PROOF tutorial
PoD Basics

Gerardo Ganis, CERN, PH-SFT
gerardo.ganis@cern.ch

PoD – Proof On Demand

- Tool-set to setup PROOF on any Resource Management System
- Developed at GSI, Darmstadt, Germany
  - Author: Anar Manafov (A.Manafov@gsi.de)
- Documented at http://pod.gsi.de/
- Plug-in based interaction with RMS
  - gLite, LSF, PBS, OGE, Condor, LoadLeveler
  - Special SSH plug-in for pwd-less SSH connected clusters (e.g clouds ...)
- We will learn out to setup PoD with LSF@CERN

PoD terminology

- Client
  - Client machine, RMS unaware
  - Requires: ROOT; Pod for remote control only
- Server or PoD server
  - Master node, RMS-aware
  - Requires: ROOT, PoD
- Workers
  - Nodes chosen by the RMS for being workers
  - Require: ROOT

In examples ...

- Client
  - Your laptop or desktop
- Server
  - Any lxplus.cern.ch machine
    - Can be you desktop if enabled to submit to LSF@CERN
  - ATLAS UI enabled to submit via gLite
    - E.g atlasui.lnf.infn.it
- Workers
  - LSF lxbatch.cern.ch pool
  - Tier2 resources

Remarks

- PoD installation strictly required only on the server machine
  - E.g. lxplus.cern.ch or atlasui.lnf.infn.it
- PoD allows remote control of the server
  - Requires installation on the client machine, i.e. your laptop or desktop

PoD w/ LSF @lxplus.cern.ch

- Installation and configuration
- Server start-up
- Worker submission
- Remote control

Installing PoD

NB: very complete documentation at
http://pod.gsi.de/doc/nightly/index.html

- Logging on lxplus.cern.ch

```
mylap:~ $ ssh -Y lxplus.cern.ch
ganis@lxplus.cern.ch's password:
...
lxplus:~ $
```

- Create a build area, *e.g local/pod*, and an area for source tarballs, *e.g. local/pod/tar*; download the latest (or the wanted) source tarball

```
lxplus:~ $ mkdir -p local/pod/tar
lxplus:~ $ cd local/pod/tar
lxplus:~ $ wget http://pod.gsi.de/releases/pod/nightly/PoD-3.11-Source.tar.gz
```

Building the source

- Untar the source, create the build directory and configure the build

```
lxplus:~ $ cd ..; tar xzf tar/PoD-3.11-Source.tar.gz
lxplus:~ $ mkdir -p Pod-3.11-Source/build; cd Pod-3.11-Source/build
lxplus:~ $ cmake -C ../BuildSetup.cmake ..
```

- Build and install (if no errors occurred ...)

```
lxplus:~ $ make
lxplus:~ $ make install
```

- PoD is installed by default in $ROOTSYS/PoD

```
lxplus:~ $ ls ~/PoD
total 3
drwxr-xr-x 7 gganis zp 2048 Apr  2 08:44 3.11
```

Enable PoD

- To enable PoD, we need to source the file PoD_env.sh under the installation directory

```
lxplus:~ $ source ~/PoD/3.11/PoD_env.sh
lxplus:~ $ which pod-info
~/PoD/3.11/bin/pod-info
```

- The command 'pod-info' is one of the commands to operate PoD
- Next step is to start the PoD server, but for that we need to make sure we have ROOT
- On lxplus.cern.ch, ROOT is available under AFS

/afs/cern.ch/sw/lcg/app/releases/ROOT

Enable ROOT

- Easy, just source <root_path>/bin/thisroot.sh

```
lxplus:~ $ source /afs/cern.ch/sw/lcg/app/releases/ROOT/ \
  5.32.02/x86_64-slc5-gcc43-opt/root/bin/thisroot.sh
```

- But ...
  - On lxplus need to change the default compiler

```
lxplus:~ $ source /afs/cern.ch/sw/lcg/contrib/gcc/4.3/ \
   x86_64-slc5/setup.sh
```

  - ROOT >= 5.32 requires an external XROOTD
    - source <root_path>/bin/setxrd.sh <xrootd_on_afs>

```
lxplus:~ $ source /afs/cern.ch/sw/lcg/app/releases/ROOT/ \
  5.32.02/x86_64-slc5-gcc43-opt/root/bin/setxrd.sh \
  /afs/cern.ch/sw/lcg/external/xrootd/3.1.0p2/x86_64-slc5-gcc43-opt
Using XRD at /afs/cern.ch/sw/lcg/external/xrootd/3.1.0p2/x86_64-slc5-gcc43-opt
```

Ready to start the PoD server

- The PoD server is managed by *pod-server*:

```
lxplus:~ $ pod-server -h
Usage: pod-server {start|restart|stop|status|getbins}
```

  - See also http://pod.gsi.de/doc/nightly/pod-server.html

- Just start it

```
lxplus:~ $ pod-server start
Starting PoD server...
updating xproofd configuration file...
starting xproofd...
starting PoD agent...
preparing PoD worker package...
select user defined environment script to be added to worker package...
selecting pre-compiled bins to be added to worker package...
PoD worker package will be repacked because "/tmp/gganis/etc/xpd.cf" was
updated
PoD worker package: /afs/cern.ch/user/g/gganis/.PoD/wrk/pod-worker
------------------------
XPROOFD [9391] port: 21001
PoD agent [9414] port: 22001
PROOF connection string: gganis@lxplus257.cern.ch:21001
------------------------
```

