Gondor : Making HTCondor DAGman Groovy

Jim White
Department of Linguistics
University of Washington
jimwhite@uw.edu
github.com/jimwhite

Computational Linguistics @ UW

- http://www.compling.uw.edu/
- Computational Linguistics Masters program est. 2001
  Department of Linguistics established in 1963
- 25 ~ 30 new CLMS students each Fall
- Classes can be attended on-line
- Departmental cluster (~100 nodes) runs Condor (7.8.8)
- Most class assignments and projects must use Condor

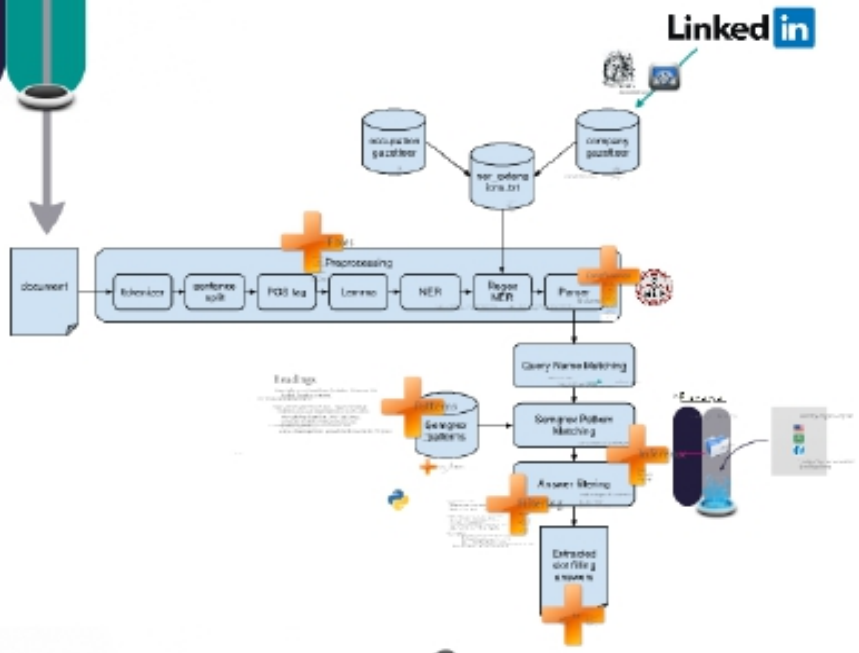
CLMS Courses using Condor

- LING 473: Computational Linguistics Fundamentals
- LING 570: Shallow Processing Techniques for Natural Language Processing
- LING 571: Deep Processing Techniques for Natural Language Processing
- LING 572: Advanced Statistical Methods in Natural Language Processing
- LING 573: Natural Language Processing Systems and Applications

LING 573 - Natural Language Processing Systems and Applications

Information Retrieval

Knowledge Base Population Task
Text Analysis Conference (NIST.gov)
TAC 2010



Information Extraction

LING 473, 57{0-2} Programs

```
$ condor_submit myjob.cmd
```

```
universe        = vanilla
executable  = /usr/bin/python
getenv          = true
input           = myinput.in
output          = myoutput.out
error           = myerror.err
log             = mylogfile.log
arguments   = "myprogram.py -x"
transfer_executable = false
queue
```

The system will send you email when your job is complete.

Grading Student Programs

- Issues
  - Student programs must run using Condor as another user (TA)
  - Rubric points for "Runs as-is" and related requirements
  - Students don't have a way to run their job that way
- Solutions
  - Dedicated grading user accounts
  - Scripts to run the jobs
  - Student accessible checking program
- CheckIt!

```
$ ~ling572_00/bin/check_it project1 <project1.tar >results.html
$ lynx results.html
```

CheckIt! project2 for jimwhite

Copied 1365456 bytes successfully.

```
tar xf /home2/ling572_01/project1/jimwhite_8852931575133087009
```

Contents

Name	Size
TF.java	3371
TF.class	4466
compile.sh	24
error.txt	0
log.txt	1125
run.sh	37
TF$1.class	949
output.txt	3059227
condor.cmd	124
readme.txt	855

