GLUE v. 2.0 – Reference Realisation to LDAP Schema

Status of This Document

This document provides information to the Grid community regarding the LDAP Schema realisation of the GLUE information model (v.2.0). Distribution is unlimited. This implementation is derived from the proposed recommendation GFD.147 “GLUE Specification v. 2.0”.

Copyright Notice

Copyright © Open Grid Forum (2014). All Rights Reserved.

Trademark

Open Grid Services Architecture and OGSA are trademarks of the Open Grid Forum.

Abstract

The GLUE specification is an information model for Grid entities described in natural language enriched with a graphical representation using UML Class Diagrams. This document presents a realisation of this information model as an LDAP Schema, and includes explanations of the major design decisions made during the rendering process.
Contents

1. Introduction ............................................................................................................. 3
2. Notational Conventions .......................................................................................... 3
3. LDAP Schema Realisation ....................................................................................... 3
   3.1 Approach ............................................................................................................. 3
   3.2 Prefix conventions .............................................................................................. 3
   3.3 object class and attribute naming conventions .................................................. 4
   3.4 object class types and inheritance ...................................................................... 4
   3.4.1 Composition of LDAP object classes into entries ............................................. 5
   3.5 Data types .......................................................................................................... 6
   3.6 Relationships ..................................................................................................... 7
   3.7 Structuring the LDAP entries ............................................................................. 10
   3.8 Aggregation ....................................................................................................... 12
   3.9 OID Assignments ............................................................................................... 12
   3.9.1 Adding new object classes or attributes ......................................................... 13
4. Security Considerations ......................................................................................... 14
5. Author Information ................................................................................................. 15
6. Contributors & Acknowledgements ..................................................................... 15
7. Intellectual Property Statement ........................................................................... 15
8. Disclaimer ............................................................................................................ 15
9. Full Copyright Notice ........................................................................................... 15
10. References .......................................................................................................... 16
11. Appendix A: Example LDAP tree structures in existing server side implementations .... 16
1. Introduction
The GLUE 2.0 information model defined in [glue-2] is an abstract information model of Grid entities. For technology providers to adopt it, it has to be realised into a concrete data model.

This document describes the normative realisation of the GLUE 2.0 abstract information model in terms of an LDAP schema. The approach followed to map the entities and relationships in the abstract information model to the LDAP data model is also described.

2. Notational Conventions
The key words ‘MUST,” “MUST NOT,” “REQUIRED,” “SHALL,” “SHALL NOT,” “SHOULD,” “SHOULD NOT,” “RECOMMENDED,” “MAY,” and “OPTIONAL” are to be interpreted as described in RFC 2119 (see http://www.ietf.org/rfc/rfc2119.txt).

This document adopts the following terminology:

“Entities” and “Classes” as defined in the abstract model of GFD.147 are referred with the terms “entity” or “entities”; “attributes” as defined in GFD.147 are called “attribute” or “attributes”. Entity and attribute names as defined in GFD.147 are shown in italic. For example, the entity with name "Service" is represented as Service.

Note that the term “entity” with no typesetting is used in this document for any generic entity or class defined in GFD.147, and must not be confused with the term “Entity” with italic typesetting, that identifies the specific entity in GFD.147 whose name is “Entity”.

All the terminology relative to LDAP is taken from definitions in RFC 4510 (see http://tools.ietf.org/html/rfc4510); object classes and attributes in the concrete LDAP schema will have typesetting bold. For example, GLUE2Entity is an LDAP object class and GLUE2ServiceID is a LDAP attribute.

3. LDAP Schema Realisation

3.1 Approach
There are many possible approaches to realise the GLUE 2.0 abstract information model as an LDAP schema. The approach followed here is informed by practical experience with the LDAP implementation of the GLUE 1 schema [glue-1], and by general considerations relating to the efficiency and simplicity of likely queries. Conversely, the GLUE 2 schema itself was designed with the expectation that LDAP would be one of the main implementation technologies, and hence there is in many cases a natural way to translate the schema concepts into LDAP.

The GLUE LDAP implementation needs to map each entity in the GLUE information model to a specific LDAP entry defined in terms of object classes. We have chosen the most straightforward mapping in which there is a one-to-one correspondence between LDAP object classes and GLUE entities, with inheritance in the abstract schema represented explicitly by object class inheritance in LDAP.

In the following sections we discuss the detailed design decisions that have been made while implementing the GLUE model into LDAP.

3.2 Prefix conventions
LDAP allows the same descriptor to refer to different object identifiers in certain cases and the LDAP server supports multiple registrations of the same descriptor (each indicating a different kind of schema element and different object identifier). However, multiple registrations of the same descriptor are to be avoided if possible [rfc4520].
In practical experience with version 1 of the GLUE schema it has generally been the case that the schema does not need to coexist with other schemas, but it nevertheless seems useful to allow for this as a possibility. As object classes and attributes might have the same names in different schemas (there is only a global namespace), in order to make schemas compatible and able to coexist with other schemas in the same LDAP server we have decided that all object class and attribute names should be prefixed with a concrete string.

Given that GLUE 2.0 represents a major version change which may be required to cohabit with older versions for some time, GLUE2 is used as a clean short prefix for all schema elements in the model – this compares with the prefix of Glue used for the version 1.x schemas.

3.3 Object class and attribute naming conventions

The name of each LDAP object class is simply the name of the model entity prefixed as described above, e.g. the object class representing the Service entity is called GLUE2Service. Each attribute of an entity in the abstract model is rendered as an LDAP attribute with a name composed from the names of the entity and the attribute, for example GLUE2ServiceType corresponds to the Type attribute of the Service entity. This gives a clear separation of attributes per object class, making it less prone to mistakes if changes are made and in the construction of queries.

3.4 Object class types and inheritance

The LDAP schema objects in the rendering are defined by following the most straightforward mapping in which there is a one-to-one correspondence between the model entities and the LDAP object classes. Every entity is represented by an LDAP object class whose set of mapped attributes is constituted only by those attributes explicitly defined in the model as belonging to that entity, and NOT those that are inherited from other entities.

The abstract model uses inheritance to derive some entities from others. LDAP is not object-oriented in the usual sense, but it allows inheritance to be represented explicitly by composing object classes [rfc4512] into LDAP entries. However, it would also be possible to define standalone object classes including all inherited attributes directly. We have chosen to use explicit inheritance, both as the most natural representation of the schema and because it simplifies some queries. For example, it enables a generic query to be made for the URL attribute of every Endpoint without any special treatment for Computing Endpoints, Storage Endpoints or any other specialised entity which may be defined in the future. The main disadvantage of this approach is more complexity in the naming of attributes within an LDAP entry, for example a GLUE2ComputingEndpoint entry can include attributes called GLUE2ComputingEndpoint-RunningJobs, GLUE2EndpointURL and GLUE2EntityName, but in practice this seems unlikely to cause significant problems.

An alternative approach could be made for the Entity entity, since it is unlikely that queries for the attributes of all LDAP entries will be common. In general, we conclude that consistency both with the abstract model and the general principles for the LDAP schema nevertheless make an explicit GLUE2Entity object class the best solution.

We have however made an exception for the ID attribute. All LDAP entries have an attribute that is used to construct its Distinguished Name (DN), and for the GLUE2 schema the natural attribute to use is clearly the ID. If we simply followed the rules described above the name of the ID attribute for every object would be GLUE2EntityID, and the DN of every object would be of the form "GLUE2EntityID=x, GLUE2EntityID=y, GLUE2EntityID=z". We consider that this would be unduly opaque, and therefore introduce an additional rule that the ID attribute is defined in the object classes representing the classes derived immediately from Entity, and the naming then follows the standard rules. So, for example, the ID attribute for all of the GLUE2Service, GLUE2ComputingService and GLUE2StorageService objects is called GLUE2ServiceID.

One final point is that the schema document defines Policy, Domain, Share, Manager and Resource as being abstract classes which MUST NOT be instantiated, but should only be used to
derive specialised entities. However, this rule is based on the fact that these objects in themselves contain no useful information, rather than that there is any structural flaw caused by instantiating them. Modifying an LDAP schema is a complex and time-consuming operation, so it may be useful at some point to prototype a new class derived from, for example, Share using a concrete GLUE2Share object class together with GLUE2Extension objects to carry the putative new attributes, and only define a new specialised object class once the definition of the new entity is stable. We have therefore decided to make these object classes concrete and instantiateable. However, it should be emphasised that such objects MUST NOT be regarded as strictly compliant with the schema, that schema validation tools SHOULD reject such objects, and that tools to translate the LDAP schema to another representation MAY reject or ignore them.

To summarise, the following rules are employed:

- The GLUE2Entity object class in LDAP should carry all attributes defined in Entity except ID.
- All entities immediately deriving from Entity will have their own ID attribute named after the entity name. For example, the object class GLUE2Location will have the attribute GLUE2LocationID.
- All classes deriving from Entity in GLUE2 will also inherit by composition from the GLUE2Entity object class in LDAP.
- The GLUE2Entity object class will be of type “Abstract”, meaning that it cannot be instantiated as a standalone entry.
- All classes except the GLUE2Entity will be STRUCTURAL, although in practice only classes with an ID attribute will be used to form DNs, and hence all entries must have exactly one such class as described below.

3.4.1 Composition of LDAP object classes into entries
An LDAP entry is formed out of a collection of LDAP object classes. Note that since GLUE2Entity is abstract and no GLUE2EntityID is defined for it (see discussion above) there is no need to include such an object class explicitly in an entry, although it does no harm to do so. Inheritance in the schema allows to instantiate its attributes into any GLUE 2 LDAP object class (except GLUE2Extension)

Each entry which is a realisation of an entity in GFD.147 MUST include:

- Exactly one object class that has a defined GLUE2<entityname>ID attribute, e.g. the object class GLUE2Service has attribute GLUE2ServiceID attribute.
- Any additional object class MUST be consistent with respect to inheritance as defined in GFD.147. For example, a GLUE2ComputingEndpoint entry MUST have GLUE2Endpoint object class as well, and MUST NOT have a GLUE2StorageEndpoint object class, as these would break the model.
- Attributes can belong to any of the included object classes, plus those of the abstract object class GLUE2Entity.
- Attributes included in the RDN MUST be of the form GLUE2<entityname>ID, that is, ONLY GLUE2 ID or LocalIDs as defined in GFD.147 MUST appear in the RDN, with the exception of the special o=glue suffix defined below.

Example of a GLUE2ComputingService entry:

dn: GLUE2ServiceID=<URI>,...,o=glue
objectClass GLUE2Service
ObjectClass GLUE2ComputingService
# from GLUE2Entity
3.5 Data types

LDAP does not have an extensive range of data types, and there is little overlap with the types defined in the GLUE schema. For the implementation of the different data types, just two different types of the standard LDAP v3 attribute set referred to in [rfc4517] are used:

- DirectoryString, with OID 1.3.6.1.4.1.1466.115.121.1.15
- Integer, with OID 1.3.6.1.4.1.1466.115.121.1.27

"Integer" is used for types UInt32 and UInt64 of the original GLUE 2.0 Specification and "DirectoryString" is used for every other type.

DirectoryString type does NOT allow empty strings. This means that if an attribute has no value the corresponding LDAP attribute entry will NOT be published by any LDAP server. Information consumers and validation tools MUST take this into account.

This also means that data type integrity (for example in the case of enumerations) will largely not be checked in the LDAP implementation itself, but must be ensured by other means, for example external validation tools.

Due to the fact that this document has been in DRAFT status for long time, some of the existing implementations of the LDAP schema prior to the current revision of this document do not yet follow the above recommendations. In particular:

- The Boolean object type (OID 1.3.6.1.4.1.1466.115.121.1.7) was used for all attributes with type ExtendedBoolean_t in GFD147
- The string type IA5String (OID 1.3.6.1.4.1.1466.115.121.1.26) was used for all objects of different type from GFD.147 UInt32, UInt64, and ExtendedBoolean_t

The GLUE WG recommends updating all renderings and adopting version 2.0 of the schema as soon as possible after this document is approved.

The attribute multiplicity in the model maps naturally to LDAP since it supports both optional and multi-valued attributes directly, and hence the constraints implied by the model (MUST/MAY and SINGLE-VALUE) are imposed directly in the LDAP attribute definitions.

Note that there are two principle changes from the LDAP representation used for GLUE 1. First, in the case of choosing IA5String (OID=1.3.6.1.4.1.1466.115.121.1.26) as the string type. However, this is basically 7-bit ASCII that does not allow text in various non-English languages to

---

glue-wg@ogf.org
be represented, and moreover the presence of such strings may cause the entire object to be rejected by an LDAP server. We have therefore decided to use DirectoryString for GLUE 2, which is basically the UTF-8 encoding of Unicode that includes ASCII as a subset. Potentially it would be possible to use IA5String for the majority of attributes where the permitted values could be restricted and only use DirectoryString for attributes that represent free text, but in practice it seems simpler to use a uniform representation. We note that the schema document itself does not define the string type in any detail, which also implies that we should use the broadest possible type.

The second change concerns case sensitivity. The GLUE 1 schema defines strings not to be case-sensitive (a matching rule of caselgnoreIA5Match), and to some extent this makes queries simpler. However, many external tools are case-sensitive, and for the GLUE 2 schema we explicitly defined strings to be case-sensitive. We have therefore followed this in the LDAP schema by defining the matching rules to be caseExact. This also supports the change to DirectoryString, since case-matching rules are more complex for extended character sets. However, this will be the most visible change in behaviour relative to GLUE 1, and hence may require some education for users.

The existence of mandatory attributes also represents a partial change from GLUE 1 which had essentially all attributes as optional. This may require more care in the implementation of information providers, but also helps to ensure the quality of the published data.

3.6 Relationships

LDAP is not a relational database, but a directory. Thus, LDAP neither provides nor ensures relationships other than the parent-child relations implied by the hierarchical DN.

To implement relationships between objects in LDAP, for each relationship a new attribute must be defined. In the GLUE 1 schema we defined two such attributes, GlueChunkKey pointing to parent objects in the DN hierarchy and GlueForeignKey pointing to objects outside the hierarchy. These attributes contain ID-value constraints of the form GlueClusterUniqueID=xyz.

In GLUE 2 we have two differences that imply a change in the way that relationships are represented. In GLUE 1 the need for the ChunkKey is related to the fact that some objects have only a non-unique LocalID, and there is therefore a need to relate those objects explicitly to their parent in order for them to be identified. For example, a GlueSA object can only be identified relative to its parent GlueSE object. By contrast, in GLUE 2 all entities (other than Extension) have a unique ID and hence can be identified uniquely, which removes the need for something similar to the ChunkKey.

Secondly, in GLUE 1 the unique ID attributes are only unique within objects of the same type, so for example a GlueClusterUniqueID and a GlueSubClusterUniqueID may be identical. However, in GLUE 2 we require ID attributes to be globally unique even across object types. It is therefore possible for the relationship value to simply be the ID.

In terms of the attribute names we felt that it would be clearer and more explicit for the name to specify the relation it represents, rather than using a generic name such as GLUE2ForeignKey. This also prevents the accidental publication of relationships not defined in the schema. The naming convention chosen is to have the prefix and object class name as for other attributes, followed by the name of the object class to which the reference points (i.e. the one containing the relevant ID attribute), and finally a suffix ForeignKey. (We also considered using FK as a more compact suffix, but decided that the longer string is likely to be easier to understand.) As an example, this means that a relation from GLUE2Endpoint to GLUE2Service is called GLUE2EndpointServiceForeignKey, and will have a value which is the corresponding GLUE2ServiceID. These attributes are inherited in the same way as any other attribute, so that for example a GLUE2StorageEndpoint will be related to a GLUE2StorageService via an attribute with the same name.

