Subject: **RECOMMENDATIONS FOR DECOUPLING OF FQANS AND SHARES AT THE CE**

Author(s): **Christoph Witzig**

Distribution **EGEE Collaboration**

1. INTRODUCTION

As described in the EGEE MJRA1.7 document “authorization in gLite” [R1], the CEs currently deployed in EGEE determine the Unix GID based on the primary FQAN. In addition, the Unix GID is used to assign a job to a given share of the batch system. Thus the authorization data is tied to the shares, and the site has to be reconfigured whenever the assignment between FQAN and shares has to be changed, i.e. whenever the VO wants to assign different CPU shares to different groups of jobs.

In the note “recommendations for changes in the gLite authorization” [R2] we proposed that this tight coupling between FQANs and shares at the CE should be replaced with a mechanism that allows the VO assign different types of jobs to different shares without site reconfiguration. This note puts forward a proposal, how this can be achieved.

2. IDEA OF FIXED SERVICE CLASSES

The proposed setup is based on assigning FQANs to a limited number of service classes at the VO level. It is shown in Figure 1. In this section, we describe the idea behind this proposal and then comment on various implementation issues in the next section.

Under this proposal every CE configures its batch system such that every supported VO gets a fixed fraction of its resources. (E.g. in Figure 1 VO1 has 60% and VO2 40% of the CPU power of the site.) In addition, for every VO a fixed number of shares are created. A fixed ratio between these shares is imposed. In Figure 1 these shares are labelled “Gold”, “Silver”, “Bronze” and “Hydrogen” and their relative weight within the share of that VO is 60%, 25%, 10% and 5% respectively. These shares represent “service classes”.

In VOMS the VO administrator assigns these service classes (respectively the corresponding fixed shares) to FQANs. E.g. in Figure 1 `/VO1/role=production` is assigned to the Gold share, `/VO1/ana/higgs` to the Silver share and `/VO1/role=lcg-admin` to the hydrogen share. More than one FQAN may be assigned to a given share. There are several solutions how this assignment can be distributed in the grid (see section 3).

Note that this proposal actually contains three issues, which can be discussed individually:

1. The scheduling should be decoupled from the Unix GIDs
2. The scheduling should be decoupled from the FQANs
3. The VO should obtain the means to dynamically adjust the share assignment (within the limits of pre-configured fixed-sized shares).

We point out that the dependencies between these three options are such that each relies on the preceding one:

The first option (decoupling scheduling from Unix GID) is an issue on its own. It can be considered to implement it without implementing the other two.

The second (decoupling scheduling from FQANs) and third (dynamic adjustment of share assignment) option requires the first one to be implemented in this proposal. In addition, the third one depends on the second one. However, implementing the second option without implementing the third one raises the question about the value added given the work invested.