Submission Inventory

Item	Present?	OK?	Pattern	Full Path
Exec	yes	ok	run.sh	project1/jimwhite_8852931575133087009.dir/content/run.sh
Condor	yes	ok	condor.cmd	project1/jimwhite_8852931575133087009.dir/content/condor.cmd
Compile	yes	ok	compile.sh	project1/jimwhite_8852931575133087009.dir/content/compile.sh
Output	yes	ok	output.txt	project1/jimwhite_8852931575133087009.dir/content/output.txt
README	yes	ok	(?i)readme\.(txt|pdf)	project1/jimwhite_8852931575133087009.dir/content/readme.txt

Running Condor Job

/condor/bin/condor_submit condor.cmd

```
Submitting job(s).
1 job(s) submitted to cluster 111871.
```

/condor/bin/condor_wait -wait 3600 log.txt

```
All jobs done.
```

Job Results: Log (log.txt)

```
000 (111871.000.000) 08/09 08:24:29 Job submitted from host: <192.168.100.50:53229>
...
001 (111871.000.000) 08/09 08:24:29 Job executing on host: <192.168.100.51:52838>
...
006 (111871.000.000) 08/09 08:24:38 Image size of job updated: 1
        3  -  MemoryUsage of job (MB)
        2076  -  ResidentSetSize of job (KB)
...
```

```
006 (111871.000.000) 08/09 08:29:39 Image size of job updated: 1881436
        1554  -  MemoryUsage of job (MB)
        1590988  -  ResidentSetSize of job (KB)
...
005 (111871.000.000) 08/09 08:29:47 Job terminated.
        (1) Normal termination (return value 0)
                Usr 0 00:05:31, Sys 0 00:00:02  -  Run Remote Usage
                Usr 0 00:00:00, Sys 0 00:00:00  -  Run Local Usage
                Usr 0 00:05:31, Sys 0 00:00:02  -  Total Remote Usage
                Usr 0 00:00:00, Sys 0 00:00:00  -  Total Local Usage
        0  -  Run Bytes Sent By Job
        0  -  Run Bytes Received By Job
        0  -  Total Bytes Sent By Job
        0  -  Total Bytes Received By Job
        Partitionable Resources :    Usage  Request
           Cpus                 :                1
           Disk (KB)            :        1       1
           Memory (MB)          :     1554    2048
...
```

Job Results: Error (error.txt)

Empty

Job Results: Output (output.txt)

```
the     4398031
a       1909523
to      1893178
of      1888349
and     1759666
in      1486078
that    814646
for     793612
is      712493
on      564755
by      559398
with    512396
he      494957
it      484400
at      463586
said    442322
was     439316
as      431831
his     373389
but     347712
be      337067
from    328710
are     328488
have    314716
i       307228

...

tecial  1
athenaeum's     1
encrusting      1
apostolidis     1
faraints        1
beatlemaniac    1
stelmakhova     1
rosser's        1
kafandaraki     1
tapahura        1
mashour's       1
sleazos 1
mudo    1
quarzsite       1
mimose  1
hildegarde's    1
killoh's        1
comrade's       1
bulkies 1
burmeister      1
leprino 1
mugg    1
claramente      1
randerson       1
muha    1
```

Condor Job Completed

This tar file conforms to the "Runs As-Is" rubric for the Condor Job portion of Project 1. This version of CheckIt! does not yet test your compile.sh (if any). Note that this is not any sort of check on whether your output is correct. Also note that if the file inventory showed missing items that you intend to include (such as README), then you should fix that before submitting.

Writing Condor Programs

flexible.job

```
file_ext        = $(depth)_$(gain)
universe        = vanilla
executable      = /opt/mono/bin/mono
getenv          = true
output          = acc_file.$(file_ext)
error           = q4.err
log             = q4.log
arguments       = "myprog.exe model_file.$(file_ext) sys_file.$(file_ext)"
transfer_executable = false
queue
```

```
$ condor_submit -append "depth=20" -append "gain=4" flexible.job
```

versus

$ mono myprog.exe model_file.20_4 sys_file.20_4 >acc_file.20_4

Gondor v1

ensemble_parse.groovy

```
1     // James White mailto:jimwhite@uw.edu
2
3     ////////////////////////////
4     // Environmental Dependencies
5     ////////////////////////////
6
7     // If there are environment variables you want to copy from the current process,
         use clone_environment:
8     // gondor.clone_environment('PATH', 'ANT_HOME', 'JAVA_HOME')
9     // If you want to copy *all* of the the current environment variables,
         omit the variable names (not recommended):
10    // gondor.clone_environment()
11
12    gondor.environment =
         [PATH:"/usr/local/bin:/bin:/usr/bin:/opt/git/bin:/opt/scripts:/condor/bin"
13        , LC_COLLATE:'C'
15    ]
```

