# Harvester: non-HEP experiments

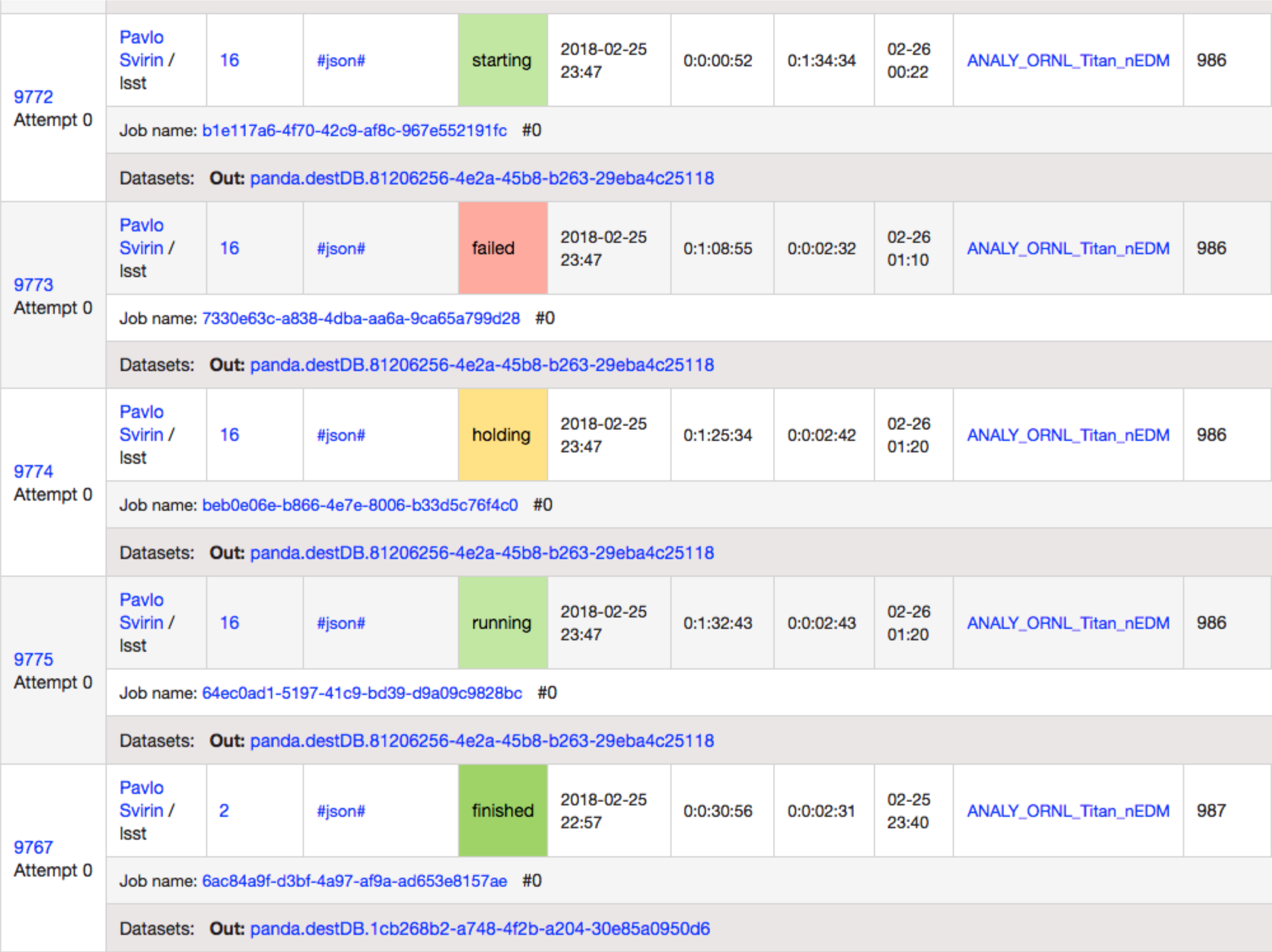
Pavlo Svirin, Sergey Panitkin

BigPanDA TIM, April 26, 2018, BNL

# pandawms.org update

- PanDA Server has been upgraded to a latest version
- Database has been updated with tables necessary for support of Harvester
- Some queries originally written for Oracle were ported to MySQL
- job submission works correctly, correct communication with Harvester (jobs fetching, status update)
- tested for compatibility with plain old pilot launcher (tested with LQCD jobs on BNL Institutional Cluster)

