Glexec: status overview (and LCAS and LCMAPS)

Gerben Venekamp & Oscar Koeroo

About

- End of life for EDG support
- Status on the preview test-beds
- OSG and glexec
- Latest developments
- SAML-XACML
- TODO

EDG: End of life (and lessons learned)

"We will no-longer support EDG-LCAS and EDG-LCMAPS and their plugins"

- **Reasons:**
  - We (developers of…) have no control over the code that is currently in production.
  - We see all kinds of complication rise from VOMS and we've made the move to FQANs two years ago.
    - LCG did not move with our newer versions and missed our move to FQANs (for the last three years)
  - It's an evolutionary process but we decided to make it official
    - It has been taken over by LCG; contact point Maarten Litmaath
  - Big bang of software… → Glite 3.1
    - *Featuring GT4 and VOMS in the FQAN format*
  - Warnings were declined because there is not enough manpower for certification in SA3

Preview test-bed

- **Not much has changed since Catania**
  - CREAM CE seems to run without any complaint @ NIKHEF
  - HIP cluster is reinstalled; failing qsub should work
- **Holidays**
  - We both had our holidays ☺
  - No time left to spend on the Preview Testbed in the past months ☹
- **NIKHEF Preview Testbed nodes will be restructured**
  - Some downtime when our local sys-admins give the *Go*
  - Currently running as 4 VMs on a dual-dualcore high-end machine
  - Will move to a lower end machine, due to usage rates and demand for the current occupied high-end machine
    - May move back when usage rates go up…

OSG and glexec

- **Glexec distribution to OSG has improved**
  - More in sync with recent version
  - Using our build infrastructure (Etics) can build their own versions
  - VDT 1.7.1 includes glexec (and LCAS+LCMAPS in setuid mode *ON*)
    - Building with Etics at OSG
    - If still required: We will send an Etics free script soon

- **Glexec code in better shape**
  - Code hardening/clean up
  - Tracking down potentially dangerous code and fixing it
  - Sudden Seg. Fault conditions experienced only at Fermilab are now gone (forever…?)

- **We're unhappy about the rpaths in all our binaries**
  - Because of our "non-GNU" build method → Etics issue

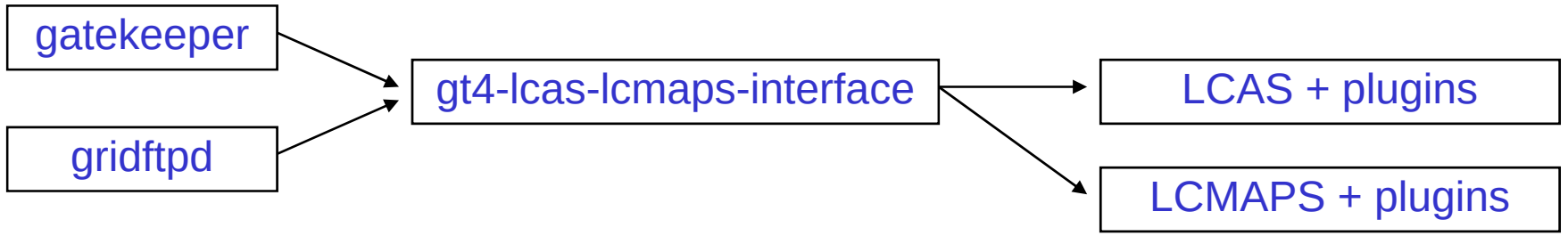
Developments

- **Added the `linger` keyword to the glexec config file**
  - it controls the fork behaviour.
  - `linger yes` is the default and makes glexec fork and wait
  - `linger no` makes glexec just exec()
  - done on request of Condor (and LCG)

- **Changed the glexec execution whitelist selection mechanism**
  - If the whiltelist contained: ‘pool*’
    - … then it would also match the Unix account ‘pooladmin’ besides ‘pool001’ to ‘pool999’
  - now ‘pool*’ matches ‘pool[0-9]*’ only
  - introduced .pool to comply with the syntax used in gridmap files
  - .pool and pool* are handled equally
  - preference of .pool over pool* (might get deprecated)