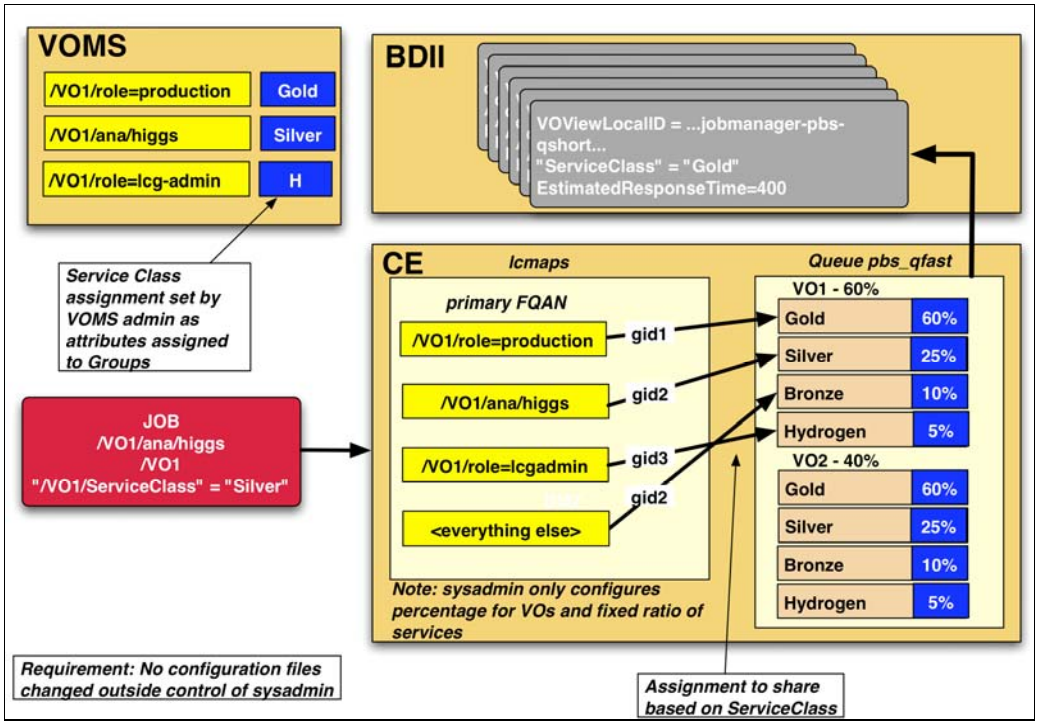
VOMS
/VO1/role=production
Gold
/VO1/ana/higgs
Silver
/VO1/role=lcg-admin
H
BDII
VOViewLocalID = ...jobmanager-pbs-qshort...
"ServiceClass" = "Gold"
EstimatedResponseTime=400
Service Class assignment set by VOMS admin as attributes assigned to Groups
JOB
/VO1/ana/higgs
/VO1
"/VO1/ServiceClass" = "Silver"
CE
lcmaps
primary FQAN
/VO1/role=production
gid1
/VO1/ana/higgs
gid2
/VO1/role=lcgadmin
gid3
<everything else>
gid2
Queue pbs_qfast
VO1 - 60%
Gold
60%
Silver
25%
Bronze
10%
Hydrogen
5%
VO2 - 40%
Gold
60%
Silver
25%
Bronze
10%
Hydrogen
5%
Note: sysadmin only configures percentage for VOs and fixed ratio of services
Requirement: No configuration files changed outside control of sysadmin
Assignment to share based on ServiceClass

Figure 1 Decoupling of FQAN's and shares through predefined service classes

3. IMPLEMENTATION ISSUES

We describe in this section implementation issues of our proposal. Please note that this description is not a detailed design. Further input from the developers of the various middleware components is needed. For the time being we comment on the following issues:

1. How shall the service information be distributed?
2. Should the service class be added to the JDL file?
3. What are the implications for the WMS?
4. What are the implications for LCAS/LCMAPS?

5. What are the implications for blah?
6. What are the implications for the GIP/IS?
7. Is it possible for a site to assign special shares to special classes of jobs?

How shall the service information be distributed?

The assignment of FQAN's to service classes is under the control of the VO, and the user must not be able to change it. Several solutions can be envisaged to make this information known:

1. Use of generic attributes (GA) tied to VOMS groups. This is currently supported by VOMS. VOMS will put the GA into the AC, which is signed by VOMS, thus it is outside the control of the user. However, the there must be a grid-wide convention of the name and value range of this GA, such that grid service can evaluate this information. Figure 1 hints at this solution.
2. Use of a specific parameter (yet to be defined) in the AC set to the service class. This parameter may be a complete listing of the FQAN to service class mapping or it may equally well be an extension of the current FQAN format.
3. VOMS publishes a mapfile that lists the current assignment of FQAN's to service classes. This mapfile is distributed in the middleware similar to the gridmapfile.

In our view, option 1 or option 3 is preferred.

Should the service class be added to the JDL?

