

AAF status and plans

Martin Vala

July 14, 2011

Outline

- 1 AAF status
- 2 AAF problems
- 3 First look to PEAC
- 4 PEAC demo

AAF Proof clusters

Cluster						ROOT	Aggregated disk space			AF xrootd		xrootd
Name	Online	Status	Proof master	Workers	Users	Version	Total	Free	Used	Running	Latest	Version
1. CAF		Stable	alice-caf.cern.ch	114	4	v5-28-00e-1	162.3 TB	19.54 TB	142.7 TB	1.0.43	1.0.43	20100510-1509_dbg
2. JRAF		Maintenance sin...	jraf.jinr.ru	8	0	v5-28-00d	2.014 TB	438.8 GB	1.586 TB	1.0.43	1.0.43	20100510-1509_dbg
3. KIAF		Stable	kiaf.sdfarm.kr	48	0	v5-28-00e-1	20.1 TB	17.2 TB	2.897 TB	1.0.43	1.0.43	20100510-1509_dbg
4. SAF		Maintenance sin...	nansafmaster.in2p3.fr	48	0	v5-28-00e-1	12.07 TB	1.572 TB	10.5 TB	1.0.43	1.0.43	20100510-1509_dbg
5. SKAF		Stable	skaf.saske.sk	60	1	v5-28-00e-1	50.14 TB	1.346 TB	48.8 TB	1.0.43	1.0.43	20100510-1509_dbg
6. SKAF_TEST		Testing	skaf-test.saske.sk	2	0	v5-28-00e-1	815.9 GB	561.6 GB	254.3 GB	1.0.43	1.0.43	20100510-1509_dbg
7. TAF		Open to local u...	pmaster.to.infn.it	16	0	v5-28-00e-1	3.914 TB	1.254 TB	2.66 TB	1.0.42	1.0.42	20100510-1509_dbg
Total				296	5		251.3 TB	41.89 TB	209.4 TB			

AAF Software

Latest software on AAF

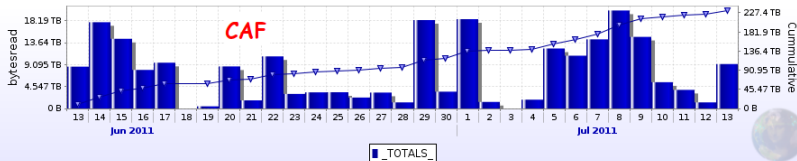
- AliEn v2-19
- ROOT 5.28.00e
- AliRoot v4-21-29-AN [sync at 5:00 and 20:00]

All packages available

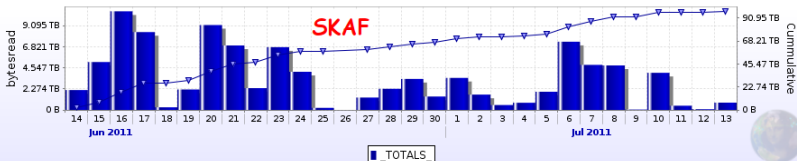
- <http://alimonitor.cern.ch/packages/>

CAF and SKAF data read

CAF



SKAF



AAF user's problems

- TProof::Open() problem
 - maybe due to AliEn packages (see demo)
- gProof->EnablePackage("VO_ALICE@AliRoot::v4-XX-YY-AN")
 - problem with sync of AliEn Packages
- Accessing OCDB (xml problem)
 - AliEn + ROOT version mismatch
- Merging of big output
 - Still waiting from PROOF dev

Upgrade of AAF setup needed

- XROOTD upgrade
 - Xrootd - 3.0.4
 - FRM scripts for better data management
 - CAF will get more then 64 workers
 - supervisors needed
- Port AAF setup for non-static proof cluster
 - Start using PoD
- Execute commands over cluster in parallel
 - PoD and custom command
- Easy setup
 - Other experiments can use it
 - Users will be able to
 - create small proof clusters (2-3 machines)
 - download files from AliEn (same as AAF)

New AAF setup : What is PEAC?

