



Enabling Grids for E-scienceE

Policy management and fair share in gLite

Andrea Guarise

HPDC 2006

Paris June 19th, 2006

www.eu-egee.org



Information Society



- **Considering a Resource Provider offering computing resources to VOs locally managed by a batch system (LRMS):**
 - The way to manage inter-VO sharing of computing resources is stable and in place
 - Intra-VO sharing is the new challenge to be faced
- **A VO is asking:**
 - How do I manage priorities between different groups of users and/or roles?
 - How do I assign different share of resources?

- **Fair share and priority:** different concepts
 - **Fair share:** to set system utilization targets for VO groups/role
 - **Priority:** to set which job execute first among the queued ones

- 1. Service Classes (SC) definition**
 - description of the service classes that each VO would like to assign to its users
- 2. Service Classes (SC) implementation**
 - description of mechanisms to set up these service classes to local batch systems (e.g., PBS/MAUI, LSF)
- 3. Service Classes (SC) discovery**
 - description of how the service classes can be advertised in the Grid Information Service for discovery and matchmaking purposes
- 4. GridUser2ServiceClass assignment**
 - description of mechanisms to let the VO dynamically assign groups to service classes
- 5. GridUser2ServiceClass mapping**
 - description of how Grid credentials are mapped to local credential in order to access the appropriate service class

- **In order to implement shares for groups within a VO.**
 - *Short term:*
 - *Static shares assignment*
 - *Medium term:*
 - *Dynamic share assignment based on G-PBox*

- **Common-agreed fixed shares and queues on each site of the collaboration:**
 - VOMS groups should be statically mapped to UNIX GIDs on CEs
 - LRMS shares are defined statically according to UNIX GIDs
 - Two service class defined: long and short
 - Discovery based on GLUE Schema, VOView feature. The WMS should be adapted to be aware of this information in job scheduling and is used to send jobs.
- **Pro & Con:**
 - Pro: enable initial intra-VO fair share
 - Con: changes in share assignment have to be arranged via phone or e-mail among VO manager(s) and sites; it therefore requires a non-negligible amount of work on both sides (VO and sites) for the management of the system

- **Components:**

- **VOMS**

- Allows to create *group* of users within a VO and assign them roles and capabilities. These are stored in the user proxy certificate used to submit a given job.

