

An Introduction to the Condor HTC Framework

S. Hosking
Parallel Programming, 159735, Massey University

May 2009

Contents

1	About Condor	3
2	High Throughput Computing	3
3	Condor Introduction	3
4	Universes	4
5	ClassAds	4
6	Usage	4
7	Related Technologies	7
7.1	Flocking	7
7.2	DAGMan	7
7.3	Condor-G	7
7.4	Condor Pool Tools	7
7.5	Condor File Transfer	7
8	References	8

1 About Condor

Starting in late 1980s and developed at the University of Wisconsin-Madison, Condor is a freely available project designed to encapsulate and run large collections of distributed computing resources with the ultimate aim of giving people and computer scientists more access to available computing power.

Condor specializes in high throughput computing (HTC), giving users the ability to run huge numbers of tasks over long periods of time, not necessarily running one job very very fast.

2 High Throughput Computing

As stated above one of the abilities of Condor is High Throughput computing. Many problems that scientists are trying to solve require weeks or even more of computing power to run, getting use of a extremely fast high performance computer (HPC) for only and small a amount of time is of no interest to them. They would rather have access to a large collection of slower computers for much longer periods, and thus increasing there overall throughput.

3 Condor Introduction

The main arm of Condors toolkit is to set up a management system that overseas all of it computer resources and assigns jobs that have been submitted by users to them. Some of the main services that Condor provides are...

- Job Queuing
- Job Scheduling
- Resource Monitoring
- Resource Management

When users need to submit a job all that they need to do is specify in a small file, called a ClassAd, the kind of environment there job needs and the Condor system will take care of the rest.

A typical Condor setup might make use of all the wasted computing power that many facilities have tied up in idle workstations. Condor can manage these workstations and run jobs on them while the mouse or keyboard has no input and if input is detected, store the jobs state, shift to a idle workstation and continue running again.

4 Universes

Inside Condor there are several different environment user can choose when submitting there jobs. These Condors universes allow us to specify even more about what type of job to run. Users specify the Universe they require for there job in a submit description file. If a universe is not specified, the default is standard. The Universes Condor defines are...

Standard - For running jobs that you wish to be able to store there state and shift to a different machine if interrupted. Requires source code to be specially linked with `condor_compile`.

Vanilla - A anything goes environment, if for some reason you code will not relink under `condor_compile` try to run under vanilla.

Parallel - For running a number of jobs at the same time, or running a MPI job.

Grid - Used to submit jobs onto remote grids.

Java - Allows users to run jobs written for the Java Virtual Machine.

5 ClassAds

Because Condor can manage such a diverse kinds of hardware and software it needs a simple yet very effective way of delegating particular jobs to particular machines. Condors way of talking this problem is ClassAds. A ClassAd is simply as is sounds, the machines all have a ClassAd, and the job submit files all have a ClassAd. Condors job is to match them appropriately.

Machine ClassAds - All Condor machines have a very verbose and highly configurable ClassAd describing the entire machine. RAM, CPU, loads, usage hours, conditions i.e. only use when no keyboard or mouse interaction for 15 minutes, and many other properties are available in a machines ClassAd.

Job ClassAds - Every jobs that is submitted also has a associated ClassAd, any the user may specify what type of architecture, operating system, amount of RAM and also the type of Universe that they would like to run in.

Condor will then play the role of a matchmaker by continuously reading all the job ClassAds and all the machine ClassAds, matching and ranking job ads with machine ads.

6 Usage

Using Condor is very simple once it is configured and running. The basic commands are...

condor_status - A summary of information from ClassAds about the machines available.

condor_q - The current job queue.

condor_submit <filename> - Creates a job ClassAd based upon the submit file information and runs the job.

In order to use condor_submit you must write a submit file with some information about the job to be run, a simple example is in the figure below.

