Education - User School 15 Thu 40 Chirp

Education

DocumentClassification	
DocumentType	

Table of Contents

1 Using Chirp to transfer output from running jobs

1.1 Goals

- This tutorial will show you how to get output from your application as it is running using Remote I/O tools.

1.2 Transferring output while your job is running

The HTCondor file transfer mechanism transfers only transfer files when a job is completed or while it is being evicted. Sometimes, it's useful to get files from a running job while it's still running (e.g. checkpoint files). This part of the tutorial will show how to transfer files while your job is still running.

HTCondor integrates with a set of tools called Parrot and Chirp that allows your job to transfer files back to the the submit node while the job is still running. In order to do this, your submit files need to change slightly and you'll need to do a few more things in your job.

1. First, you'll need to add the following to your submit file:

   ```
   +WantIOProxy = True
   ```
2. Second, you'll need to transfer the `parrot_run` binary with your file, you can do this by using a transfer_input_files command

   ```
   transfer_input_files = /usr/bin/parrot_run
   ```
3. Finally, you'll need to make sure that you save your file at `/chirp/CONDOR/` followed by the path on the submit node where the file should go. E.g., if you want the ouput file to go to `/stash/user/username/job1/test`, the path should be `/chirp/CONDOR/stash/user/username/job1/test`.

Let's try creating a few files in a job and transfer them while the job is running:

1. First log in to the OSG Connect submit node (login.osgconnect.net)
2. Create a directory for this tutorial :

   ```
   $> mkdir chirp_tutorial
   $> cd chirp_tutorial
   ```
3. Create a file named `job.sh` with the following:

   ```
   #!/bin/bash
   echo "test1" > checkpoint_file
   ./parrot_run cp checkpoint_file $1/checkpoint_file1
   sleep 30
   echo "test2" > checkpoint_file
   ./parrot_run cp checkpoint_file $1/checkpoint_file2
   sleep 30
   echo "test3" > checkpoint_file
   ./parrot_run cp checkpoint_file $1/checkpoint_file3
   rm checkpoint_file
   ```

   4. Make the file executable:

   ```
   $> chmod 755 job.sh
   ```
4. Create a submit file (`chirp.submit`) with the following:

   ```
   output        = job.out.$(Process)
   error         = job.error.$(Process)
   ```

```
log            = job.log.$(Process)

universe = vanilla
executable = job.sh
transfer_input_files = /usr/bin/parrot_run
+WantIOProxy = True
arguments = /chirp/CONDOR/$ENV(PWD)
queue 1
```

5. Submit the job:

```
$> condor_submit chirp.submit
```

6. While your job is running, you should see files named `checkpoint_file1`, `checkpoint_file2`, and `checkpoint_file3` appear in your submit directory.

7. You can also run `cat` or view the files to verify that they contain the string `test1`, `test2`, or `test3`

As an exercise, try modifying the submit file and script so that the checkpoint files are saved in `/stash/user/username/checkpoints` where username is replaced by your OSG Connect username.

1.3 Further information

- Using Condor with Parrot/Chirp
- Manual for condor_chirp

This topic: Education > UserSchool15Thu40Chirp
Topic revision: r5 - 30 Jul 2015 - 19:08:20 - TimCartwright

TWiki | Report Bugs | Privacy Policy