

Computing On Demand: Dynamic Analysis Model

Peter Malzacher, Anar Manafov (GSI Darmstadt, Germany)

Constant changes in computational infrastructure like the current interest in Clouds, imply conditions on the design of applications. We must make sure that our analysis infrastructure, including source code and supporting tools, is ready for the on demand computing era. PROOF on Demand is a unique tool set providing you with a dynamic PROOF cluster for your dynamic ROOT based analysis.

Native PROOF connections

Whenever possible, PoD setups direct connections between nodes. It results in a full functional PROOF cluster. Users get native speed and the whole range of PROOF features. To use native connections an incoming traffic must be allowed on PoD workers for a defined port. Otherwise PoD uses packet-forwarding algorithms.

Multiuser/-core environment

PoD implements automatic port mapping algorithms to properly handle cases when several users start PoD instances (servers/workers) on the same machine. PoD also automatically manages situations when multiple PoD workers are started on the same node. Private PoD instances can't disturb each other.

Easy to use

The process of installation is very simple and fully automated. PoD works out of the box. Its distribution contains preconfigured modules and everything users need to just immediately start to work with it right after the installation.



Packet-forwarding

When worker nodes are behind a firewall then PoD uses its packet-forwarding algorithms to maintain the PROOF traffic. The algorithms are very efficient, there is no speed penalty, but some PROOF functions are limited.

Remote clusters

Using a new pod-remote command users are able to control remote PoD servers. The pod-remote command implements a thin client concept and disentangles PoD UI and Server. PROOF and PoD protocols are automatically forwarded via SSH tunnels, which are created and maintained by PoD.

Different job managers

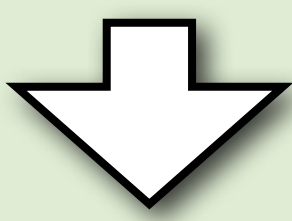
PoD is shipped with a number of plug-ins to cover all major RMSs, such as local cluster systems and Grid: SSH, LSF, PBS Pro/OpenPBS/Torque, Grid Engine (Oracle/Sun), Condor, LoadLeveler, gLite. If there is no RMS available, then the SSH plug-in can be used. The SSH plug-in is also used to setup PROOF clusters on Clouds.

3 steps to start your private PROOF cluster

START PoD SERVER

pod-server start

The host, where a user starts the PoD server, will be acting as a PROOF master. Stopping of the PoD server will trigger automatic shutdown of PoD workers.



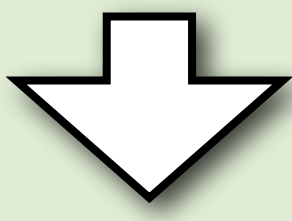
SUBMIT PoD JOBS USING PoD JOB MANAGER

pod-submit ...

or

pod-ssh ...

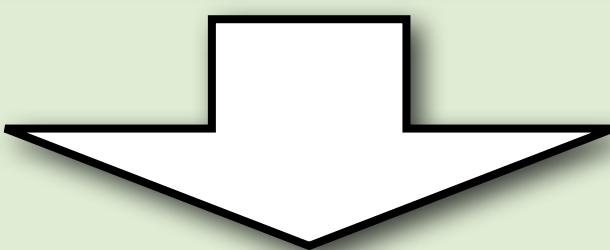
The user chooses a job manager plug-in to distribute PoD workers to a resource. It's also possible to accumulate workers from different resources in one PoD session, i.e. submit from different plug-ins.



CHECK AVAILABLE PROOF WNS

pod-info ...

PROOF workers are appearing online. The start-up time of jobs (Grid/local) ranges from a few seconds to hours depending on the chosen resource.



Your dynamic PROOF cluster is ready!

You can now connect to your cluster to process a PROOF analysis as if it would be a regular static PROOF cluster. If you close your ROOT session or an analysis crashed there is no need to resubmit PoD workers. PoD automatically reuses workers. At any time you can add more PROOF workers (step #2). PoD will automatically shutdown the cluster or a particular worker when idle for a pre-defined amount of time.



MORE INFO

PoD: <http://pod.gsi.de>

PROOF: <http://root.cern.ch/drupal/content/proof>

[Work supported by BMBF, FK 01AK802G]