- **G-PBox**

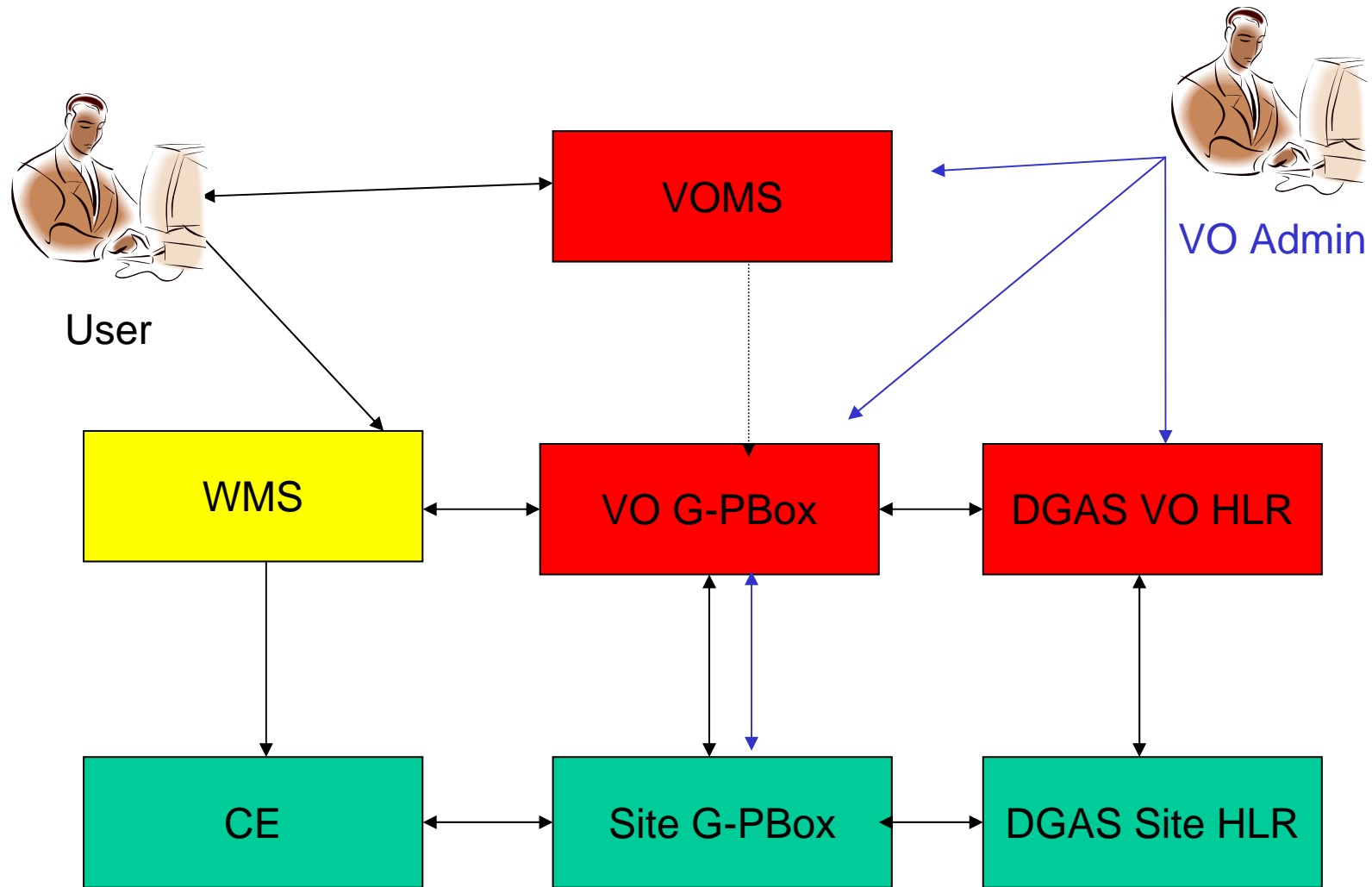
- To map users to service classes.
- To dynamically change the association between users and classes.

- **DGAS**

- Distributed Grid Accounting System. Allows to retrieve grid accounting information from both sites and VOs point of view.