In an early version of the schema we made a different decision: that relations between derived objects would use an attribute defined in the derived object class. For example, a relation

GlueClusterUniqueID=xyz.
between Computing Endpoint and Computing Service would be carried by an attribute called GLUE2ComputingEndpointComputingServiceForeignKey. Since some software still publishes such attributes they are still defined as optional attributes in the schema, but they are deprecated and MUST NOT be relied on.

Relational attributes need to be defined in the LDAP schema corresponding to every relation defined in the abstract model, and with multiplicities as defined in the model document. Relations are bidirectional, but there is no general need to define an attribute for both ends of the relation since LDAP queries can be performed in either direction. That is, it is possible either to query for an object which has a particular ID in its ForeignKey attribute, or for an object with an ID which has been extracted from a ForeignKey. Depending on the circumstances there may be differences in efficiency or ease of use, for example queries which return multiple IDs are likely to be more complex, but in general we decided to define a ForeignKey only for one end of a relationship.

There were two main considerations taken into account in deciding which end of the relationship to use. In many cases there is a natural parent-child relation, for example Service is logically a parent of Endpoint, and it is likely to be better for the relation to point from child to parent. This is both for likely ease of coding of information providers – create the parent and then loop over the children – and because the most likely query direction is to find the children of a given parent rather than vice versa.

The second consideration is multiplicity. For one-to-many relations it will normally be better to have one attribute per object than many, and even for many-to-many relations it will often be the case that one of the multiplicities is likely to be substantially more than the other. For example, the relation between Share and Endpoint is many-to-many, but in most cases there will be many more Shares than Endpoints.

In general terms this mechanism is similar to the one used in relational databases with foreign keys, except for a few key points:

- In a relational database, when implementing a one-to-many relationship, the foreign key attribute is included in the “many” object since a database cell can only have one value. In LDAP attributes can be multivalued, so this may depend on the needs for each object.

- In a relational database, when implementing a many-to-many relationship, a new table is created that holds all relations due to the fact that a table cell cannot hold multivalued attributes. LDAP supports multivalued attributes directly so there is no need for any intermediate table.

- Relational databases ensure relationship integrity, but LDAP does not.

We considered each of the schema relations individually to decide which end should carry the ForeignKey attribute in the light of the considerations described above, and the result is shown in Table 1. In the vast majority of cases the decision was obvious. The only exception to the “one end” rule is for the two peer relations Service-Service and Activity-Activity where the keys need to be at both ends.

<table>
<thead>
<tr>
<th>Relation 1</th>
<th>Mult 1</th>
<th>Mult 2</th>
<th>Relation 2</th>
<th>Object with key</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity</td>
<td>1</td>
<td>0..*</td>
<td>Extension</td>
<td>Extension</td>
<td>GLUE2ExtensionEntityForeignKey</td>
</tr>
<tr>
<td>Location</td>
<td>0..1</td>
<td>0..*</td>
<td>Service</td>
<td>Location</td>
<td>GLUE2LocationServiceForeignKey</td>
</tr>
<tr>
<td>Location</td>
<td>0..1</td>
<td>0..*</td>
<td>Domain</td>
<td>Location</td>
<td>GLUE2LocationDomainForeignKey</td>
</tr>
<tr>
<td>Contact</td>
<td>0..*</td>
<td>0..*</td>
<td>Service</td>
<td>Contact</td>
<td>GLUE2ContactService</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------------</td>
<td>----------------------</td>
<td>------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contact</td>
<td>0..*</td>
<td>Domain</td>
<td>Contact</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ContactDomainForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AdminDomain</td>
<td>1</td>
<td>Service</td>
<td>Service</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ServiceAdminDomainForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AdminDomain</td>
<td>0..1</td>
<td>AdminDomain</td>
<td>AdminDomain(child)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2AdminDomainAdminDomainForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UserDomain</td>
<td>1..*</td>
<td>Policy</td>
<td>Policy</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2PolicyUserDomainForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UserDomain</td>
<td>0..1</td>
<td>Activity</td>
<td>Activity</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ActivityUserDomainForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UserDomain</td>
<td>0..1</td>
<td>UserDomain</td>
<td>UserDomain(child)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2UserDomainUserDomainForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service</td>
<td>1</td>
<td>Endpoint</td>
<td>Endpoint</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2EndpointServiceForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service</td>
<td>1</td>
<td>Share</td>
<td>Share</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ShareServiceForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service</td>
<td>1</td>
<td>Manager</td>
<td>Manager</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ManagerServiceForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service</td>
<td>0..*</td>
<td>Service (both)</td>
<td>Service (both)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ServiceServiceForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Endpoint</td>
<td>0..*</td>
<td>Share</td>
<td>Share</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ShareEndpointForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Endpoint</td>
<td>1</td>
<td>AccessPolicy</td>
<td>AccessPolicy</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2AccessPolicyEndpointForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Endpoint</td>
<td>0..1</td>
<td>Activity</td>
<td>Activity</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ActivityEndpointForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Share</td>
<td>0..*</td>
<td>Resource</td>
<td>Share</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ShareResourceForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Share</td>
<td>0..1</td>
<td>Activity</td>
<td>Activity</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ActivityShareForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Share</td>
<td>1</td>
<td>MappingPolicy</td>
<td>MappingPolicy</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2MappingPolicyShareForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manager</td>
<td>1</td>
<td>Resource</td>
<td>Resource</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ResourceManagerForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Resource</td>
<td>0..1</td>
<td>Activity</td>
<td>Activity</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ActivityResourceForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Activity</td>
<td>0..*</td>
<td>Activity</td>
<td>Activity (both)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ActivityActivityForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ComputingService</td>
<td>1</td>
<td>ToStorageService</td>
<td>ToStorageService</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ToStorageServiceComputingServiceForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ComputingManager</td>
<td>1</td>
<td>ApplicationEnvironment</td>
<td>ApplicationEnvironment</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2ApplicationEnvironmentComputingManagerForeignKey</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ComputingManager</td>
<td>0..1</td>
<td>Benchmark</td>
<td>Benchmark</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GLUE2BenchmarkComputingManagerForeignKey</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
3.7 Structuring the LDAP entries

In LDAP, object instances (entries) are arranged in a hierarchical structure called the Directory Information Tree (DIT). An LDAP entry consists of set of attributes taken from the object classes associated with the object. An LDAP entry MUST be composed of at least one structural object class and MAY use several auxiliary object classes. Each entry MUST have a unique Distinguished Name (DN) constructed from an ordered series of Relative Distinguished Names (RDNs), each of which consists of an attribute name and its value taken from a structural object class. Each RDN must correspond to an entry which actually exists. The RDNs are required to be unique only to the extent that the full DN of every object needs to be unique. The DNs through their hierarchically ordered RDNs define the tree structure.

In the GLUE 2 rendering approach the GLUE 2 entries are built from object classes that are grouped into the same entry only if there is an inheritance relationship in the model. This results in an LDAP tree of GLUE 2 entries where each instance of a model entity has its own separate node in the tree, and it is natural to use the unique ID of the model entity to form the RDN. In
case of entries with multiple object classes, the ID of the object class derived immediately from the abstract object class GLUE2Entity is used, as described earlier.

For example, a ComputingEndpoint entity would be realised as an LDAP GLUE2ComputingEndpoint entry that consists of attributes of the object classes GLUE2Entity, GLUE2Endpoint, and GLUE2ComputingEndpoint. Following the example the unrelated model entities ComputingEndpoint and ComputingManager rendered as GLUE2ComputingEndpoint and GLUE2ComputingManager, are separate nodes in the tree. The RDNs of GLUE2ComputingEndpoint and GLUE2ComputingManager are GLUE2EndpointID=X and GLUE2ManagerID=Y.

It was also decided to define an additional structural object class not derived from the model to facilitate grouping of same type of entries under a single node in the tree. This grouping node SHOULD have a GLUE2Group object class with a single attribute GLUE2GroupID. The attribute specifies the class of the entries to be grouped. Note that this attribute is NOT a GLUE 2 ID as defined in GFD.147 (i.e. does not have to be universally unique and not a URI) but is rather a Local ID, meant to identify the grouping, and follows the restrictions of LocalID as in GFD.147.

More explicitly, the only requirement is that the GLUE2GroupID needs to be different for each GLUE2Group entry at a particular point in the tree, such that the full DN of each such entry is unique. A GLUE2Group grouping entry MAY be inserted at any point in the DIT. The grouping facilitates queries by restricting the query to the subtree below the GLUE2Group entry and also improves visual presentation (e.g. avoiding very long object lists in an LDAP browser). Implementations MAY define circumstances in which Groups will always be used, and MAY also define how the GroupIDs are constructed. However it should be emphasised that Groups are specific to an LDAP implementation.

For example, ComputingActivity objects represent jobs in a computing system, and hence may have a very large multiplicity. It may therefore be useful to introduce a GLUE2Group entry with attribute GLUE2GroupID=ComputingActivities as their parent in the tree to allow them to be manipulated and displayed as a unit. It would also be possible to group subsets of jobs, for example by virtual organisation.

When deciding the principles governing construction of the DIT for the GLUE 2 rendering the following considerations were taken into account: it may be convenient for different LDAP implementations to use a different DIT, and for the tree to be restructured as information is aggregated for different purposes. The GLUE 2 abstract information model ensures that any entity can be uniquely referenced by its unique ID, and the rendering choices for the associations in the model via the ForeignKey attributes described earlier allows all entity relations to be followed directly. This means that invariant DNs are not needed to identify entries or to express associations.

Therefore it was agreed that as part of the GLUE 2 LDAP rendering the DIT structure is not mandated. Implementers are free to build their own DIT, and some of the existing production implementations are presented in Appendix A. Implementations SHOULD document the choices they make, and MAY impose further restrictions which are guaranteed within the implementation.

Nevertheless one element of the DIT is mandated: the LDAP tree requires a root DN, also called the base DN. For the DIT of an LDAP tree containing GLUE 2 information the root DN o=glue MUST be used. This enables both GLUE 2 and GLUE 1 information to be present in the same LDAP server since the GLUE 1 tree has the root o=grid. Having different base DNs means that it is not possible to perform single queries across both trees, but this is unlikely to be useful, and the separation also ensures that GLUE 1 clients are not affected by the presence of GLUE 2 information in the same server.

An additional restriction is MANDATORY:

- All GLUE2Extension entries MUST appear immediately below the object they extend, since they are logically part of the object and have only a LocalID and not a globally unique ID.
3.8 Aggregation

In this context, aggregation means composition of different LDAP trees under the same tree. At the time of writing, there is no defined way of aggregating different GLUE 2 documents into one. In this section we will attempt to define a minimal structure that has proven to be effective in some implementations.

The **GLUE2Group** object class can also be used to perform aggregation, instead of being used to group objects of the same kind. This object can be seen as an “insertion point”, that is, a node of the LDAP DIT under which subtrees of other LDAP trees can be merged.

For the purpose of reaching a minimum common aggregation strategy, it is decided that the special **GLUE2Group** with **GLUE2GroupID=grid** is reserved for merging **AdminDomains** hosting **grid Services** from different sources. **GLUE2GroupID=grid** is an insertion point for **Domain** aggregation.

Additionally, two other special **GLUE2GroupID** values “resource” and “services” are reserved for aggregation of services coming from different sources. A "resource" can be used for backward compatibility with GLUE 1 service aggregation, while "services" can be used by those who are not bound to any GLUE 1 implementation. **GLUE2GroupID=resource/services** is an insertion point for **Service** aggregation.

Hence, an implementer wishing to realise **Domain** aggregation MUST follow the restrictions written below:

- The aggregation **GLUE2Group** entry MUST appear directly below the o=glue root. That is, the DN of such an entry MUST be exactly: **GLUE2GroupID=grid,o=glue**
- The aggregated trees should appear directly below such an entry. That is, each LDAP entry belonging to the aggregated tree with its own RDN MUST be concatenated with the suffix **GLUE2GroupID=grid,o=glue**, i.e. <RDN>,GLUE2GroupID=grid,o=glue

Examples of how production implementers realise aggregation are shown in Appendix A.

3.9 OID Assignments

The GLUE 2.0 LDAP implementation utilises the OID subtree of 1.3.6.1.4.1.6757 which is assigned to the Global Grid Forum. The subtree OIDs mapped to each LDAP object class are shown in Table 2. Table 3 gives an example of how OIDs are assigned to attributes for **AdminDomain** and its ancestor LDAP object classes.

OIDs are important for the LDAP server, to uniquely identify and validate object classes and attributes contained in the LDIF entries that will populate the LDAP database. These are placed in the LDAP schema as part of the definition of each LDAP object class and attribute.

Information consumers and LDAP clients will search data using more human readable query parameters, and can therefore completely avoid knowing these OIDs.

It is however important on the server side to have a simple and consistent way of assigning these OIDs so that future extensions of the GLUE 2 schema are easy to implement.

For this reason the numbering scheme in previous versions of this document has been revised. LDAP object classes and their attributes OIDs follow these simple rules:

- For each entity $E$ in GFD.147, an OID is assigned, in the form: 1.3.6.1.4.1.6757.100.1.1.X
  Where $X$ uniquely identifies the entity $E$, $X$ is a positive integer, $X>0$.
- For each attribute $A$ that belongs to the entity $E$ with OID 1.3.6.1.4.1.6757.100.1.1.X, an OID is assigned of the form:
Where Y uniquely identifies the attribute A. Y is a positive integer, Y>0.

3.9.1 Adding new object classes or attributes
Within this framework, an implementer wishing to add a new attribute to an existing LDAP object class 1.3.6.1.4.1.6757.100.1.1.X has to look for the attribute with OID 1.3.6.1.4.1.6757.100.1.1.X.Y with the highest Y, and define a new attribute with OID 1.3.6.1.4.1.6757.100.1.1.X.Z where Z=Y+1. Always add attributes increasing the Y number.

Example: Since GLUE2AdminDomain OID is 1.3.6.1.4.1.6757.100.1.1.6 (see table 2), the new attribute will have OID 1.3.6.1.4.1.6757.100.1.1.6.Y. To choose Y, look at the defined attributes for GLUE2AdminDomain: the highest Y is 3 (see table 3), hence the new attribute will have OID 1.3.6.1.4.1.6757.100.1.1.6.4.

Similarly, an implementer wishing to add a new LDAP object class and its attributes will have to define a new OID 1.3.6.1.4.1.6757.100.1.1.W where W=X+1, X being the highest used value within all the existing object class OIDs 1.3.6.1.4.1.6757.100.1.1.X. Then he will use the OIDs 1.3.6.1.4.1.6757.100.1.1.W.Y, Y>0, for each of the new attributes of the object class.

Example: Check table 2. Let OIDs in that table be 1.3.6.1.4.1.6757.100.1.1.X. The highest X used for object class OIDs is 36. A new object class must have W=X+1, hence one can define the new LDAP object class to have OID 1.3.6.1.4.1.6757.100.1.1.37 and all its attributes to be OID 1.3.6.1.4.1.6757.100.1.1.37.1, OID=1.3.6.1.4.1.6757.100.1.1.37.2, … , and so on.
Table 2: OIDs assigned to LDAP object classes and their corresponding entity in GFD.147.

