Enabling Grids for E-science in Europe

www.eu-egee.org

JRA1 all-hands meeting, June 29 2004

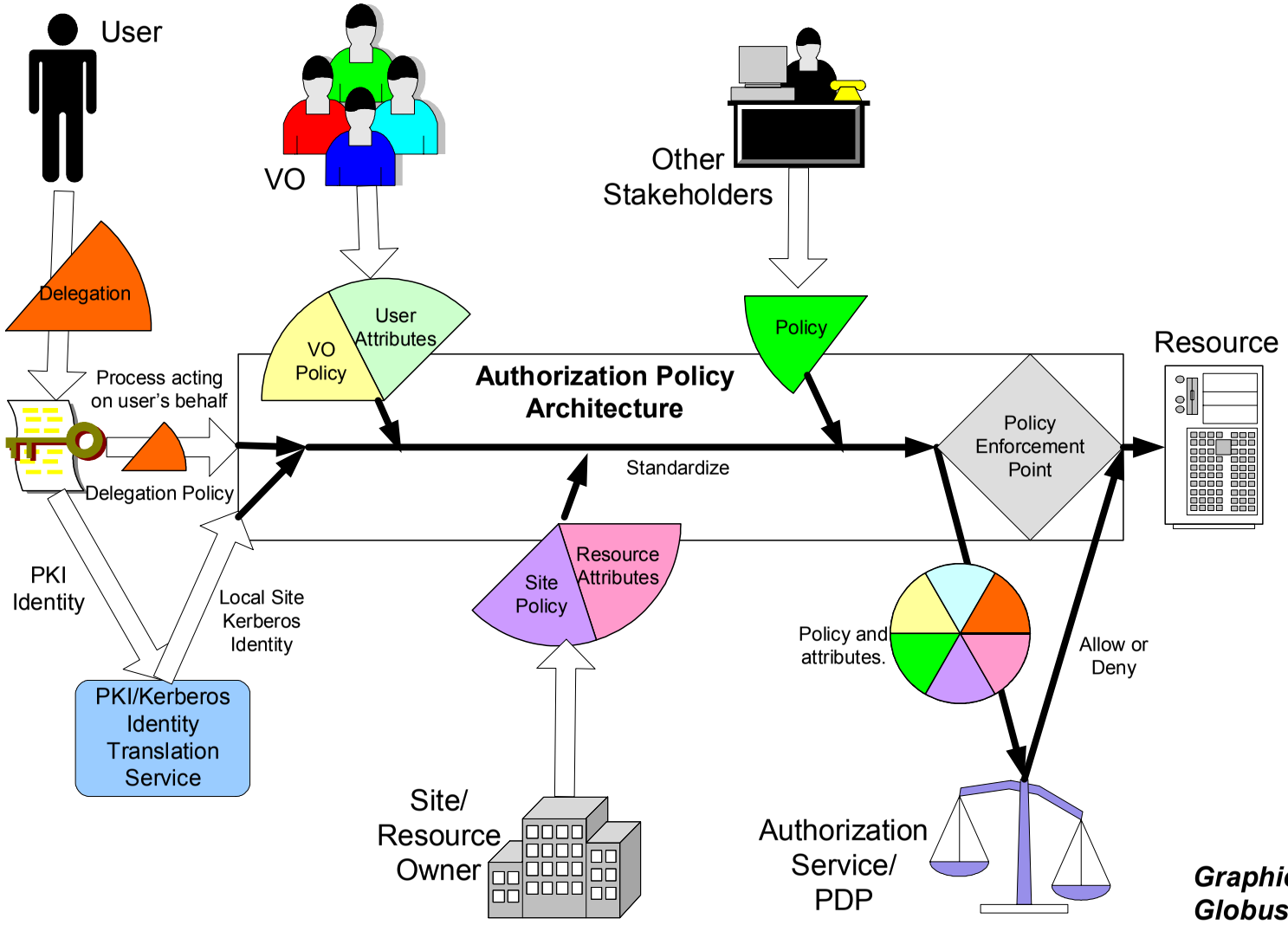
Common Security Components

Olle Mulmo
JRA3

EGEE is a project funded by the European Union under contract IST-2003-508833

Our goal: it's all about policy enforcement

Enabling Grids for E-science in Europe



Graphics from Globus Alliance & GGF OGSA-WG

Software development at JRA3

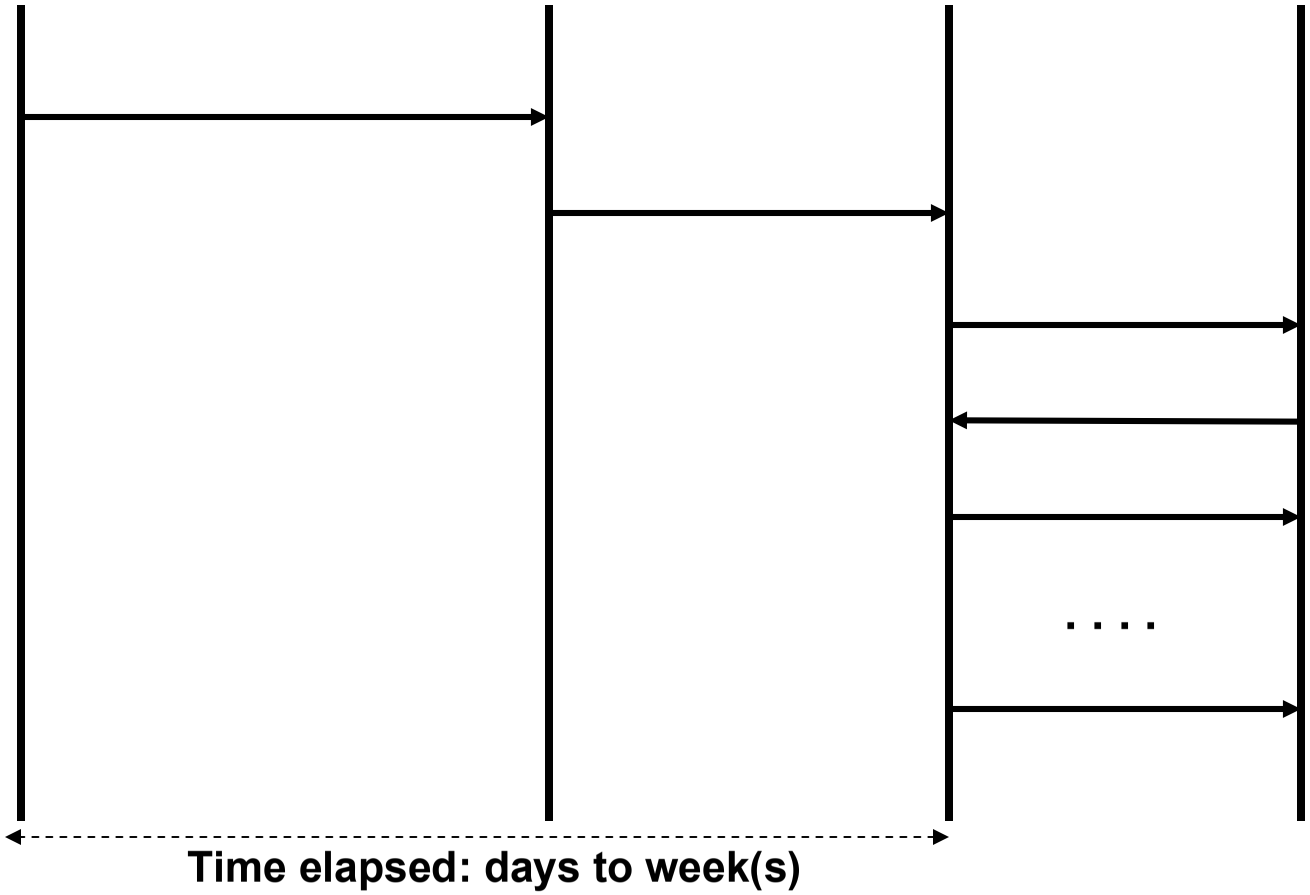
Enabling Grids for E-science in Europe

Planning
-Requirements Development
-Project estimates
-Project plan

High Level Design
-Architectural design

Stage 1,..,N
-Detailed design
-Construction
-Release

JRA1 SCM



Software development at JRA3

Enabling Grids for E-science in Europe

Deliverable	Description
Requirement specifications	Document describing the identified requirements for the deliverable. The success of the task will be measured against this. Iterated with JRA1. Handled by Change Control…
Project estimates	Setting the scope for the project. Resources needed – people and time. Will be revised after each phase in the project.
(QA, see JRA1 and JRA3 QA plan, and metrics)	
Architecture design	High-level specification, the plan to build the right solution to the problem.
Detailed design	Detailed-level design, the plan to build the right solution in the right way.
Staged Delivery Plan	Defining the order in which the software will be constructed. Maximizing JRA1 value (e.g. choose the most important modules first if possible), minimizing risk.
Construction and release	(See developers guidelines from JRA2)

... but HOW?

Enabling Grids for E-science in Europe

- Libraries?
  - All functionality must be integrated in the service implementation
- Handlers / Filters?
  - Orthogonality (wrap/unwrap)
  - Extensibility (delegation, mutual authZ)
- Languages? Priorities still somewhat unclear
  - Want JRA1 “customers” - all not identified yet

SOAP over HTTPS

Enabling Grids for E-science in Europe