# nEDM challenge fragment



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 9772<br>Attempt 0 | Pavlo Svirin / lsst | 16 | #json# | starting | 2018-02-25 23:47 | 0:0:00:52 | 0:1:34:34 | 02-26 00:22 | ANALY_ORNL_Titan_nEDM | 986 |
| | Job name: b1e117a6-4f70-42c9-af8c-967e552191fc #0 | | | | | | | | | |
| | Datasets: **Out:** panda.destDB.81206256-4e2a-45b8-b263-29eba4c25118 | | | | | | | | | |
| 9773<br>Attempt 0 | Pavlo Svirin / lsst | 16 | #json# | failed | 2018-02-25 23:47 | 0:1:08:55 | 0:0:02:32 | 02-26 01:10 | ANALY_ORNL_Titan_nEDM | 986 |
| | Job name: 7330e63c-a838-4dba-aa6a-9ca65a799d28 #0 | | | | | | | | | |
| | Datasets: **Out:** panda.destDB.81206256-4e2a-45b8-b263-29eba4c25118 | | | | | | | | | |
| 9774<br>Attempt 0 | Pavlo Svirin / lsst | 16 | #json# | holding | 2018-02-25 23:47 | 0:1:25:34 | 0:0:02:42 | 02-26 01:20 | ANALY_ORNL_Titan_nEDM | 986 |
| | Job name: beb0e06e-b866-4e7e-8006-b33d5c76f4c0 #0 | | | | | | | | | |
| | Datasets: **Out:** panda.destDB.81206256-4e2a-45b8-b263-29eba4c25118 | | | | | | | | | |
| 9775<br>Attempt 0 | Pavlo Svirin / lsst | 16 | #json# | running | 2018-02-25 23:47 | 0:1:32:43 | 0:0:02:43 | 02-26 01:20 | ANALY_ORNL_Titan_nEDM | 986 |
| | Job name: 64ec0ad1-5197-41c9-bd39-d9a09c9828bc #0 | | | | | | | | | |
| | Datasets: **Out:** panda.destDB.81206256-4e2a-45b8-b263-29eba4c25118 | | | | | | | | | |
| 9767<br>Attempt 0 | Pavlo Svirin / lsst | 2 | #json# | finished | 2018-02-25 22:57 | 0:0:30:56 | 0:0:02:31 | 02-25 23:40 | ANALY_ORNL_Titan_nEDM | 987 |
| | Job name: 6ac84a9f-d3bf-4a97-af9a-ad653e8157ae #0 | | | | | | | | | |
| | Datasets: **Out:** panda.destDB.1cb268b2-a748-4f2b-a204-30e85a0950d6 | | | | | | | | | |

# Installation script for Harvester

- It takes ~2 mins to deploy and provide a simple configuration for Harvester with one PanDA queue:

  ```
  ./install-harvester.sh -d ~/harvesters -h TJLab -q ANALY-TJLAB-LQCD -b torque -p /tmp/proxy -c /etc/grid-security/certificates
  ```

- Uses latest Harvester from “OLCF_validation” branch
- Uses updated saga_monitor and saga_submitter modules
- Some manual intervention still needed to “submitter” section to tune queue/projectname/etc.
- In case of installation on machine without access to superuser:
  - Steps described how to compile a personal installation for python 2.7.14, pip, virtualenv, sqlite3, curl 7.58.0

