

Ibergrid O-E-9 notes and issues.

draft 0.1 Mario David 18 Feb. 10

Middleware

“Note:

MU Middleware Unit

OU Operations Unit

PS Pilot Service

SR Staged-Rollout”

There are several layers:

1. Node type
2. Service
3. Client
4. Transversal to all services and/or node types: information service, yaim configuration.

More over a service or node type can be divided into:

1. Stateless: CE's, WMSs, clients (UI, WNs)
2. State dependent: SE's and Catalog services (LFC, AMGA, etc.), VOMS server.

Stateless services are easier to put into a production environment with a special tag for testing, there are no major issues, for example, if a given user has mistakenly used the service.

State dependent services have to be dealt more carefully, specially avoiding a user of mistakenly using that service.

There are presently some node types which are being dealt in a special manner, due to it's complexity:

1. FTS
2. dCache
3. more??

We will need to understand the path that new versions of these services are put into production. For the FTS, CERN is doing the testing as EA. For dCache some large sites do the rollout in very close collaboration with the developers.

We will need to know if/when the versioning scheme will change from major gLite distributions, to node-type, or even service based version.

Should the gLite version continue to be dependent solely on the OS (and arch): gLite 3.0 was supported only for SLC3, gLite 3.1 is only supported in SL4 32 and SL4 64 bit arch, while gLite 3.2 is supported in SL5 64bit arch, plus some more OSs for the clients.

Should we consider a major change affecting several node types and/or services to earn a new major glite release, not necessarily having to do with changing the OS; for example going from the GlueSchema 1.3 to GlueSchema 2.0.

Will have to check the roadmap.

Sites, VO's and user communities

A new version of some middleware component falls on one of the following categories:

1. bug or security vulnerability.
2. new functionality with backwards compatibility.
3. new functionality not necessarily with backwards compatibility.
4. new service or node type.

Item 1, it is expected that a certified component is delivered from the Product Team through the MU directly into the production repositories. It is assumed that no staged rollout is done, and that the certification and verification process is robust enough.

Item 2 to 4, it is assumed that “Someone” has requested the new functionality, new service or node type. Where “Someone” is:

1. Site admin, operations.
2. User community or VO.

The approval for new functionality should be dependent, among other things, from a clear commitment of the requester to provide or arrange the necessary resources needed for the testing, either a staged rollout or a pilot service. We should have additionally, a rather stable set of sites that participate in the staged-rollout.

Presently there are several sites which are in the Pre-Production service, which are expected to become the EA in the production, to perform the stage-rollout.

We will need to understand how the current sites participating in this activity, will be committed taking into account their NGIs structure.

A list of sites participating in the SR,

<http://egee-pre-production-service.web.cern.ch/egee-pre-production-service/index.php?dir=./panel/>

- [PreGR-01-UoM](#) : glite-FTS_oracle(SL5/x86_64)
- [CERN PPS](#) : glite-FTS_oracle(SL4/i386), glite-FTA_oracle(SL4/i386), glite-LSF_utils(SL4/i386), afs_UI(SL4/i386), afs_UI(SL5/x86_64), top_BDII(SL4/i386), top_BDII(SL5/x86_64), WN_GLEXEC(SL5/x86_64)
- [PPS-IFIC](#) : glite-LB(SL4/i386), glite-WMS(SL4/i386)
- [PPS-CNAF](#) : glite-LB(SL4/i386), glite-WMS(SL5/x86_64)
- [DESY-PPS](#) : glite-LFC_mysql(SL5/x86_64), glite-WMS(SL5/x86_64), glite-VOBOX(SL5/x86_64)
- [INFN-PADOVA](#) : glite-LFC_mysql(SL5/x86_64), gLite-CREAM_LSF(SL5/x86_64)
- [CESGA-PPS](#) : glite-MON(SL4/i386), glite-SE_dpm_mysql(SL4/i386), glite-SGE_utils(SL4/i386), lcg-CE(SL4/i386), IC-Info_registry(SL4/i386), glite-WN
- [ITWM-PPS](#) : glite-SE_dpm_disk(SL5/x86_64), glite-SE_dpm_mysql(SL5/x86_64), WN_GLEXEC(SL4/x86_64)
- [RU-Moscow-KIAM-PPS](#) : glite-SE_dpm_mysql(SL4/i386), glite-UI(SL4/i386)

