STAR Scheduler – A tutorial

Lee Barnby – Kent State University

Introduction

- What is the scheduler and what are the advantages?
- What it does and what you need use it.

Examples

- Breakdown of a simple real life example.
- A more advanced example.
- Templates.

Future developments…?

What is the scheduler?
I. Background

- The scheduler is an application allowing the user to submit jobs to multiple machines using only a short description of what she/he wants to do.
- Original contributions from Claude Pruneau plus Masters student (Wayne) and current implementation by Gabriele Carcassi (BNL).
- I am giving the talk from a user perspective!

What is the scheduler
II. The basics

- The scheduler application is written in Java, however you do not need to know anything about Java to use it.
- The user's job description file is written in xml (eXtensible Markup Language)
- Does this mean you have to learn a new language…

- … no you only need a few elements of the language.
- You may already be familiar with another markup language, HTML, so the notation will seem familiar.

The advantages

- Simplifies user task of submitting to many machines in parallel.
- Allows us to use a *distributed disk model*.
- Insulates users from changes to underlying farm infrastructure and software.

Basic operation.

- Scheduler creates list of user-specified files.
- Splits into several jobs and creates a valid script to run a job.
  - Job runs on a particular machine if the files are only resident there.

What you have to provide

- A description of the files you want from the catalog OR the locations of the files
- Your job must be know what to do with a list of files rather than a simple file name
  - For the most common user application, a MuDst-based analysis, this is already taken care of.

A first example

```
<?xml version="1.0" encoding="utf-8" ?>
<job maxFilesPerProcess="17" >
 <command>root4star -b -q
    /star/u/lbarnby/Analysis/dAuTest/runMakerFromMuDst.C\(\"$FILELIST\",\"/star/u/lbarnby/swgdisk/d
    AuTest/V0AnalysisdAu_MinBias_Full.$JOBID.root\",2E5,\"\"\)</command>

 <stdout URL="file:/star/u/lbarnby/Analysis/dAuTest/V0AnalysisdAu_MinBias_Full.$JOBID.out"/>
 <stderr URL="file:/star/u/lbarnby/Analysis/dAuTest/V0AnalysisdAu_MinBias_Full.$JOBID.out"/>

 <input URL="file:/star/data16/reco/dAuMinBias/FullField/P03ia/2003/025/*.MuDst.root" />
 <input URL="file:/star/data16/reco/dAuMinBias/FullField/P03ia/2003/026/*.MuDst.root" />
 <input URL="file:/star/data16/reco/dAuMinBias/FullField/P03ia/2003/027/*.MuDst.root" />
 <input URL="file:/star/data16/reco/dAuMinBias/FullField/P03ia/2003/028/*.MuDst.root" />
 <input URL="file:/star/data16/reco/dAuMinBias/FullField/P03ia/2003/033/*.MuDst.root" />
 <input URL="file:/star/data16/reco/dAuMinBias/FullField/P03ia/2003/034/*.MuDst.root" />

</job>
```

Invoke with: “star-submit MyTemplate.xml”

File catalog(ue) example

```
<?xml version="1.0" encoding="utf-8" ?>
<job maxFilesPerProcess="25" >
  <command>root4star -b -q
    /star/u/lbarnby/Analysis/dAuTest/runMakerFromMuDst.C\(\"$FILELIST\",\"$SCRATCH/V0Anal
    ysisdAu_MinBias_Rev.$JOBID.root\",2E5,\"\"\)</command>

  <output fromScratch="*.root" toURL="file:/star/u/lbarnby/swgdisk/dAuTest/"/>

  <stdout URL="file:/star/u/lbarnby/Analysis/dAuTest/V0AnalysisdAu_MinBias_Rev.$JOBID.out"/>
  <stderr URL="file:/star/u/lbarnby/Analysis/dAuTest/V0AnalysisdAu_MinBias_Rev.$JOBID.out"/>

<input
    URL="catalog:star.bnl.gov?production=P03ia,filetype=daq_reco_MuDst,storage!=hpss,trgsetu
    pname=dAuMinBias,magscale=ReversedFullField" nFiles="all"/>

</job>
```

More advanced example

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE job [
<!ENTITY DAY "056">
<!ENTITY TRGNAME "dAuCombined">
<!ENTITY FIELD "Reversed">
]>
<job maxFilesPerProcess="25" simulateSubmission="false" >
 <command>root4star -b -q
      /star/u/lbarnby/Analysis/dAuTest/runMakerFromMuDst.C\(\"$FILELIST\",\"$SCRATCH/V0Analysis.$J
      OBID.root\",2E5,\"\"\)</command>

  <output fromScratch="*.root" toURL="file:/star/u/lbarnby/swgdisk/dAuTest/"/>
  <stdout URL="file:/star/u/lbarnby/Analysis/dAuTest/V0Analysisd_&TRGNAME;_&DAY;.$JOBID.out"/>
  <stderr URL="file:/star/u/lbarnby/Analysis/dAuTest/V0Analysisd_&TRGNAME;_&DAY;.$JOBID.out"/>

<input
      URL="catalog:star.bnl.gov?production=P03ia,filetype=daq_reco_MuDst,storage!=hpss,trgsetupname
      =&TRGNAME;,magscale=&FIELD;FullField,filename~physics_4&DAY;" nFiles="all"/>

</job>
```

A template example

```
<?xml version="1.0" encoding="utf-8" ?>
<job maxFilesPerProcess="30" simulateSubmission="false" >
 <command>starnew;root4star -b -q
    /star/u/lbarnby/Analysis/dAuFilt/macros/runMakerFromMuDst.C\(\"$FILELIST\",\"$SCRATCH/V
    0Analysis_&FIELD;_&SET;_Filt.$JOBID.root\",2E7,\"\"\)</command>

  <output fromScratch="*.root" toURL="file:/star/u/lbarnby/scratch/dAuFilt/"/>

  <stdout URL="file:/star/u/lbarnby/Analysis/dAuFilt/V0Analysis_Filt.$JOBID.out"/>
  <stderr URL="file:/star/u/lbarnby/Analysis/dAuFilt/V0Analysis_Filt.$JOBID.out"/>

<input
    URL="file:/star/data05/scratch/suire/strange/YearIII/CommonMuDst/PRODUCTION/DATA/&FI
    ELD;Field/*&SET;Set.*MuDst.root" />

</job>
```

Invoke with e.g.: “star-submit-template –template MyTemplate.xml –entities FIELD=Reversed SET=One”

Further resources

- Extensive documentation on web:

http://www.star.bnl.gov/comp/Grid/scheduler/

Includes full description of options and some examples

- Also hypernews Forum “Scheduler discussions”

Recent/Future Developments

- Scheduler available at PDSF
  - Tested, works well for users if using local library setup (as explained by Iwona).
  - Minor change, need to modify URL of file file catalogue, details being worked out…
- Making scheduler work with GRID
  - Hopefully minor user changes
  - Major change in scripts produced by scheduler (not ‘bsub’ for LSF but something else)