http://groovy.codehaus.org/

Gondor v1

```
17 //////////////
18 // Data Files
19 //////////////
20
21 workspace_dir = new File('/home2/jimwhite/workspace/parsers')
22
23 // Each parser has it's own binary, but we'll use the one in base for them all.
24 bllip_dir = new File(workspace_dir, 'base/bllip-parser')
25
26 ensemble_dir = new File(workspace_dir, 'ensemble')
27
28 ycorpus_dir = new File(workspace_dir, 'ycorpus')
29
30 ////////////////////////////
31 // Condor Command Definitions
32 ////////////////////////////
33
34 // first-stage/PARSE/parseIt -l399 -N50 first-stage/DATA/EN/ $*
35 parse_nbest = gondor.condor_command(
      new File(bllip_dir, 'first-stage/PARSE/parseIt')
      , ['-K.flag', '-l400.flag', '-N50.flag', 'model.in', 'input.in'])
36
37 // second-stage/programs/features/best-parses" -l "$MODELDIR/features.gz"
       "$MODELDIR/$ESTIMATORNICKNAME-weights.gz"
38 rerank_parses = gondor.condor_command(
      new File(bllip_dir, 'second-stage/programs/features/best-parses')
      , ['-l.flag', 'features.in', 'weights.in', 'infile.in'])
```

Generated Submit Description

_home2_jimwhite_workspace_parsers_base_bllip-parser_second-stage_programs_features_best-parses.condor

```
####################
#
# James White (mailto:jimwhite@uw.edu)
#
####################

Universe   = vanilla
Environment= PATH=/usr/local/bin:/bin:/usr/bin:/opt/git/bin:
                                  /opt/scripts:/condor/bin;LC_COLLATE=C
Executable = /home2/jimwhite/workspace/parsers/base/bllip-parser/
                           second-stage/programs/features/best-parses
Arguments  = -l $(_features) $(_weights)
Log        = jimwhite__home2_jimwhite_workspace_parsers_base_bllip-
               parser_second-stage_programs_features_best-parses.log
Input      = $(_MyJobInput)
Output     = $(_MyJobOutput)
Error      = $(_MyJobError)
Request_Memory=5*1029
Notification=Error
Queue
```

Gondor v1

```
40 //////////////////////
41 // Job DAG Definitions
42 //////////////////////
43
44 ['brown-train.mrg'].each { String file_path ->
45   ensemble_dir.eachFileMatch(~/parser_.*/) { File parser_dir ->
46      def PARSER_MODEL=new File(parser_dir, 'first-stage/DATA/EN/')
47      def MODELDIR=new File(parser_dir, 'second-stage/models/ec50spnonfinal')
48      def ESTIMATORNICKNAME='cvlm-l1c10P1'
49      def RERANKER_WEIGHTS = new File(MODELDIR, ESTIMATORNICKNAME + '-weights.gz')
50      def RERANKER_FEATURES = new File(MODELDIR, 'features.gz')
51
53      def sysout_dir = new File(parser_dir, 'tmp/parsed')
54      sysout_dir.deleteDir()
55      sysout_dir.mkdirs()
56
57      def nbest_output = new File(sysout_dir, file_path + '.nbest')
58      def reranker_output = new File(sysout_dir, file_path + '.best')
59
60      def charniak_input = new File(ycorpus_dir, file_path + ".sent")
63
64      parse_nbest(model:PARSER_MODEL, input:charniak_input, outfile:nbest_output)
65      rerank_parses(features: RERANKER_FEATURES, weights: RERANKER_WEIGHTS
                , infile:nbest_output, outfile:reranker_output)
66   }
67 }
```

Generated DAGman DAG File

JOB _home2_jimwhite_..._parseIt_J1 _home2_jimwhite_..._parseIt.condor
VARS _home2_jimwhite_..._parseIt_J1
_model="/workspace/ensemble/parser_19/first-stage/DATA/EN/"
_input="/workspace/ycorpus/brown-train.mrg.sent"
_MyJobOutput="/workspace/parser_19/tmp/parsed/brown-train.mrg.nbest"
_MyJobError="ensemble_parse_jobs/_home2_jimwhite_..._parseIt_J1.err"

