OpenGridForum

OPEN FORUM | OPEN STANDARDS

GLUE2 – XML Rendering – Update

GLUE Working Group

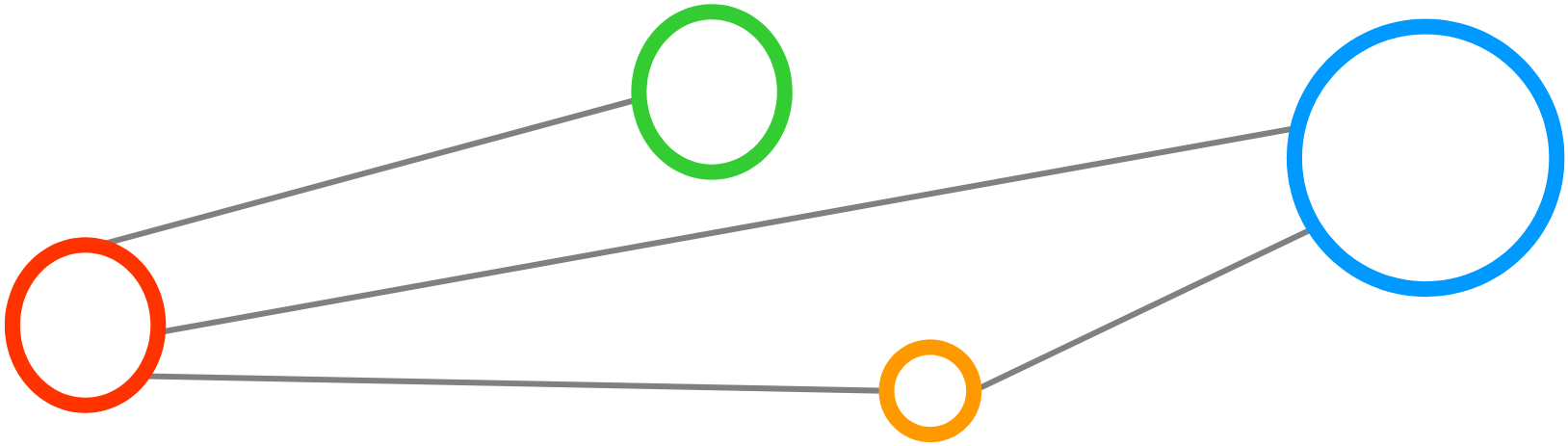
Morris Riedel et al.

OGF IPR Policies Apply

- "I acknowledge that participation in this meeting is subject to the OGF Intellectual Property Policy."
- Intellectual Property Notices Note Well: All statements related to the activities of the OGF and addressed to the OGF are subject to all provisions of Appendix B of GFD-C.1, which grants to the OGF and its participants certain licenses and rights in such statements. Such statements include verbal statements in OGF meetings, as well as written and electronic communications made at any time or place, which are addressed to:
    - the OGF plenary session,
    - any OGF working group or portion thereof,
    - the OGF Board of Directors, the GFSG, or any member thereof on behalf of the OGF,
    - the ADCOM, or any member thereof on behalf of the ADCOM,
    - any OGF mailing list, including any group list, or any other list functioning under OGF auspices,
    - the OGF Editor or the document authoring and review process
- Statements made outside of a OGF meeting, mailing list or other function, that are clearly not intended to be input to an OGF activity, group or function, are not subject to these provisions.
- Excerpt from Appendix B of GFD-C.1: "Where the OGF knows of rights, or claimed rights, the OGF secretariat shall attempt to obtain from the claimant of such rights, a written assurance that upon approval by the GFSG of the relevant OGF document(s), any party will be able to obtain the right to implement, use and distribute the technology or works when implementing, using or distributing technology based upon the specific specification(s) under openly specified, reasonable, non-discriminatory terms. The working group or research group proposing the use of the technology with respect to which the proprietary rights are claimed may assist the OGF secretariat in this effort. The results of this procedure shall not affect advancement of document, except that the GFSG may defer approval where a delay may facilitate the obtaining of such assurances. The results will, however, be recorded by the OGF Secretariat, and made available. The GFSG may also direct that a summary of the results be included in any GFD published containing the specification."

- OGF Intellectual Property Policies are adapted from the IETF Intellectual Property Policies that support the Internet Standards Process.

