate this so that you are referring to the correct
# PBS Professional commands. Reduces some hardcoding, too.
if [ -f /etc/pbs.conf ] ; then
    . /etc/pbs.conf
else
    echo "Could not find /etc/pbs.conf on host: `hostname`"
fi

#####

# Define some GLOBAL VARIABLES

#####

# Prologue arguments
PBS_JOBID=${1}          # $1 = job id
USER=${2}               # $2 = uid of execution user
GROUP=${3}              # $3 = gid of execution user

# Intel Cluster Checker hook
INTEL_CLCK_HOOK=${PBS_HOME}/mom_priv/clck.py

if [ -f ${INTEL_CLCK_HOOK} ] ; then
    echo "Executing Intel Cluster Checker..."
    ${INTEL_CLCK_HOOK} $PBS_JOBID $USER $GROUP
    status=$?

# Check the error
if [ $status -eq 0 ] ; then
    exit 0
else
    echo "Intel Cluster Checker detected FAILURE(S), your job is
    getting requeued on different machines."
```



```
    echo "Provide your OU and ER files to your administrator for
further investigation."
    exit $status
fi
else
    echo "Intel Cluster Checker was NOT present... skipping setup
and tests."
fi
```

The Intel Cluster Checker plug-in (clck.py) will execute the Intel Cluster Checker tool and parse the output for failures. If failures are detected, the plug-in will begin cataloging the test modules and effected nodes. The effected nodes will be marked offline and the node's comment attribute will be updated with a short description of the failure.

## Demonstration

A simple demonstration of a user submitting a small cluster with 4 compute nodes illustrates the integration of the Intel Cluster Checker with PBS Professional prologue capabilities.

The PBS Professional `qsub[5]` command is used for submitting the job.

```
> qsub -l select=4 qsub.script
277.master
```

After the job is successfully submitted, PBS Professional will evaluate the request, consider it for scheduling, apply the scheduling policies, and then dispatch it to the execution nodes. The `qstat[5]` command is used to monitor the status of the job.

```
> qstat
Job id      Name           User      Time Use S Queue
277.master  qsub.script    scott           0 R workq
```

The MOM daemon will launch the prologue, which contains the code to execute the Intel Cluster Checker tool and evaluate the success and failures. If failures are detected, the prologue will mark the failed node offline, update the node comment with short summary of what failed, and will requeue the job.

The `pbsnodes[5]` command is used to monitor the status of the nodes. Using the `pbsnodes` command and a few `grep` options illustrates that the nodes were marked offline because job 277 failed the Intel Cluster Checker test module “clean\_ipc”.

```
> pbsnodes -av | grep -e '^[[:alnum:]]' -e 'state' -e 'comment'

node00
    state = offline
    comment = Job 277 FAILED Intel ClCk clean_ipc
node01
    state = offline
    comment = Job 277 FAILED Intel ClCk clean_ipc
node02
    state = offline
    comment = Job 277 FAILED Intel ClCk clean_ipc
node03
    state = offline
    comment = Job 277 FAILED Intel ClCk clean_ipc
```

The job is requeued, to give the job another opportunity to be scheduled on different nodes. The user will also receive the ER and OU files from the failed jobs which will contain the report from the Intel Cluster Checker, which can be provided to the administrator for further investigation.

Check has FAILED.

Intel Cluster Checker detected FAILURE(S), your job is getting requeued on different machines. Provide your OU and ER files to your administrator for further investigation.

Typically, administrators are monitoring the node status with a cluster management tool, which can be configured to trigger an event that emails the administrator when a node is offline or down.

## Conclusions

Providing a feedback mechanism, such as Intel Cluster Checker, into the scheduling decisions of PBS Professional will help ensure jobs are being dispatched to healthy execution nodes. By utilizing the real-time information provided by Intel Cluster Checker, PBS Professional can:

- Ensure no computing cycles are wasted on execution on nodes that may fail
- Maximize the overall operational efficiency of the HPC system

## References

- [1] "Intel Cluster Checker 1.8 User's Guide", <http://software.intel.com/file/38436>
- [2] "Intel Cluster Checker 1.8 Developer's Guide", <http://software.intel.com/file/38438>
- [3] "Intel Cluster Checker 1.8 Test Modules Reference Guide",  
<http://software.intel.com/file/38437>
- [4] "PBS Professional 11.1 Administrator's Guide",  
<http://www.pbsworks.com/documentation/support/PBSProAdminGuide11.1.pdf>
- [5] "PBS Professional 11.1 User's Guide",  
<http://www.pbsworks.com/documentation/support/PBSProUserGuide11.1.pdf>

[www.pbsworks.com](http://www.pbsworks.com) • [intel.com/go/cluster](http://intel.com/go/cluster)

Copyright © 2011 Altair Engineering, Inc. All rights reserved. PBS™, PBS Works™, PBS Professional®, PBS Analytics™, PBS Catalyst™, e-BioChem™, e-Compute™, and e-Render™ are trademarks of Altair Engineering, Inc. and are protected under U.S. and international law and treaties. All other marks are the property of their respective owners. This paper is for informational purposes only, and may contain errors; the content is provided as is, without express or implied warranties of any kind.