# panda_harvester.cfg template

```
[master]
uname = ${USERNAME}
gname = ${GROUPNAME}
loggername = harvester
harvester_id=${HARVESTERID}

[db]
database_filename = ${BASE_DIR}/var/harvester/test.db
verbose = False
nConnections = 5
# database engine : sqlite or mariadb
engine = sqlite
# user name
user = harvester
# password
password = harvester@olcf
# schema
schema = HARVESTER
```

```
[pandacon]
nConnections = 5
timeout = 180

ca_cert = ${PATH_TO_CERTIFICATES}
cert_file = ${PATH_TO_PROXY}
key_file = ${PATH_TO_PROXY}
pandaURL = http://pandawms.org:25080/server/panda
pandaURLSSL = https://pandawms.org:25443/server/panda
pandaURLProxy = http://pandawms.org:25080/server/panda
verbose = True

[qconf]
configFile = ${BASE_DIR}/etc/panda/EC2_queueconfig.json

queueList =
        ${QUEUENAME}
```

# Queueconfig template

```
{
    "${QUEUENAME}": {
        "prodSourceLabel":"user",
        "nQueueLimitJob":5,
        "nQueueLimitWorker":5,
        "walltimeLimit" : 10,
        "maxWorkers":5,
        "mapType":"OneToOne",
        "preparator":{
            "name":"RseDirectPreparator",
            "module":"pandaharvester.harvesterpreparator.rse_direct_preparator",
            "basePath":"${TOP_DIR}/harvester-preparator"
        },
        "submitter":{
            "name":"SAGAYAMLSubmitter",
            "module":"pandaharvester.harvestersubmitter.saga_yaml_submitter",
            "nCorePerNode": 16,
            "adaptor": "${BATCH_SYSTEM}://localhost",
            "localqueue": "debug",
            "projectname": "lqcd17q1"
        }, .....
```

# Harvester file hierarchy

```
|-- /home/user_home/harvesters
    |-- harvester-messenger
    |-- harvester-preparator
    |-- harvester-worker-maker
    |-- harvester-TJLab
        |-- bin
        |-- etc
        |-- include
        |-- lib
        |-- lib64 -> lib
        |-- share
        |-- start_harvester.sh
        |-- stop_harvester.sh
        |-- clean_logs.sh
        `-- var
            |-- harvester
            |   `-- test.db
            `-- log
                `-- panda
```

# Job descriptions in YAML

- we introduced job description in Yet Another Markup Language (YAML), so users don’t have any need to do Python programming

- LQCD and LSST expressed interest in this format

```
seqname: Test_Seq_LQCD

jobs:
    JOB1:
        walltime: "20:00:00"
        nodes: “8000"
        command: |+
          export PMI_NO_FORK=1
          export CRAY_CUDA_MPS=1

          cd $PBS_O_WORKDIR

          conf_num=0

          for i in {0..31}; do
           aprun –n 12 –N 1 ./wrapper1.sh \
                 $((i*12+conf_num*396)) &
           sleep 2s
          done
    JOB2:
       …

sequence:
  JOB1 : JOB2
```