- PEAC - Proof Enabled Analysis Cluster
- Common setup (independent on experiment)
- PoD
- Xrootd SE
- MLSensor for monitoring
- Software managment

What is PEAC made of?

- PoD
 - start xproofd using ssh plugin
 - execute custom command
- xrootd SE
 - using frm scripts
 - handle more then 64 workers
- Monalisa for monitoring
 - MLSensor
 - Master will be ML service (not in ALICE case)
- afdsmgrd
 - data staging
 - dataset managment
 - data redistribution (maybe)
- xrddm
 - copy files from different sources

PEAC software management [1]

- Gentoo style
 - pbuild (PROOF build)
 - pre-defined functions
 - compile
 - download binary from external source (AliEn)
- Different pbuild for different software
 - VO_PEAC@ROOT.pbuild
 - VO_ALICE@ROOT.pbuild
 - VO_ATLAS@AtlasSoft.pbuild
 - VO_MY@MySoft.pbuild
- Software redistribution on Cluster using xrootd
 - redirector and dataserver
 - xrdcp : Extreme copy (torrent way copy)
- Dependencies are handled
- MD5 checksum of binaries and dependencies [TODO]

PEAC software management [2]

```
[aliceskaf@prf000-iep-grid PEAC]$ ./peac sw install VO_ALICE@AliRoot::v4-21-28-AN
Checking for updates [ OK ]
[VO_PEAC@Cmake::2.8.4] [ OK ]
[VO_ALICE@OpenSSL::0.9.8g] [ OK ]
[VO_ALICE@Xrootd::3.0.4] [ OK ]
[VO_ALICE@Xalienfs::1.0.12c] [ OK ]
[VO_ALICE@ROOT::v5-28-00e] [ OK ]
[VO_ALICE@Geant3::1-11-25] [ OK ]
[VO_ALICE@AliRoot::v4-21-28-AN] [ OK ]
[aliceskaf@prf000-iep-grid PEAC]$ ls /pool/PEAC/sw/sl-x86_64-4.1.2/VO_ALICE/AliRoot/
v4-21-28-AN
[aliceskaf@prf000-iep-grid PEAC]$ _
```

- Different linux distribution support
 - sl-x86_64-4.1.2
 - ubuntu-x86_64-4.4.5
 - gentoo-x86_64-4.5.2
 - and so on
- Workers can run on different linux distributions
 - Worker will compile if binary not found
 - Only one worker will compile

Demo

- TProof:Open()
 - AAF vs. PEAC
 - same cluster (CAF)
 - same PROOF setting
 - using security

Demo [2]

```
mvala@pcalike04:~$ root -b -l
root [0] .x public/testProof.C("mvala@alice-caf.cern.ch","workers=2x")
Starting master: opening connection ...
Starting master: OK
+++ Starting max 2 workers per node following the setting of PROOF_NWORKERS
Opening connections to workers: OK (114 workers)
Setting up worker servers: OK (114 workers)
PROOF set to parallel mode (114 workers)
Real time 0:00:27, CP time 0.140
root [1] .q
mvala@pcalike04:~$ root -b -l
*[[A*[[Aroot [0] .x public/testProof.C("mvala@alice-caf.cern.ch:1099","workers=2x")
Starting master: opening connection ...
Starting master: OK
+++ Starting max 2 workers per node following the setting of PROOF_NWORKERS
Opening connections to workers: OK (114 workers)
Setting up worker servers: OK (114 workers)
PROOF set to parallel mode (114 workers)
Real time 0:00:07, CP time 0.140
root [1] .q
mvala@pcalike04:~$ root -b -l
root [0] .x public/testProof.C("mvala@alice-caf.cern.ch:1099","workers=8x")
Starting master: opening connection ...
Starting master: OK
+++ Starting max 8 workers per node following the setting of PROOF_NWORKERS
Opening connections to workers: OK (456 workers)
Setting up worker servers: OK (456 workers)
PROOF set to parallel mode (456 workers)
Real time 0:00:11, CP time 0.190
root [1] _
```