OGF GLUE2 Enumerations procedures and best practices

Author: Florido Paganelli florido.paganelli@hep.lu.se

Last revision: 2014-03-25

1 Introduction

This work is an attempt to summarize and generalize the usage of Open Enumerations in GLUE2, and an attempt to set some rules on how the management of these should be carried on.

1.1 Acknowledgements:

Thanks to Stephen Burke, David Meredith, Paul Millar and Balazs Konya for kickstart of these ideas and their further development. Thanks to John-Paul Navarro and Shiraz Memon for coordinating this work within the OGF GLUE2 Working Group.

Table of Contents

2 Naming schemes

The following rules apply:

1) Every string SHOULD be lowercase.

2) Clients SHOULD consider these strings case-insensitive.

2.1 Services:

See also GFD.147, Appendix B.31

`<organization_name>.<product_name>.<service_detail>`

where

`<organization_name>`
  is the reversed domain name of the organization.
  Presence: **MANDATORY** for new names

`<product_name>`
  free form string for product names
  Presence: **MANDATORY**

`<service_detail>`
  free form string that SHOULD identify a subservice or a service implementation. It can contain dots.
  Presence: **NOT MANDATORY**

Decision: minumum requirement is

`<organization_name>.<product_name>`

Examples: `org.nordugrid.arex`, `org.glite.ce.cream`

Exisiting names not following the above rules can be kept, but should be Deprecated (see Section 4.1)

2.2 Interfaces:

GFD.147 Appendix B.18 does not define any clear format. The following should be RECOMMENDED:

`<organization_name>.<interface_name>`

where:

`<organization_name>`
  is the reversed domain name of the organization
  Presence: **MANDATORY** for new names

`<interface_name>`
  free form string for product names

Presence: **MANDATORY**

Examples: `org.nordugrid.xbes, org.ogf.glue.emies.activitycreation,`
`org.glite.voms`

2.3 Capabilities:

See GFD.147, appendix B.5, on *Capability_t*.

2.4 Sorting out special classes of names

In case the organization name is unknown/undefined/short-lived

Use the reserved prefix:

`org.ogf.glue.`

Decision: org.ogf.glue.* will be used as there are issues with domain names in some cases, they must be officially registered.

Example:
For EMIR, developed during EMI project that now is over, *ServiceType_t* is:
`org.ogf.glue.emir`

Simplified naming for Services with only one Interface

A specific Endpoint interface can be bound to a specific Service. In such case, to simplify name creation, *InterfaceName_t* MAY be the same of a corresponding *ServiceType_t* .

3 Requesting new Open Enumerations

A community willing to submit new open enumeration MUST:

1. Provide the following information about the enumeration to be submitted:

    1. **Type** of the enumeration to be added (i.e. one of the types described in GFD.147, Appendix B)

    2. **Name** of the enumeration as in the guidelines in Section 2;

    3. **Description** of the enumeration. This description is a textual, human readable summary of what the object described by the open enumeration does.

    4. **Reference email** to be contacted by the OGF group during evaluation of the proposed strings. This email will not be published and will be internal to the group.

2. Send the above information to the OGF group at the address glue-wg@ogf.org with subject "[Enumerations] Request for addition/deletion/modification"

NOTE: Requests that are missing the information listed in 1 will be automatically rejected. The proposer will be asked to provide missing information.

4 Management of Open Enumerations

The group preserves the current list of Open Enumerations in the git repository at the link

https://github.com/OGF-GLUE/Enumerations

 in the form of CSV files. Section 4.1 contains a description of the content of the CSV files.

The groups keeps them updated using the procedures listed in this document, defined in Section 4.2.

4.1 Description of Open Enumerations CSV files

CSV filenames in github are of the form:

Filename	Purpose
`<EnumerationName_t>.csv` *Example:* `ServiceType_t.csv`	Contains Open Enumerations officially approved by the group.
`<EnumerationName_t>-draft.csv` *Example:* `ServiceType_t-draft.csv`	Contains Open Enumerations pending approval from the group

Records contained in the CSV files are as follows:

1. The first row always contain the following strings:
   `<EnumerationName_t>,Description,Status`
   where `<EnumerationName_t>` must be the same as the CSV filename, i.e.
   `ServiceType_t`

2. Rows from the second on contains the following:

Field name	Purpose	Example
`<EnumerationName>`	It's the enumeration value.	`org.nordugrid.arex`
`Description`	Contains the textual description of the object represented by the enumeration name.	`NorduGrid Resource Coupled Execution Service`
`Status`	See discussion below.	`Recommended`

Statuses

Recommended: name is choosen and proposed for everybody to use.
Obsolete: name is old and might not be used in the future.
A name can be **only one** of Obsolete OR Recommended, but can always be
Deprecated: name SHOULD NOT be used.
So far there was no agreement on **Deprecated**. But I think is important to have for

"Recommended Obsolete" and "Deprecated Obsolete" can co-exist. The CSV will contain a multi-line field for those.

4.2 GLUE2 Working Group Procedures

1. When a request arrives, a selected member of the group adds the names to <EnumerationType_t>-draft.csv document. (e.g. ServiceType_t-draft.csv). These values will have no *Status* in the -draft.csv document.

2. Promplty upon arrival the group decides about the feasibility of adding the values. Enumerations in draft state can be used by users but are in the process of being accepted. Consumers can use them at their own risk. If the values are accepted, the *Status* field SHOULD contain `Recommended`. Evaluation rules are defined later in this section.

3. As soon as possible, the group decides and merges the -draft files with the official ones.

4. If the above is not possible, at least once a year the group meets and does the merge on all documents. This MUST be a scheduled venue.

Evaluation rules of thumb

1. Namespace - if it uses a DNS-style name does the requestor reasonably justify relating it to that domain, i.e. do they speak for the relevant project or is it clearly related to a standard service provided by the project. See also Section 2.

2. Checking that we don't already have something suitable in the existing list.

3. Sanity checking that the value is naming something that matches the conceptual purpose of the enumeration.