As mentioned above the service class may be added to the AC as the VO issues it. This raises the question is whether it is desirable to also put it into the JD. By doing so, it must be guaranteed (outside the user control) that it has the same values between the AC and JDL are the same. However, this option has the disadvantage that vital information is kept in two places, and therefore we do not recommend it.

What are the implications for the WMS?

Obviously the matchmaking must be changed. The VOViews must contain a field expressing the service class of the share, and the WMS must do the matchmaking using these service classes instead of the FQAN's .

What are the implications for LCAS/LCMAPS?

There are no implications for LCAS/LCMAPS. They perform the user authorization and mapping as usual.

What are the implications for blah?

blah must submit the job now to the correct share, thus it must know the service class. There are two ways how this can be accomplished:

1. blah receives this information as an input parameter. In this case the caller must be modified to obtain the service class.
2. blah must extract this information– either by examining the proxy certificate (if the information is located in the AC) or by consulting an authorization module (yet to be defined[1]). Currently, blah is not a security related component, so the security implications of this solution must be reviewed within JRA1.

What are the implications for the GIP/IS?

The VOView concept must be expanded to accommodate this service class and the IS must contain this information. There are two possibilities:

1. The Glue schema is not changed and the ACBR simply contains the value of the service class
2. The Glue schema is changed and a parameter describing the service class is added.

In both cases the GIP must also be changed in order to publish this information.

Is it possible for a site to assign special shares to special classes of jobs?

There may be sites that want to assign special shares to special VO groups. E.g. a site may dedicate a share of its CE to calibration programs and would want to assign an FQAN to a specific share, say the FQAN "`/VO1/analysis/emcal_calibration`" to a share "EMCAL". However, this mapping should only be valid at this particular site.

Today, this requirement is easily fulfilled by the current CE setup. In order to fulfil this requirement within the solution described here, the local system administrator would also setup the share as today. The code extracting the service class for this job (be it blah or not) would have to consult the local configuration in order to find out whether this particular FQAN should be mapped to a special share or not. The GIP in turn must also publish the FQAN mapping of this share (and not the service class). The WMS must take into account that the matchmaking may be done on the service class or the FQAN.

4. COMMENTS

We add the following comments to this proposal:

1. The configuration burden of the site administrator is limited to configuring the relative weight of the different shares. However, the local site administrator must modify its local configuration as soon as the VO wants to change these ratios (e.g. increase the "Gold" share from 60% to 70%).
2. The local site administrator retains the full control of his site. In particular he is in full control of the mapping of FQANs to Unix credentials.
3. The naming of the service attribute ('gold", "silver", …) has to be coordinated throughout the middleware.

[1] This would be a requirement for the new authorization service currently under design.

4. This mechanism can also be extended to not only assign the service class to the FQAN but also a priority, which job to run first once it has arrived on a CE. In this case every FQAN gets two additional attributes: the service class and the priority within that service class.
5. If recommendation 4 is accepted, then the service class for the chosen FQAN has to be taken into account – especially if the service information is put into the AC.
6. Some institutions have already implemented a similar solution in test-beds. Thus, some operating experience already exists, and it should be possible to take their experiences into account and implement this solution within a reasonable amount of time.

5. OUTLOOK

We described in document a high level design how FQANs and shares can be decoupled at the CE. Next, we invite feedback from the members of the EGEE collaboration before Feb 28, 2008. In particular, feedback is requested on the relative priority of the three issues identified in this proposal, namely:

1. The scheduling should be decoupled from the Unix GIDs
2. The scheduling should be decoupled from the FQANs
3. The VO should obtain the means to dynamically adjust the share assignment (within the limits of pre-configured fixed-sized shares).

In early March the proposal will be revised according to the comments received and presented to the TCG.

6. REFERENCES

R1	C.Witzig, Authorization in gLite https://edms.cern.ch/document/887174/1
R2	C.Witzig, Recommendations for changes in gLite authorization https://edms.cern.ch/document/887174/1