# User environment

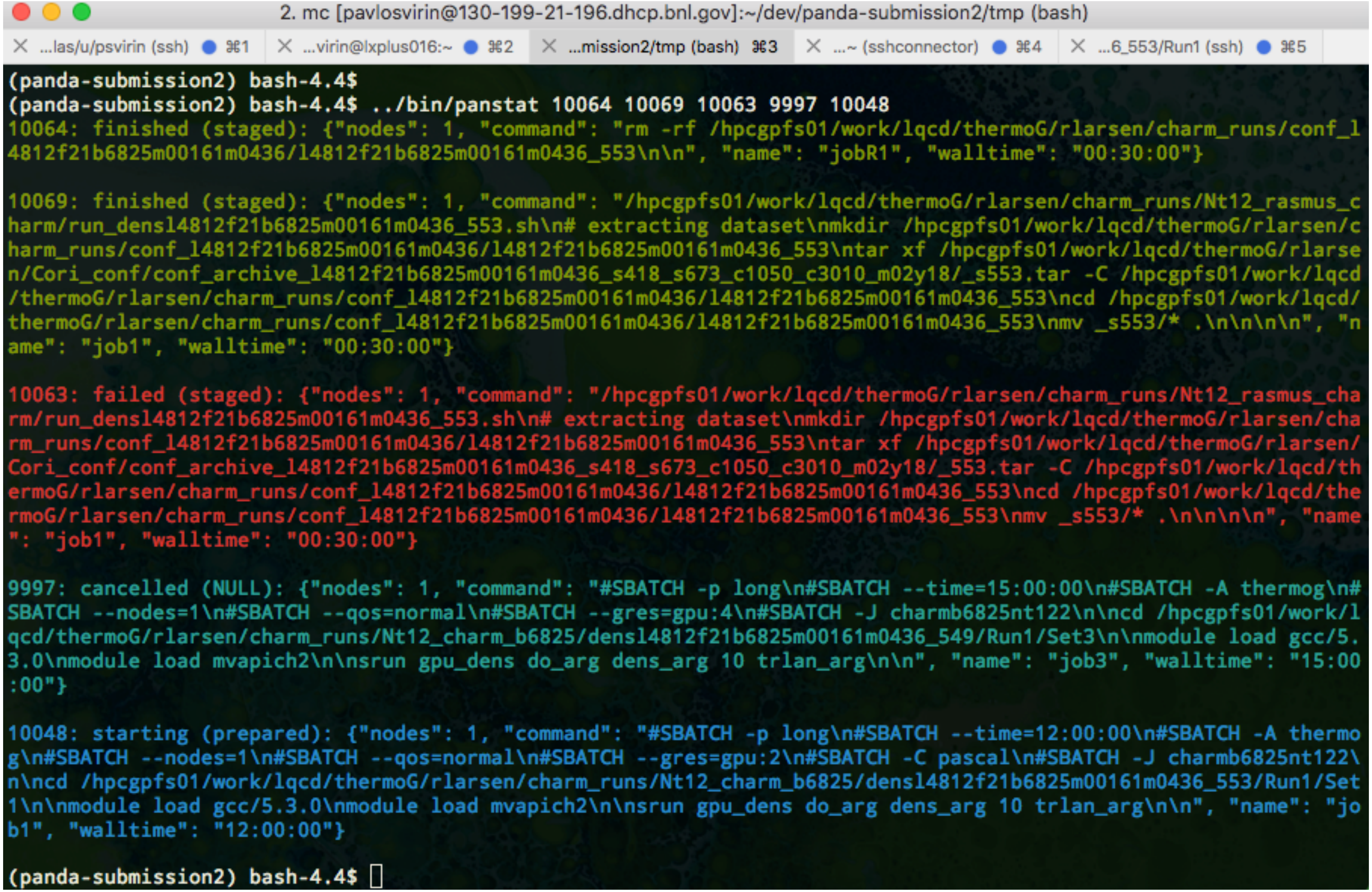
- Simple user environment developed, independent from experiment