<table>
<thead>
<tr>
<th>OID</th>
<th>LDAP object class NAME</th>
<th>GFD.147 entity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.27</td>
<td>GLUE2StorageService</td>
<td>StorageService</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.28</td>
<td>GLUE2StorageServiceCapacity</td>
<td>StorageServiceCapacity</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.29</td>
<td>GLUE2StorageAccessProtocol</td>
<td>StorageAccessProtocol</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.30</td>
<td>GLUE2StorageEndpoint</td>
<td>StorageEndpoint</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.31</td>
<td>GLUE2StorageShare</td>
<td>StorageShare</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.32</td>
<td>GLUE2StorageShareCapacity</td>
<td>StorageShareCapacity</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.33</td>
<td>GLUE2StorageManager</td>
<td>StorageManager</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.34</td>
<td>GLUE2DataStore</td>
<td>DataStore</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.35</td>
<td>GLUE2ToComputingService</td>
<td>ToComputingService</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.36</td>
<td>GLUE2Group</td>
<td>Group</td>
</tr>
</tbody>
</table>

Table 3: OIDs assigned to LDAP attributes of the AdminDomain entity and its ancestors. Note that GLUE2EntityID does not exist, as it is an abstract entity, so there is no corresponding LDAP attribute and OID.

<table>
<thead>
<tr>
<th>OID</th>
<th>LDAP Attribute Name</th>
<th>GFD.147 Attribute name</th>
</tr>
</thead>
<tbody>
<tr>
<td>--</td>
<td>GLUE2EntityID</td>
<td>ID</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.1.2</td>
<td>GLUE2EntityName</td>
<td>Name</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.1.3</td>
<td>GLUE2EntityOtherInfo</td>
<td>OtherInfo</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.1.4</td>
<td>GLUE2EntityCreationTime</td>
<td>CreationTime</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.1.5</td>
<td>GLUE2EntityValidity</td>
<td>Validity</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.5.1</td>
<td>GLUE2DomainID</td>
<td>ID</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.5.2</td>
<td>GLUE2DomainDescription</td>
<td>Description</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.5.3</td>
<td>GLUE2DomainWWW</td>
<td>WWW</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.6.1</td>
<td>GLUE2AdminDomainDistributed</td>
<td>Distributed</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.6.2</td>
<td>GLUE2AdminDomainOwner</td>
<td>Owner</td>
</tr>
<tr>
<td>1.3.6.1.4.1.6757.100.1.1.6.3</td>
<td>GLUE2AdminDomainAdminDomainForeignKey</td>
<td>AdminDomain to AdminDomain association</td>
</tr>
</tbody>
</table>

4. Security Considerations

Using LDAP to implement the GLUE 2.0 specification raises several considerations especially in the field of data integrity.

LDAP is not a relational database, thus it cannot ensure relationship integrity. This must be ensured by other means.

LDAP cannot ensure most data types referred in the GLUE 2.0 specification, thus this implementation uses the generic types “DirectoryString” and “Integer” specified in [rfc4517].
5. Author Information
Sergio Andreozzi, INFN
Stephen Burke, RAL
Felix Ehm, CERN
Laurence Field, CERN
Gerson Galang, ARCS
David Horat, CERN
Balazs Konya, Lund University
Maarten Litmaath, CERN
Paul Millar, DESY
JP Navarro, ANL
Florido Paganelli, Lund University

6. Contributors & Acknowledgements
We gratefully acknowledge the contributions made to this document.

7. Intellectual Property Statement
The OGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the OGF Secretariat.

The OGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the OGF Executive Director.

8. Disclaimer
This document and the information contained herein is provided on an “As Is” basis and the OGF disclaims all warranties, express or implied, including but not limited to any warranty that the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

9. Full Copyright Notice
Copyright (C) Open Grid Forum (2008). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works.
However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the OGF or other organizations, except as needed for the purpose of developing Grid Recommendations in which case the procedures for copyrights defined in the OGF Document process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the OGF or its successors or assignees.

10. References


[gLite] http://gridinfo.web.cern.ch/

[GOCDB] https://wiki.egi.eu/wiki/GOCDB_Documentation_Index

[ARC] http://www.nordugrid.org/arc

11. Appendix A: Example LDAP tree structures in existing server side implementations

This section gives a detailed explanation of how the DIT has been realised by the gLite [gLite] and ARC [ARC] Grid middlewares. Historically, these two decided to follow different approaches, but through agreements on how aggregation is done, it has been possible to connect the different trees.

Aggregation is used to combine information taken from different trees into a single DIT. When aggregating DITs from different LDAP servers, entries MAY be added or removed as part of the aggregation process but the DIT relations between existing entries in that tree SHOULD be preserved.

Following the presentation of the general considerations and the minimal restrictions on the DIT as stated in Section 3.7, gLite and ARC defined the following GLUE 2 LDAP DIT structure. The GLUE 2 model can be used to describe Grid information on different levels, from lower to top:

1) gLite “Resource” or ARC “local” level
The bottom level, where no aggregation is usually done, contains a set of services operated on local resources within the same server or local network.

In the gLite model, individual Services and their dependent entities are aggregated under the GLUE2GroupID=resource,o=glue LDAP entry. The software component that generates such
data is called the Resource BDII. Resource BDII publish information about the software installed on a given node – this is often a single complete Service, but may have multiple Services or only a component of a Service distributed over several nodes. Resource BDII are usually not exposed to information consumers directly, and the Resource BDII does NOT contain GLUE2AdminDomain entries, or the related GLUE2Contact and GLUE2Location entries. These entries are added later by another component, the Site BDII, described later. Resource BDII do not publish ComputingActivities for performance reasons.

In the ARC model, all Services are aggregated under the GLUE2GroupID=services,o=glue LDAP entry. The software component that generates such data is called ARIS. ARIS is usually exposed to information consumers directly. This is a key difference with the gLite BDII, as this service level tree MAY contain an LDAP entry with DN GLUE2DomainID=<domainID>,o=glue which contains information about the AdminDomain to which the services belong, where ID is a URI as mandated by GFD.147. The ARIS tree includes the following groupings: GLUE2GroupID=ComputingActivities, GLUE2GroupID=ExecutionEnvironments, GLUE2GroupID=ApplicationEnvironments. These are placed in the trees according to Figure 1.

2) gLite “Site” or Domain level
In the gLite model, local information from different sources like Resource BDII or ARIS local trees representing all Services at a single Grid site (Admin Domain) are then merged or aggregated at an intermediate level by a software component called the Site BDII. The root of its tree is an entry with the DN

GLUE2DomainID=<domainName>,o=glue
where domainName is the unique name of the site as registered in the GOC DB [GOCDB]. This GLUE2AdminDomain entry is created directly by the Site BDII, together with its associated GLUE2Location and GLUE2Contact entries.

Under this DN, the contents of the Resource BDIIIs published under the entry

GLUE2GroupID=resource,o=glue
are aggregated with the DN suffix

GLUE2GroupID=resource,GLUE2DomainID=<domainName>,o=glue.

As a consequence of agreements on the structure of the DIT finalised in 2012, ARIS’s local content under

GLUE2GroupID=services,o=glue
is also aggregated as

GLUE2GroupID=services,GLUE2DomainID=<domainName>,o=glue.

The GLUE 2 information model allows AdminDomains to form a hierarchical structure, and Site BDII MAY follow the same structure, with the information in the Site BDIIIs of sub-sites being aggregated into a single overall Site BDII. In this case the DIT hierarchy directly follows the AdminDomain hierarchy.

All the DN suffixes listed above are usually referred to as “insertion points”, to indicate that a tree coming from a different source with DN GLUE2GroupID=resource,o=glue or GLUE2GroupID=services,o=glue will be inserted starting from the entity with the same GLUE2GroupID attribute value.

The content aggregated by a Site BDII is available for information consumers to use directly, although this is an unusual case. For this reason the site BDII is treated as a Service in its own right, and hence has a Resource BDII which publishes information about it and is aggregated into the Site BDII in the same way as any other Service.

ARC has no equivalent of a Site BDII and therefore no Domain level. The same information is distributed in ARIS, at the local level.
3) gLite “Top” or Global level

In the gLite model, information taken from the Site (or potentially the Resource/local) levels is aggregated on a global level by a component called a Top BDII. The DIT representing the global level naturally accommodates the Domain and local level trees as subtrees. The Top BDII is not a unique service, there may be many instances, and the sites and services included in a Top BDII may vary according to its configuration. The standard configuration aggregates all information from certified sites as defined in the GOC DB.

In this context, the insertion point GLUE2GroupID=grid,o=glue aggregates GLUE2DomainID=<domainName>,o=glue trees coming from different Site BDIIIs, as

GLUE2DomainID=<domainName>,GLUE2GroupID=grid,o=glue.

Services contained in such trees will be aggregated with suffix

GLUE2GroupID=resource,GLUE2DomainID=<domainName>,GLUE2GroupID=grid,o=glue

A Top BDII is again treated as a Service in its own right. It therefore has a Resource BDII to publish information about itself, which is aggregated into the Site BDII for the site which hosts it, and thence into all the Top BDIIIs.

ARC does not have an equivalent to a Top BDII. All the information needed for the aggregation should be fetched by a Top BDII directly from ARIS. While an agreement on how to achieve this has been already established, work in this respect is still ongoing.

The DITs for the three levels are shown on the figures below.

Notable properties of the common DIT structure are as follows:

- **GLUE2AdminDomain** entries only aggregate Services as their child entries in the tree when those Services are managed by that Domain.

- A **GLUE2Service** entry SHOULD aggregate all Service related entries describing the specific Service via placing all those entries under its subtree; unrelated entries MUST NOT appear there.

**Local-level DIT (gLite Resource BDII or ARC ARIS):**

- gLite and ARC: Right beneath the o=glue root it contains a GLUE2Group entry with GLUE2GroupID=resource or GLUE2GroupID=services. This entry usually accommodates all the local Services with their complete subtrees.

- In ARC, right beneath the o=glue root it MAY contain GLUE2AdminDomain or GLUE2UserDomain entries.
Figure 1: Example DIT structure of a local information source (gLite Resource BDII or ARIS ARC) publishing Service, Computing Service and Storage Service together with Domain information.
Domain-level DIT (gLite Site-BDII):

- Right beneath the o=glue root it contains only one GLUE2AdminDomain entry and might have a GLUE2UserDomain
- The GLUE2AdminDomain entry can have a GLUE2Group entry with GLUE2GroupID=resource or GLUE2GroupID=services attributes that contains all the Service trees from the local level belonging to that Domain.
- Right beneath the o=glue root it contains a GLUE2Group entry with GLUE2GroupID=resource. This entry is used to describe the LDAP service itself.

**Figure 2:** Example DIT structure for Site BDII aggregation of local information sources. Aggregation of the services from multiple local sources is done by moving every subtree of the GLUE2GroupID=resource or GLUE2GroupID=services entries under the GLUE2AdminDomain entry.
Global-level DIT (gLite Top BDII):

- Right beneath the o=glue root it contains a GLUE2Group entry with GLUE2GroupID=grid. This entry accommodates all the local AdminDomains and UserDomains with their complete subtrees.
- Right beneath the o=glue root it contains a GLUE2Group entry with GLUE2GroupID=resource. This entry is used to describe the LDAP service itself.
Figure 3: Example DIT structure for Top BDII aggregation from both Site BDII and local level information sources.
12. Appendix B: LDAP Schema

A standalone file of the following LDAP schema is available at:
http://schemas.ogf.org/glue/2014/07/GLUE20_LDAP.schema

```plaintext
# Copyright (C) Open Grid Forum (2009). All Rights Reserved.
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#
# File:    schema/00-Version.schema
# URL:     http://redmine.ogf.org/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: - Version file
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
#
# Schema Version: 2.0
# Last updated: 2014-07-02
#
# File:    schema/01-Group.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: - Group
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
#
# Group

attributetype ( 1.3.6.1.4.1.6757.100.1.1.36.1
   NAME ( 'GLUE2GroupID' 'GLUE2GroupName' 'GLUE2GroupLocalID' )
   DESC 'Local ID, not unique'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE
 )

objectclass ( 1.3.6.1.4.1.6757.100.1.1.36
   NAME 'GLUE2Group'
   DESC 'Group object'
   STRUCTURAL
   MUST GLUE2GroupID
 )
```

# File:    schema/5-01-Entity.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 5.1 Entity
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# Entity <<abstract>>

attributetype ( 1.3.6.1.4.1.6757.100.1.1.1.2
NAME 'GLUE2EntityName'
DESC 'Human-readable name'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.1.3
NAME 'GLUE2EntityOtherInfo'
DESC 'Placeholder to publish info that does not fit in any other attribute'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.1.4
NAME 'GLUE2EntityCreationTime'
DESC 'Timestamp describing when the entity instance was generated'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.1.5
NAME 'GLUE2EntityValidity'
DESC 'The duration of the information presented in this object after its creation time'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )

objectclass ( 1.3.6.1.4.1.6757.100.1.1.1
NAME 'GLUE2Entity'
DESC 'Base class'
ABSTRACT
MAY ( GLUE2EntityName $ GLUE2EntityOtherInfo $ GLUE2EntityCreationTime $ GLUE2EntityValidity ) )

# File: schema/5-02-Extension.schema
# URL: http://forge.gridforum.org/sf/projects/glue-wg
# Doc: GLUE Specification 2.0 (March 3, 2009)
# Section: 5.2 Extension
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
# Extension
<table>
<thead>
<tr>
<th>attributetype</th>
<th>OID</th>
<th>NAME</th>
<th>DESC</th>
<th>EQUALITY</th>
<th>SUBSTR</th>
<th>SYNTAX</th>
<th>SINGLE-VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.3.6.1.4.1.6757.100.1.1.2.1</td>
<td>GLUE2ExtensionLocalID</td>
<td>A local identifier unique within the class instance to which it is associated</td>
<td>caseExactMatch</td>
<td>caseExactSubstringsMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.15</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.6757.100.1.1.2.2</td>
<td>GLUE2ExtensionKey</td>
<td>An identifier local to the container class instance</td>
<td>caseExactMatch</td>
<td>caseExactSubstringsMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.15</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.6757.100.1.1.2.3</td>
<td>GLUE2ExtensionValue</td>
<td>A value for the attribute named by the key</td>
<td>caseExactMatch</td>
<td>caseExactSubstringsMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.15</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.6757.100.1.1.2.4</td>
<td>GLUE2ExtensionEntityForeignKey</td>
<td>ForeignKey</td>
<td>caseExactMatch</td>
<td>caseExactSubstringsMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.15</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>objectclass</th>
<th>OID</th>
<th>NAME</th>
<th>DESC</th>
<th>STRUCTURAL</th>
<th>MUST</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.3.6.1.4.1.6757.100.1.1.2</td>
<td>GLUE2Extension</td>
<td>A key/value pair enabling the association of extra information</td>
<td></td>
<td>GLUE2ExtensionLocalId $ GLUE2ExtensionKey $ GLUE2ExtensionValue $ GLUE2ExtensionEntityForeignKey</td>
<td></td>
</tr>
</tbody>
</table>

# File: schema/5-03-Location.schema
# URL: http://forge.gridforum.org/sf/projects/glue-wg
# Doc: GLUE Specification 2.0 (March 3, 2009)
# Section: 5.3 Location
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
# Location inherits from Entity