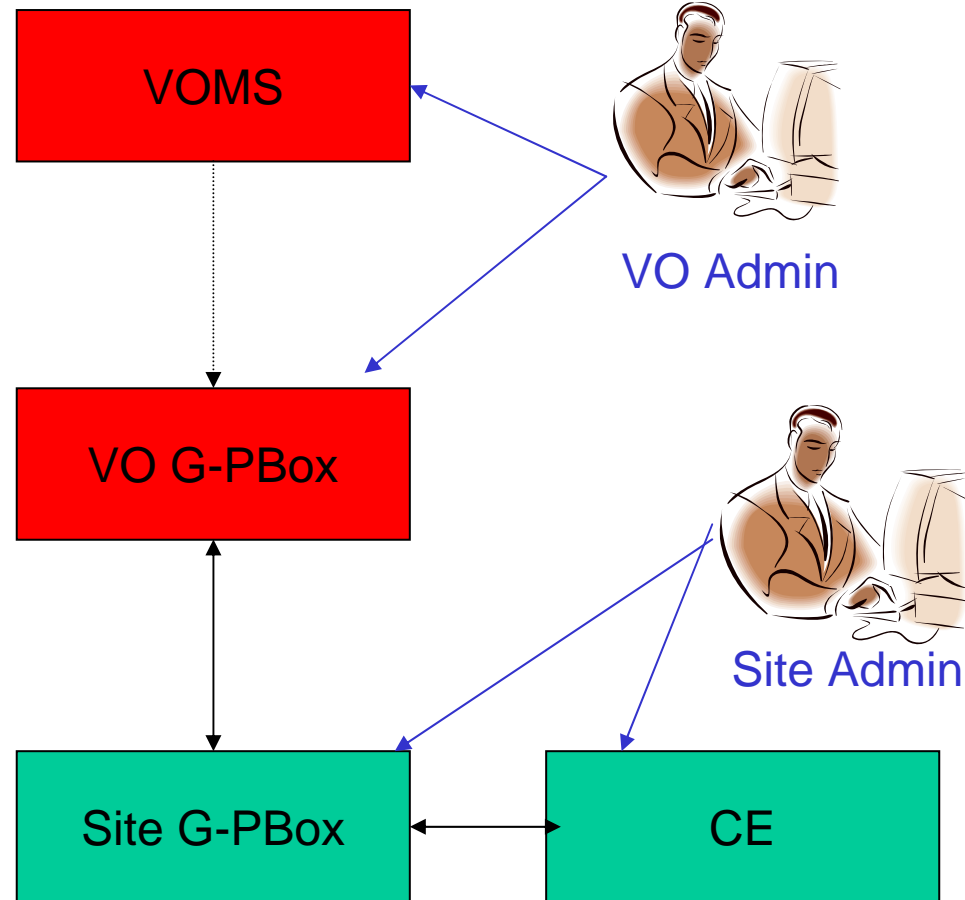
- **WMS**

- Workload Management System. Responsible for analysing the user job requests, find a suitable computing resource, submit the job request to such resource and follow the job's life until its conclusion.



- The working principle is simple: G-PBox allows to define via a proper language (XACML) a generic set of computing policies. The policies are then automatically propagated to the interested entities where these are evaluated by a Policy Decision Point (PDP) and enforced by a Policy Enforcement Point (PEP).
- In this scenario policies defining the mapping between group/roles and service classes are defined by the VO administrator on the VO G-PBox. The VO G-PBox then forwards such policies to all the necessary site G-PBoxes. The site administrator on his side defines a set of policy in the site G-PBox defining the mapping between service classes, local unix groups and LRMS shares.
- When the CE receives a job submitted by a VO User belonging to a certain Group/Role the PDP will assign it to a given local group/share according to such policies, thus implementing the desired fair share.
- Moreover G-PBox is able to define policies with parameters coming from external source of information such as an Accounting system (DGAS in our example). This way it is also possible to define usage quotas.

- VO manager defines Group, Roles and capabilities within a VO. Then she assigns users to groups and grants them the possibility to ask for roles.
- (e.g. /atlas/analysis, /atlas/production)
- VO manager assigns Groups and Roles to some pre-defined service classes, e.g. Gold, Silver or Bronze.
- Site Admin defines the local mapping for service levels Gold, Silver and Bronze, these get mapped to local unix groups and shares.
- Example: GOLD -> gid: vonameGOLD, share: 70% of the VO-queue available resources.