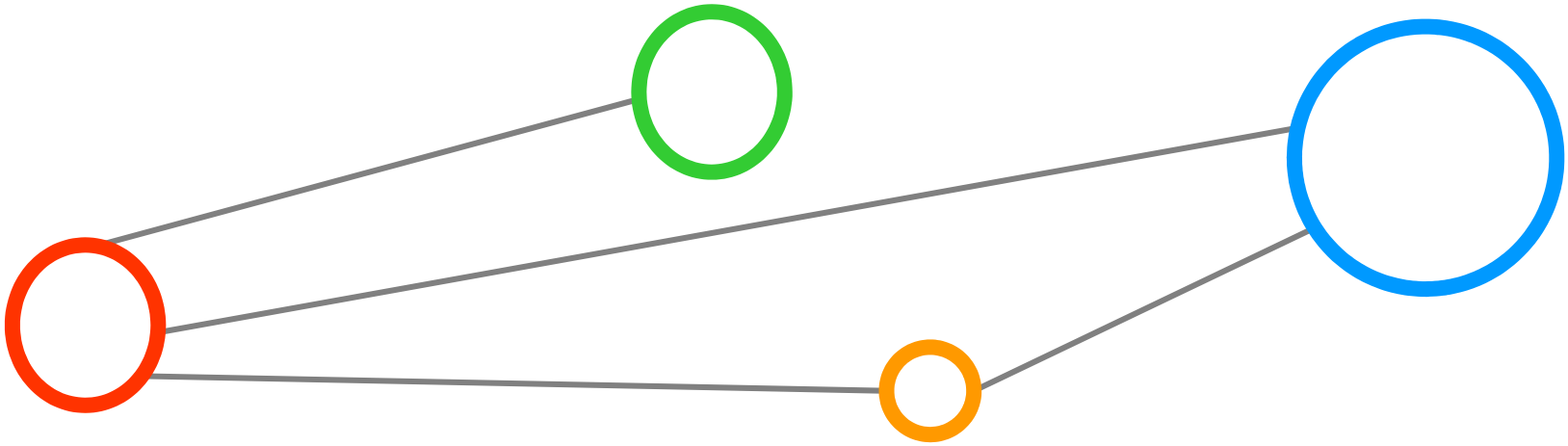
Outline



Outline

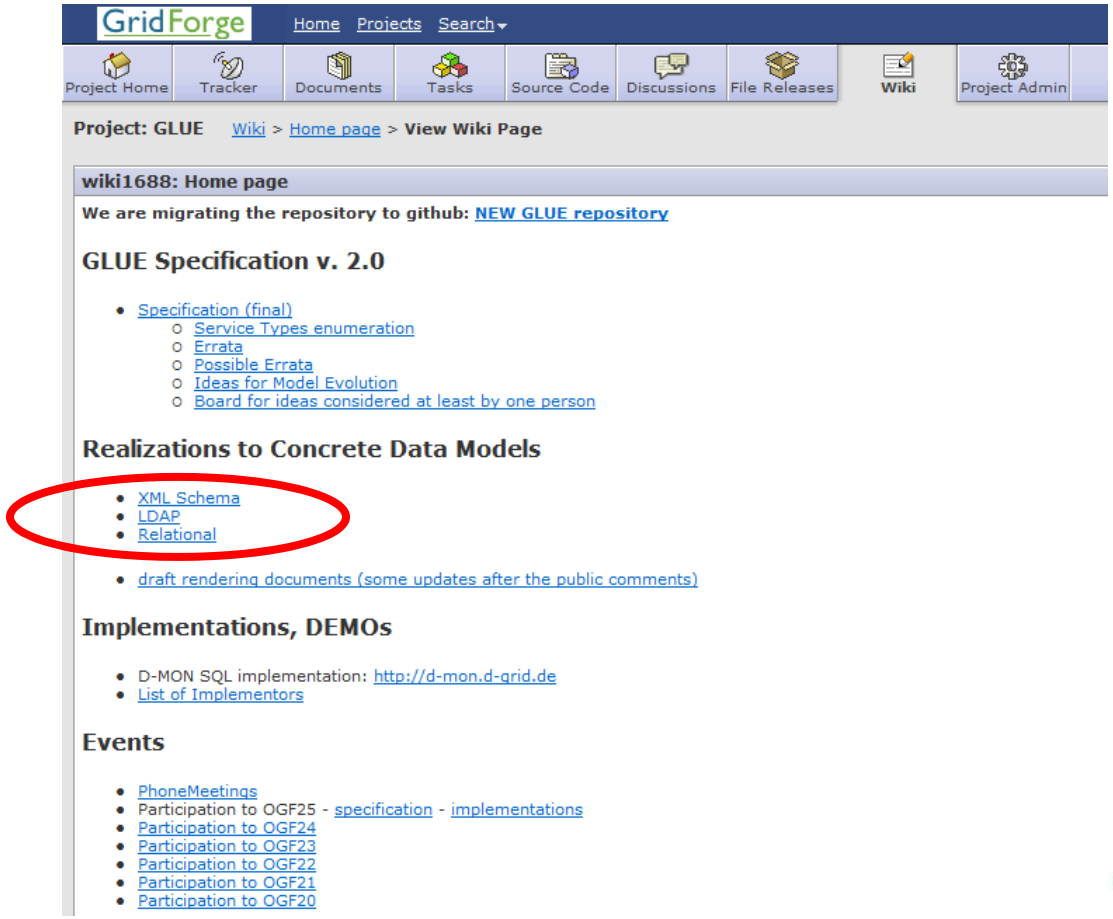
- Welcome & Progress
- Stakeholder PGI
- Other stakeholders?
- Next Steps