- No programming needed to control or define jobs

- Offers basic job control: pansub, panstat, pankill, panretry

    - Also includes a tool to spawn multiple jobs from template

# User environment



2. mc [pavlosvirin@130-199-21-196.dhcp.bnl.gov]:~/dev/panda-submission2/tmp (bash)
...las/u/psvirin (ssh) ⌘1
...virin@lxplus016:~ ⌘2
...mission2/tmp (bash) ⌘3
...~ (sshconnector) ⌘4
...6_553/Run1 (ssh) ⌘5

```
(panda-submission2) bash-4.4$
(panda-submission2) bash-4.4$ ../bin/panstat 10064 10069 10063 9997 10048
10064: finished (staged): {"nodes": 1, "command": "rm -rf /hpcgpfs01/work/lqcd/thermoG/rlarsen/charm_runs/conf_l
4812f21b6825m00161m0436/l4812f21b6825m00161m0436_553\n\n", "name": "jobR1", "walltime": "00:30:00"}

10069: finished (staged): {"nodes": 1, "command": "/hpcgpfs01/work/lqcd/thermoG/rlarsen/charm_runs/Nt12_rasmus_c
harm/run_densl4812f21b6825m00161m0436_553.sh\n# extracting dataset\nmkdir /hpcgpfs01/work/lqcd/thermoG/rlarsen/c
harm_runs/conf_l4812f21b6825m00161m0436/l4812f21b6825m00161m0436_553\ntar xf /hpcgpfs01/work/lqcd/thermoG/rlarse
n/Cori_conf/conf_archive_l4812f21b6825m00161m0436_s418_s673_c1050_c3010_m02y18/_s553.tar -C /hpcgpfs01/work/lqcd
/thermoG/rlarsen/charm_runs/conf_l4812f21b6825m00161m0436/l4812f21b6825m00161m0436_553\ncd /hpcgpfs01/work/lqcd/
thermoG/rlarsen/charm_runs/conf_l4812f21b6825m00161m0436/l4812f21b6825m00161m0436_553\nmv _s553/* .\n\n\n\n", "n
ame": "job1", "walltime": "00:30:00"}

10063: failed (staged): {"nodes": 1, "command": "/hpcgpfs01/work/lqcd/thermoG/rlarsen/charm_runs/Nt12_rasmus_cha
rm/run_densl4812f21b6825m00161m0436_553.sh\n# extracting dataset\nmkdir /hpcgpfs01/work/lqcd/thermoG/rlarsen/cha
rm_runs/conf_l4812f21b6825m00161m0436/l4812f21b6825m00161m0436_553\ntar xf /hpcgpfs01/work/lqcd/thermoG/rlarsen/
Cori_conf/conf_archive_l4812f21b6825m00161m0436_s418_s673_c1050_c3010_m02y18/_553.tar -C /hpcgpfs01/work/lqcd/th
ermoG/rlarsen/charm_runs/conf_l4812f21b6825m00161m0436/l4812f21b6825m00161m0436_553\ncd /hpcgpfs01/work/lqcd/the
rmoG/rlarsen/charm_runs/conf_l4812f21b6825m00161m0436/l4812f21b6825m00161m0436_553\nmv _s553/* .\n\n\n\n", "name
": "job1", "walltime": "00:30:00"}

9997: cancelled (NULL): {"nodes": 1, "command": "#SBATCH -p long\n#SBATCH --time=15:00:00\n#SBATCH -A thermog\n#
SBATCH --nodes=1\n#SBATCH --qos=normal\n#SBATCH --gres=gpu:4\n#SBATCH -J charmb6825nt122\n\ncd /hpcgpfs01/work/l
qcd/thermoG/rlarsen/charm_runs/Nt12_charm_b6825/densl4812f21b6825m00161m0436_549/Run1/Set3\n\nmodule load gcc/5.
3.0\nmodule load mvapich2\n\nsrun gpu_dens do_arg dens_arg 10 trlan_arg\n\n", "name": "job3", "walltime": "15:00
:00"}

10048: starting (prepared): {"nodes": 1, "command": "#SBATCH -p long\n#SBATCH --time=12:00:00\n#SBATCH -A thermo
g\n#SBATCH --nodes=1\n#SBATCH --qos=normal\n#SBATCH --gres=gpu:2\n#SBATCH -C pascal\n#SBATCH -J charmb6825nt122\
n\ncd /hpcgpfs01/work/lqcd/thermoG/rlarsen/charm_runs/Nt12_charm_b6825/densl4812f21b6825m00161m0436_553/Run1/Set
1\n\nmodule load gcc/5.3.0\nmodule load mvapich2\n\nsrun gpu_dens do_arg dens_arg 10 trlan_arg\n\n", "name": "jo
b1", "walltime": "12:00:00"}

(panda-submission2) bash-4.4$
```