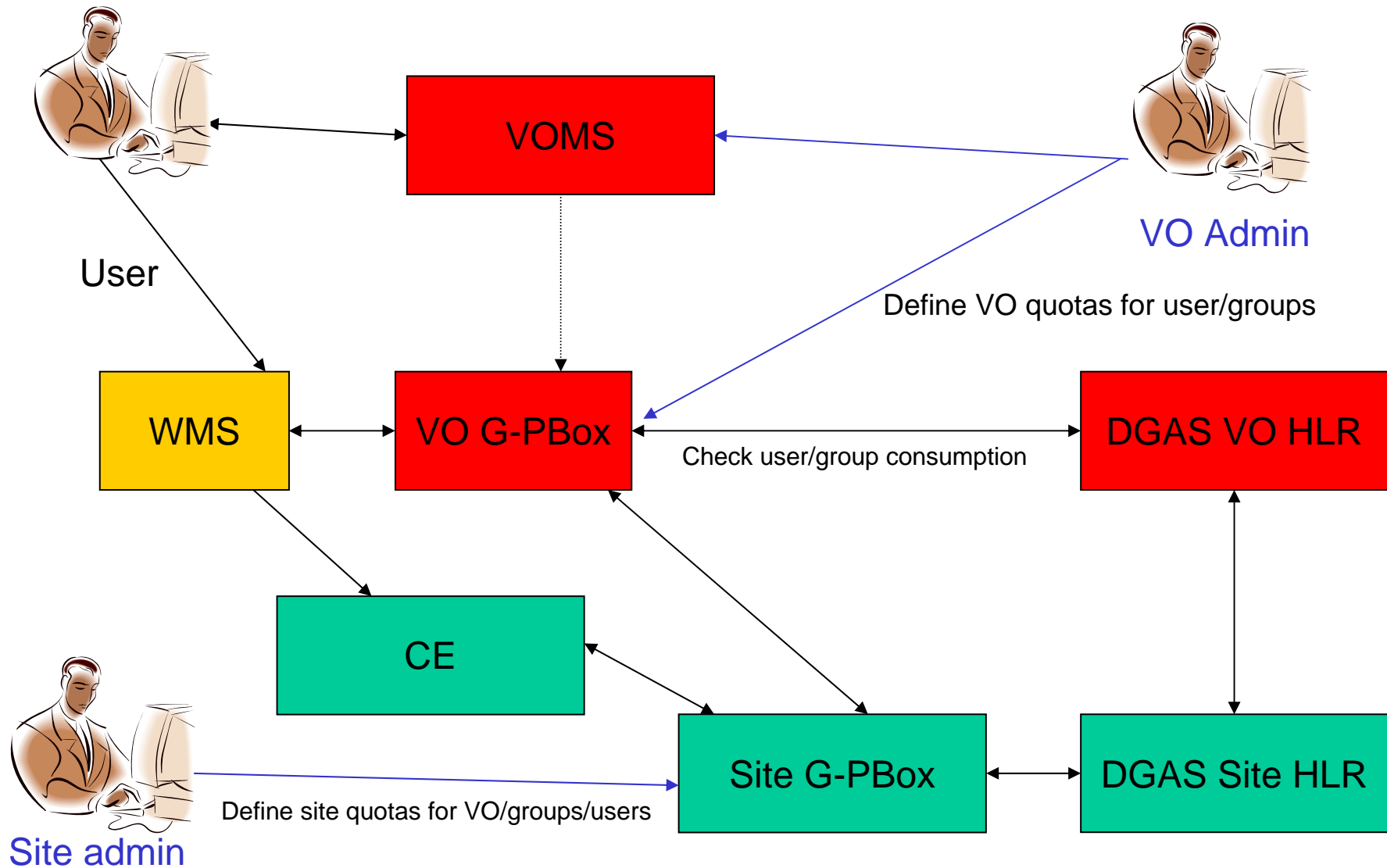


- **Computing Element setup:**
 - Create one queue for VO.
 - Create several local pools for VO, each with its own fair share.
 - Publish the supported service classes in the Information System.
- **Site G-PBox:**
 - Policies mapping service classes to the corresponding local accounts. (private)
 - Policies mapping groups/roles to service classes. (public, read from VO G-PBox)
- **VO G-PBox:**
 - Policies mapping groups/roles to service classes (public, transmitted to Site G-PBoxes)
 - Policies to filter CEs on the base of the mapping policies and the service classes implemented by the CE

- **Abstraction between VO-level shares definition and local site queue/account configuration.**
- **Mapping of users/groups/roles to service classes can be changed dynamically.**
- **Easy to discover what service classes are supported by each CE.**
- **No need to rely on publication of FQAN for CE selection from WMS.**
- **XACML semantics allow much more complex policies not just related to fair share, such as for usage quotas.**

- **Another open issues is how to implement Usage Quotas in grid computing:**
 - By using the policy enforcement system framework implemented by G-PBox it is possible to define and enforce usage quotas over the Grid.
 - This can be done considering grid accounting information while defining and evaluating some policies.
 - We briefly illustrate an on-going activity aimed to use accounting information coming from DGAS accounting system to define G-PBox “usage quotas” policies.

- **For example a scenario can be a Virtual Organisation that allows a student to use no more than 100 Hours of CPU time in a month.**
 - In this case the policy is defined at G-PBox VO level.
 - When the WMS receives a job submission request for that student, it would contact the VO G-PBox for evaluating the request. G-PBox would contact DGAS asking for the amount of cpuTime consumed by the user in that month. It can then decide whether to allow the submission or not.
- **Another example would be a site willing to accept jobs for a given VO/group for no more than 10 hours of CPU per day.**
 - In this case the site G-PBox can contact the site HLR (site accounting information) and ask for the daily consumption for that VO to evaluate the policy.



- **Information about G-Pbox can be found at:**
 - <http://infnforge.cnaf.infn.it/gpbox/>
- **Information about DGAS can be found here:**
 - <http://www.to.infn.it/grid/accounting/>
- **References:**
 - <http://www.cnaf.infn.it/~andreoizzi/wiki/pub/doc/JobPrioritiesWG-20060531.pdf>
 - <https://grid-it.cnaf.infn.it/cdsagenda/fullAgenda.php?ida=a0657>