Welcome & Progress



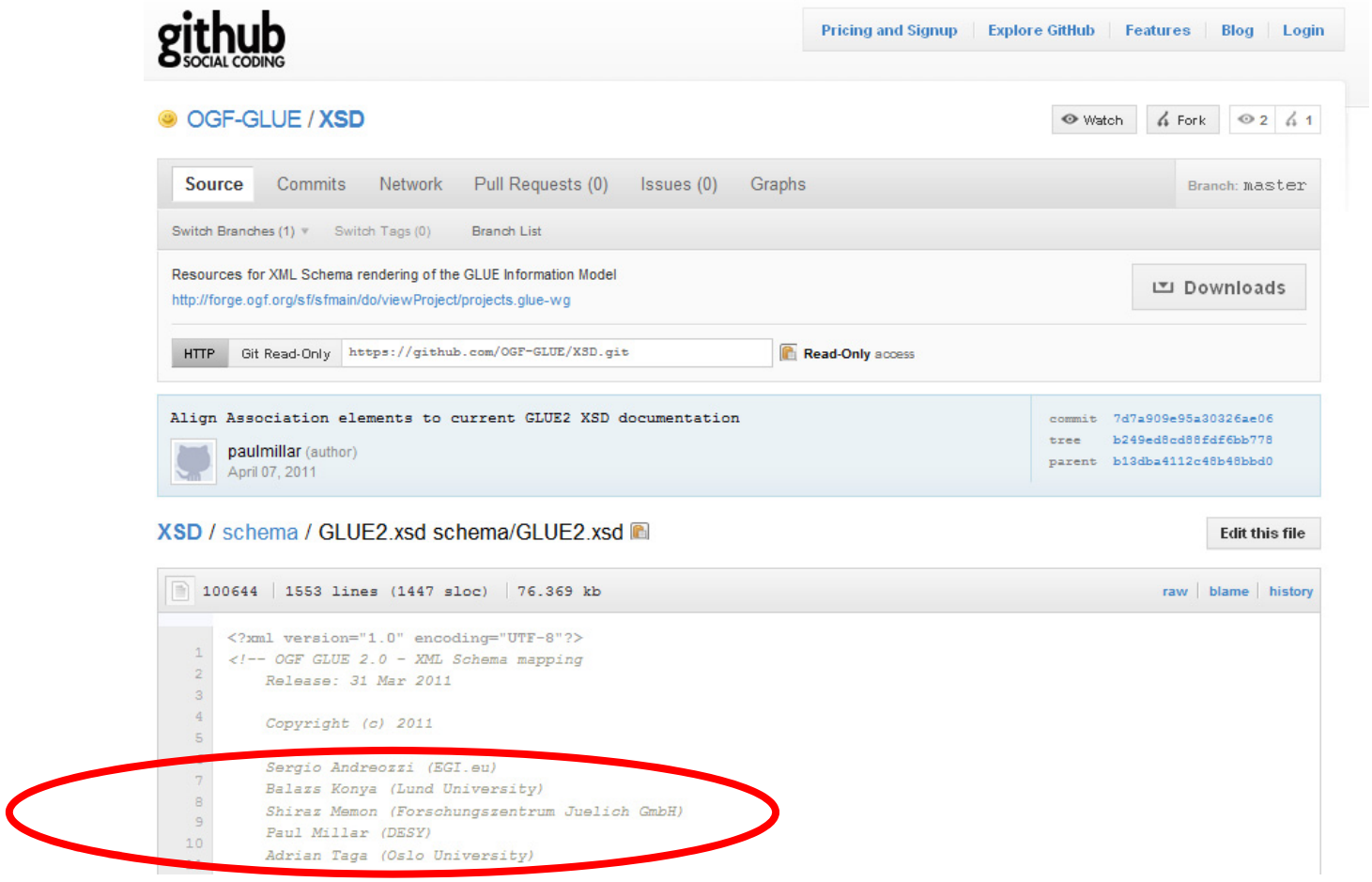
Welcome & Progress

- All material online
  - http://forge.ogf.org/sf/wiki/do/viewPage/projects.glue-wg/wiki/HomePage



XML Rendering Schema

- XSD is in GitHub
  - https://github.com/OGF-GLUE/XSD/blob/master/schema/GLUE2.xsd



XML Rendering Specification

- Specification in GridForge
  - http://forge.ogf.org/sf/go/doc15514

April 4, 2011

GLUE v. 2.0 – Reference Realization to XML Schema

Status of This Document

This document provides information to the Grid community regarding the realization of the GLUE information model (v.2.0) as XML Schema. Distribution is unlimited. This realization is derived from the proposed recommendation of the specification document [glue-2].

Copyright Notice

Copyright © Open Grid Forum (2011). All Rights Reserved.

