Installing LittleJohn

Download LittleJohn and Unpack

Download Page:

http://www.cac.cornell.edu/matlab/downloads

Latest Release:

http://www.cac.cornell.edu/matlab/downloads/littlejohn-distro-0.3.0.zip

Snapshots (For the brave and NOT for this workshop):

http://consultrh5.cac.cornell.edu/littlejohn/snapshots/

Unpack somewhere you have write permissions, some good choices:

Windows: c:\cac

Mac: /Users/cac

Linux: /home/username/cac

Lab Computers: USB disk

You should end up with a littlejohn directory created in this folder which contains a number of .m files and a set of subdirectories. We will refer to the littlejohn directory as LITTLEJOHNHOME

Modify cacsched.m

Most of our examples make use of a scheduler object created by cacsched.m, you'll need to edit this file to insert your username. You can use any editor to do this, but it's easy to do inside of MATLAB which is what we'll show here.

1) Start MATLAB.

2) Edit LITTLEJOHNHOME\cacsched.m and change line 24 to include your CAC username (mine is naw47)

>> addpath('/home/naw47/cac/');

```
20 -    LJHome = fileparts(which('runTests'));
21
22      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
23      %CHANGE THIS LINE TO INCLUDE YOUR CAC USERNAME
24 -    userHome ='\\storage01\matlab\naw47';
25      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
26
```

Optional Modify cacsched.m

- The communication between the client MATLAB and the scheduler is file based. This means each job submission creates a large number of files which need to be stored somewhere on the client machine. By default this is set to LITTLEJOHNHOME\jobs in cacsched.m, but you can change this if you'd like.
- The top line sets it to the default, the commented second line is an example of manually setting it.

```
47 -    set(sched,'DataLocation',fullfile(LJHome, 'jobs'));
48      %set(sched,'DataLocation','/home/naw47/LJJobs');
49 -    get(sched);
50
```

Modify classpath.txt

LittleJohn is heavily dependant on a series of Java libraries for functionality. The jars in LITTLEJOHNHOME/lib need to be added MATLAB's java classpath. This is controlled by text file, which is normally in $matlabroot/toolbox/local/classpath.txt. The easiest way to find the full path is from the matlab prompt, typical paths shown below.

Windows: *>> which classpath.txt*
C:\MATLAB\R2009a\toolbox\local\classpath.txt

Unix: *>> which classpath.txt*
/usr/local/matlab/toolbox/local/classpath.txt

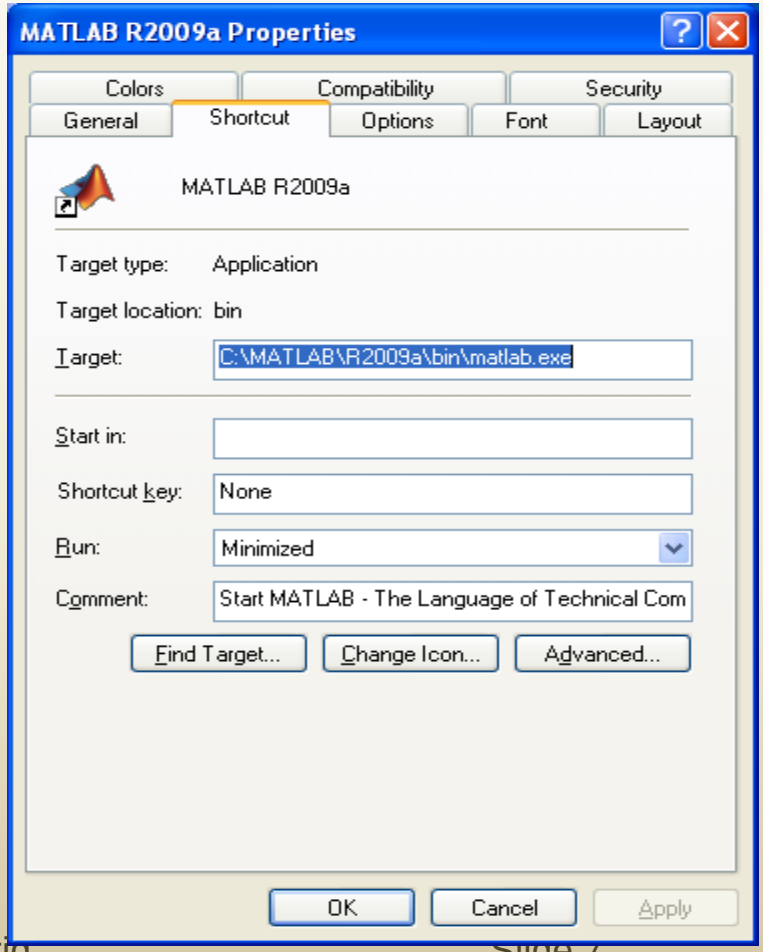
Mac: *>> which classpath.txt*
/Applications/Matlab_R2009b.app/toolbox/local/classpath.txt

Modify classpath.txt

- In many situations you'll need root (administrator on Windows) privileges to modify this file, which affects the global MATLAB install. If this is not feasible (or if you are on a multi-user system) you may alternatively place a classpath.txt file in your startup directory. This file is user-specific and will only affect your MATLAB environment.
- MATLAB looks first in the startup directory for a classpath.txt file, then the default directory, using whichever it finds first.
- On most platforms the startup directory can be found by using the "userpath" function. For info: http://www.mathworks.com/access/helpdesk/help/techdoc/matlab_env/f8-10506.html

Workshop Path

- For the lab computers, there are issues.
- On the Windows lab computers we get around this by creating a shortcut and placing it on the USB disk.
- Find the MATLAB 2009a and copy the shortcut over to the USB Drive.
- Right click on the MATLAB icon and select Properties.
- Modify “Start in” to reflect the USB Drive (probably i:/).
- Copy the system classpath.txt file to the USB drive.