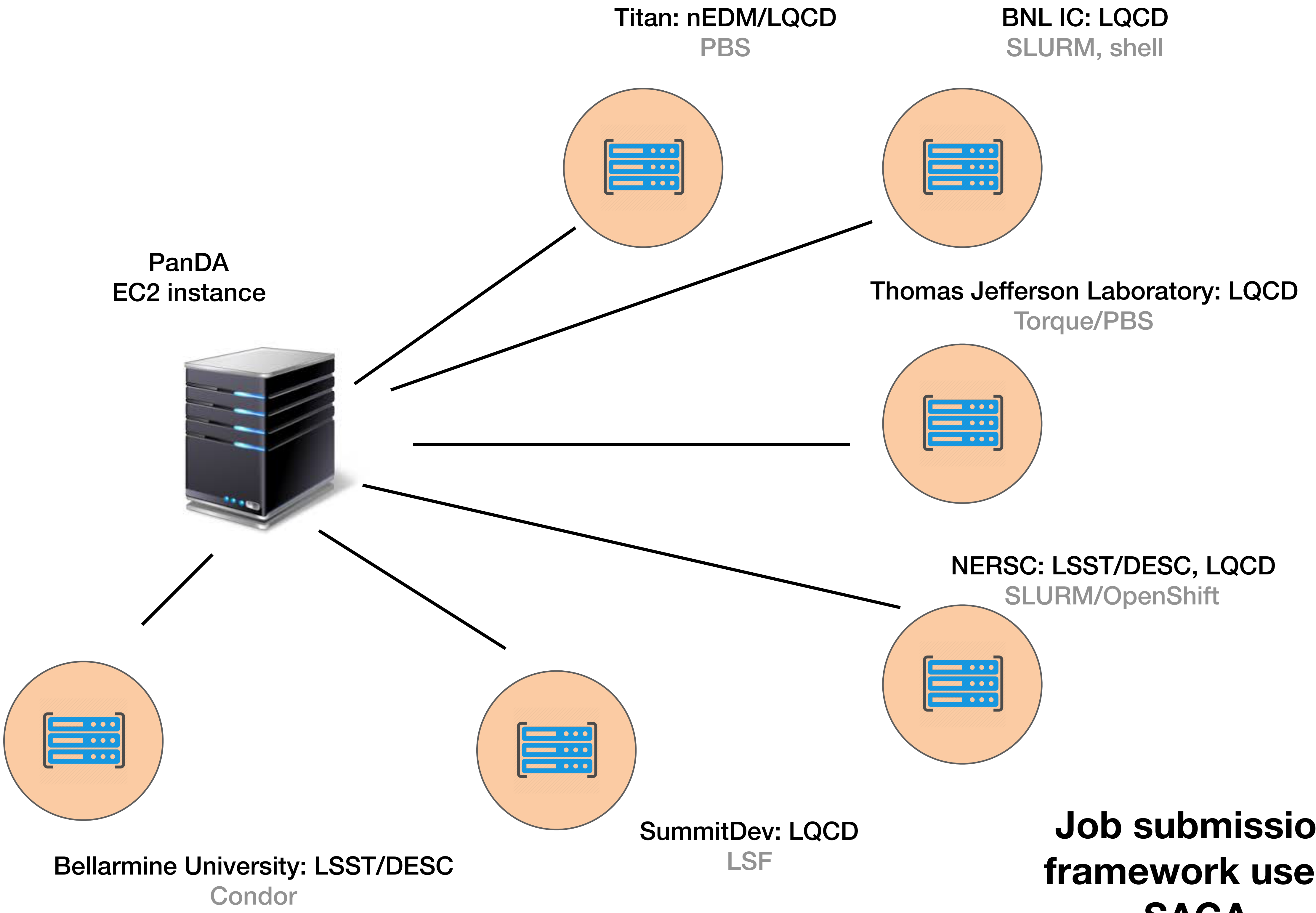
# Job templates

```
variables:
    nodes: "1"
    walltime: "00:30:00"
    om : "[0.3:0.5:0.1]"
    seed : "[1:3]"
    rafts : "['01', '02', '03', '10', '11', '12']"

command: l+
    #!/bin/bash -l
    #SBATCH --partition debug
    #SBATCH --image=docker:slosar/desc_lss:v0.21
    #SBATCH --nodes {{nodes}}
    #SBATCH --time={{walltime}}
    #SBATCH --job-name=CoLoRe_test_{{seed}}_{{om}}
    #SBATCH -C haswell
    #SBATCH --volume="/global/cscratch1/sd/psvirin/run_one_test:/predir;/global/cscratch1/sd/psvirin/
run_one_test/test0-{{seed}}:/rundir"

    export OMP_NUM_THREADS=64
    gen_config {{seed}} {{om}}
    srun -n {{nodes}} -c 64 shifter /home/lss/CoLoRe/runCoLoRe /rundir/param_files/param_colore_.cfg
```