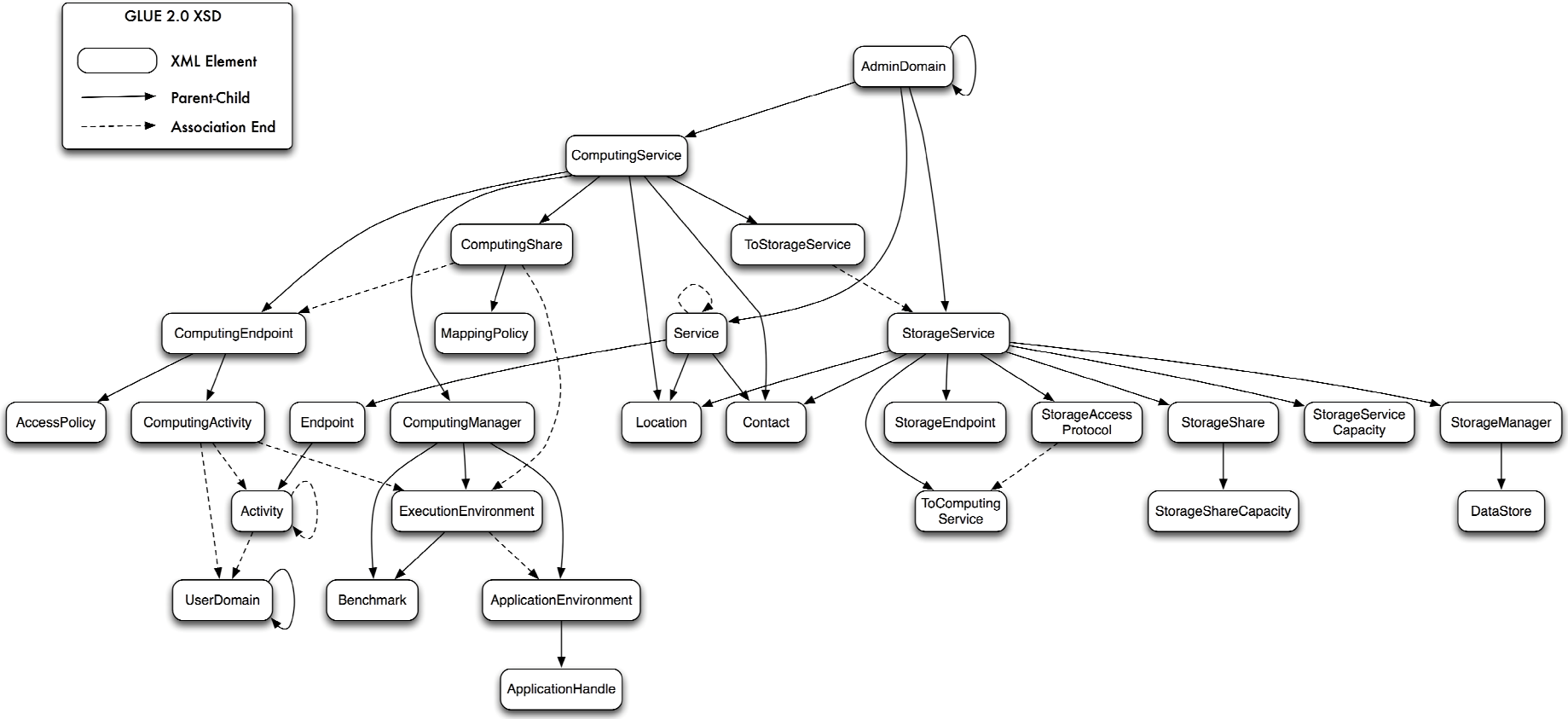
Trademark

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 realization of this information model as XML Schema.

XML Elements Overview



Remaining Work & Challenges

- 99% completeness of specification and schema
- Many editorial comments from Paul
  - Paul Millar and Sergio Andreozzi work on putting them into the document → just work, no issues

- Also not addressed TeraGrid comments still
  - Warren Smith (2nd May) comments
    - http://www.ogf.org/pipermail/glue-wg/2011-May/001119.html
  - Neither discussed nor addressed so far
  - Generally XSD ok, but several improvements & corrections
  - Performance in processing information issues (hierarchy)
  - Part of older discussions „flat vs. not-flat“

Open Comments (1)

- “I've been looking at the XML schema a bit and I don't see anything in there that would mean we couldn't do what we're doing now on TeraGrid.”
- I do see a few things that could be improved with this schema. In some places, it specifies a structure like:

- <Foos>
- <Foo> ... </Foo>
- <Foo> ... </Foo>
- </Foos>

Open Comments (2)

- “I don't see any need for the Foos element to contain multiple Foo elements.”

- I also see that a number of the top-level elements don't have a minOccurs="0", so they are required, but probably shouldn't be.
- Similarly, they don't have a maxOccurs, so each one can only occur once (which doesn't make sense for some of them).

Open Comments (3)

- I assume these are just mistakes. One thing to think about is whether or not you want to have a top level element to contain everything.
- For example, an element named glue2 (for the TeraGrid GLUE 2 schema I used a top level element of Entities, which is a bit generic).
- The main benefits I can see to having a single top level element are that it might be easier to search for GLUE 2 documents in an XML database and it might be easier to embed a GLUE 2 document inside another XML document.