<table>
<thead>
<tr>
<th>attributetype</th>
<th>OID</th>
<th>NAME</th>
<th>DESC</th>
<th>EQUALITY</th>
<th>SUBSTR</th>
<th>SYNTAX</th>
<th>SINGLE-VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.3.6.1.4.1.6757.100.1.1.3.1</td>
<td>GLUE2LocationID</td>
<td>Globally unique ID</td>
<td>caseExactMatch</td>
<td>caseExactSubstringsMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.15</td>
<td></td>
</tr>
</tbody>
</table>
attributetype (1.3.6.1.4.1.6757.100.1.1.3.2
  NAME 'GLUE2LocationAddress'
  DESC 'Street address'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype (1.3.6.1.4.1.6757.100.1.1.3.3
  NAME 'GLUE2LocationPlace'
  DESC 'Name of town/city'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype (1.3.6.1.4.1.6757.100.1.1.3.4
  NAME 'GLUE2LocationCountry'
  DESC 'Country name'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype (1.3.6.1.4.1.6757.100.1.1.3.5
  NAME 'GLUE2LocationPostCode'
  DESC 'Postal code'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype (1.3.6.1.4.1.6757.100.1.1.3.6
  NAME 'GLUE2LocationLatitude'
  DESC 'The position of a place north or south of the equator measured from -90 degrees to +90 degrees with positive values going north and negative values going south'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype (1.3.6.1.4.1.6757.100.1.1.3.7
  NAME 'GLUE2LocationLongitude'
  DESC 'The position of a place east or west of Greenwich, England measured from -180 degrees to +180 degrees with positive values going east and negative values going west'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype (1.3.6.1.4.1.6757.100.1.1.3.8
  NAME 'GLUE2LocationServiceForeignKey'
)
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15

attributetype ( 1.3.6.1.4.1.6757.100.1.1.3.9
NAME 'GLUE2LocationDomainForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.3
NAME 'GLUE2Location'
SUP GLUE2Entity
DESC 'A geographical position'
STRUCTURAL
MUST GLUE2LocationID
MAY ( GLUE2LocationAddress $ GLUE2LocationPlace $ GLUE2LocationCountry $ GLUE2LocationPostCode $ GLUE2LocationLatitude $ GLUE2LocationLongitude $ GLUE2LocationServiceForeignKey $ GLUE2LocationDomainForeignKey )
)

# File:    schema/5-04-Contact.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 5.4 Contact
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# Contact inherits from Entity

attributetype ( 1.3.6.1.4.1.6757.100.1.1.4.1
NAME 'GLUE2ContactID'
DESC 'Globally unique ID'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.4.2
NAME 'GLUE2ContactDetail'
DESC 'URL embedding the contact information. The syntax of URI depends on the communication channel'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.4.3
NAME 'GLUE2ContactType'
DESC 'Type of contact'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)
SINGLE-VALUE
}

attributetype ( 1.3.6.1.4.1.6757.100.1.1.4.4
NAME 'GLUE2ContactServiceForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.4.5
NAME 'GLUE2ContactDomainForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.4
NAME 'GLUE2Contact'
SUP GLUE2Entity
DESC 'Information enabling to establish a communication with a person or group of persons part of a domain'
STRUCTURAL
MUST ( GLUE2ContactID $ GLUE2ContactDetail $ GLUE2ContactType )
MAY ( GLUE2ContactServiceForeignKey $ GLUE2ContactDomainForeignKey )
)

# File:    schema/5-05-Domain.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 5.5 Domain, 5.5.1 AdminDomain and 5.5.2 UserDomain
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
# Domain <<abstract>> inherits from Entity
attributetype ( 1.3.6.1.4.1.6757.100.1.1.5.1
NAME 'GLUE2DomainID'
DESC 'A description of the domain'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.5.2
NAME 'GLUE2DomainDescription'
DESC 'A description of the domain'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.5.3
NAME 'GLUE2DomainWWW'
DESC 'The URL identifying a web page with more information about the domain'
EQUALITY caseExactMatch
)
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.5
NAME 'GLUE2Domain'
SUP GLUE2Entity
DESC 'A Domain'
STRUCTURAL
MUST GLUE2DomainID
MAY ( GLUE2DomainDescription $ GLUE2DomainWWW )
)

#########################################################
# AdminDomain inherits from Domain
#########################################################
attributetype ( 1.3.6.1.4.1.6757.100.1.1.6.1
NAME 'GLUE2AdminDomainDistributed'
DESC 'True if the services managed by the AdminDomain are considered
geographically distributed by the administrators themselves'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.6.2
NAME 'GLUE2AdminDomainOwner'
DESC 'Owner of the managed resources'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.6.4
NAME 'GLUE2AdminDomainAdminDomainForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.6
NAME 'GLUE2AdminDomain'
DESC 'A collection of actors that can be assigned with
administrative
roles and privileges to services via policies. An AdminDomain manages
services that can be geographically distributed, nevertheless a
primary
location should be identified.'
SUP GLUE2Domain
STRUCTURAL
MAY ( GLUE2AdminDomainDistributed $ GLUE2AdminDomainOwner $
GLUE2AdminDomainAdminDomainForeignKey )
)

#########################################################
# UserDomain inherits from Domain
#########################################################
attributetype ( 1.3.6.1.4.1.6757.100.1.1.7.1
NAME 'GLUE2UserDomainLevel'
DESC 'The number of hops to reach the root for hierarchically
organized
domains described by the composed by association (0 is for the root).'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.7.2
NAME 'GLUE2UserDomainUserManager'
DESC 'The Endpoint ID managing the users part of the domain and the
related attributes such as groups or roles'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.7.3
NAME 'GLUE2UserDomainMember'
DESC 'An identifier for a user in this user domain'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.7.5
NAME 'GLUE2UserDomainUserDomainForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.7
NAME 'GLUE2UserDomain'
DESC 'A collection of actors that can be assigned with user roles
and
privileges to services or shares via policies'
SUP GLUE2Domain
STRUCTURAL
MAY ( GLUE2UserDomainLevel $ GLUE2UserDomainUserManager $ GLUE2UserDomainUserDomainMember $ GLUE2UserDomainUserDomainForeignKey )
)

# File:    schema/5-06-Service.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 5.6 Service
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
# Service inherits from Entity

attributetype ( 1.3.6.1.4.1.6757.100.1.1.8.1
NAME 'GLUE2ServiceID'
DESC 'Globally unique ID'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.8.2
NAME 'GLUE2ServiceCapability'

DESC 'The provided capability according to the OGSA architecture (it is given by the sum of all the capabilities provided by the related endpoints)'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
}

attributetype ( 1.3.6.1.4.1.6757.100.1.1.8.3
NAME 'GLUE2ServiceType'
DESC 'The type of service according to a middleware classification'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.8.4
NAME 'GLUE2ServiceQualityLevel'
DESC 'Maturity of the service in terms of quality of the software components'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.8.5
NAME 'GLUE2ServiceStatusInfo'
DESC 'A URL providing additional information like monitoring aspects'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.8.6
NAME 'GLUE2ServiceComplexity'
DESC 'Human-readable summary description of the complexity in terms of the number of endpoint types, shares and resources. The syntax should be: EndpointType=X, Share=Y, Resource=Z.'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.8.7
NAME 'GLUE2ServiceAdminDomainForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.8.8
NAME 'GLUE2ServiceServiceForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)
objectclass ( 1.3.6.1.4.1.6757.100.1.1.8
  NAME 'GLUE2Service'
  SUP GLUE2Entity
  DESC "An abstracted, logical view of actual software components that participate in the creation of an entity providing one or more functionalities useful in a Grid environment. A service exposes zero or more endpoints having well-defined interfaces, zero or more managers and the related resources. The service is autonomous and denotes a weak aggregation among endpoints, the underlying managers and the related resources, and the defined shares. The service enables to identify the whole set of entities providing the functionality with a persistent name.'
  STRUCTURAL
  MUST ( GLUE2ServiceID $ GLUE2ServiceType $ GLUE2ServiceQualityLevel
  $ GLUE2ServiceAdminDomainForeignKey )
  MAY ( GLUE2ServiceCapability $ GLUE2ServiceStatusInfo $ GLUE2ServiceComplexity $ GLUE2ServiceServiceForeignKey )
)