Modify classpath.txt

- Add the following lines to classpath.txt replacing LITTLEJOHNHOME with the your install path. Note these are Windows slashes, reverse them for Mac and Unix.

  LITTLEJOHNHOME\lib\littlejohn.jar
  LITTLEJOHNHOME\lib\bcprov-jdk15-1.43.jar
  LITTLEJOHNHOME\lib\bcprov-jdk16-143.jar
  LITTLEJOHNHOME\lib\cog-jglobus-1.7.0.jar
  LITTLEJOHNHOME\lib\commons-logging-1.1.1.jar
  LITTLEJOHNHOME\lib\cryptix-asn1.jar
  LITTLEJOHNHOME\lib\cryptix.jar
  LITTLEJOHNHOME\lib\cryptix32.jar
  LITTLEJOHNHOME\lib\cxf-2.2.4.jar
  LITTLEJOHNHOME\lib\log4j-1.2.15.jar
  LITTLEJOHNHOME\lib\not-yet-commons-ssl-0.3.11.jar
  LITTLEJOHNHOME\lib\puretls.jar

Examine classpath.txt

- Restart MATLAB in order to pick up changes to classpath.txt. Use the javaclasspath function to check that your changes have been picked up:

>> javaclasspath

STATIC JAVA PATH

C:\MATLAB\R2009a\java\patch
C:\MATLAB\R2009a\java\jar\util.jar
C:\MATLAB\R2009a\java\jar\widgets.jar
c:\projects\matlab\lib\littlejohn.jar ← *Ensure file exists!*
c:\projects\matlab\lib\bcprov-jdk16-143.jar
…

Test install with runTests

- In the LittleJohn distribution is a tool call “runTests” which performs a series of tests to test functionality at different levels. Level 1 tests test that the basic install appears functional, run that now.

>> cacsched

Configuration: ''

Type: 'generic‘ ….

>> runtests(sched,1);

Running basic install tests to ensure install is OK*

Classpath entries check out, Java looks good, trying to load LittleJohn classes

Retrieved R2009a as current matlab version

[2010-06-10 21:21:45,109] Using certificate issued by:CN=Cornell Center for Advanced Computing CA, OU=Center for Advanced Computing, OU=myproxy.cac.cornell.edu

Successfully loaded LittleJohn version 0.3.0-b299. Java is OK.

DataLocation is writable and ready to go

Basic install tests complete. Resolve any errors before continuing.****

Register your certificate

- In order to complete the install you'll need to download the CAC server certificates and register your certificate with us.
- Run cacRegisterCertificate() and follow the instructions in the pop-up dialogs. This process will take approximately 2 minutes to propagate through our system.

>> cacRegisterCertificate();

Found existing Cert path of:C:\Documents and Settings\naw47\.globus\certificates

Fetching https://myproxy.cac.cornell.edu/CA/fb4341f6.0 to C:\Documents and Settings\naw47\.globus\certificates\fb4341f6.0

Fetching https://myproxy.cac.cornell.edu/CA/fb4341f6.signing_policy to C:\Documents and Settings\naw47\.globus\certificates\fb4341f6.signing_policy

[2010-06-10 21:27:11,000] Using certificate issued by:CN=Cornell Center for Advanced Computing CA, OU=Center for Advanced Computing, OU=myproxy.cac.cornell.edu

Functional Testing

- Level 2 tests ensure the file transfer system is working and that communication with the scheduler is ok.

  >> runtests(sched,2);

  Running functional tests of LittleJohn components.*

  gridFTP appears to be working OK!

  Attempting to submit a simple test job to TUC.

  Service communication looks OK

  *.***Functional component tests passed****

- The contrib folder contains additional utilities that add functionality to LittleJohn

  >> addpath('LITTLEJOHNHOME/contrib');

  >> updateContrib();

Complete Testing

- Level 3 tests submit actual jobs to the cluster and ensure that both parallel and distributed tests are functioning.

>> runtests(sched,3);
Running integration tests that simulate actual use of TUC by a user.
Current Number of Free Cores: 359
Current Number of running jobs: 7
Next downtime(when your jobs MAY be cancelled): June 9, 2010, 8am-5pm.
Running distributed job test...
Downloading completed job: Job65.
Running parallel job test...
Downloading completed job: Job66.
Running unique directory distributed test....
4 jobs not yet complete, pausing then rechecking.
3 jobs not yet complete, pausing then rechecking.
Downloading completed job: Job67.
Tests passed, things appear to be working for you!

Next Steps

- At this point, you have a fully operational install of LittleJohn. Your next steps should be to take a look in the examples directory to start seeing how to take advantage of TUC.
- **cacsubmit** – super simple distributed job example
- **cacparsubmit** – simple example of a parallel (MPI) job.
- **pooljobremote** – matlab pool example
- **cacNonBlockSubmit** – example of submitting a non-blocking job (avoiding waitForState)