Open Comments (4)

- At a higher level, I'll throw out my preference again for a "flat" rendering approach (like the TeraGrid one).
- I see that this schema is about half way to a totally flat approach by representing the many-to-many associations with IDs.
- I see that a fair amount of analysis (the table and diagram in the document) had to be done to get to the point and it might be simpler all around to just go all the way and use IDs for all associations and make things consistent.

Open Comments (5)

- I also still do prefer the flat approach so that it is easier to construct documents with a subset of the GLUE 2 information.
- We wouldn't have to try to construct a hierarchy of GLUE 2 information (that we don't care about in that context) to have a valid GLUE 2 document with the information we are interested in.
- For example, if we just want to publish information about the ComputingEndpoints for a cluster.

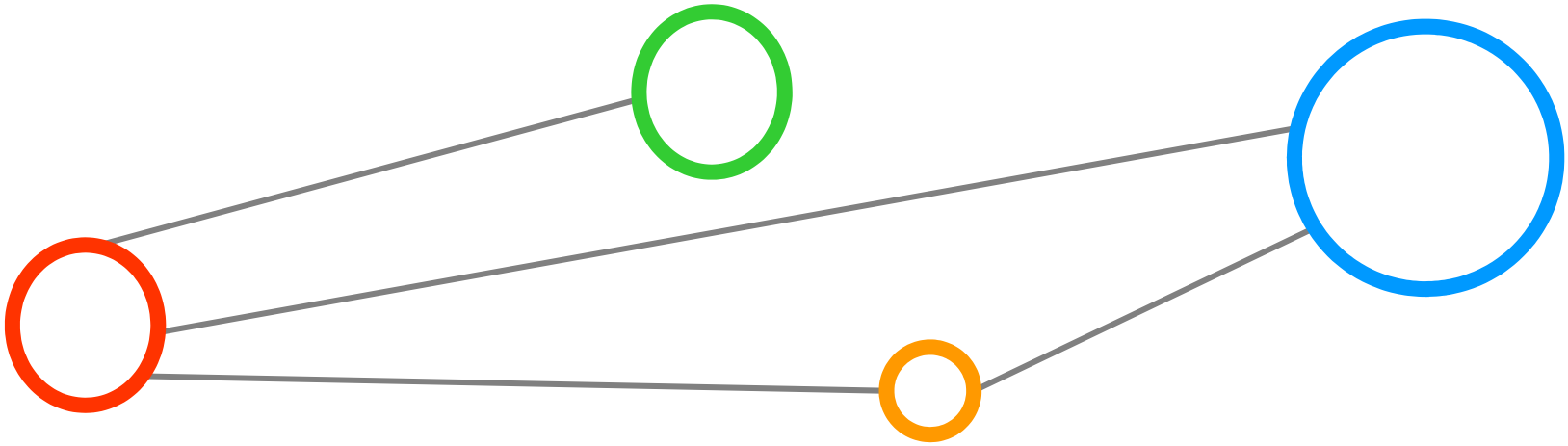
Open Comments (6)

- There is also an implication to a flat hierarchy that I like - a single GLUE 2 document may only have partial information about a resource or grid and may need to be composed with other documents for the full picture.
- This makes sense to me because some parts of GLUE 2 are relatively static and probably entered manually (domains, locations, contacts), some parts are dynamic but can be discovered automatically from the right system (compute-related stuff for clusters from a login node), and others are somewhat dynamic and may be discoverable from the right system (storage-related stuff from the storage system).

