PoD

PROOF on Demand

Version 2.1.1

Anar Manafov, GSI Darmstadt

pod-agent

made use of the thread pool pattern

- x14 reduced virtual memory consumption*
- 4 threads instead of 80 threads*
- x10 faster on network intensive operations

* when 4 thread in the pool and 80 WNs

PoD
PROOF on Demand

PoD idle monitor

- shuts a node down if it's idle for a defined amount of time
- is implemented for both PoD server and PoD worker
- prevents blocking of resources, which are not being used

PoD
PROOF on Demand

a helper header

Problem: PoD's automatic port mapping helps a lot in multi-user environment, but makes it difficult for a user to track new ports for XPROOF in analysis scripts.

Solution: every time PoD starts, it generates a C/C++ header file which defines the current settings of the environment.

```
#include "pod-master.h"
TProof::Open( Form("%s:%s", POD_MASTER_HOST, POD_XPROOF_PORT) );
```

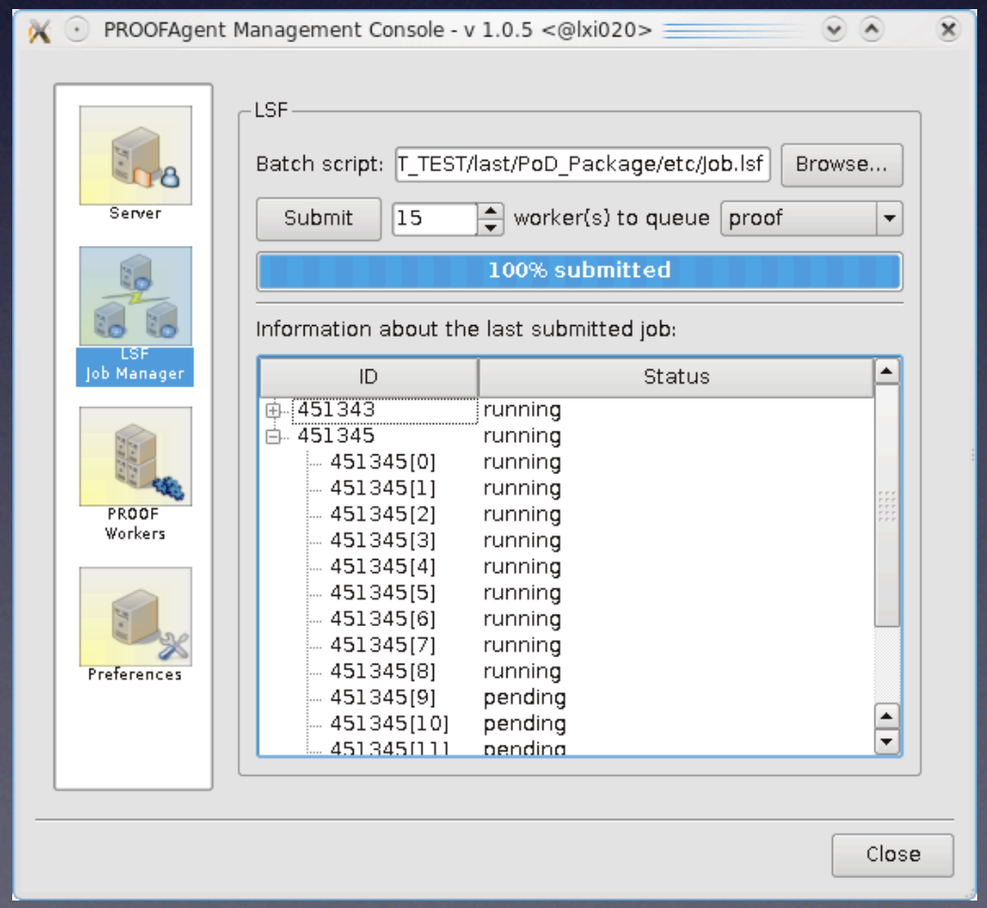
PoD user defaults

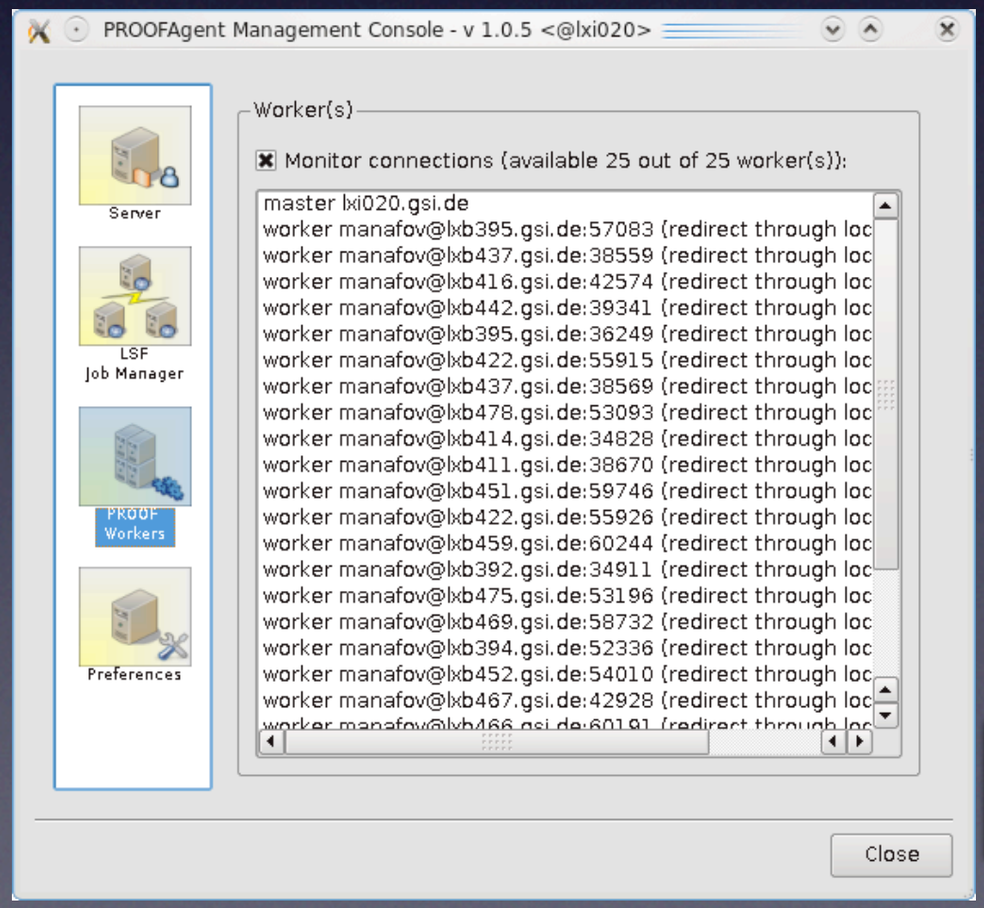
$POD_LOCATION/etc/PoD.cfg

- a value/key configuration file
- a configuration entry point for all PoD modules
- makes it easy to relocate PoD distribution

pod-console

- monitoring sensors are less aggressive now
- update only what is visible
- sleep if it's idle - currently under development





PoD
PROOF on Demand

PoD utilities

- introduced a new algorithm for the automatic port mapping
- a better handling of concurrent PoD workers or servers on a single multi-core machine
- a job wrapper script got an updated output format

PoD
PROOF on Demand

PoD vs GSI's static PROOF

Tests processed by Markus Fasel <M.Fasel@gsi.de>
This is a typical complex user analysis task.
Number of files: **10511** (1048184 events), on lustre.
Number of PoD workers: **152**
Number of GSIAF workers: **152**

	Start-up time	Library load	File validation	Init	Processing rate	Merging	Total analysis time
PoD	0m 30s*	0m 28s	0m 14s	0m 36s	1271 evt/s	5m 16s	19m 38s
GSIAF	0m 0s	0m 16s	0m 13s	0m 17s	1080 evt/s	24m 3s	39m 0s