Why?	A transport-level security solution to enable authenticity, integrity, and confidentiality to SOAP-based interactions.
How?	ProxyCert-aware TLS implementation, plugged into a SOAP engine. J: Axis handler, reuse EDG java-sec + CoG C: gSOAP plugin from IT (license?)

Message Level Security

Enabling Grids for E-science in Europe

Why?	Allows for WS-I compliance, enables endpoint trust, and in some cases, eliminates the need for delegation
How?	Make use of existing implementations that aim at WS-I compliance. J: Refit Globus WS-Sec handlers with more recent WSS4J implementation from Apache C: ???

Delegation portType

Enabling Grids for E-science in Europe

Why?	Disentangle authentication from delegation, making it an optional feature, reducing overhead and enabling use of “normal” HTTPS.
How?	J: Axis handler, client+server C: Apache filter (GridPP), client plugin

AuthZ framework

Enabling Grids for E-science in Europe

Why?	To allow for combination of policies from multiple sources, and to enforce them
How?	J: framework + Axis handler wrapper C: framework + globus authz callout wrapper

Workload Management, “LCAS”

Enabling Grids for E-science in Europe

Why?	Enables a CE to make complex decisions based on job management specific domain knowledge
How?	Evolve/wrap current implementation as AuthZ framework module(s), work closely with IT+CZ cluster.

Mutual AuthZ

Enabling Grids for E-science in Europe

Why?	*Insert soon-to-appear real-world requirement here*
How?	*Infosys?* *VOMS in server certs?* ***getCreds() portType?***

Key Management for Biomed

Enabling Grids for E-science in Europe

Why?	Biomed has strong requirement on runtime access to (encrypted data) stores
How?	Standalone add-on service for key management that may reuse DM infrastructure. Work with NA4 affiliated experts to gather knowledge and detailed requirements. (No software product promised!!!)

Auditing

Enabling Grids for E-science in Europe

Why?	
How?	Coding conventions, use existing log functionality. No special service at this point in time.