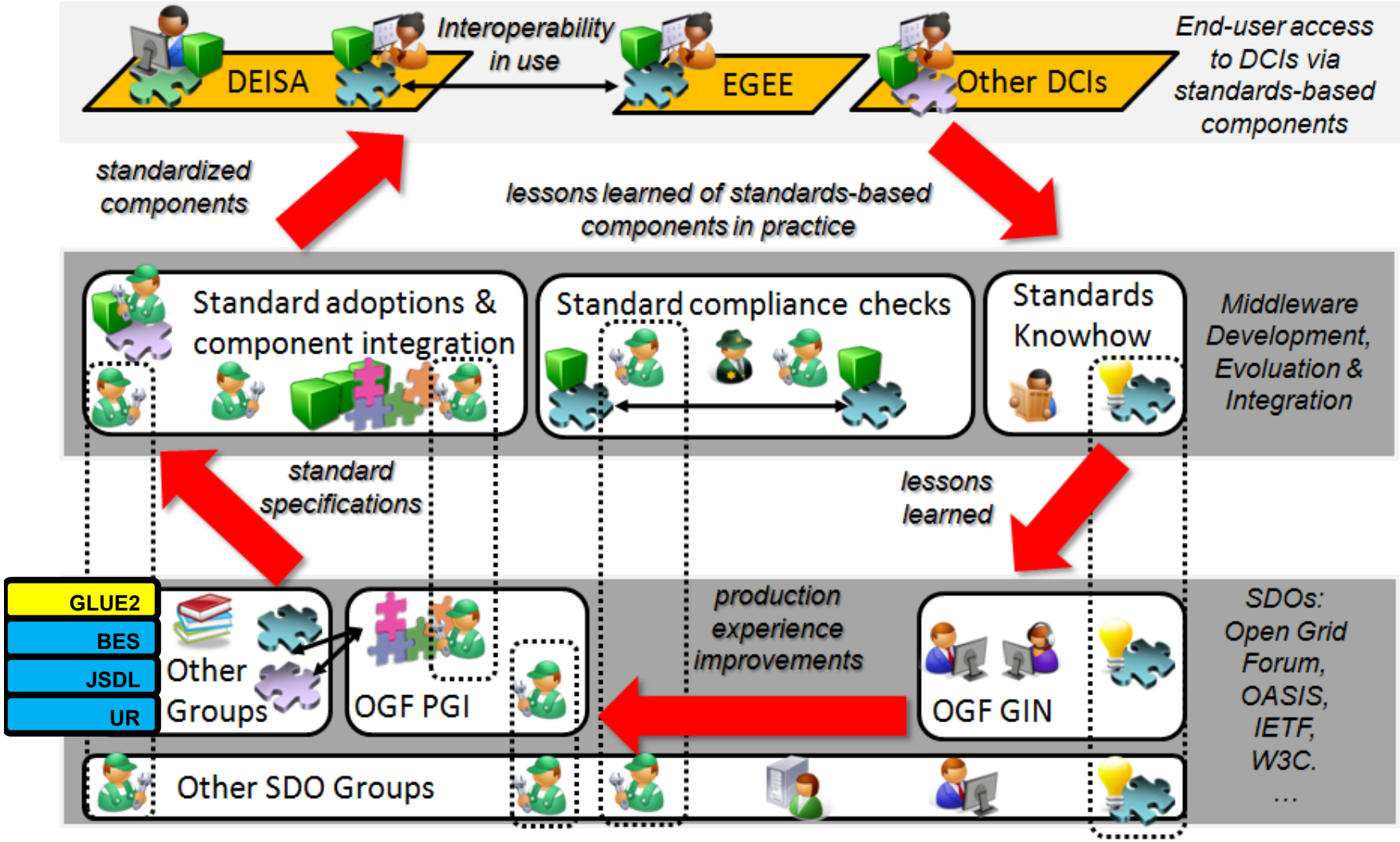
Open Comments (7)

- I think it would be nice if these documents can be constructed independently, not have extraneous information, and still be valid GLUE 2 documents before they get merged (if they do get merged).