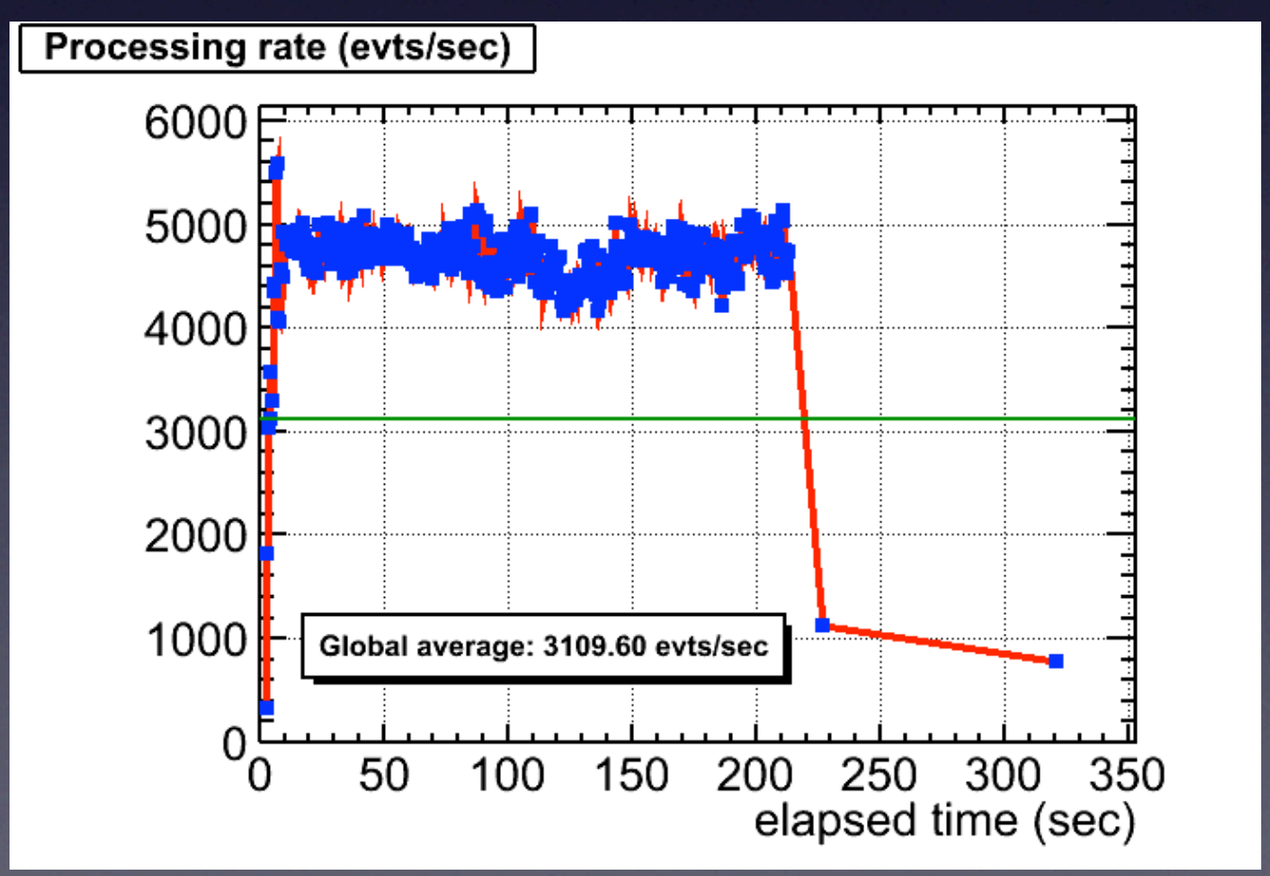
* the start-up time for jobs ranges from a few seconds to a several minutes, depending on the occupation of job queues, priorities. There is a dedicated queue in GSI for PoD, which is preemptive. When in this queue users get workers extremely fast.