# File:    schema/5-07-Endpoint.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 5.7 Endpoint
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# Endpoint inherits from Entity
attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.1
  NAME 'GLUE2EndpointID'
  DESC 'Globally unique ID'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.2
  NAME 'GLUE2EndpointURL'
  DESC 'Network location of the endpoint to contact the related service'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.3
  NAME 'GLUE2EndpointCapability'
  DESC 'The provided capability according to the OGSA architecture'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.4
NAME 'GLUE2EndpointTechnology'
DESC 'Technology used to implement the endpoint'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.5
NAME 'GLUE2EndpointInterfaceName'
DESC 'Identification of a type and version of the interface'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.6
NAME 'GLUE2EndpointInterfaceVersion'
DESC 'The version of the primary interface protocol'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.7
NAME 'GLUE2EndpointInterfaceExtension'
DESC 'Identification of an extension to the interface'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.8
NAME 'GLUE2EndpointWSDL'
DESC 'URL of the WSDL document describing the offered interface (applies to Web Services endpoint)'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.9
NAME 'GLUE2EndpointSupportedProfile'
DESC 'URI identifying a supported profile'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.10
NAME 'GLUE2EndpointSemantics'
DESC 'URI of a document providing a human-readable description of the semantics of the endpoint functionalities'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.11
NAME 'GLUE2EndpointImplementor'

```
DESC 'Main organization implementing this software component'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.12
NAME 'GLUE2EndpointImplementationName'
DESC 'Name of the implementation'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.13
NAME 'GLUE2EndpointImplementationVersion'
DESC 'Version of the implementation (e.g., major version.minor version.patch version)'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.14
NAME 'GLUE2EndpointQualityLevel'
DESC 'Maturity of the endpoint in terms of quality of the software components'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.15
NAME 'GLUE2EndpointHealthState'
DESC 'A state representing the health of the endpoint in terms of its capability of properly delivering the functionalities'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.16
NAME 'GLUE2EndpointHealthStateInfo'
DESC 'Textual explanation of the state endpoint'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.17
NAME 'GLUE2EndpointServingState'
DESC 'A state specifying if the endpoint is accepting new requests and if it is serving the already accepted requests'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.18
NAME 'GLUE2EndpointStartTime'
DESC 'The timestamp for the start time of the endpoint'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.19
NAME 'GLUE2EndpointIssuerCA'
DESC 'Distinguished name of Certification Authority issuing the certificate for the endpoint'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.20
NAME 'GLUE2EndpointTrustedCA'
DESC 'Distinguished name of the trusted Certification Authority'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.21
NAME 'GLUE2EndpointDowntimeAnnounce'
DESC 'The timestamp for the announcement of the next scheduled downtime'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.22
NAME 'GLUE2EndpointDowntimeStart'
DESC 'The starting timestamp of the next scheduled downtime'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.23
NAME 'GLUE2EndpointDowntimeEnd'
DESC 'The ending timestamp of the next scheduled downtime'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.24
NAME 'GLUE2EndpointDowntimeInfo'
DESC 'Description of the next scheduled downtime'
EQUALITY caseExactMatch
attributetype ( 1.3.6.1.4.1.6757.100.1.1.9.25
   NAME 'GLUE2EndpointServiceForeignKey'
   DESC 'ForeignKey'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE
 )

objectclass ( 1.3.6.1.4.1.6757.100.1.1.9
   NAME 'GLUE2Endpoint'
   SUP GLUE2Entity
   DESC 'A network location having a well-defined interface and exposing the service functionalities'
   STRUCTURAL
   MUST ( GLUE2EndpointID $ GLUE2EndpointURL $ GLUE2EndpointInterfaceName $ GLUE2EndpointQualityLevel $ GLUE2EndpointHealthState $ GLUE2EndpointServingState $ GLUE2EndpointServiceForeignKey )
   MAY ( GLUE2EndpointCapability $ GLUE2EndpointTechnology $ GLUE2EndpointInterfaceVersion $ GLUE2EndpointInterfaceExtension $ GLUE2EndpointWSDL $ GLUE2EndpointSupportedProfile $ GLUE2EndpointSemantics $ GLUE2EndpointImplementor $ GLUE2EndpointImplementationName $ GLUE2EndpointImplementationVersion $ GLUE2EndpointHealthStateInfo $ GLUE2EndpointStartTime $ GLUE2EndpointIssuerCA $ GLUE2EndpointTrustedCA $ GLUE2EndpointDowntimeAnnounce $ GLUE2EndpointDowntimeStart $ GLUE2EndpointDowntimeEnd $ GLUE2EndpointDowntimeInfo )
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.10.1
   NAME 'GLUE2ShareID'
   DESC 'Globally unique ID'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE
 )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.10.2
   NAME 'GLUE2ShareDescription'
   DESC 'Description of this share'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE
 )
attributetype ( 1.3.6.1.4.1.6757.100.1.1.10.3
  NAME 'GLUE2ShareEndpointForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.10.4
  NAME 'GLUE2ShareResourceForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.10.5
  NAME 'GLUE2ShareServiceForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )

objectclass ( 1.3.6.1.4.1.6757.100.1.1.10
  NAME 'GLUE2Share'
  SUP GLUE2Entity
  DESC 'A share'
  STRUCTURAL
  MUST ( GLUE2ShareID $ GLUE2ShareServiceForeignKey )
  MAY ( GLUE2ShareDescription $ GLUE2ShareEndpointForeignKey $ GLUE2ShareResourceForeignKey )
  )

# File:     schema/5-09-Manager.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 5.9 Manager
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
# Manager <<abstract>> inherits from Entity

attributetype ( 1.3.6.1.4.1.6757.100.1.1.11.1
  NAME 'GLUE2ManagerID'
  DESC 'Globally unique ID'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.11.2
  NAME 'GLUE2ManagerProductName'
  DESC 'The name of the software product which implements the Manager functionality'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )
attributetype ( 1.3.6.1.4.1.6757.100.1.1.11.3
    NAME 'GLUE2ManagerProductVersion'
    DESC 'The version of the software product which implements the Manager'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.11.4
    NAME 'GLUE2ManagerServiceForeignKey'
    DESC 'ForeignKey'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

objectclass ( 1.3.6.1.4.1.6757.100.1.1.11
    NAME 'GLUE2Manager'
    SUP GLUE2Entity
    DESC 'A manager'
    STRUCTURAL
    MUST ( GLUE2ManagerID $ GLUE2ManagerProductName $ GLUE2ManagerServiceForeignKey )
    MAY GLUE2ManagerProductVersion )

# File: schema/5-10-Resource.schema
# URL: http://forge.gridforum.org/sf/projects/glue-wg
# Doc: GLUE Specification 2.0 (March 3, 2009)
# Section: 5.10 Resource
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
# Resource <<abstract>> inherits from Entity
# No attributes

attributetype ( 1.3.6.1.4.1.6757.100.1.1.12.1
    NAME 'GLUE2ResourceID'
    DESC 'Globally unique ID'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.12.2
    NAME 'GLUE2ResourceManagerForeignKey'
    DESC 'ForeignKey'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

objectclass ( 1.3.6.1.4.1.6757.100.1.1.12
    NAME 'GLUE2Resource'
    SUP GLUE2Entity
    DESC 'A resource'
    STRUCTURAL
    MUST ( GLUE2ResourceID $ GLUE2ResourceManagerForeignKey )
    MAY GLUE2ResourceProductVersion )
SUP GLUE2Entity
DESC 'A resource'
STRUCTURAL
MUST ( GLUE2ResourceId $ GLUE2ResourceManagerForeignKey )

# File: schema/5-11-Activity.schema
# URL: http://forge.gridforum.org/sf/projects/glue-wg
# Doc: GLUE Specification 2.0 (March 3, 2009)
# Section: 5.11 Activity
# Authors: Laurence Field (laurence.field@cern.ch), CERN
# David Horat (david.horat@cern.ch), CERN
# Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# Activity inherits from Entity
# No attributes

attributetype ( 1.3.6.1.4.1.6757.100.1.1.13.1
    NAME 'GLUE2ActivityID'
    DESC 'Globally unique ID'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.13.2
    NAME 'GLUE2ActivityUserDomainForeignKey'
    DESC 'ForeignKey'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.13.3
    NAME 'GLUE2ActivityEndpointForeignKey'
    DESC 'ForeignKey'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.13.4
    NAME 'GLUE2ActivityShareForeignKey'
    DESC 'ForeignKey'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.13.5
    NAME 'GLUE2ActivityResourceForeignKey'
    DESC 'ForeignKey'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.13.6
NAME 'GLUE2ActivityActivityForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15

objectclass ( 1.3.6.1.4.1.6757.100.1.1.13
NAME 'GLUE2Activity'
SUP GLUE2Entity
DESC 'An activity is a unit of work managed by a Service and submitted via an Endpoint'
STRUCTURAL
MUST GLUE2ActivityID
MAY ( GLUE2ActivityUserDomainForeignKey $ GLUE2ActivityEndpointForeignKey $ GLUE2ActivityShareForeignKey $ GLUE2ActivityResourceForeignKey $ GLUE2ActivityActivityForeignKey )
)

# File:    schema/5-12-Policy.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 5.12 Policy, 5.12.1 AccessPolicy, 5.12.2 MappingPolicy
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelili@hep.lu.se), Lund University
# Policy <<abstract>> inherits from Entity
attributetype ( 1.3.6.1.4.1.6757.100.1.1.14.1
NAME 'GLUE2PolicyID'
DESC 'Scheme adopted to define the policy rules'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.14.2
NAME 'GLUE2PolicyScheme'
DESC 'Scheme adopted to define the policy rules'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.14.3
NAME 'GLUE2PolicyRule'
DESC 'A policy rule'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.14.4
NAME 'GLUE2PolicyUserDomainForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
objectclass ( 1.3.6.1.4.1.6757.100.1.1.14
  NAME 'GLUE2Policy'
  SUP GLUE2Entity
  DESC 'A policy'
  STRUCTURAL
  MUST ( GLUE2PolicyID $ GLUE2PolicyScheme $ GLUE2PolicyRule )
  MAY ( GLUE2PolicyUserDomainForeignKey )
)

#################################################################
# AccessPolicy inherits from Policy
#################################################################
attributetype ( 1.3.6.1.4.1.6757.100.1.1.15.1
  NAME 'GLUE2AccessPolicyEndpointForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.15
  NAME 'GLUE2AccessPolicy'
  DESC 'Statements, rules or assertions that provide coarse-granularity information about the authorization of access by groups of actors to an Endpoint'
  SUP 'GLUE2Policy'
  STRUCTURAL
  MUST GLUE2AccessPolicyEndpointForeignKey
)

#################################################################
# MappingPolicy inherits from Policy
#################################################################
attributetype ( 1.3.6.1.4.1.6757.100.1.1.16.1
  NAME 'GLUE2MappingPolicyShareForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.16
  NAME 'GLUE2MappingPolicy'
  DESC 'Statements, rules or assertions that provide coarse-granularity information about the mapping or UserDomain requests to a Share'
  SUP 'GLUE2Policy'
  STRUCTURAL
  MUST GLUE2MappingPolicyShareForeignKey
)

# File:    schema/6-01-ComputingService.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 6.1 ComputingService
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#         David Horat (david.horat@cern.ch), CERN
#         Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
# ComputingService inherits from Service

attributetype ( 1.3.6.1.4.1.6757.100.1.1.17.1
    NAME 'GLUE2ComputingServiceTotalJobs'
    DESC 'Number of total jobs'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.17.2
    NAME 'GLUE2ComputingServiceRunningJobs'
    DESC 'Number of running jobs'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.17.3
    NAME 'GLUE2ComputingServiceWaitingJobs'
    DESC 'Number of jobs waiting in the underlying computing managers (i.e., Local Resource Manager System or LRMS)'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.17.4
    NAME 'GLUE2ComputingServiceStagingJobs'
    DESC 'Number of jobs that are staging files in/out'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.17.5
    NAME 'GLUE2ComputingServiceSuspendedJobs'
    DESC 'Number of jobs which started their execution, but are suspended (e.g., for preemption)'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.17.6
    NAME 'GLUE2ComputingServicePreLRMSWaitingJobs'
    DESC 'Number of jobs that are in the Grid layer waiting to be passed to the underlying computing manager (i.e., LRMS)'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.17
    NAME 'GLUE2ComputingService'
)
The computing service is autonomous and denotes a weak aggregation among computing endpoints, the underlying computing managers and related execution environments, and the defined computing shares. The computing service enables to identify the whole set of entities providing the computing functionality with a persistent name.

SUP GLUE2Service
STRUCTURAL
MAY ( GLUE2ComputingServiceTotalJobs $
$ GLUE2ComputingServiceRunningJobs $
$ GLUE2ComputingServiceWaitingJobs $
$ GLUE2ComputingServiceStagingJobs $
$ GLUE2ComputingServiceSuspendedJobs $
$ GLUE2ComputingServicePreLRMSWaitingJobs )

# File:    schema/6-02-ComputingEndpoint.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 6.2 ComputingEndpoint
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#         David Horat (david.horat@cern.ch), CERN
#         Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# ComputingEndpoint inherits from Endpoint
attributetype ( 1.3.6.1.4.1.6757.100.1.1.18.1
NAME 'GLUE2ComputingEndpointStaging'
DESC 'Supported staging functionalities'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.18.2
NAME 'GLUE2ComputingEndpointJobDescription'
DESC 'Supported type of job description language'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.18.3
NAME 'GLUE2ComputingEndpointTotalJobs'
DESC 'Number of total jobs'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.18.4
NAME 'GLUE2ComputingEndpointRunningJobs'
DESC 'Number of running jobs'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.18.5

NAME 'GLUE2ComputingEndpointWaitingJobs'
DESC 'Number of jobs waiting in the underlying computing managers (i.e., Local Resource Manager System or LRMS)'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.18.6
NAME 'GLUE2ComputingEndpointStagingJobs'
DESC 'Number of jobs that are staging files in/out'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.18.7
NAME 'GLUE2ComputingEndpointSuspendedJobs'
DESC 'Number of jobs which started their execution, but are suspended (e.g., for preemption)'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.18.8
NAME 'GLUE2ComputingEndpointPreLRMSWaitingJobs'
DESC 'Number of jobs that are in the Grid layer waiting to be passed to the underlying computing manager (i.e., LRMS)'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.18.9
NAME 'GLUE2ComputingEndpointComputingServiceForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE

objectclass ( 1.3.6.1.4.1.6757.100.1.1.18
NAME 'GLUE2ComputingEndpoint'
DESC 'Endpoint for creating, monitoring, and controlling computational activities called jobs; it can be used to expose also complementary capabilities (e.g., reservation, proxy manipulation)'
SUP GLUE2Endpoint
STRUCTURAL
MAY ( GLUE2ComputingEndpointStaging $
GLUE2ComputingEndpointJobDescription $
GLUE2ComputingEndpointTotalJobs $
GLUE2ComputingEndpointRunningJobs $
GLUE2ComputingEndpointWaitingJobs $
GLUE2ComputingEndpointStagingJobs $
)
## GLUE2ComputingEndpointSuspendedJobs

GLUE2ComputingEndpointPreLRMSWaitingJobs

GLUE2ComputingEndpointComputingServiceForeignKey

---

| File:    | schema/6-03-ComputingShare.schema |
| URL:     | http://forge.gridforum.org/sf/projects/glue-wg |
| Doc:     | GLUE Specification 2.0 (March 3, 2009) |
| Section: | 6.3 ComputingShare |
| Authors: | Laurence Field (laurence.field@cern.ch), CERN |
|          | David Horat (david.horat@cern.ch), CERN |
|          | Florido Paganelli (florido.paganelli@hep.lu.se), Lund University |

### ComputingShare inherits from Share

| attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.1 |
| NAME 'GLUE2ComputingShareMappingQueue' |
| DESC 'Name of a queue available in the underlying computing manager (i.e., LRMS) where jobs of this share are submitted (different shares can be mapped into the same queue; it is not foreseen that a single share can be mapped into many different queues)' |
| EQUALITY caseExactMatch |
| SUBSTR caseExactSubstringsMatch |
| SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 |
| SINGLE-VALUE |

| attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.2 |
| NAME 'GLUE2ComputingShareMaxWallTime' |
| DESC 'The maximum obtainable wall clock time per slot that can be granted to the job upon user request (unnormalized value)' |
| EQUALITY integerMatch |
| ORDERING integerOrderingMatch |
| SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 |
| SINGLE-VALUE |

| attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.3 |
| NAME 'GLUE2ComputingShareMaxMultiSlotWallTime' |
| DESC 'The maximum obtainable wall clock time limit that MAY be granted to a multi-slot job upon user request' |
| EQUALITY integerMatch |
| ORDERING integerOrderingMatch |
| SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 |
| SINGLE-VALUE |

| attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.4 |
| NAME 'GLUE2ComputingShareMinWallTime' |
| DESC 'The minimum wall clock time per slot for a job (unnormalized value); if a job requests a lower time, than it can be rejected; if a job requests at least this value, but runs for a shorter time, than it might be accounted for this value' |
| EQUALITY integerMatch |
| ORDERING integerOrderingMatch |
| SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 |
| SINGLE-VALUE |

| attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.5 |
| NAME 'GLUE2ComputingShareDefaultWallTime' |