Test the master connection

- The PoD server (i.e. the master) is now up: we can test the connection, e.g from your laptop

```
mylap:~ $ root -l
root [0] TProof *p = TProof::Open("gganis@lxplus257.cern.ch:21001",
                                  "masteronly")
Starting master: opening connection ...
Starting master: OK
PROOF set to sequential mode
(class TProof*)0x10188e400
root [1]
```

  - The option “masteronly” starts the master only
  - Try `p->Print()` to check the settings
- The connection is OK: we need some workers ...

pod-submit

- To have workers we need to *submit* jobs to the RMS: the pod-submit command is done for that

```
lxplus:~ $ pod-submit -h
Usage: pod-submit [OPTION]
     -l           Show all available RMS plug-ins.
     -r rms       Name of the resource management system (supported:
condor, ge, glite, loadleveler, lsf, pbs).
     -q name      Submit the jobs to specified queue.
     -n X         Desired number or PROOF workers.
     -h           Show summary of options.

Report bugs to http://pod.gsi.de
```

- In our case the RMS if 'lsf' and for testing we use the queue '8nm', 10 workers

```
lxplus:~ $ pod-submit -r lsf -q 8nm -n 10
Job <230454960> is submitted to queue <8nm>
```

pod-info

- This command allows to get all the relevant information about the status of the server and of the job submission.
- Several options, two more frequently used:
  - pod-info -c
    - Returns the connection string

```
lxplus:~ $ pod-info -c
gganis@lxplus257.cern.ch:21001
```

  - pod-info -n
    - Returns the number of workers up and running

```
lxplus:~ $ pod-info -n
6
```

Now the Pod cluster is ready

- Some PoD worker daemons are now up: we can connect and start our PROOF work

```
mylap:~ $ root -l
root [0] TProof *p = TProof::Open("gganis@lxplus257.cern.ch:21001")
Starting master: opening connection ...
Starting master: OK
Opening connections to workers: OK (10 workers)
Setting up worker servers: OK (10 workers)
PROOF set to parallel mode (10 workers)
(class TProof*)0x101866600
root [1]
```

Shutting down the cluster

- To shutdown the cluster, stop the server

```
lxplus:~ $ pod-server stop
Stopping PoD server...
Gracefully shut down PoD server process(es): 9391 9414
```

- And check the status with 'pod-info'

```
lxplus:~ $ pod-info -n
pod-info: PoD server is NOT running.
```

Controlling PoD remotely

- PoD allows also to control the server remotely, e.g. from your laptop
- The command pod-remote allows you to run all the commands which we saw so far

```
lxplus:~ $ pod-remote -h
...
Commands options:
  --start                 Start remote PoD server
  --stop                  Stop remote PoD server
  --restart               Restart remote PoD server
  --command arg           Execute arbitrary commands
```

  - See also http://pod.gsi.de/doc/nightly/pod-remote.html
- Some additional options allow to control the way the connection are open and the remote setup

Starting the remote Pod server

- Starting PoD on lxplus.cern.ch

```
mylap:~ $ pod-remote --start \
            --remote lxplus.cern.ch:~/PoD/current \
            --env-local lsf_local_setup.sh
```

  - --remote *<host>:<pod_location>*
    defines the server address and the location of PoD on the server
  - --env-local *<local_setup_script>*
    local script to be transferred and executed remotely to setup the environment
  - --env-remote *<remote_setup_script>*
    remote script to be executed remotely to setup the environment

Starting the remote server (2)

- The password is asked 3 times

```
mylap:~ $ pod-remote --start \
  --remote lxplus.cern.ch:~/PoD/current \
  --env-remote /afs/cern.ch/user/g/gganis/local/lsf_pod_setup.sh
gganis@lxplus.cern.ch's password:
** Starting remote PoD server on lxplus.cern.ch:~/PoD/current
gganis@lxplus.cern.ch's password:
gganis@lxplus.cern.ch's password:
** Server is started. Use "pod-info -sd" to check the status ...
```

to connect and to setup the relevant tunnels

```
mylap:~ $  ps ax | grep ssh
11126 s000  S      0:00.04 ssh -o StrictHostKeyChecking=no \
  -T lxplus.cern.ch
11131 s000  S      0:00.04 ssh -o StrictHostKeyChecking=no \
  -o ConnectTimeout=30 -N -L 22001:localhost:22001 lxplus.cern.ch
11136 s000  S      0:00.03 ssh -o StrictHostKeyChecking=no \
  -o ConnectTimeout=30 -N -L 21001:localhost:21001 lxplus.cern.ch
```

- Pwd-less SSH may be setup to automatize this

Submit workers with pod-remote

- Use pod-remote –command

```
mylap:~ $ pod-remote -d \
          --command "pod-submit -r lsf -q 8nm -n 10"
```

- The pod-info command is transparent

```
mylap:~ $ pod-info -c
gganis@localhost:21001
mylap:~ $ pod-info -n
10
```

PoD w/ gLite @atlasui.cern.ch

- Installation and configuration
- Server start-up
- Worker submission
- Remote control

Remember

PoD User's Guide:

http://pod.gsi.de