PoD
PROOF on Demand

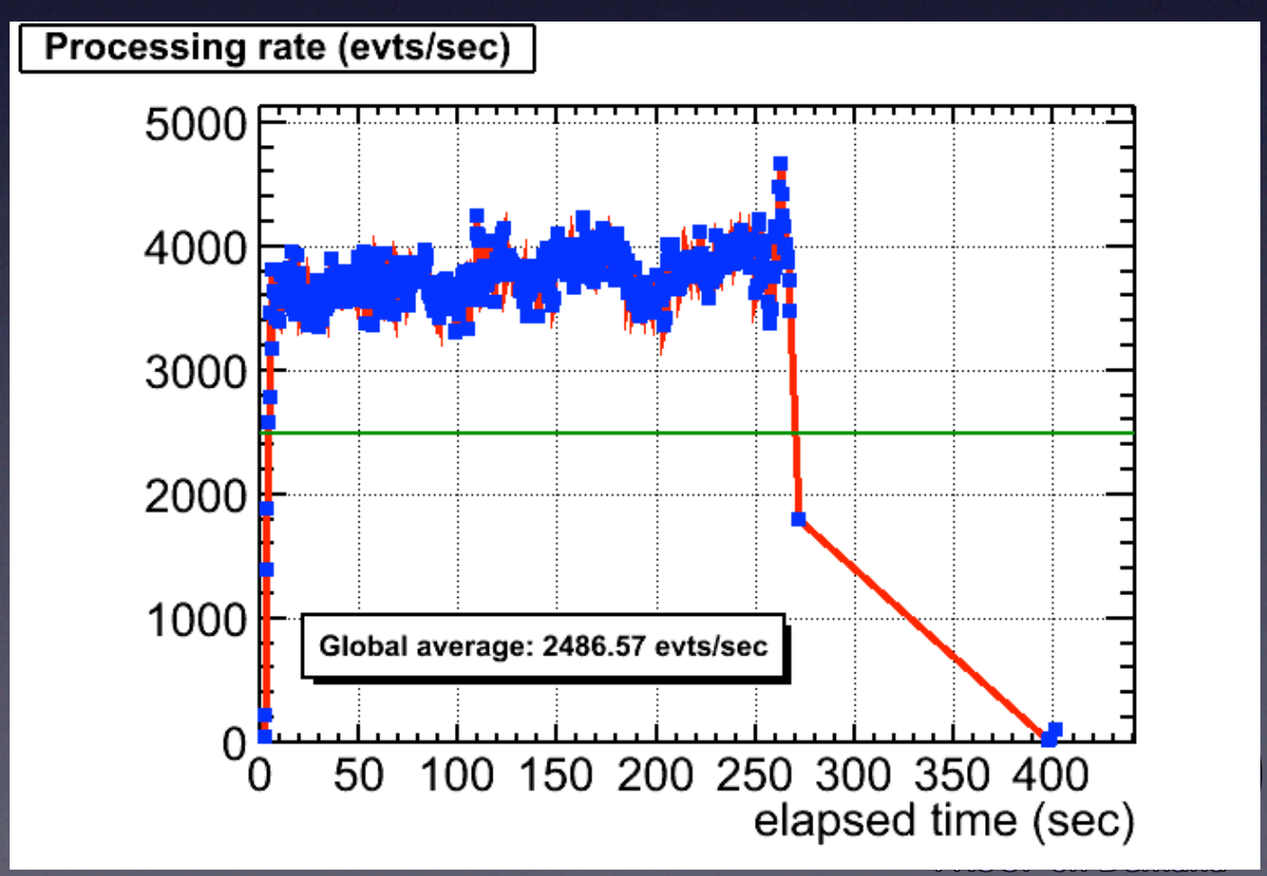
PoD vs GSI's static PROOF

Tests processed by Jacek Otwinowski <J.Otwinowski@gsi.de>
This is a Pt analysis task.
Number of files: **10000**, on lustre.
Number of PoD workers: **152**
Number of GSIAF workers: **152**

PoD



GSIAF



ToDo

- pod-console: sleep if there is no interaction
- pod-console: a slight redesign of the GUI
- pod-agent: use the packet-forwarder only if it's needed
- pod-agent: looking forward for some more speed improvements
- an SSH plug-in development

documentation: http://www-linux.gsi.de/~manafov/D-Grid/docz/
source browser: http://depc218.gsi.de:22222/git/
trac: https://subversion.gsi.de/trac/dgrid
blog: http://proof-on-demand.blogspot.com/

PoD
PROOF on Demand