JOB _home2_jimwhite_..._best-parses_J2 _home2_jimwhite_..._best-parses.condor
VARS _home2_jimwhite_..._best-parses_J2
_features="…/parser_19/…/ec50spnonfinal/features.gz"
_weights="…/parser_19/…/ec50spnonfinal/cvlm-l1c10P1-weights.gz"
_MyJobInput="…/parser_19/tmp/parsed/brown-train.mrg.nbest"
_MyJobOutput="…/parser_19/tmp/parsed/brown-train.mrg.best"
_MyJobError="ensemble_parse_jobs/_home2_jimwhite_..._best-parses_J2.err"

... MANY MORE LIKE THAT ...

PARENT _home2_jimwhite_..._parseIt_J1 **CHILD** _home2_jimwhite_..._best-parses_J2
PARENT _home2_jimwhite_..._parseIt_J3 **CHILD** _home2_jimwhite_..._best-parses_J4
PARENT _home2_jimwhite_..._parseIt_J5 **CHILD** _home2_jimwhite_..._best-parses_J6
...

Gondor v1

```
40 //////////////////////
41 // Job DAG Definitions
42 //////////////////////
43
44 ['brown-train.mrg'].each { String file_path ->
45   ensemble_dir.eachFileMatch(~/parser_.*/) { File parser_dir ->
46      def PARSER_MODEL=new File(parser_dir, 'first-stage/DATA/EN/')
47      def MODELDIR=new File(parser_dir, 'second-stage/models/ec50spnonfinal')
48      def ESTIMATORNICKNAME='cvlm-l1c10P1'
49      def RERANKER_WEIGHTS = new File(MODELDIR, ESTIMATORNICKNAME + '-weights.gz')
50      def RERANKER_FEATURES = new File(MODELDIR, 'features.gz')
51
53      def sysout_dir = new File(parser_dir, 'tmp/parsed')
54      sysout_dir.deleteDir()
55      sysout_dir.mkdirs()
56
57      def nbest_output = new File(sysout_dir, file_path + '.nbest')
58      def reranker_output = new File(sysout_dir, file_path + '.best')
59
60      def charniak_input = new File(ycorpus_dir, file_path + ".sent")
63
64      parse_nbest(model:PARSER_MODEL, input:charniak_input, outfile:nbest_output)
65      rerank_parses(features: RERANKER_FEATURES, weights: RERANKER_WEIGHTS
                , infile:nbest_output, outfile:reranker_output)
66   }
67 }
```

My Development Principles

- Work Independently
  - Hip, Hip, Hooray for Leo Singer and LIGO!
    HTCondor MacPort:

    ```
    sudo port install htcondor
    sudo port load htcondor
    ```

- Brevity is Beautiful
- Don't Repeat Yourself (DRY)
- Integrate with Other Current and Future Compute Systems
- Compile-time Provenance

DRMAA Java Binding

- Been Around a Long Time
- Supported by Many Systems
- Constrain the Design to Ease Future Interoperability
- New Implementation for Condor sans JNI
  https://github.com/jimwhite/condor-jrmaa
  - Generates Submit Description Files and uses `condor_submit`
- DAGman Workflow Extension
  - Generates DAGman DAG File (and Submit Files)
  - Uses DRMAA and pretends all jobs succeed
  - Add Dependency Method:

```
void addToParentJobIds(String childJobId, String parentJobId);
```

Gondor v3

GoodGondor.groovy

```
import org.ifcx.gondor.Command

@groovy.transform.BaseScript org.ifcx.gondor.WorkflowScript workflowScript

def parse_nbest = command(path:'first-stage/PARSE/parseIt') {
    flag "-K"  ;  flag "-l400" ; flag "-N50"
    infile "model"
    infile "input"
    outfile "output"
    jobTemplate { softRunDurationLimit = 100 }
}

def rerank_parses = command(path:'second-stage/programs/features/best-parses') {
    flag '-l' ; infile 'features' ; infile 'weights' ; infile 'stdin' ; outfile
'stdout'
}

def modelFile = new File("model.dat")
def inputFile = new File("input.txt")
def parsedFile = new File("output1.ptb")
def p = parse_nbest(n:15, model:modelFile, input:inputFile, output:parsedFile, m:2)

def RERANKER_FEATURES = new File('RERANKER_FEATURES')
def RERANKER_WEIGHTS = new File('RERANKER_WEIGHTS')

def reranker_output = new File("best_parse.ptb")

(parse_nbest(model: modelFile) << new File("in2.txt")) |
    rerank_parses(features: RERANKER_FEATURES, weights: RERANKER_WEIGHTS) >> new
File("out2.tree")
```