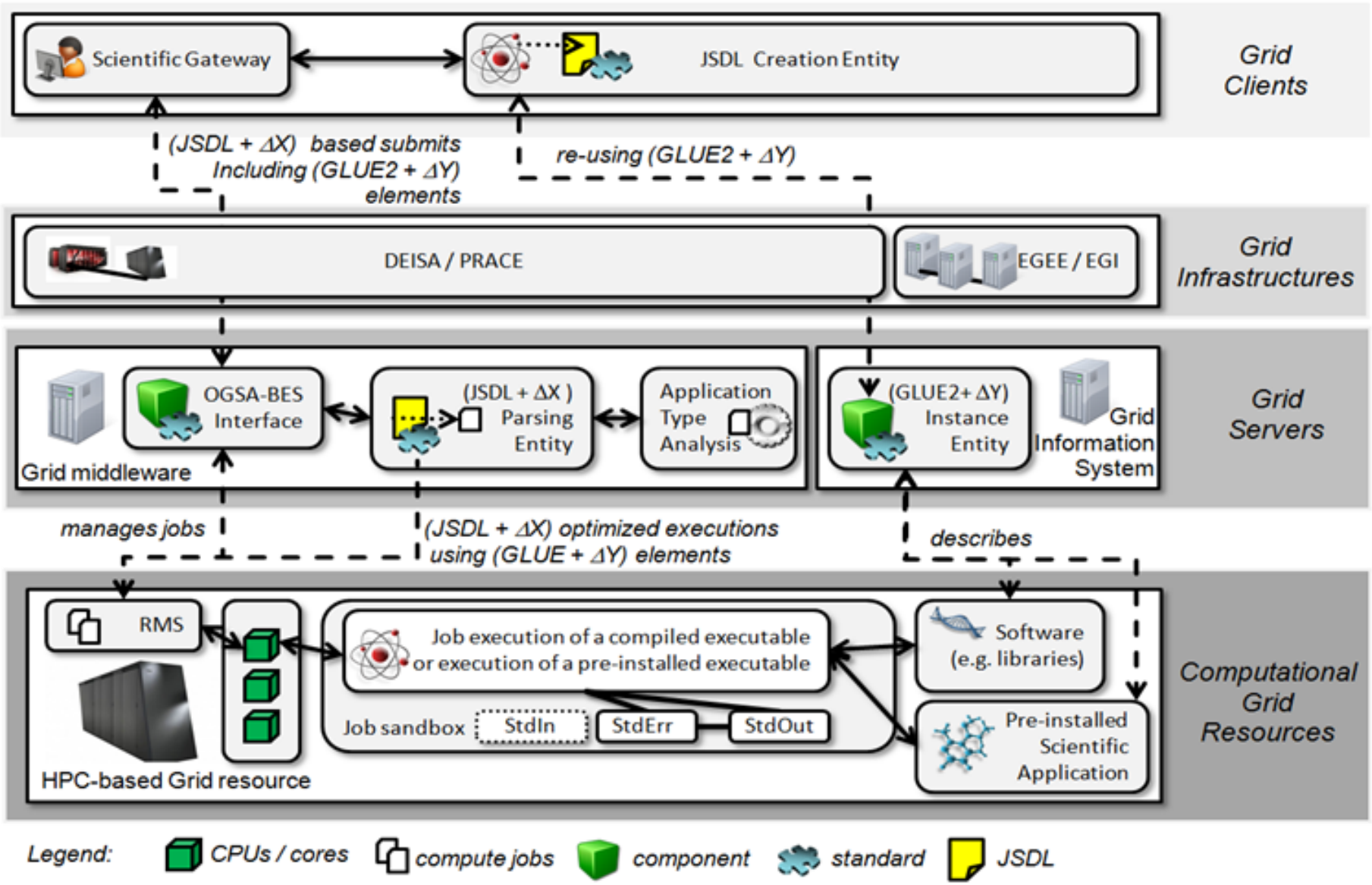
Stakeholder PGI



PGI & GIN Ecosystem



PGI GLUE2 Overall Ecosystem



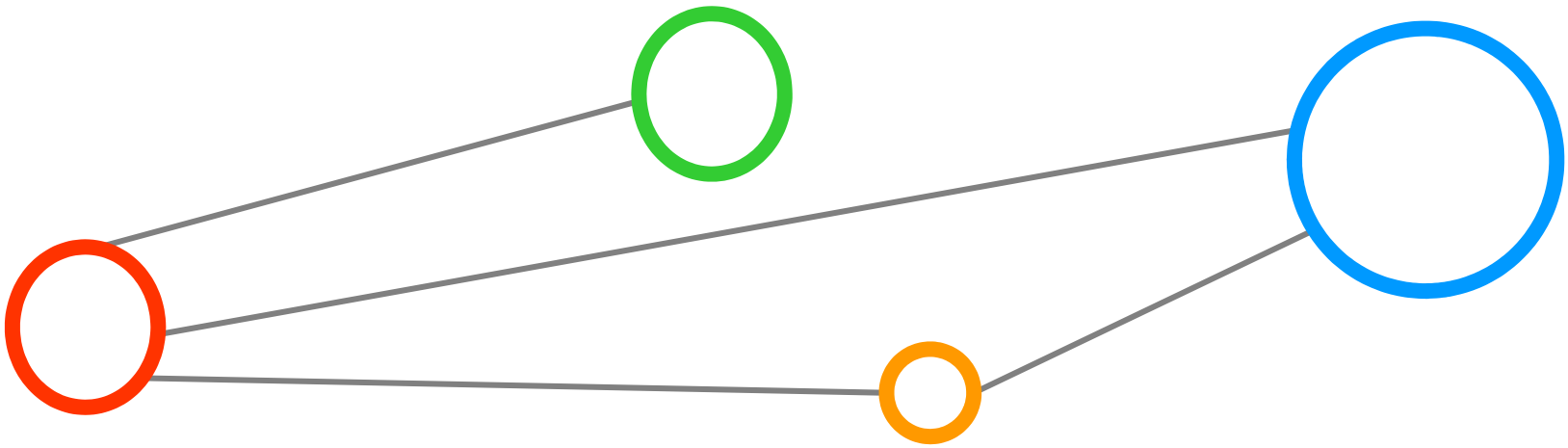
Major Dependencies

- A PGI Execution Service should have the capability of...
  - publishing information about the resource characteristic exposed by the service
  - and publishing information of the detailed properties of the activities being managed by the service
  - “the ComputingService and the generic Service elements of the GLUE2 XML rendering” make sense

- Bottom line: Use of GLUE2 XML rendering specification
  - Resource and Activity information
  - E.g. GLUE2 OSFamily_t type, OSName_t type, Platform_t type,...

- (Not forgetting: we need more than GLUE2 → + Delta Y!)

