

Central Computing at Jefferson Lab

Alexander Austregesilo

GlueX Software Tutorial
May 23, 2022



Outline

1 The Common User Environment (CUE)

- Interactive and batch nodes
- Recommendations for File Storage

2 The GlueX Software Stack

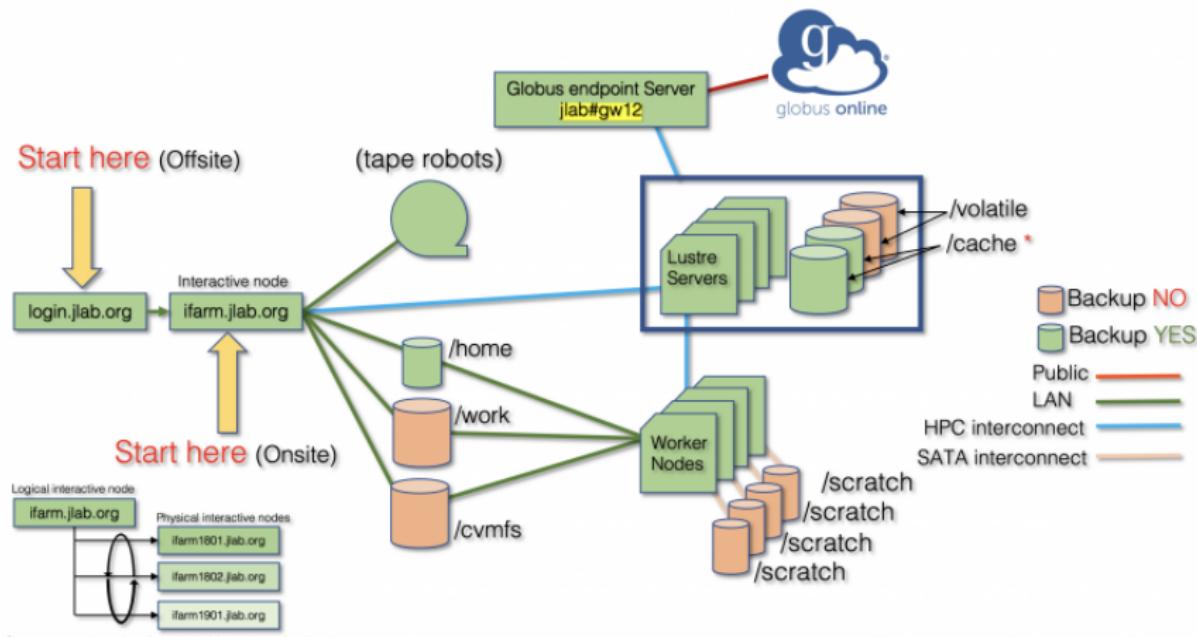
- Packages
- Default: Using the current release
- Developer mode

3 Batch Workflow Management

- SWIF2
- Batch Submission Scripts
- Example: How to run a plugin?
- Example: How to submit DSelector jobs?

4 How to Ask for Help

CUE Overview



https://scicomp.jlab.org/docs/getting_started

Computing Resources



Interactive nodes

- Login to ifarm1801, ifarm1802 or ifarm1901
- For software development and testing
- From offsite, only reachable through login.jlab.org

Batch farm

- Different nodes with 32-128 cores, 32-256BG RAM

Nodes with GPU

- 4 NVidia TitanRTX and 40 T4 cards
- Accessible thorough slurm

https://halldweb.jlab.org/wiki/index.php/HOWTO_use_AmpTools_on_the_JLab_farm_GPUs

File Storage

/home/USER

- Very small
- Use for scripts (environment, batch, ...)

/work/halld/home/USER

- Shared workspace, 400TB total
- Use for software builds
- **Do not use for analysis output**

/volatile/halld/home/USER

- Shared temporary storage, 250TB total
- Files are deleted after 6 months, possibly earlier if above quota

/cache/halld/home/USER

- "Write-through" cache, 1.5PB
- Files > 1MB migrated to tape
- Backed-up files deleted when full
- **Overwrite produces conflicts**

/mss/halld/home/USER

- Representation of tape library
- Use jcache command (next slide)

/farm_out/USER

- Optimized for batch log files

More info: <https://scicomp.jlab.org/docs/node/632>

Cache and Tape

Useful Commands



/cache/halld/home/USER

- jcache pin: keep on disk (7 days)
- jcache put: force to tape

/mss/halld/home/USER

- jcache get: cache file

More info:

<https://scicomp.jlab.org/docs/write-through-cache>

Important Packages

Active Development

- hdds: Hall D detector specifications
- halld_recon: reconstruction, monitoring, analysis
- halld_sim: generators, detector simulation, amplitudes
- hdgeant4: MC simulation
- AmpTools: amplitude analysis
- gluex_root_analysis: DSelector analysis
- gluex_MCWrapper: scripts to run all simulation
- hd_utilities: many useful scripts

All packages have repositories on github: Guide to Using Git Wiki

Dependencies

- ccdb, cernlib, diracxx, evio, evtgen, geant4, hepmc, jana, lapack, photon, rcdb, root, sqlite, sqlitecpp, xerces-c

All versions specified in xml file: `version.xml`

Default Scenario

bash and tcsh supported, this talk uses bash for all examples (.sh instead of .csh)

```
source /group/halld/Software/build_scripts/glueX_env_boot_jlab.sh
```

- Provides gxenv command and defines some directories
- Does not yet set up environment
- Add this to your .bashrc or .profile

gxenv

- Without argument: set up current default environment
Warning: default versions change over time
- Set up fixed version with xml file as argument, e.g.:
gxenv \$HALLD VERSIONS/version_5.7.1.xml
- gxclean: cleans environment

Developer Mode

\$BUILD_SCRIPTS/my_halld_build_jlab

- Use case: develop new features or debug existing software
- Starts with default version set
- Checks out local versions of (a subset of):
hdds, halld_recon, halld_sim, hdgeant4, gluex_root_analysis, amptools

\$BUILD_SCRIPTS/my_halld_update.py

- Update all private packages as identified in the xml file and build them

https://halldweb.jlab.org/docs/build_scripts_web/

Workflow Management

SWIF: Jefferson Lab's Scientific Workflow Indefatigable Factotem

- *it will work tirelessly on your behalf so that you need not expend unnecessary effort to make good use of the compute farm*
 - <https://scicomp.jlab.org/docs/swif2>

- Efficiently handles I/O with tape library
- Book-keeping, classification of errors, resubmission
- Possibility to handle job dependencies
- Submission to computing resources off-site, e.g. NERSC, PSC

Useful Commands

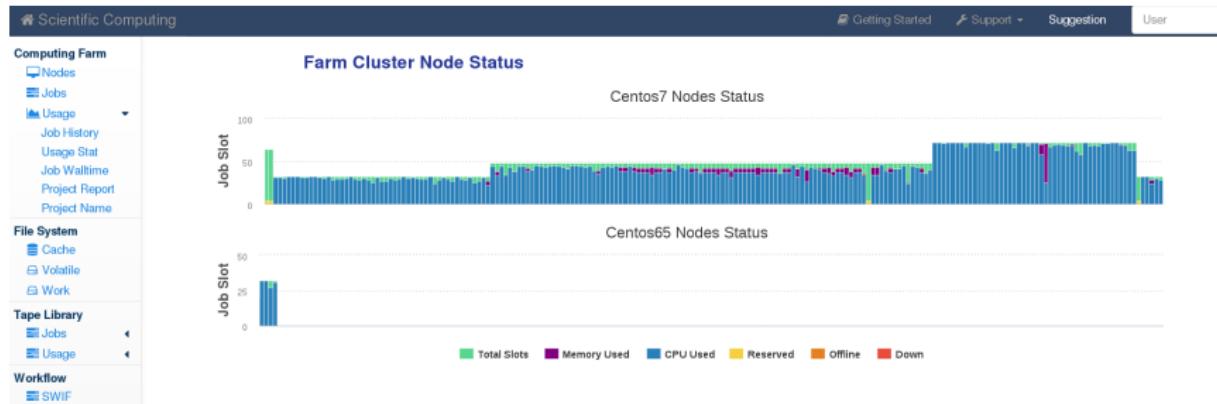
- `swif2 help` and `swif2 <command> help`
prints usage instructions
- `swif2 list (-archived)`
displays all your workflows (archived after 2 weeks)
- `swif2 status -workflow [WORKFLOW]`
shows more info about workflow, problem jobs
- `swif2 create -workflow [WORKFLOW]`
creates new workflow
- `swif2 add-job`
adds jobs to workflow, should be done with script
- `swif2 run -workflow [WORKFLOW]`
runs workflow
- `swif2 retry-jobs -workflow [WORKFLOW] -problems [PROBLEM]`
retries problems jobs