The Road Ahead for Gondor

- Self-describing Command Line Scripts
- Dynamic SubDAG Workflow Scripts
- Persistent Workflow Results
- Workflow Reduction
- Provenance
- Reproducible Research

Workflow Persistence & Reduction

- Put Everything in Git
- All Intermediate Artifacts including Condor Control Files
- Previous Results Reused If Desired Based on Object IDs
  - See for example Nix – The Functional Package Manager https://nixos.org/nix/

    Eelco Dolstra. Secure Sharing Between Untrusted Users in a Transparent Source/Binary Deployment Model. In *20th IEEE/ACM International Conference on Automated Software Engineering* (ASE 2005), pages 154–163, Long Beach, California, USA. ACM Press, November 2005.
- File Transfer via Pull or Push As Desired
- git-annex (or similar) for very big blobs

Provenance

- Git2PROV.org
  - Generates PROV-O (and –N, -JSON, and SVG) from Git commits

Git2PROV: Exposing Version Control System Content as W3C PROV
by Tom De Nies, Sara Magliacane, Ruben Verborgh, Sam Coppens, Paul Groth, Erik Mannens, and Rik Van de Walle

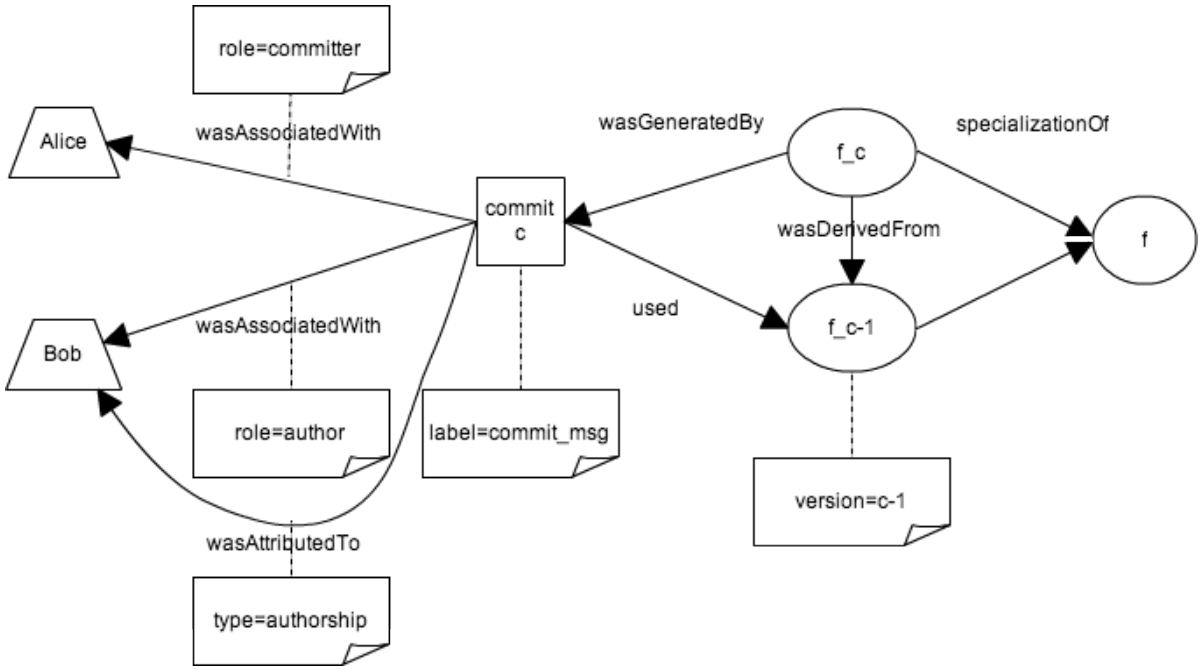


Fig. 1. Mapping of Git operations to PROV concepts. Note that the Activity *Start* and *End* concepts of PROV are not depicted, and correspond to, respectively, the author time and the commit time of each commit.

Thank You!

http://depts.washington.edu/newscomm/photos/the-spring-cherry-blossoms-in-the-quad/