



HTCondor Advanced Job Submission

John (TJ) Knoeller
Center for High Throughput Computing

Overview

- Development of an advanced submit file
- Using as many techniques and tricks as possible. 😊

The Problem

- I have a lot of media files that I have collected over the years.
- I want to convert them all to .mp4
(Sounds like a high-throughput problem...)

Basic submit file for conversion

```
Executable = ffmpeg
```

```
Transfer_executable = false
```

```
Should_transfer_files = YES
```

```
file = S1E2 The Train Job.wmv
```

```
Transfer_input_files = $(file)
```

```
Args = "-i '$(file)' '$(file).mp4'"
```

Queue

Converting a set of files

```
Transfer_input_files = $(file)
Args = "-i '$(file)' '$(file).mp4' "
```

```
Queue FILE from (
  S1E1 Serenity.wmv
  S1E2 The Train Job.wmv
  S1E3 Bushwhacked.wmv
  S1E4 Shindig.wmv
  ...
)
```

Output filename problems

- › Output is `$(file).wmv.mp4`. So output files are named

S1E1 Serenity.wmv.mp4

S1E2 The Train Job.wmv.mp4

S1E3 Bushwhacked.wmv.mp4

S1E4 Shindig.wmv.mp4

\$F() to the rescue

- `$Fqpdnx()` expands to parts of a filename

```
file = "./Video/Firefly/S1E4 Shindig.wmv"
```

```
$Fp(file)    -> ./Video/Firefly/
```

```
$Fqp(file)  -> "./Video/Firefly"
```

```
$Fd(file)   -> Firefly/
```

```
$Fn(file)   -> S1E4 Shindig
```

```
$Fx(file)   -> .wmv
```

```
$Fnx(file)  -> S1E4 Shindig.wmv
```

\$Fn() is name without extension

```
Transfer_Input_Files = $(file)
```

```
Args = "-i '$Fnx(file)' '$Fn(file).mp4'"
```

Resulting files are now

```
S1E1 Serenity.mp4
```

```
S1E2 The Train Job.mp4
```

```
S1E3 Bushwhacked.mp4
```

```
S1E4 Shindig.mp4
```


\$Fq() and Arguments

- \$Fq(file) expands to quoted "filename"
 - Gives "parse error" with Arguments statement
- For Args use '\$F(file)' instead.
 - Becomes 'filename' on LINUX
 - Becomes "filename" on Windows

"new" Args preserves spaces

```
FILE = The Train Job.wmv
```

```
Args = "-i '$Fn(file)' -w640 '$Fn(file).mp4' "
```

```
# Tool Tip* see it before you submit it.
```

```
condor_submit test.sub -dump test.ads
```

```
condor_status -ads test.ads -af Arguments
```

```
-i The' 'Train' 'Job.wmv -w640 The' 'Train' 'Job.mp4
```

```
# On *nix the job sees
```

```
-i 'The Train Job.wmv' -w640 'The Train Job.mp4`
```

```
# on Windows the job sees
```

```
-i "The Train Job.wmv" -w640 "The Train Job.mp4"
```

When you can't use Args

- Argument quoting not portable across operating systems
 - LINUX needs space and ' escaped
 - Windows needs double quotes around filenames that have space or ^
- Transfer_input_files will not transfer a file with a comma in the name.
- What the job sees can be hard to predict

Add custom attributes to the job

```
Executable = xcode.pl
Args = -s 640x360
Transfer_executable = true
Should_transfer_files = true
# +WantIOProxy = true
+SourceDir = $Fqp(FILE)
+SourceFile = $Fqnx(FILE)
+OutFile = "$Fn(FILE).mp4"
Batch_name = $Fd(FILE)
Queue FILE matching files Firefly/*.wmv
```

Use a script to query the .job.ad

```
#!/usr/bin/env perl
# xcode.pl
# Pull filenames from job ad
my $src = `condor_status -ads .job.ad -af SourceFile`;
my $out = `condor_status -ads .job.ad -af OutFile`;
# find condor_chirp (also need +WantIOProxy in job)
my $lib = `condor_config_val libexec`;
chomp $src; chomp $out; chomp $lib;
# fetch the input file
system("$lib/condor_chirp fetch '$src' '$src'")
# do the conversion
system("ffmpeg -i '$src' @ARGV '$out'");
```

See how it's going..

```
condor_q -batch
```

```
OWNER BATCH_NAME ... DONE RUN IDLE TOTAL JOB_IDS
Tj      Firefly/           _   2   2   _ 104.0-4
```

```
condor_q -af:jh JobStatus SourceFile SourceDir
```

ID	JobStatus	SourceFile	SourceDir
104.0	2	S1E1 Serenity	Firefly/
104.1	2	S1E2 The Train Job	Firefly/
104.2	1	S1E3 Bushwhacked	Firefly/
104.4	1	S134 Shindig	Firefly/

Use a custom print format

- `-print-format <format-file>`
 - control attributes, headings, format, constraint
 - like `-autoformat` on steroids
 - `condor_status`, `condor_q`, and `condor_history`
 - Config to make it your default output
 - An "experimental" feature right now

htcondor-wiki.cs.wisc.edu/index.cgi/wiki?p=ExperimentalFeatures

Custom print format xcode.cpf

SELECT

ClusterId AS " ID" PRINTAS JOB_ID

JobStatus AS ST PRINTAS JOB_STATUS

(time()-EnteredCurrentStatus)/60.0 AS MIN PRINTF %7.2f

JobBatchName AS BATCH

SourceFile AS SOURCE

RemoteHost!=undefined ? RemoteHost : "_" AS SLOT

WHERE SourceFile!=undefined

SUMMARY STANDARD

ID	ST	MIN	BATCH	SOURCE	SLOT
104.0	R	5.02	Firefly/	S1E2 The Train Job	slot1@crane

Make it your default output

- In your personal config
 - ~/.condor/user_config
 - %USERPROFILE%\condor\user_config
- Save the xcode.cpf file and add this knob

```
# PERSONAL = $ENV (HOME) / .condor
```

```
# PERSONAL = $ENV (USERPROFILE) \ .condor
```

```
Q_DEFAULT_PRINT_FORMAT_FILE=$(PERSONAL) /xcode.cpf
```

Test using a subset of jobs

```
Queue FILE from (  
    S1E1 Serenity.wmv  
    # S1E2 The Train Job.wmv  
    # S1E3 Bushwhacked.wmv  
    ...  
)
```

use a python-style slice to define a subset

```
Queue FILE matching files [:1] *.wmv
```

Even easier if you prepare

- › Put `$(slice)` in your submit file

```
Queue FILE matching files $(slice) *.wmv
```

- › Then control the slice from the command line

```
condor_submit 'slice=[:1]' firefly.sub
```

Queue N with \$CHOICE

```
size=-s $CHOICE (Step, 640x360, 800x450)
```

```
Args="-i '$Fn(file)' $(size) '$Fn(file).mp4' "
```

```
Queue 2 file from (
```

```
  S1E1 Serenity.wmv
```

```
  S1E2 The Train Job.wmv
```

```
  S1E3 Bushwhacked.wmv
```

```
  S1E4 Shindig.wmv
```

```
  ...
```

```
)
```

Questions?