Possible problems

- `SLURM_FAILED` : your program crashed
- `SLURM_TIMEOUT`, `SLURM_OUT_OF_MEMORY` : check resources
- `SITE_LAUNCH_FAIL`, `SWIF_INPUT_FAIL`, `SWIF_SYSTEM_ERROR` : just retry
- `swif2 cancel -delete -workflow [WORKFLOW]`
deletes workflow

Scientific Computing

<https://scicomp.jlab.org/>



- Useful information on
 - Batch jobs and SWIF2 (more details coming soon)
 - File system and tape library
 - Usage statistics

Scientific Computing



<https://scicomp.jlab.org/>

Scientific Computing

Getting Started Support Suggestion aaustreg

Outstanding (Pending/Active) Batch Farm Jobs

User	Org	Depend	Pending	PbsPending	Stagein	Running	StageOut	Total
aaustreg	halld	0	1	0	0	0	0	1
acernst	halld	188	0	22	0	673	0	883
clas12-2	clas12	0	0	0	0	9	0	9
efuchey	halla	0	2,383	372	0	175	0	2,930
ellie	halla	6	0	303	0	0	0	309
igorko	clas	2,493	0	349	0	0	0	2,842
jhoekins	eic	0	0	0	0	10	0	10
jzarlino	halld	0	740	325	0	51	0	1,116
kageya	clas	0	0	0	0	1	0	1
mkamel	halld	331	0	95	0	50	0	476
morozov	casa	0	0	1	0	0	0	1
obrecht	halla	1	69,051	400	0	414	0	69,866
randika	casa	0	0	0	0	41	0	41
roark	clas	0	678	1	0	14	0	693
rradloff	halld	0	0	0	0	13	0	13
scole	halld	2,338	0	276	0	323	0	2,937
shankar	clas	0	88	0	0	16	0	104
tylern	clas12	0	0	0	0	1	0	1
xiongw	halld	0	3,148	304	0	882	0	4,334
		5,357	76,089	2,448	0	2,673	0	86,567

- User specific information by “log-in” in the box on the top right

GlueX Scripts



https://github.com/JeffersonLab/hd_utilities/tree/master/launch_scripts/launch

https://github.com/JeffersonLab/hd_utilities/tree/master/launch_scripts/root_analysis

- Easy-to-use framework
- In use for official production and analysis launches
- Can also be used for DSelector analysis
- Actively supported

- `launch.py` : generic python program to submit jobs to workflow
- `script.sh` : bash scripts handles I/O and running on farm node
- `jobs.config` : steered with simple config file, **only user modification here!**

Usage: `launch.py <job.config> <minrun> <maxrun>`

halld_recon on the farm

jobs.config file



```
/group/halld/Software/workshops/tutorial_2022/session2a/launch/jobs_analysis.config
```

```

# SCI COMP JOB ACCOUNTING
PROJECT          gluex           # http://scicomp.jlab.org/scicomp/#/projects
TRACK            analysis         # https://scicomp.jlab.org/docs/batch_job_tracks
OS               centos7

# JOB RESOURCES
NCORES           12
DISK             10GB
RAM              10GB
TIMELIMIT        4hrs

# WORKFLOW DEFINITION
# RUNPERIOD, VERSION: Not needed by launch.py: Only used to replace variables in this config
RUNPERIOD        2017-01
VERSION          05
WORKFLOW         analysis_[RUNPERIOD]_ver[VERSION]

# JOB, SCRIPT CONTROL
ENVFILE          /group/halld/Software/workshops/tutorial_2018/session1a/env_t18.sh
SCRIPTFILE        /group/halld/Software/workshops/tutorial_2018/session1d/launch/script.sh
RDB_QUERY         "@is_production and @status_approved"      # comment-out for all runs
JANA_CONFIG       /group/halld/Software/workshops/tutorial_2018/session1d/launch/jana_analysis.config
CACHE_PIN_DAYS    0                                         # max is 60, 0 or comment-out for none

# FILE INPUT, OUTPUT BASE DIRECTORIES
INDATA_TOPDIR    /mss/halld/RunPeriod-[RUNPERIOD]/recon/ver02/REST/

# FILE OUTPUT
OUTDIR_LARGE     /volatile/halld/home/tutorial/RunPeriod-[RUNPERIOD]/analysis/ver[VERSION]
OUTDIR_SMALL      /volatile/halld/home/tutorial/RunPeriod-[RUNPERIOD]/analysis/ver[VERSION]      # REST, skims, hists, trees
                                                               # log files