DESC 'The default wall clock time per slot allowed to a job by the computing manager (i.e., LRMS) if no limit is requested in the job submission description. Once this time is expired the job will most likely be killed or removed from the queue (unnormalized value)'

EQUALITY  integerMatch
ORDERING  integerOrderingMatch
SYNTAX  1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.6
  NAME 'GLUE2ComputingShareMaxCPUTime'
  DESC 'The maximum obtainable CPU time that can be granted to the job upon user request per slot (unnormalized value)'
  EQUALITY  integerMatch
  ORDERING  integerOrderingMatch
  SYNTAX  1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.7
  NAME 'GLUE2ComputingShareMaxTotalCPUTime'
  DESC 'The maximum obtainable CPU time that can be granted to the job upon user request across all assigned slots; this property is a limit for the sum of the CPU time used in all the slots occupied by a multi-slot job (unnormalized value)'
  EQUALITY  integerMatch
  ORDERING  integerOrderingMatch
  SYNTAX  1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.8
  NAME 'GLUE2ComputingShareMinCPUTime'
  DESC 'The minimum CPU time per slot for a job (unnormalized value); if a job requests a lower time, than it can be rejected; if a job requests at least this value, but uses the CPU for a shorter time, than it might be accounted for this value'
  EQUALITY  integerMatch
  ORDERING  integerOrderingMatch
  SYNTAX  1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.9
  NAME 'GLUE2ComputingShareDefaultCPUTime'
  DESC 'The default CPU time per slot allowed to each job by the computing manager (i.e., LRMS) if no limit is requested in the job submission description (unnormalized value)'
  EQUALITY  integerMatch
  ORDERING  integerOrderingMatch
  SYNTAX  1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.10
  NAME 'GLUE2ComputingShareMaxTotalJobs'
  DESC 'The maximum allowed number of jobs in this share'
  EQUALITY  integerMatch
  ORDERING  integerOrderingMatch
  SYNTAX  1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.11
  NAME 'GLUE2ComputingShareMaxRunningJobs'
  DESC 'The maximum allowed number of jobs in running state in this
  share'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.12
  NAME 'GLUE2ComputingShareMaxWaitingJobs'
  DESC 'The maximum allowed number of jobs in waiting state in this
  share'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.13
  NAME 'GLUE2ComputingShareMaxPreLRMSStructingJobs'
  DESC 'The maximum allowed number of jobs that are in the Grid layer
  waiting to be passed to the underlying computing manager (i.e., LRMS)
  for this share'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.14
  NAME 'GLUE2ComputingShareMaxUserRunningJobs'
  DESC 'The maximum allowed number of jobs in running state per Grid
  user
  in this share'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.15
  NAME 'GLUE2ComputingShareMaxSlotsPerJob'
  DESC 'The maximum number of slots which could be allocated to a
  single
  job (defined to be 1 for a computing service accepting only single-
  slot
  jobs)'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.16
  NAME 'GLUE2ComputingShareMaxStageInStreams'
  DESC 'The maximum number of streams to stage files in'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SINGLE-VALUE
)
<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Description</th>
<th>Equality</th>
<th>Ordering</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>GLUE2ComputingShareMaxStageOutStreams</code></td>
<td>The maximum number of streams to stage files out</td>
<td>integerMatch</td>
<td>integerOrderingMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.27</td>
</tr>
<tr>
<td><code>GLUE2ComputingShareSchedulingPolicy</code></td>
<td>Implied scheduling policy of the share</td>
<td>caseExactMatch</td>
<td>caseExactSubstringsMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.15</td>
</tr>
<tr>
<td><code>GLUE2ComputingShareMaxMainMemory</code></td>
<td>The maximum RAM that a job can use</td>
<td>integerMatch</td>
<td>integerOrderingMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.27</td>
</tr>
<tr>
<td><code>GLUE2ComputingShareGuaranteedMainMemory</code></td>
<td>The amount of physical RAM that a job is guaranteed to have available for its use</td>
<td>integerMatch</td>
<td>integerOrderingMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.27</td>
</tr>
<tr>
<td><code>GLUE2ComputingShareMaxVirtualMemory</code></td>
<td>The maximum virtual memory that a job can use</td>
<td>integerMatch</td>
<td>integerOrderingMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.27</td>
</tr>
<tr>
<td><code>GLUE2ComputingShareGuaranteedVirtualMemory</code></td>
<td>The amount of virtual memory that a job is guaranteed to have available for its use</td>
<td>integerMatch</td>
<td>integerOrderingMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.27</td>
</tr>
<tr>
<td><code>GLUE2ComputingShareMaxDiskSpace</code></td>
<td>The maximum disk space that a job can use excluding shared</td>
<td>integerMatch</td>
<td>integerOrderingMatch</td>
<td>1.3.6.1.4.1.1466.115.121.1.27</td>
</tr>
</tbody>
</table>
area

such as cache'

EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE

).attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.24
  NAME 'GLUE2ComputingShareDefaultStorageService'
  DESC 'ID of the default Storage Service to be used to store files by
  jobs in case no destination Storage Service is explicitly stated'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.25
  NAME 'GLUE2ComputingSharePreemption'
  DESC 'True if the computing manager (i.e., LRMS) enables preemption
  of jobs; a preempted job is supposed to be automatically resumed'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.26
  NAME 'GLUE2ComputingShareServingState'
  DESC 'A state specifying if the share is open to place new requests
  and if it is open to offer the already present requests for execution'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.27
  NAME 'GLUE2ComputingShareTotalJobs'
  DESC 'Number of total jobs in any state'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.28
  NAME 'GLUE2ComputingShareRunningJobs'
  DESC 'Number of running jobs submitted via any type of interface
  (local
  and Grid)'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.29
  NAME 'GLUE2ComputingShareLocalRunningJobs'
  DESC 'Number of running jobs submitted via a local interface'
attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.30
NAME 'GLUE2ComputingShareWaitingJobs'
DESC 'Number of jobs waiting in the underlying computing managers
(i.e., LRMS) submitted via any type of interface (local and
Grid)'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.31
NAME 'GLUE2ComputingShareLocalWaitingJobs'
DESC 'Number of jobs waiting in the underlying computing managers
(i.e., LRMS) submitted via a local interface'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.32
NAME 'GLUE2ComputingShareSuspendedJobs'
DESC 'Number of jobs which started their execution, but are
suspended (e.g., for preemption)'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.33
NAME 'GLUE2ComputingShareLocalSuspendedJobs'
DESC 'Number of local jobs which started their execution, but are
suspended (e.g., for preemption)'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.34
NAME 'GLUE2ComputingShareStagingJobs'
DESC 'Number of jobs that are staging files in/out'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.35
NAME 'GLUE2ComputingShareLocalStagingJobs'
DESC 'Number of local jobs that are staging files in/out'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE )
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.36
  NAME 'GLUE2ComputingSharePreLRMSWaitingJobs'
  DESC 'Number of jobs that are in the Grid layer waiting to be passed
to
  the underlying computing manager (i.e., LRMS)'
  EQUALITY    integerMatch
  ORDERING    integerOrderingMatch
  SYNTAX      1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.37
  NAME 'GLUE2ComputingShareEstimatedAverageWaitingTime'
  DESC 'Estimated time to last for a new job from the acceptance to
the
  start of its execution'
  EQUALITY    integerMatch
  ORDERING    integerOrderingMatch
  SYNTAX      1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.38
  NAME 'GLUE2ComputingShareEstimatedWorstWaitingTime'
  DESC 'Estimated worst waiting time assuming that all jobs run for
the
  maximum wall time'
  EQUALITY    integerMatch
  ORDERING    integerOrderingMatch
  SYNTAX      1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.39
  NAME 'GLUE2ComputingShareFreeSlots'
  DESC 'Number of free slots'
  EQUALITY    integerMatch
  ORDERING    integerOrderingMatch
  SYNTAX      1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.40
  NAME 'GLUE2ComputingShareFreeSlotsWithDuration'
  DESC 'Number of free slots with their time limits. Syntax: ns[:t]
  [ns:t]* where the pair ns:t means that there are ns free slots for the
duration of t (expressed in seconds); the time limit information is
optional'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstring
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.41
  NAME 'GLUE2ComputingShareUsedSlots'
  DESC 'Number of slots used by running jobs'
  EQUALITY    integerMatch
  ORDERING    integerOrderingMatch
  SYNTAX      1.3.6.1.4.1.1466.115.121.1.27
}
SINGLE-VALUE

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.42
  NAME 'GLUE2ComputingShareRequestedSlots'
  DESC 'Number of slots which are needed to execute all waiting and staging jobs'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.43
  NAME 'GLUE2ComputingShareReservationPolicy'
  DESC 'Type of reservation policy'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.44
  NAME 'GLUE2ComputingShareTag'
  DESC 'UserDomain-defined tag (the values SHOULD use namespace to avoid collision)'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.45
  NAME 'GLUE2ComputingShareComputingEndpointForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.46
  NAME 'GLUE2ComputingShareExecutionEnvironmentForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.19.47
  NAME 'GLUE2ComputingShareComputingServiceForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.19
  NAME 'GLUE2ComputingShare'
  DESC 'A utilization target for a set of execution environments defined by a set of configuration parameters and characterized by status information'
SUP GLUE2Share
STRUCTURAL

# File:    schema/6-04-ComputingManager.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 6.4 ComputingManager
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
# ComputingManager inherits from Manager
attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.1 NAME 'GLUE2ComputingManagerReservation' DESC 'True if the computing manager (i.e, LRMS) supports advance reservation' EQUALITY caseExactMatch SUBSTR caseExactSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.2
NAME 'GLUE2ComputingManagerBulkSubmission'
DESC 'True if the computing manager (i.e., LRMS) supports the bulk submission'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.3
NAME 'GLUE2ComputingManagerTotalPhysicalCPUs'
DESC 'Number of managed physical CPUs accessible via any of the available endpoints (there is one physical CPU per socket)'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.4
NAME 'GLUE2ComputingManagerTotalLogicalCPUs'
DESC 'Number of managed logical CPUs accessible via any of the available endpoints (a logical CPU corresponds to a CPU visible to the operating system)'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.5
NAME 'GLUE2ComputingManagerTotalSlots'
DESC 'Number of managed slots'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.6
NAME 'GLUE2ComputingManagerSlotsUsedByLocalJobs'
DESC 'Number of slots used by jobs submitted via local interface'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.7
NAME 'GLUE2ComputingManagerSlotsUsedByGridJobs'
DESC 'Number of slots used by jobs submitted via a Grid interface'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.8
NAME 'GLUE2ComputingManagerHomogeneous'
DESC 'True if the computing manager has only one type of execution environment'

EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.9
  NAME 'GLUE2ComputingManagerNetworkInfo'
  DESC 'Type of internal network available among all the managed
  execution environment instances'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.10
  NAME 'GLUE2ComputingManagerLogicalCPUDistribution'
  DESC 'Syntax: X1:Y1, …, Xn:Yn where Xi is the number of logical CPUs
  and Yi is the number of boxes for the execution environment i'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.11
  NAME 'GLUE2ComputingManagerWorkingAreaShared'
  DESC 'A working area is an allocated storage extent that holds the
  home directories of the Grid jobs; this property is true if the working
  area
  is shared across different execution environment instances (i.e.,
  cluster nodes)'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.12
  NAME 'GLUE2ComputingManagerWorkingAreaGuaranteed'
  DESC 'True if the job is guaranteed the full extent of the
  WorkingAreaTotal attribute'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.13
  NAME 'GLUE2ComputingManagerWorkingAreaTotal'
  DESC 'Total size of working area available to all the Grid jobs
  either
  as a shared area across all the execution environments
  (WorkingAreaShared is true) or local to a certain execution
  environment (WorkingAreaShared is false); even if individual quota
  per job is
  enforced, this is not advertised'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.14
    NAME 'GLUE2ComputingManagerWorkingAreaFree'
    DESC 'Free size of working area available to all the Grid jobs
    either as a shared area across all the execution environments
    (WorkingAreaShared is true) or local to a certain execution
    environment (WorkingAreaShared is false); (even if individual quota
    per job is enforced, this is not advertised)'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.15
    NAME 'GLUE2ComputingManagerWorkingAreaLifeTime'
    DESC 'Lifetime of the Grid job files present in the working area; the
    lifetime is related to the end time of the job'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.16
    NAME 'GLUE2ComputingManagerWorkingAreaMultiSlotTotal'
    DESC 'Total size of working area available to all the multi-slot
    Grid jobs either as a shared area across all the execution environments
    (WorkingAreaShared is true) or local to a certain execution
    environment (WorkingAreaShared is false); even if individual quota per job is
    enforced, this is not advertised' 
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.17
    NAME 'GLUE2ComputingManagerWorkingAreaMultiSlotFree'
    DESC 'Free size of working area available to all the multi-slot Grid
    jobs either as a shared area across all the execution environments
    (WorkingAreaShared is true) or local to a certain execution
    environment (WorkingAreaShared is false); (even if individual quota per job is
    enforced, this is not advertised)'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.18
    NAME 'GLUE2ComputingManagerWorkingAreaMultiSlotLifeTime'
    DESC 'Lifetime of the multi-slot Grid job files present in the
    working area; the lifetime is related to the end time of the job'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
SYNTAX      1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.19
  NAME 'GLUE2ComputingManagerCacheTotal'
  DESC 'Total size of a temporary storage area where frequently
  accessed
data can be stored for rapid access by consequent Grid jobs'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.20
  NAME 'GLUE2ComputingManagerCacheFree'
  DESC 'Free size of a temporary storage area where frequently
  accessed
data can be stored for rapid access by consequent Grid jobs; in the
  computation of the free size, files which are not claimed by any job
  can be considered as deleted'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.21
  NAME 'GLUE2ComputingManagerTmpDir'
  DESC 'The absolute path of a temporary directory local to an
  execution
  environment instance (i.e., worker node). This directory must be
  available to programs using the normal file access primitives
  (open/read/write/close)'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.22
  NAME 'GLUE2ComputingManagerScratchDir'
  DESC 'The absolute path for a shared directory available for
  application data. Typically a POSIX accessible transient disk space
  shared between the execution environment instances. It may be used by
  MPI applications or to store intermediate files that need further
  processing by local jobs or as staging area, specially if the
  execution
  environment instances have no Internet connectivity'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.20.23
  NAME 'GLUE2ComputingManagerApplicationDir'
  DESC 'The path of the directory available for application
  installation.
  Typically a PO-SIX accessible disk space with transient to permanent
  allocation to the users'
  EQUALITY caseExactMatch
attributetype ( 1.3.6.1.4.1.6757.100.1.1.21.1
    NAME 'GLUE2BenchmarkID'
    DESC 'Globally unique ID'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.20
    NAME 'GLUE2ComputingManager'
    DESC 'A software component locally managing one or more execution environments. It can describe also aggregated information about the managed resources. The computing manager is also known as Local Resource Management System (LRMS).'
    SUP GLUE2Manager
    STRUCTURAL
    MAY ( GLUE2ComputingManagerReservation $ GLUE2ComputingManagerBulkSubmission $ GLUE2ComputingManagerTotalPhysicalCPUs $ GLUE2ComputingManagerTotalLogicalCPUs $ GLUE2ComputingManagerTotalSlots $ GLUE2ComputingManagerSlotsUsedByLocalJobs $ GLUE2ComputingManagerSlotsUsedByGridJobs $ GLUE2ComputingManagerHomogeneous $ GLUE2ComputingManagerNetworkInfo $ GLUE2ComputingManagerLogicalCPUDistribution $ GLUE2ComputingManagerWorkingAreaShared $ GLUE2ComputingManagerWorkingAreaGuaranteed $ GLUE2ComputingManagerWorkingAreaTotal $ GLUE2ComputingManagerWorkingAreaFree $ GLUE2ComputingManagerWorkingAreaLifeTime $ GLUE2ComputingManagerWorkingAreaMultiSlotTotal $ GLUE2ComputingManagerWorkingAreaMultiSlotFree $ GLUE2ComputingManagerWorkingAreaMultiSlotLifeTime $ GLUE2ComputingManagerCacheTotal $ GLUE2ComputingManagerCacheFree $ GLUE2ComputingManagerTmpDir $ GLUE2ComputingManagerScratchDir $ GLUE2ComputingManagerApplicationDir $ GLUE2ComputingManagerComputingServiceForeignKey )