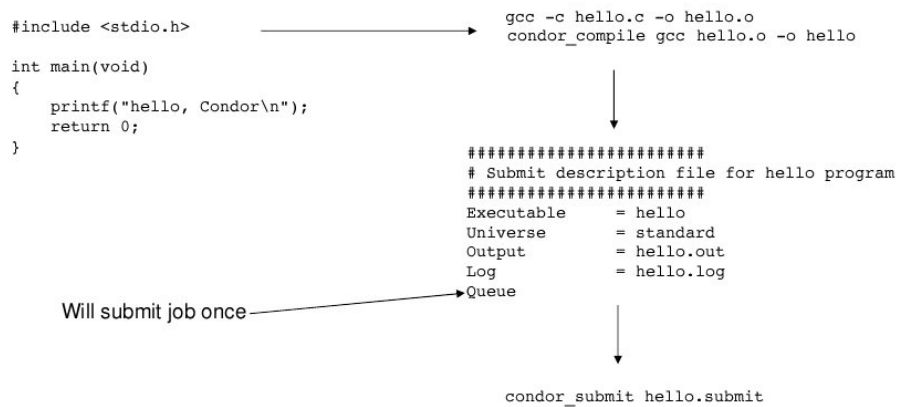


Figure 1: A simple job submit.

A more complex job may require us to specify the type of architecture and operating system, or even memory. The following two figures show this.

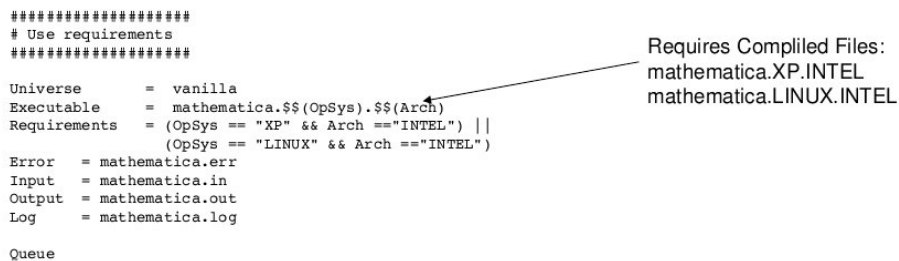


Figure 2: A complex job submit specifying architectures and operating systems.

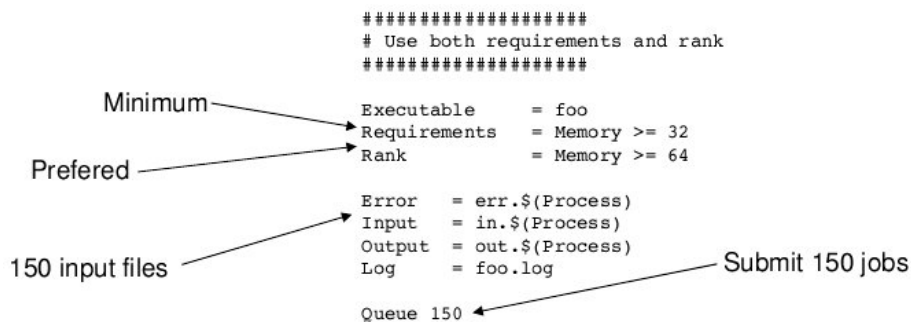


Figure 3: Another complex job submit specifying memory requirements.

We can also submit to the parallel universe to run concurrent jobs or MPI jobs. Note that the Condor machines may not be setup to run fast MPI jobs, but can be. Example in the next two figures.

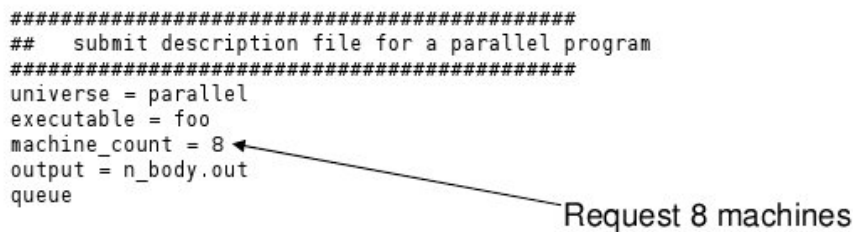


Figure 4: A parallel job submit specifying 8 jobs to run at the same time.