```

- RESOURCES : modify for efficient use of farm
- CACHE_PIN_DAYS : only effective if OUTDIR_LARGE on cache disk

halld_recon on the farm

Example

/group/halld/Software/workshops/tutorial_2022/session2a/launch/

1 Set up workflow:

```
swif create bggen_2017-01_ver03_batch01
```

Name has to match entry in jobs.config exactly

2 Register jobs:

```
./launch.py jobs_analysis_bggen.config 30274 31057 -f '00[0-4]',
```

with runs in range 30274 - 31057 that fulfil RCDB query

-f : optional, only first five files in this example

3 Run workflow:

```
swif run -workflow bggen_2017-01_ver03_batch01
```

4 Periodically check workflow:

- swif2 list and swif2 status bggen_2017-01_ver03_batch01
- <https://scicomp.jlab.org/>
- output folder

To avoid interference, copy config file and edit USER

DSelector on the farm config file



/group/halld/Software/workshops/tutorial_2022/session2a/root_analysis/jobs_root_analysis.config

```

# SCICOMP JOB ACCOUNTING
PROJECT          glueX           # http://scicomp.jlab.org/scicomp/#/projects
TRACK           analysis         # https://scicomp.jlab.org/docs/batch_job_tracks
OS              centos7

# JOB RESOURCES
NCORES          6
DISK            50GB
RAM             10GB
TIMELIMIT       1hrs

# WORKFLOW DEFINITION
RUNPERIOD      2017-01
WORKFLOW        analysis_[RUNPERIOD]_omega_3pi           # MUST start with a letter!

# JOB, SCRIPT CONTROL
ENVFILE         /group/halld/Software/workshops/tutorial_2018/session1a/env_t18.sh
SCRIPTFILE      /group/halld/Software/workshops/tutorial_2018/session1d/root_analysis/script.sh
#CACHE_PIN_DAYS 21                                     # max is 60, 0 or comment-out for none

# ROOT CONFIG
ROOT_SCRIPT     /group/halld/Software/workshops/tutorial_2018/session1d/root_analysis/Run_Selector.C
TREE_NAME       pi0pi0im_B4_Tree
SELECTOR_NAME   /group/halld/Software/workshops/tutorial_2018/session1c/DSelector_omega_3pi      # Leave .C off of the end!
                                                # Leave .C off of the end!

# FILE INPUT, OUTPUT BASE DIRECTORIES
INDATA_TOPDIR   /cache/halld/RunPeriod-2017-01/analysis/ver12/tree_pi0pi0im_B4/merged/

# FILE OUTPUT
OUTDIR_LARGE    /volatile/halld/home/tutorial/RunPeriod-[RUNPERIOD]/DSelector/omega/      # REST, hist
OUTDIR_SMALL    /volatile/halld/home/tutorial/RunPeriod-[RUNPERIOD]/DSelector/omega/      # log

```

- TREE_NAME : name of tree in input file
- SELECTOR_NAME : your DSelector (without .C)

DSelector on the farm

Example

/group/halld/Software/workshops/tutorial_2022/session2a/root_analysis/

- 1 Set up workflow:

```
swif2 create dselector_2017-01_ver52_batch01
```

Name has to match entry in jobs.config exactly

- 2 Register jobs (same executable as before):

```
.. /launch/launch.py jobs.root_analysis.config 30274 31057
```

with runs in range 30274 - 31057 that fulfil RCDB query

- 3 Run workflow:

```
swif2 run -workflow dselector_2017-01_ver52_batch01
```

- 4 Periodically check workflow:

- swif2 list and swif2 status dselector_2017-01_ver52_batch01
- <https://scicomp.jlab.org/>
- output folder

To avoid interference, copy config file and edit USER

How to Ask for Help

- Useful wiki pages: Getting Started at Gluex, Software Overview, Offline Software, Offline HOWTO and Analysis HOWTO
- Computer Center, SciComp and Service Now
- halld-offline email list and gluex software google group
- slack, open analysis discussion meetings

Report inconsistencies, contribute to the documentation efforts!