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.21.2
   NAME 'GLUE2BenchmarkType'
   DESC 'Type of benchmark'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.21.3
   NAME 'GLUE2BenchmarkValue'
   DESC 'Value'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.21.4
   NAME 'GLUE2BenchmarkExecutionEnvironmentForeignKey'
   DESC 'ForeignKey'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.21.5
   NAME 'GLUE2BenchmarkComputingManagerForeignKey'
   DESC 'ForeignKey'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE )

objectclass ( 1.3.6.1.4.1.6757.100.1.1.21
   NAME 'GLUE2Benchmark'
   SUP GLUE2Entity
   DESC 'Benchmark information about an entity providing computing
capacity'
   STRUCTURAL
   MUST ( GLUE2BenchmarkID $ GLUE2BenchmarkType $ GLUE2BenchmarkValue )
   MAY ( GLUE2BenchmarkExecutionEnvironmentForeignKey $ GLUE2BenchmarkComputingManagerForeignKey )
)

# File:    schema/6-06-ExecutionEnvironment.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 6.6 ExecutionEnvironment
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
# ExecutionEnvironment inherits from Resource
attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.1
   NAME 'GLUE2ExecutionEnvironmentPlatform'
   DESC 'The architecture platform of this execution environment'
   EQUALITY caseExactMatch
attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.2
   NAME 'GLUE2ExecutionEnvironmentVirtualMachine'
   DESC 'True if the execution environment is based on a virtual
   machine
   (in this case, the values of the other attributes are related to the
   virtualized environment and not to the hosting environment)'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE
 )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.3
   NAME 'GLUE2ExecutionEnvironmentTotalInstances'
   DESC 'Number of execution environment instances'
   EQUALITY integerMatch
   ORDERING integerOrderingMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
   SINGLE-VALUE
 )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.4
   NAME 'GLUE2ExecutionEnvironmentUsedInstances'
   DESC 'Number of used execution environment instances; an instance is
   used when, according to the policies of the Manager (i.e., LRMS), it
   cannot accept new jobs because it already runs the maximum number of
   allowed jobs'
   EQUALITY integerMatch
   ORDERING integerOrderingMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
   SINGLE-VALUE
 )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.5
   NAME 'GLUE2ExecutionEnvironmentUnavailableInstances'
   DESC 'Number of unavailable execution environment instances because
   of
   failures or maintenance'
   EQUALITY integerMatch
   ORDERING integerOrderingMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
   SINGLE-VALUE
 )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.6
   NAME 'GLUE2ExecutionEnvironmentPhysicalCPUs'
   DESC 'Number of physical CPUs in an execution environment instance'
   EQUALITY integerMatch
   ORDERING integerOrderingMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
   SINGLE-VALUE
 )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.7
   NAME 'GLUE2ExecutionEnvironmentLogicalCPUs'
   DESC 'Number of logical CPUs in an execution environment instance'
   EQUALITY integerMatch
   ORDERING integerOrderingMatch
 )
attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.8
    NAME 'GLUE2ExecutionEnvironmentCPUMultiplicity'
    DESC 'Information about the multiplicity of both physical CPUs and
cores available in an execution environment instance'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.9
    NAME 'GLUE2ExecutionEnvironmentCPUVendor'
    DESC 'Name of the physical CPU vendor'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.10
    NAME 'GLUE2ExecutionEnvironmentCPUModel'
    DESC 'Physical CPU model as defined by the vendor'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.11
    NAME 'GLUE2ExecutionEnvironmentCPUVersion'
    DESC 'Physical CPU version as defined by the vendor'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.12
    NAME 'GLUE2ExecutionEnvironmentCPUClockSpeed'
    DESC 'Nominal clock speed of the physical CPU'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.13
    NAME 'GLUE2ExecutionEnvironmentCPUTimeScalingFactor'
    DESC 'Factor used by the Manager (i.e., LRMS) to scale the CPU time
(CPU Time divided by CPUTimeScalingFactor); for the reference
execution
environment, this attribute is equal to 1'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.14

<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>GLUE2ExecutionEnvironmentWallTimeScalingFactor</td>
</tr>
<tr>
<td>DESC</td>
<td>Factor used by the Manager (i.e., LRMS) to scale the Wall time (Wall Time divided by WallTimeScalingFactor)</td>
</tr>
<tr>
<td>EQUALITY</td>
<td>caseExactMatch</td>
</tr>
<tr>
<td>ORDERING</td>
<td>integerOrderingMatch</td>
</tr>
<tr>
<td>SYNTAX</td>
<td>1.3.6.1.4.1.1466.115.121.1.15</td>
</tr>
<tr>
<td>SINGLE-VALUE</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>GLUE2ExecutionEnvironmentMainMemorySize</td>
</tr>
<tr>
<td>DESC</td>
<td>Amount of RAM (if many jobs run in the same execution environment, they compete for the total RAM)</td>
</tr>
<tr>
<td>EQUALITY</td>
<td>integerMatch</td>
</tr>
<tr>
<td>ORDERING</td>
<td>integerOrderingMatch</td>
</tr>
<tr>
<td>SYNTAX</td>
<td>1.3.6.1.4.1.1466.115.121.1.27</td>
</tr>
<tr>
<td>SINGLE-VALUE</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>GLUE2ExecutionEnvironmentVirtualMemorySize</td>
</tr>
<tr>
<td>DESC</td>
<td>The amount of Virtual Memory (RAM+Swap)</td>
</tr>
<tr>
<td>EQUALITY</td>
<td>integerMatch</td>
</tr>
<tr>
<td>ORDERING</td>
<td>integerOrderingMatch</td>
</tr>
<tr>
<td>SYNTAX</td>
<td>1.3.6.1.4.1.1466.115.121.1.27</td>
</tr>
<tr>
<td>SINGLE-VALUE</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>GLUE2ExecutionEnvironmentOSFamily</td>
</tr>
<tr>
<td>DESC</td>
<td>Family of the operating system</td>
</tr>
<tr>
<td>EQUALITY</td>
<td>caseExactMatch</td>
</tr>
<tr>
<td>SUBSTR</td>
<td>caseExactSubstringsMatch</td>
</tr>
<tr>
<td>SYNTAX</td>
<td>1.3.6.1.4.1.1466.115.121.1.15</td>
</tr>
<tr>
<td>SINGLE-VALUE</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>GLUE2ExecutionEnvironmentOSName</td>
</tr>
<tr>
<td>DESC</td>
<td>Name of the operating system</td>
</tr>
<tr>
<td>EQUALITY</td>
<td>caseExactMatch</td>
</tr>
<tr>
<td>SUBSTR</td>
<td>caseExactSubstringsMatch</td>
</tr>
<tr>
<td>SYNTAX</td>
<td>1.3.6.1.4.1.1466.115.121.1.15</td>
</tr>
<tr>
<td>SINGLE-VALUE</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>GLUE2ExecutionEnvironmentOSVersion</td>
</tr>
<tr>
<td>DESC</td>
<td>Version of the operating system</td>
</tr>
<tr>
<td>EQUALITY</td>
<td>caseExactMatch</td>
</tr>
<tr>
<td>SUBSTR</td>
<td>caseExactSubstringsMatch</td>
</tr>
<tr>
<td>SYNTAX</td>
<td>1.3.6.1.4.1.1466.115.121.1.15</td>
</tr>
<tr>
<td>SINGLE-VALUE</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>GLUE2ExecutionEnvironmentConnectivityIn</td>
</tr>
<tr>
<td>DESC</td>
<td>Permission for direct inbound connectivity, even if limited</td>
</tr>
<tr>
<td>EQUALITY</td>
<td>caseExactMatch</td>
</tr>
<tr>
<td>SUBSTR</td>
<td>caseExactSubstringsMatch</td>
</tr>
<tr>
<td>SYNTAX</td>
<td>1.3.6.1.4.1.1466.115.121.1.15</td>
</tr>
<tr>
<td>SINGLE-VALUE</td>
<td></td>
</tr>
</tbody>
</table>
attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.21
  NAME 'GLUE2ExecutionEnvironmentConnectivityOut'
  DESC 'Permission for direct outbound connectivity, even if limited'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.22
  NAME 'GLUE2ExecutionEnvironmentNetworkInfo'
  DESC 'Type of internal network available among the execution environment instances'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.22.23
  NAME 'GLUE2ExecutionEnvironmentComputingManagerForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )

objectclass ( 1.3.6.1.4.1.6757.100.1.1.22
  NAME 'GLUE2ExecutionEnvironment'
  DESC 'A description of hardware, operating system and network characteristics that defines the environment available to and requestable by a Grid job when submitted to a Service via a Endpoint; the description also includes information about the total/available/used instances of the execution environment'
  SUP GLUE2Resource
  STRUCTURAL
  MUST ( GLUE2ExecutionEnvironmentPlatform $ GLUE2ExecutionEnvironmentMainMemorySize $ GLUE2ExecutionEnvironmentOSFamily $ GLUE2ExecutionEnvironmentConnectivityIn $ GLUE2ExecutionEnvironmentConnectivityOut )
  MAY ( GLUE2ExecutionEnvironmentVirtualMachine $ GLUE2ExecutionEnvironmentTotalInstances $ GLUE2ExecutionEnvironmentUsedInstances $ GLUE2ExecutionEnvironmentUnavailableInstances $ GLUE2ExecutionEnvironmentPhysicalCPUs $ GLUE2ExecutionEnvironmentLogicalCPUs $ GLUE2ExecutionEnvironmentCPUMultiplicity $ GLUE2ExecutionEnvironmentCPUTimeScalingFactor $ GLUE2ExecutionEnvironmentWallTimeScalingFactor $ GLUE2ExecutionEnvironmentVirtualMemorySize $ GLUE2ExecutionEnvironmentOSName $ GLUE2ExecutionEnvironmentOSVersion $ GLUE2ExecutionEnvironmentNetworkInfo $ GLUE2ExecutionEnvironmentComputingManagerForeignKey ) )
# File: schema/6-07-ApplicationEnvironment.schema
# URL: http://forge.gridforum.org/sf/projects/glue-wg
# Doc: GLUE Specification 2.0 (March 3, 2009)
# Section: 6.7 ApplicationEnvironment
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# ApplicationEnvironment inherits from Entity

attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.1
  NAME 'GLUE2ApplicationEnvironmentID'
  DESC 'Globally unique ID'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.2
  NAME 'GLUE2ApplicationEnvironmentAppName'
  DESC 'Name of the application environment'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.3
  NAME 'GLUE2ApplicationEnvironmentAppVersion'
  DESC 'Version of the application environment'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.4
  NAME 'GLUE2ApplicationEnvironmentRepository'
  DESC 'URL of a service which offers a repository and/or a name service for this application environment'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.5
  NAME 'GLUE2ApplicationEnvironmentState'
  DESC 'State about the installation'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.6
  NAME 'GLUE2ApplicationEnvironmentRemovalDate'
  DESC 'Date and time after which the application can be removed'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch

# GLUE Working Group

glue-wg@ogf.org
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.7
  NAME 'GLUE2ApplicationEnvironmentLicense'
  DESC 'The type of license'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.8
  NAME 'GLUE2ApplicationEnvironmentDescription'
  DESC 'The description of this application environment'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.9
  NAME 'GLUE2ApplicationEnvironmentBestBenchmark'
  DESC 'Type of benchmark which best identify the sensitivity of this
  application to the performance aspect'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.10
  NAME 'GLUE2ApplicationEnvironmentParallelSupport'
  DESC 'The type of supported parallel execution framework'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.11
  NAME 'GLUE2ApplicationEnvironmentMaxSlots'
  DESC 'Maximum number of slots that can be used to run jobs using the
  application environment at the same time'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.12
  NAME 'GLUE2ApplicationEnvironmentMaxJobs'
  DESC 'Maximum number of jobs that can use the application
  environment at the same time'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.13
  NAME 'GLUE2ApplicationEnvironmentMaxUserSeats'
  DESC 'Maximum number of users that can use the application
  environment at the same time'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)
DESC 'Maximum number of user seats that can use the application environment at the same time'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.14
  NAME 'GLUE2ApplicationEnvironmentFreeSlots'
  DESC 'Available number slots that can be used to run jobs using the application environment at the same time'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.15
  NAME 'GLUE2ApplicationEnvironmentFreeJobs'
  DESC 'Number of new jobs that could start their execution and use the application environment at the same time'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.16
  NAME 'GLUE2ApplicationEnvironmentFreeUserSeats'
  DESC 'Free seats for additional users that can use the application environment at the same time'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.17
  NAME 'GLUE2ApplicationEnvironmentExecutionEnvironmentForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.23.18
  NAME 'GLUE2ApplicationEnvironmentComputingManagerForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.23
  NAME 'GLUE2ApplicationEnvironment'
  SUP GLUE2Entity
  DESC 'Description of the application software or environment characteristic available within one or more execution environments'
  STRUCTURAL
  MUST ( GLUE2ApplicationEnvironmentID $
attributetype  ( 1.3.6.1.4.1.6757.100.1.1.24.1
    NAME 'GLUE2ApplicationHandleID'
    DESC 'Type of handle for an application environment'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

attributetype  ( 1.3.6.1.4.1.6757.100.1.1.24.2
    NAME 'GLUE2ApplicationHandleType'
    DESC 'Type of handle for an application environment'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

attributetype  ( 1.3.6.1.4.1.6757.100.1.1.24.3
    NAME 'GLUE2ApplicationHandleValue'
    DESC 'Actionable value to trigger the handle method'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )

attributetype  ( 1.3.6.1.4.1.6757.100.1.1.24.4
    NAME 'GLUE2ApplicationHandleApplicationEnvironmentForeignKey'
    DESC 'Foreign Key'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE )
objectclass ( 1.3.6.1.4.1.6757.100.1.1.24
    NAME 'GLUE2ApplicationHandle'
    SUP GLUE2Entity
    DESC 'Technique for bootstrapping and/or accessing the application'
    STRUCTURAL
    MUST ( GLUE2ApplicationHandleID $ GLUE2ApplicationHandleType $ GLUE2ApplicationHandleValue $ GLUE2ApplicationHandleApplicationEnvironmentForeignKey )
)

# File:    schema/6-09-ComputingActivity.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 6.9 ComputingActivity
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# ComputingActivity inherits from Activity
attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.1
    NAME 'GLUE2ComputingActivityType'
    DESC 'Type of computing activity'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.2
    NAME 'GLUE2ComputingActivityIDFromEndpoint'
    DESC 'The job ID as assigned by the computing endpoint'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.3
    NAME 'GLUE2ComputingActivityLocalIDFromManager'
    DESC 'The local ID of the job as assigned by the computing manager (i.e., LRMS)'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.4
    NAME 'GLUE2ComputingActivityJobDescription'
    DESC 'Job description language used to specify the job request'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.5
    NAME 'GLUE2ComputingActivityState'
    DESC 'The state of the job according to the Grid state model for jobs'
)
<table>
<thead>
<tr>
<th>Attribute Type Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name: GLUE2ComputingActivityRestartState</td>
</tr>
<tr>
<td>Name: GLUE2ComputingActivityExitCode</td>
</tr>
<tr>
<td>Name: GLUE2ComputingActivityComputingManagerExitCode</td>
</tr>
<tr>
<td>Name: GLUE2ComputingActivityError</td>
</tr>
<tr>
<td>Name: GLUE2ComputingActivityWaitingPosition</td>
</tr>
<tr>
<td>Name: GLUE2ComputingActivityUserDomain</td>
</tr>
<tr>
<td>Attribute Type</td>
</tr>
<tr>
<td>---------------</td>
</tr>
<tr>
<td><code>attributetype(1.3.6.1.4.1.6757.100.1.1.25.12)</code></td>
</tr>
<tr>
<td><code>attributetype(1.3.6.1.4.1.6757.100.1.1.25.13)</code></td>
</tr>
<tr>
<td><code>attributetype(1.3.6.1.4.1.6757.100.1.1.25.14)</code></td>
</tr>
<tr>
<td><code>attributetype(1.3.6.1.4.1.6757.100.1.1.25.15)</code></td>
</tr>
<tr>
<td><code>attributetype(1.3.6.1.4.1.6757.100.1.1.25.16)</code></td>
</tr>
<tr>
<td><code>attributetype(1.3.6.1.4.1.6757.100.1.1.25.17)</code></td>
</tr>
<tr>
<td><code>attributetype(1.3.6.1.4.1.6757.100.1.1.25.18)</code></td>
</tr>
</tbody>
</table>
NAME 'GLUE2ComputingActivityStdIn' DESC 'The name of the file which is used as the standard input of the job'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.19
    NAME 'GLUE2ComputingActivityStdOut'
    DESC 'The name of the file which contains the standard output of the job'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.20
    NAME 'GLUE2ComputingActivityStdErr'
    DESC 'The name of the file which contains the standard error of the job'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.21
    NAME 'GLUE2ComputingActivityLogDir'
    DESC 'The name of the directory which contains the logs related to the job and generated by the Grid layer (usually the directory is private to the job)'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.22
    NAME 'GLUE2ComputingActivityExecutionNode'
    DESC 'Hostname associated to the execution environment instance (i.e., worker node) running the job; multi-node jobs are described by several instances of this attribute'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.23
    NAME 'GLUE2ComputingActivityQueue'
    DESC 'The name of the Computing Manager (i.e, LRMS) queue to which this job was queued'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)
<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>GLUE2ComputingActivityUsedTotalWallTime</code></td>
<td>The totally consumed wall clock time by the job (in case of multi-slot jobs, this value refers to the sum of the wall clock time consumed in each slot)</td>
</tr>
<tr>
<td><code>GLUE2ComputingActivityUsedTotalCPUTime</code></td>
<td>The totally consumed CPU time by the job (in case of multi-slot jobs, this value refers to the sum of the consumed CPU time in each slot)</td>
</tr>
<tr>
<td><code>GLUE2ComputingActivityUsedMainMemory</code></td>
<td>The RAM used by the job</td>
</tr>
<tr>
<td><code>GLUE2ComputingActivitySubmissionTime</code></td>
<td>Time when the job was submitted to a computing endpoint</td>
</tr>
<tr>
<td><code>GLUE2ComputingActivityComputingManagerSubmissionTime</code></td>
<td>Time when the job was submitted to the Computing Manager (i.e., LRMS) by the Grid layer</td>
</tr>
<tr>
<td><code>GLUE2ComputingActivityStartTime</code></td>
