High Throughput Computing
Derek Weitzel - HCC
What is HTC?

• High Throughput Computing uses many cores for long periods of time.

• Not usually associated with MPI, OpenMP, or CUDA.
How does HTC look?

• Lots and lots of small jobs
  – Small ( < 8 hours) jobs

  – *Single* or small core count ( < 8)

• Workflow support is excellent
  – Job B depends on Job A

  – Advanced frameworks such as Pegasus with data dependencies
Data In HTC

- Data is usually very important in HTC
  - Small per job data (< 1GB)
  - But possibly massive total data (> 100GB)
Usual Problems in HTC

- Application Portability
  - Can the application run anywhere?
  - Is it easy to run?
  - Can it be compiled on a lowest denominator platform (redhat 5)?
HTC: What it looks like

• Submit File:

```plaintext
universe = vanilla
Executable = myjob.exe
arguments = 10
output = job.output
type = job.error
log = job.log
transfer_input_files = input1, input2
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
queue 1
```

• Submit Job:

```bash
$ condor_submit example.submit
```
HTC: What it looks like

- A Condor Submit file describes the job to the system

```plaintext
universe = vanilla
Executable = myjob.exe
arguments = 10
output = job.output
error = job.error
log = job.log
transfer_input_files = input1, input2
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
queue 1
```
HTC: What it looks like

- Executable is the (compiled?) executable that will run on the remote host

```plaintext
universe = vanilla
Executable = myjob.exe
arguments = 10
output = job.output
error = job.error
log = job.log
transfer_input_files = input1, input2
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
queue 1
```
HTC: What it looks like

- Arguments are used for the application when it runs on the worker node.

```plaintext
universe = vanilla
Executable = myjob.exe
arguments = 10
output = job.output
error = job.error
log = job.log
transfer_input_files = input1, input2
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
queue 1
```
HTC: What it looks like

• Output and Error is the file locations that the executable’s stdout and stderr should go.

```plaintext
universe = vanilla
Executable = myjob.exe
arguments = 10
output = job.output
error = job.error
log = job.log
transfer_input_files = input1, input2
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
queue 1
```
HTC: What it looks like

- Log shows the job’s transitions (idle -> running)

```plaintext
universe = vanilla
Executable = myjob.exe
arguments = 10
output = job.output
error = job.error
log = job.log
transfer_input_files = input1, input2
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
queue 1
```
HTC: What it looks like

- Log shows the job’s transitions (idle -> running)

```plaintext
universe = vanilla
Executable = myjob.exe
arguments = 10
output = job.output
transfer_input_files = input1, input2
log = job.log
when_to_transfer_output = ON_EXIT
queue = 1
```
HTC: What it looks like

- Lists the files to be transferred with the job.
- Job is not running in your home directory.

```plaintext
universe = vanilla
Executable = myjob.exe
arguments = 10
output = job.output
error = job.error
log = job.log
transfer_input_files = input1, input2
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
queue 1
```
HTC: What it looks like

- Tell Condor when to transfer the files

```
universe = vanilla
Executable = myjob.exe
arguments = 10
output = job.output
error = job.error
log = job.log
transfer_input_files = input1, input2
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
queue 1
```
HTC: What it looks like

• Submit 1 of the jobs described above

universe = vanilla
Executable = myjob.exe
arguments = 10
output = job.output
error = job.error
log = job.log
transfer_input_files = input1, input2
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
queue 1
HTC: What it looks like

• Submit 1 of the jobs described above

Interactive Submission

```plaintext
error = job.error
log = job.log
transfer_input_files = input1, input2
should_transfer_files = YES
when_to_transfer_output = ON_EXIT
queue 1
```
So what just happened?

1. Job was submitted to cluster
2. Job was scheduled onto a worker node
3. Input files and executable was transferred to the worker node
4. Executable is started on the worker node
5. Executable completes
6. Stdout and Stderr, as well as files that were changed are transferred back to submitter
How to expand access

- HTC are very well suited to going out to other clusters
How to expand access

• Submit file looks exactly the same.

• Submission can automatically overflow to other clusters.