# Map of non-ATLAS Harvesters



PanDA
EC2 instance

Titan: nEDM/LQCD
PBS

BNL IC: LQCD
SLURM, shell

Thomas Jefferson Laboratory: LQCD
Torque/PBS

NERSC: LSST/DESC, LQCD
SLURM/OpenShift

Bellarmine University: LSST/DESC
Condor

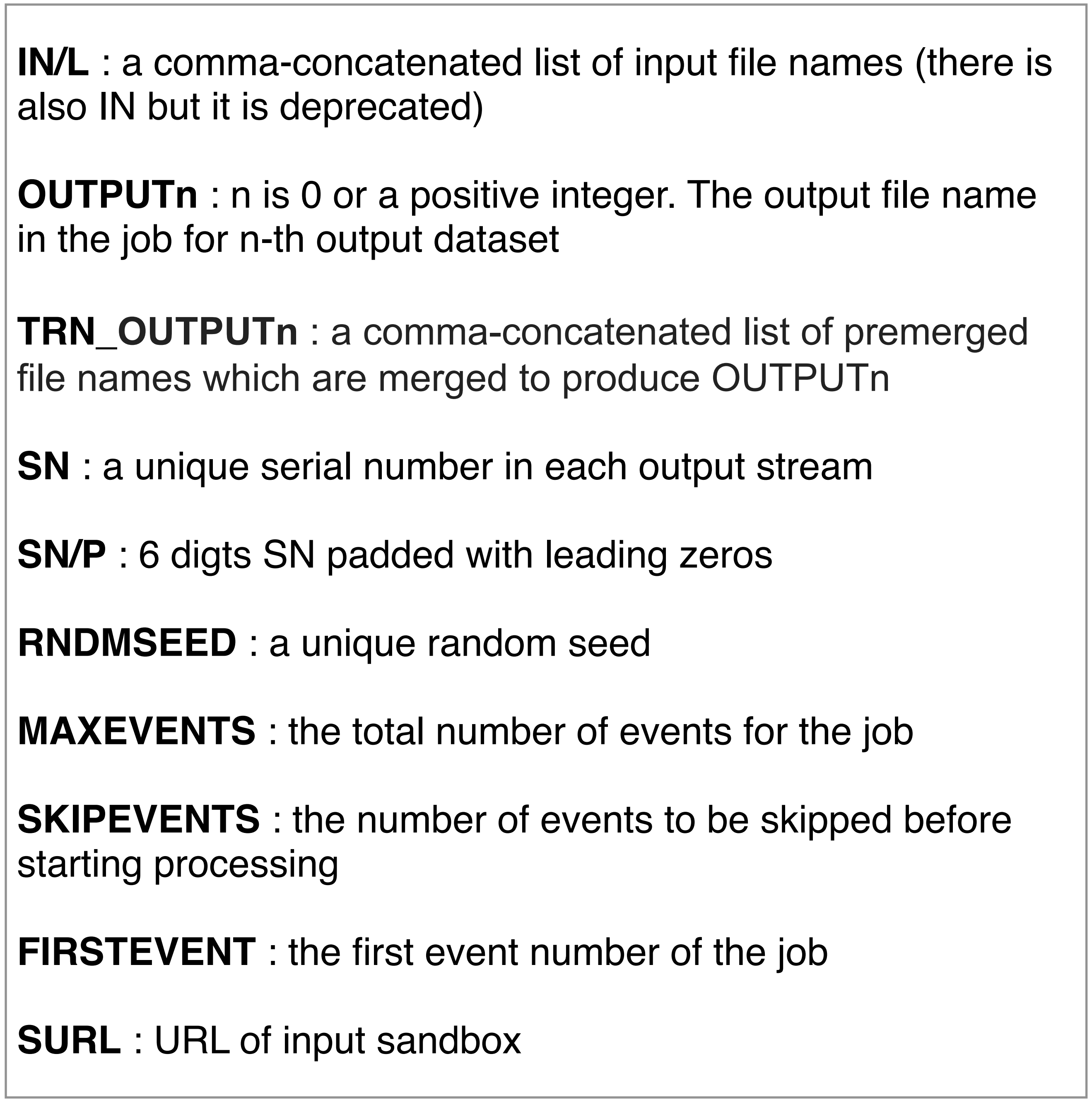
SummitDev: LQCD
LSF

**Job submission framework used: SAGA**

# JEDI

- JEDI components installed on pandavm.cern.ch instance
- Configuration was done with assistance from Ruslan Mashinistov and Alexander Novikov
- JEDI takes task description and spawns jobs into PanDA Server, jobs reach "activated" status
    - testing for the whole JEDI's workflow functionality not done yet
- JEDI also installed on pandawms.org , required python 2.7 and the latest version of MariaDB

# JEDI



**IN/L** : a comma-concatenated list of input file names (there is also IN but it is deprecated)

**OUTPUTn** : n is 0 or a positive integer. The output file name in the job for n-th output dataset

**TRN_OUTPUTn** : a comma-concatenated list of premerged file names which are merged to produce OUTPUTn

**SN** : a unique serial number in each output stream

**SN/P** : 6 digts SN padded with leading zeros

**RNDMSEED** : a unique random seed

**MAXEVENTS** : the total number of events for the job

**SKIPEVENTS** : the number of events to be skipped before starting processing

**FIRSTEVENT** : the first event number of the job

**SURL** : URL of input sandbox

- JEDI seems to be too oriented on LHC even-based experiments
  - for example, LSST does not have events
  - what if LSST runs a simulation “N rafts * M sensors * X parameters per sensor?”

# Next-generation pandawms.org

- Current pandawms.org runs Scientific Linux 6.4
- A new virtual machine with CentOS 7.4 already created and configured in Amazon Cloud
- Database already configured and data transferred from old pandawms.org
- PanDA Server and PanDA Monitor will be run in Docker containers developed by Ruslan, can be possible to run production version of PanDA Server together with an experimental one
- Transfer of domain names expected after we complete LQCD production campaign (mid-May)

# Summary

- 4 instances of Harvester configured and ready to use for non-ATLAS experiment
- An installer has been developed which will allow a fast installation for people not involved in Harvester development
- Harvester tested with nEDM, LQCD, LSST, also testing performed with Next Generation Executer (NGE)
- Production runs started with payloads from BNL LQCD team, expecting to finish this run next week

# Next steps

- Finish Harvester testing with the first LQCD production at BNL
- Support TJLab team  instance into production mode
- Finish testing client tools and provide user documentation
- Test Harvester with Summit