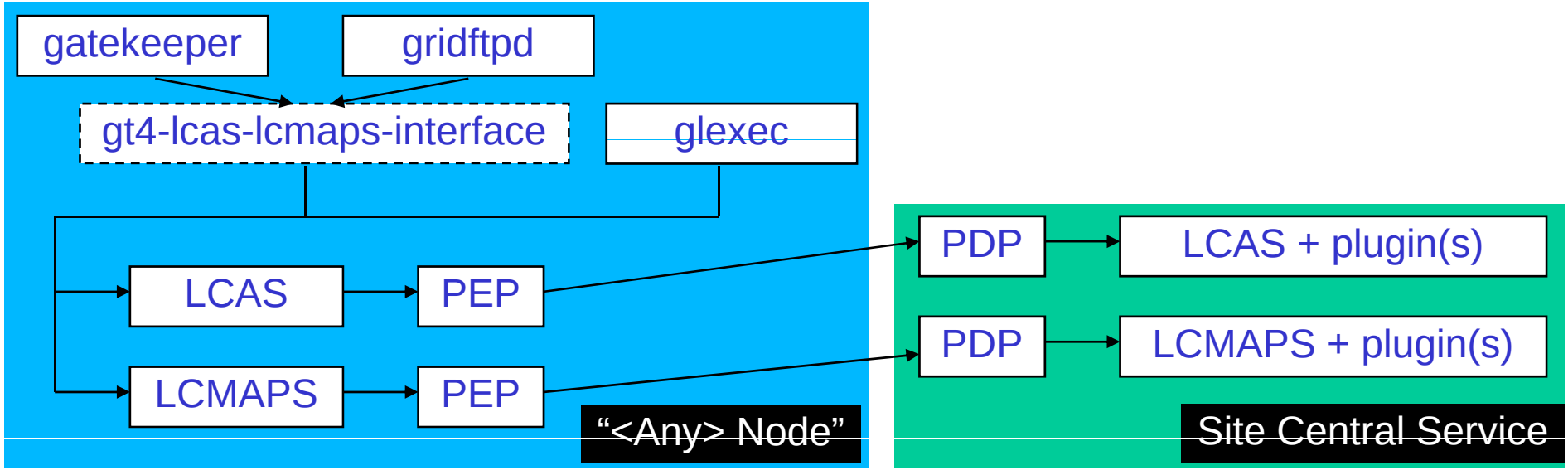
Developments (2)

- **Able to use LCAS and LCMAPS from GT4 services**
  - Like: GT4 gatekeeper, gridFTPd and GSI-OpenSSHd
- **Done by implementing the provided AuthZ (and Mapping) interface from GT4**
  - No need for an ‘edg’ or ‘glite’ variant of the GT4 versions of gatekeeper/gridftpd



SAML-XACML (1)

- **Planning to implement:**
  - LCAS and LCMAPS plugins that form a PEP
    - Goal: get the UID and GID information from a central source securely
  - Implement the PDP side as the Central Service that provides the UID and GID information
    - based on LCAS and LCMAPS



SAML-XACML (2)

- **Using SAML-XACML standards to get the job done**
  - Obligations: current topics were
    - where in the XML-structure do we put credential and environment parameters
    - How do we extract them and enforce them
  - Supporting SAML-ticket for advanced reservation
    - *Better have it in the protocol now then needing to make a change in the (near) future*
    - Make it useful for other application then LCAS+LCMAPS and SAZ+PRIMA/GUMS

- **Basic requirements**
  1. We need a C implementation of the protocol implementation
  2. The library needs to be able to function without the entire set of GT4 libraries
  3. Must be able to create our own Obligation handlers

- **Combined effort with OSG to get it right**
  - We share our basic requirements
  - We discuss how different pieces of information are packed into the SAML-XACML structures to convey interoperability

TODO

- **Handle list of existing (open) bugs**
- **When time allows or when management applies force: make glexec and L&L work natively on x86_64 (and Itanium)**
- **Make the SAML-XACML interface a successful implementation that we can use through out our infrastructures**
- **Continuous effort on code hardening**
- **Deal with other big list with small items that we must do**

- **Documentation...**
  - Between now and the delivery of the SAML-XACML-lib alfa version we have the opportunity to write something useful

- **We want a 3rd party (Poznan?) review to gain more trust in glexec**
  - Distributed at FermiGrid now
  - Review of our code