- [RAL-LCG2](#) : top_BDII(SL5/x86_64)
- [GRIF](#) : glite-SE_dpm_mysql(SL4/x86_64), glite-LFC_mysql(SL4/x86_64), glite-WMS(SL4/x86_64)
- [GUP-CERTIF-TB](#) : glite-BDII(SL4/i386), lcg-CE_TORQUE(SL4/i386)
- [WCSS-PPS](#) : glite-SE_dpm_mysql(SL4/x86_64), lcg-CE_TORQUE(SL4/x86_64)
- [SiGNET](#) : lcg-CE_TORQUE(SL4/i386), glite-CREAM_TORQUE(SL4/i386)
- [SAMPA](#) : glite-CREAM_TORQUE(SL5/x86_64)
- [AEGIS01-IPB-SCL](#) : glite-CREAM_TORQUE(SL5/x86_64)
- [FZK-PPS](#) : WN_GLEXEC(SL5/x86_64), glite-SCAS(SL5/x86_64)

A mailing list has been created containing all admins of these sites:

early-adopters@cern.ch

Will have to check the ones which are more active, and possibly start questionnaire on how they intend to proceed in EGI/NGI framework.

Ideally those sites will continue without much disruption.

Product Teams (PTs)

We will need to know for each PT, how they will interact/communicate with the MU, what tools like: web portals, twikis, bug trackers, etc. .

We will need to know how and which PTs have already been formed.

Service Level Agreements.

<https://twiki.cern.ch/twiki/bin/view/EGEE/ProductTeams>

Operations Unit (OU)

EGI is expected to provide the repositories used for the production infrastructure. Possibly, it should also provide the repository for the “beta” releases, which are in the staged rollout or pilot services.

A possible procedure could be:

1. Certified repository (provided by EGI): the PT commits the new versions to this repository, notifying the MU, while doing the verification.
2. Beta repository (provided by EGI): the MU, after doing the verification, commits the new version to this repository, and initiates the process of staged rollout.
3. Production repository (provided by EGI): the new version is committed by the OU, after the staged rollout success.

What modifications are needed in the operations tools (GOCDB, dashboard) in order to signal that a given service is in “beta” release (or in the staged-rollout process).

How will the “reliability/availability” of a given site will be affected, or “**It should not be affected**”.

EGI should provide a “portal” where sites can register to become EA, and participate in the staged-rollout.

- Web pages and twikis definition, everything is now at CERN.

- The problem tracking and communication between the PS/SR and the developers should be through GGUS, maybe a special unit inside GGUS for this, as was done in the EGEE Pre-Production Service (PPS).
- Define a Web form and a procedure for subscription of Early adopters.
 - A site subscribes as EA
 - The coordinator is notified.
 - The coordinator contacts the site for more info and planing.

For all services: services should be in the production infrastructure, but tagged somehow in the information system and operational tools as “beta”. The only exception is the “site-bdii” for which a “new site” as to be deployed.

In any case, one has to evaluate in a case by case basis: Node type, Service, Clients, what is the impact of adding that service to the prod infrastructure, and how it can be tested.

For this kind of proposal to work, the service version HAS to be properly published in the information system, and there should be a requirement for it's correctness in the validation/verification process.

There is still the issue of how this propagates into the GOCDB for example. Presently, the site admin will have to manually set it in the GOCDB. If the GOCDB gets some of it's data from Gstat, then the site admin would only have to worry to configure the service correctly, since it would propagate to the operational tools.

Presently in GlueSchema 1.3, the **GlueServiceUniqueID** entity has the following object:

GlueServiceVersion

This object should hold the correct version of the service, and this does not happen at the moment for some services.

This object can have the format:

Major.minor.patch_beta

For the service participating in the staged rollout.

In the GlueSchema 2.0 (GFD-R-P.147 “Glue specification v. 2.0”, Glue working group, March 3 2009) there is an “**Endpoint**” class with an attribute “**QualityLevel**” for which the proposed data types are:

- development
- pre-production
- production
- testing

So, this attribute could be used not only for services in the staged rollout stage, but as well as services currently in development or certification.