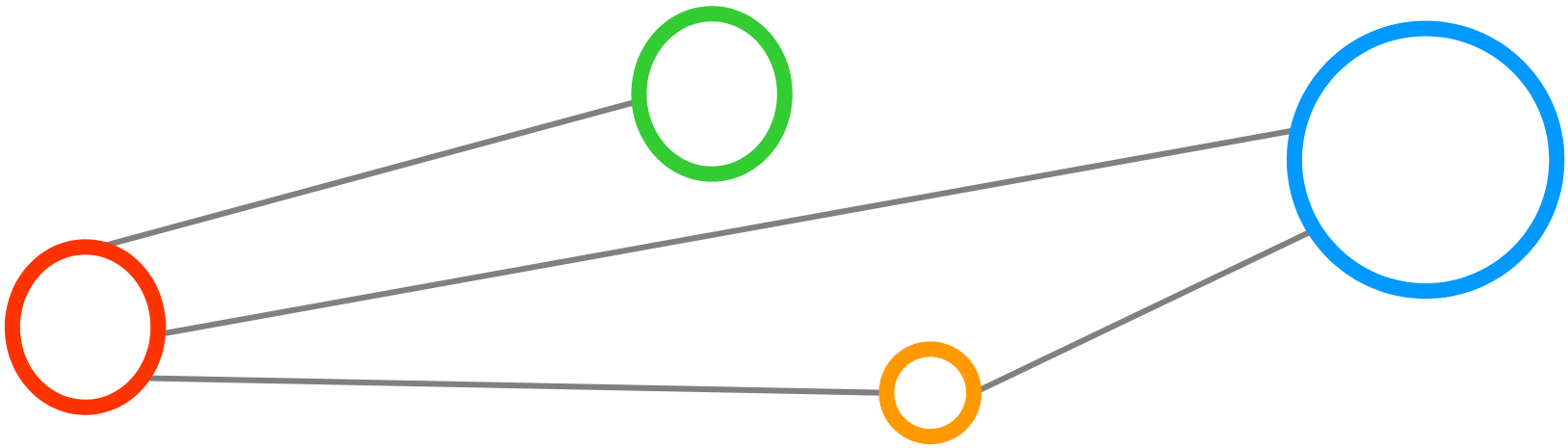
Other Stakeholders?



Other Stakeholders?

- TBD: Audience – Who implements apart from PGI & EMI
- Maybe RENKEI Follow-on
- Castor information provider (consider XML)

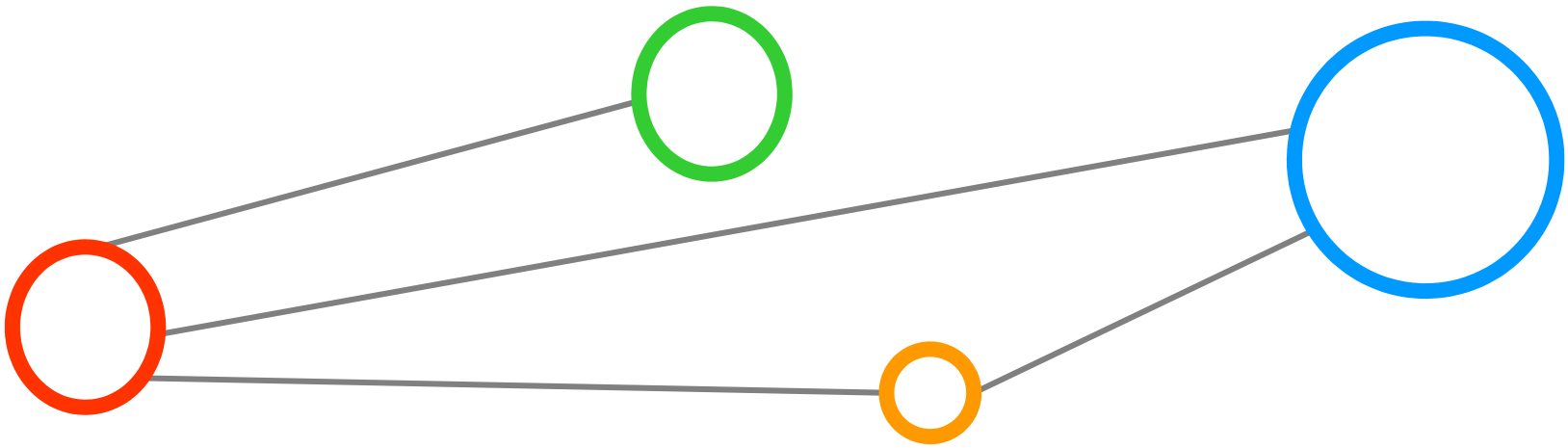
Next Steps



Next Steps

- Chairs need to work on the „non-Lundland-comments“
- Detailed inputs from the BES/JSDL/GLUE2 joint venture from PGI
- GLUE2 XML rendering will play a major role

Comments



Comments

- Production experience 44% volume is application elements (David) – overload of info systems?
- Discussions: Flat seem to have advantages
- Update challenges of ‚parts of the information‘

- Benefits von hierarchy needs to be understood

- Open Questions
  - Other service types → authN, authZ
  - Extensibility mechanisms for central services (not compute, not storage)
  - Profiling on top of GLUE2 then

Changes for the future

- The ComputingService object encapsulates all the “computing element” characteristics while the generic Service element is used to publish the associated Activity-Factory endpoints
- Some extensions neceesary
- a possibility to provide information about service features, specifically
  - notification capabilities
  - supported data staging protocols (experts ?)
  - exclusive execution of jobs on worker nodes
  - remote session directory access including supported protocols
- parallel environments
- session directory location
- More granularity resource information needed...
- Network groups Input → connectivity services

- Jens: Done in the past – capabilities
  - GLUE2 is designed to be extensible – so profiling is easy

Full Copyright Notice