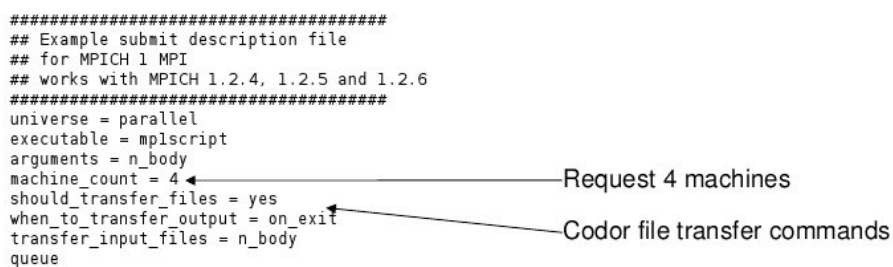


Figure 5: Another parallel job submit specifying a MPI job.

7 Related Technologies

7.1 Flocking

Flocking allows two or more sets of Condor machines to be connected. It allows us to share resources, i.e. share between the science department resources and the commerce resources.

7.2 DAGMan

Another feature that Condor can support is a functionality called DAGMan (Directed Acyclic Graph Manager). It gives Condor the ability to manage dependencies between jobs and can be very useful to manage a large batch of jobs. An example of a DAGMan submission is in the figure below.

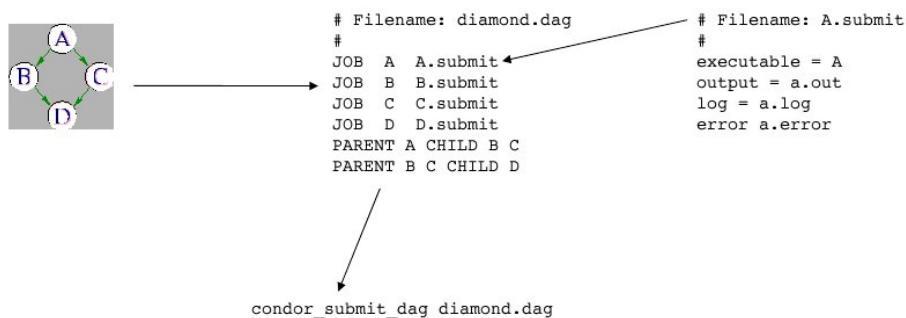


Figure 6: Another parallel job submit specifying a MPI job.

7.3 Condor-G

Another ability of Condor is Condor-G. Condor-G uses the Globus Toolkit to start the job on the remote machine and provides a way to access the 'grid' a growing area in computer science.

7.4 Condor Pool Tools

Condor Pool Tools are a set of scripts that can be used to manage your Condor environment. The tool help you create a list of machines that should be in the environment and ones that are in the environment.

7.5 Condor File Transfer

Because of the diverse nature of that Condor it can run accross large geographic areas so a secure way of transferring files is needed. Condor uses its own system called 'Condor File Transfer' it is a secure system where Condor will transfer any files needed by a job from the machine where the job was submitted to the machine the job is going to be run on.

8 References

- Condor Project Homepage. Retrieved May 10 2009, from <http://www.cs.wisc.edu/condor/>
- Condor User Manual. Retrieved May 10 2009, from <http://www.cs.wisc.edu/condor/manual/v7.2/>
- Condor High-Throughput Computing System. Retrieved May 10 2009, from http://en.wikipedia.org/wiki/Condor_High-throughput_computing_system
- High-throughput computing. Retrieved May 10 2009, from http://en.wikipedia.org/wiki/High-throughput_computing