CS554 Project Ideas


Overview
The goal of job management system (JMS) is to efficiently manage the distributed computing power of workstations, servers, and supercomputers in order to maximize job throughput and system utilization. A JMS is in charge of all the activities from submitting a job to returning a job after being executed, such as resource allocation, job scheduling, and job launching.

Over the past two decades, there are a lot of JMSs developed for the High Performance Computing (HPC) environment, where jobs require large number of compute nodes and tasks with a job are tightly coupled. Representatives of these JMSs are SLURM developed in LLNL, Condor developed in UW-Madison, SGE developed in Sun Microsystems, PBS related projects (e.g. OpenPBS in NASA, TORQUE maintained by Adaptive Computing Enterprises, Inc. and the commercial PBS Pro offered by Altair Engineering), and Cobalt developed in ANL. All these JMSs have centralized architecture and have been deployed as resource managers on various clusters and supercomputers for years. This project is to benchmark and comparing all of these JMSs with detailed profiling about their performance and resource consumption under the same workloads. The aim is to understand strengths and weaknesses of the current HPC JMSs, which will guide us to develop the next generation distributed JMS towards exascale computing.

Relevant Systems and Reading Material
SLURM: http://www.schedmd.com/slurmdocs/overview.html
Condor: http://research.cs.wisc.edu/htcondor/
SGE: http://web.njit.edu/all_topics/HPC/sge.html
OpenPBS: http://www.nas.nasa.gov/Software/PBS/docs.html
TORQUE: http://www.adaptivecomputing.com/products/open-source/torque/
Cobalt: http://trac.mcs.anl.gov/projects/cobalt

Preferred/Required Skills
Required: Linux, scripting language
Preferred: HPC, C/C++, Java

Parameters
Different workloads, different scales

Metrics
Throughput, latency, efficiency, time and memory consumption

Project Mentor