<td>Time when the job entered in the Computing Manager (i.e., LRMS) running state</td>
</tr>
</tbody>
</table>
attributetype (1.3.6.1.4.1.6757.100.1.1.25.30
  NAME 'GLUE2ComputingActivityComputingManagerEndTime'
  DESC 'Time when the job entered its final Computing Manager (i.e.,
  LRMS) state'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype (1.3.6.1.4.1.6757.100.1.1.25.31
  NAME 'GLUE2ComputingActivityEndTime'
  DESC 'Time when the job entered its final Grid state'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype (1.3.6.1.4.1.6757.100.1.1.25.32
  NAME 'GLUE2ComputingActivityWorkingAreaEraseTime'
  DESC 'A working area is an allocated storage extent that holds the
  home directories of the Grid jobs; the time when the dedicated working area
  of this job will be removed'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype (1.3.6.1.4.1.6757.100.1.1.25.33
  NAME 'GLUE2ComputingActivityProxyExpirationTime'
  DESC 'The expiration time of the proxy related to the job'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype (1.3.6.1.4.1.6757.100.1.1.25.34
  NAME 'GLUE2ComputingActivitySubmissionHost'
  DESC 'The name of the host from which the job was submitted (e.g.,
  IP address, port and host name)'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype (1.3.6.1.4.1.6757.100.1.1.25.35
  NAME 'GLUE2ComputingActivitySubmissionClientName'
  DESC 'The name of the software client which was used to submit the
  job'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.36
    NAME 'GLUE2ComputingActivityOtherMessages'
    DESC 'Optional job messages provided by either the Grid Layer or the
           Computing Manager'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.37
    NAME 'GLUE2ComputingActivityComputingEndpointForeignKey'
    DESC 'ForeignKey'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.38
    NAME 'GLUE2ComputingActivityComputingShareForeignKey'
    DESC 'ForeignKey'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.25.39
    NAME 'GLUE2ComputingActivityExecutionEnvironmentForeignKey'
    DESC 'ForeignKey'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.25
    NAME 'GLUE2ComputingActivity'
    DESC 'An activity managed by an OGSA execution capability service'
    SUP 'GLUE2Activity'
    STRUCTURAL
    MUST ( GLUE2ComputingActivityState $ GLUE2ComputingActivityOwner )

# File:    schema/6-10-ToStorageService.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 6.10 ToStorageService
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# ToStorageService inherits from Entity
attributetype ( 1.3.6.1.4.1.6757.100.1.1.26.1
  NAME 'GLUE2ToStorageServiceID'
  DESC 'Globally unique ID'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.1
  SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.26.2
  NAME 'GLUE2ToStorageServiceLocalPath'
  DESC 'The local path of the computing service enabling to access a remote path in the associated storage service (this is typically an NFS mount point)'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.1
  SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.26.3
  NAME 'GLUE2ToStorageServiceRemotePath'
  DESC 'The remote path in the storage service which is associated the local path in the computing service (this is typically an NFS exported directory)'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.1
  SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.26.4
  NAME 'GLUE2ToStorageServiceComputingServiceForeignKey'
  DESC 'ForeignKey' \n
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.26.5
NAME 'GLUE2ToStorageServiceStorageServiceForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)
objectclass ( 1.3.6.1.4.1.6757.100.1.1.26
NAME 'GLUE2ToStorageService'
SUP GLUE2Entity
DESC 'Description of a POSIX access via a file system technology enabling the computing service to access the associated storage service'
STRUCTURAL
MUST ( GLUE2ToStorageServiceID $ GLUE2ToStorageServiceLocalPath $ GLUE2ToStorageServiceRemotePath $ GLUE2ToStorageServiceComputingServiceForeignKey $ GLUE2ToStorageServiceStorageServiceForeignKey )
)

# File:    schema/7-01-StorageService.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 7.1 StorageService
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# StorageService inherits from Service
# No attributes

objectclass ( 1.3.6.1.4.1.6757.100.1.1.27
NAME 'GLUE2StorageService'
DESC 'Description of the size and state of an homogeneous storage extent'
SUP GLUE2Service
STRUCTURAL
)

# File:    schema/7-02-StorageServiceCapacity.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 7.2 StorageServiceCapacity
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# StorageServiceCapacity inherits from Entity

attributetype ( 1.3.6.1.4.1.6757.100.1.1.28.1
NAME 'GLUE2StorageServiceCapacityID'
DESC 'Globally unique ID'
EQUALITY caseExactMatch
attributetype ( 1.3.6.1.4.1.6757.100.1.1.28.2
    NAME 'GLUE2StorageServiceCapacityType'
    DESC 'Type of storage capacity'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.28.3
    NAME 'GLUE2StorageServiceCapacityTotalSize'
    DESC 'Size of dedicated storage extent'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.28.4
    NAME 'GLUE2StorageServiceCapacityFreeSize'
    DESC 'Size of free storage extent'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.28.5
    NAME 'GLUE2StorageServiceCapacityUsedSize'
    DESC 'Size of used storage extent'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.28.6
    NAME 'GLUE2StorageServiceCapacityReservedSize'
    DESC 'Size of reserved storage extent'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.28.7
    NAME 'GLUE2StorageServiceCapacityStorageServiceForeignKey'
    DESC 'ForeignKey'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.28
    NAME 'GLUE2StorageServiceCapacity'
    SUP GLUE2Entity
    DESC 'Description of the size and state of an homogeneous storage
ATTRIBUTETYPE (1.3.6.1.4.1.6757.100.1.1.29.1)
NAME 'GLUE2StorageAccessProtocolID'
DESC 'Globally unique ID'
EQUALITY caseExactMatch
 SUBSTR caseExactSubstringsMatch
 SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
 SINGLE-VALUE
 }

ATTRIBUTETYPE (1.3.6.1.4.1.6757.100.1.1.29.2)
NAME 'GLUE2StorageAccessProtocolType'
DESC 'The name of the protocol'
EQUALITY caseExactMatch
 SUBSTR caseExactSubstringsMatch
 SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
 SINGLE-VALUE
 }

ATTRIBUTETYPE (1.3.6.1.4.1.6757.100.1.1.29.3)
NAME 'GLUE2StorageAccessProtocolVersion'
DESC 'The version of the protocol'
EQUALITY caseExactMatch
 SUBSTR caseExactSubstringsMatch
 SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
 SINGLE-VALUE
 }

ATTRIBUTETYPE (1.3.6.1.4.1.6757.100.1.1.29.4)
NAME 'GLUE2StorageAccessProtocolMaxStreams'
DESC 'The number of parallel streams this protocol supports'
EQUALITY integerMatch
 ORDERING integerOrderingMatch
 SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
 SINGLE-VALUE
 }

ATTRIBUTETYPE (1.3.6.1.4.1.6757.100.1.1.29.5)
NAME 'GLUE2StorageAccessProtocolStorageServiceForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
 SUBSTR caseExactSubstringsMatch
objectclass ( 1.3.6.1.4.1.1466.115.121.1.15
       NAME 'GLUE2StorageAccessProtocol'
       SUP GLUE2Entity
       DESC 'A type of protocol available to access the available storage capacities'
       STRUCTURAL
       MUST ( GLUE2StorageAccessProtocolID $
               GLUE2StorageAccessProtocolType $
               GLUE2StorageAccessProtocolVersion $
               GLUE2StorageAccessProtocolStorageServiceForeignKey )
       MAY ( GLUE2StorageAccessProtocolMaxStreams )
     )

# File:    schema/7-04-StorageEndpoint.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 7.4 StorageEndpoint
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# StorageEndpoint inherits from Endpoint
# No attributes

attributetype ( 1.3.6.1.4.1.6757.100.1.1.30.1
       NAME 'GLUE2StorageEndpointStorageServiceForeignKey'
       DESC 'ForeignKey'
       EQUALITY caseExactMatch
       SUBSTR caseExactSubstringsMatch
       SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
       SINGLE-VALUE
     )

objectclass ( 1.3.6.1.4.1.6757.100.1.1.30
       NAME 'GLUE2StorageEndpoint'
       DESC 'A network location having a well-defined interface and
          exposing the service functionalities'
       SUP GLUE2Endpoint
       STRUCTURAL
       MAY GLUE2StorageEndpointStorageServiceForeignKey
     )

# File:    schema/7-05-StorageShare.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 7.5 StorageShare
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# StorageShare inherits from Share

attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.1
       NAME 'GLUE2StorageShareServingState'
       DESC 'A state specifying if the share is open to place new requests
          and if it is open to offer the already present requests for
     )
attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.2
    NAME 'GLUE2StorageSharePath'
    DESC 'A namespace where files are logically assigned to when they
    are
    stored into this
    share'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.3
    NAME 'GLUE2StorageShareAccessMode'
    DESC 'An identifier for the type of access and usage allowed for
    this
    share'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.4
    NAME 'GLUE2StorageShareSharingID'
    DESC 'Local ID common to the storage shares which use the same
    storage
    share capacities (dedicated is a reserved term and means that the
    storage share capacities are not shared with other storage share
    capacities part of different storage shares)'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.5
    NAME 'GLUE2StorageShareAccessLatency'
    DESC 'The maximum latency category for a file stored in this share
    to
    be made available for reading'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.6
    NAME 'GLUE2StorageShareRetentionPolicy'
    DESC 'The quality of retention, which indicates the probability of
    the
    storage system losing a file'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.7
   NAME 'GLUE2StorageShareExpirationMode'
   DESC 'Support for files with infinite and/or finite lifetimes, and what actions the storage service may take upon the expiration of a file'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.8
   NAME 'GLUE2StorageShareDefaultLifeTime'
   DESC 'The default lifetime assigned to the file if no explicit lifetime is specified'
   EQUALITY integerMatch
   ORDERING integerOrderingMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
   SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.9
   NAME 'GLUE2StorageShareMaximumLifeTime'
   DESC 'The maximum lifetime that can be requested for a file'
   EQUALITY integerMatch
   ORDERING integerOrderingMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
   SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.10
   NAME 'GLUE2StorageShareTag'
   DESC 'A user defined tag for additional information'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.11
   NAME 'GLUE2StorageShareStorageEndpointForeignKey'
   DESC 'ForeignKey'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.12
   NAME 'GLUE2StorageShareDataStoreForeignKey'
   DESC ' ForeignKey'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.31.13
   NAME 'GLUE2StorageShareStorageServiceForeignKey'
   DESC 'ForeignKey'
   EQUALITY caseExactMatch
   SUBSTR caseExactSubstringsMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
   SINGLE-VALUE
)
objectclass ( 1.3.6.1.4.1.6757.100.1.1.31
  NAME 'GLUE2StorageShare'
  DESC 'A utilization target for a set of storage resources defined by
  a
  set of configuration parameters and characterized by status
  information,'
  SUP GLUE2Share
  STRUCTURAL
  MUST ( GLUE2StorageShareServingState $
  GLUE2StorageShareSharingID $
  GLUE2StorageShareAccessLatency )
  MAY ( GLUE2StorageSharePath $
  GLUE2StorageShareAccessMode $
  GLUE2StorageShareRetentionPolicy $
  GLUE2StorageShareExpirationMode $
  GLUE2StorageShareDefaultLifeTime $
  GLUE2StorageShareMaximumLifeTime $
  GLUE2StorageShareTag $
  GLUE2StorageShareStorageEndpointForeignKey $
  GLUE2StorageShareDataStoreForeignKey $
  GLUE2StorageShareStorageServiceForeignKey )
)

# File: schema/7-06-StorageShareCapacity.schema
# URL: http://forge.gridforum.org/sf/projects/glue-wg
# Doc: GLUE Specification 2.0 (March 3, 2009)
# Section: 7.6 StorageShareCapacity
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# StorageShareCapacity inherits from Entity
attributetype ( 1.3.6.1.4.1.6757.100.1.1.32.1
  NAME 'GLUE2StorageShareCapacityID'
  DESC 'Globally unique ID'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.32.2
  NAME 'GLUE2StorageShareCapacityType'
  DESC 'Type of storage capacity'
  EQUALITY caseExactMatch
  SUBSTR caseExactSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.32.3
  NAME 'GLUE2StorageShareCapacityTotalSize'
  DESC 'Size of dedicated storage extent'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.32.4
attributetype ( 1.3.6.1.4.1.6757.100.1.1.32.5
  NAME 'GLUE2StorageShareCapacityFreeSize'
  DESC 'Size of free storage extent'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.32.6
  NAME 'GLUE2StorageShareCapacityUsedSize'
  DESC 'Size of used storage extent'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.32.7
  NAME 'GLUE2StorageShareCapacityReservedSize'
  DESC 'Size of reserved storage extent'
  EQUALITY integerMatch
  ORDERING integerOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
  SINGLE-VALUE
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.32
  NAME 'GLUE2StorageShareCapacity'
  SUP GLUE2Entity
  DESC 'Description of the size and state of an homogeneous storage
   extent'
  STRUCTURAL
   MUST ( GLUE2StorageShareCapacityID $ GLUE2StorageShareCapacityType $ GLUE2StorageShareCapacityStorageShareForeignKey )
   MAY ( GLUE2StorageShareCapacityTotalSize $ GLUE2StorageShareCapacityFreeSize $ GLUE2StorageShareCapacityUsedSize $ GLUE2StorageShareCapacityReservedSize )
)

# File: schema/7-07-StorageManager.schema
# URL: http://forge.gridforum.org/sf/projects/glue-wg
# Doc: GLUE Specification 2.0 (March 3, 2009)
# Section: 7.7 StorageManager
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# StorageManager inherits from Manager
# No attributes

attributetype ( 1.3.6.1.4.1.6757.100.1.1.33.1
  NAME 'GLUE2StorageManagerStorageServiceForeignKey'
  DESC 'ForeignKey'
  EQUALITY caseExactMatch
)
objectclass ( 1.3.6.1.4.1.6757.100.1.1.33
    NAME 'GLUE2StorageManager'
    DESC 'The primary software component locally managing one or more
    storage resources. It can describe also aggregated information about
    the managed resources.'
    SUP 'GLUE2Manager'
    STRUCTURAL
    MAY GLUE2StorageManagerStorageServiceForeignKey
)

# File:    schema/7-08-DataStore.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 7.8 DataStore
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University

# DataStore inherits from Resource
attributetype ( 1.3.6.1.4.1.6757.100.1.1.34.1
    NAME 'GLUE2DataStoreType'
    DESC 'Type of storage resource'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.34.2
    NAME 'GLUE2DataStoreLatency'
    DESC 'The maximum latency category for a file stored in this
    resource to be made available for reading'
    EQUALITY caseExactMatch
    SUBSTR caseExactSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.34.3
    NAME 'GLUE2DataStoreTotalSize'
    DESC 'Size of storage extent'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.34.4
    NAME 'GLUE2DataStoreFreeSize'
    DESC 'Size of free storage extent'
    EQUALITY integerMatch
    ORDERING integerOrderingMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
    SINGLE-VALUE
)
attributetype ( 1.3.6.1.4.1.6757.100.1.1.34.5
NAME 'GLUE2DataStoreUsedSize'
DESC 'Size of used storage extent'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.34.6
NAME 'GLUE2DataStoreStorageManagerForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )

objectclass ( 1.3.6.1.4.1.6757.100.1.1.34
NAME 'GLUE2DataStore'
DESC 'Abstraction of a sufficiently homogeneous storage device
providing a storage capacity, managed by a local software component
(storage manager), part of a storage service, reachable via one or
more endpoints and having one or more shares defined on it. A storage
resource refers to a category with summary information on the
capacity'
SUP 'GLUE2Resource'
STRUCTURAL
MUST ( GLUE2DataStoreType $ GLUE2DataStoreLatency )
MAY ( GLUE2DataStoreTotalSize $ GLUE2DataStoreFreeSize $ GLUE2DataStoreUsedSize $ GLUE2DataStoreStorageManagerForeignKey )
)

# File:     schema/7-09-ToComputingService.schema
# URL:     http://forge.gridforum.org/sf/projects/glue-wg
# Doc:     GLUE Specification 2.0 (March 3, 2009)
# Section: 7.9 ToComputingService
# Authors: Laurence Field (laurence.field@cern.ch), CERN
#          David Horat (david.horat@cern.ch), CERN
#          Florido Paganelli (florido.paganelli@hep.lu.se), Lund University
# ToComputingService inherits from Entity

attributetype ( 1.3.6.1.4.1.6757.100.1.1.35.1
NAME 'GLUE2ToComputingServiceID'
DESC 'Globally unique ID'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.6757.100.1.1.35.2
NAME 'GLUE2ToComputingServiceNetworkInfo'
DESC 'Type of network available among the storage service and computing service'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
attributetype ( 1.3.6.1.4.1.6757.100.1.1.35.3
NAME 'GLUE2ToComputingServiceBandwidth'
DESC 'The nominal bandwidth available between the storage service
and computing service'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.35.4
NAME 'GLUE2ToComputingServiceStorageAccessProtocolForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.35.5
NAME 'GLUE2ToComputingServiceComputingServiceForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

attributetype ( 1.3.6.1.4.1.6757.100.1.1.35.6
NAME 'GLUE2ToComputingServiceStorageServiceForeignKey'
DESC 'ForeignKey'
EQUALITY caseExactMatch
SUBSTR caseExactSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

objectclass ( 1.3.6.1.4.1.6757.100.1.1.35
NAME 'GLUE2ToComputingService'
SUP GLUE2Entity
DESC 'Description of the network link quality of a storage service
to a computing service'
STRUCTURAL
MUST ( GLUE2ToComputingServiceID $
GLUE2ToComputingServiceComputingServiceForeignKey $
GLUE2ToComputingServiceStorageServiceForeignKey )
MAY ( GLUE2ToComputingServiceNetworkInfo $
GLUE2ToComputingServiceBandwidth $
GLUE2ToComputingServiceStorageAccessProtocolForeignKey )
)