SAML-XACML AuthZ Interface
Analysis and design suggestions

Yuri Demchenko

SNE Group, University of Amsterdam

www.eu-egee.org

Information Society

Outline

- Goals and background
- AuthZ components in EGEE/OSG and interoperability picture
- Obligations – definition and use cases
- Reference model for Obligations Handling (OHRM)
- Obligations expression conventions
- Examples, implementations and (inter)operability tests
- Issues for discussion

Goals and background

- **Goals**
  - Common SAML-XACML AuthZ Interface to achieve interoperability between different AuthZ systems
  - Basis for the Site-Central AuthZ Service (SCAS)
- **History and lessons to be learnt**
  - Started/initiated at MWSG11 meetings March 1-2, 2007 at UCSD
  - Development stages:

Agreement – Discussion – Common understanding – (Analysis, Requirements?) - (Design?) – Alpha implementation – (Design?) – Beta Implementation (planed)

- **JRA1 commissioned AuthZ study and technical document drafting**

“SAML-XACML Authorisation Interface and XACML Obligations Handling”

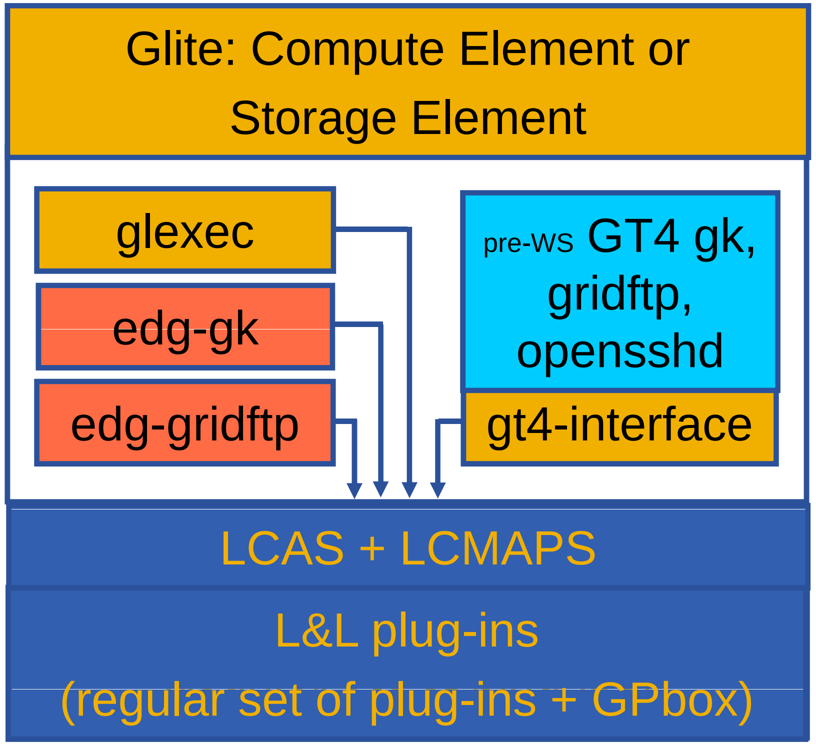
  - http://staff.science.uva.nl/~demch/projects/aaauthreach/draft-authz-saml-xacml-obligations-01.pdf

Document focus

“SAML-XACML Authorisation Interface and XACML Obligations Handling” (version 0.1)

- **Analysis of current AuthZ component**
- **Basic information on SAML2.0, XACML2.0, and SAML2.0 profile of XACML**
- **Proposed design suggestions and solutions**
  - Two basic use cases of the possible SCAS implementation – LCAS/LCMAPS based and native XACML based, that correspondently implement stateful and stateless PDP operational model
  - Description of different obligation enforcement scenarios
  - Obligations Handling Reference Model (OHRM)
  - (Conventional) agreement on the Obligations expression in the XACML policy and applicable XACML Request format
  - ObligationId format and OHRM related Obligation marking/labelling approach
  - Basic (design) requirements to the ObligationHandler API
  - SAML2.0-XACML profile conformance test definition and requirements

Current AuthZ components in EGEE/OSG



Glite: Compute Element or Storage Element

glexec

edg-gk

edg-gridftp

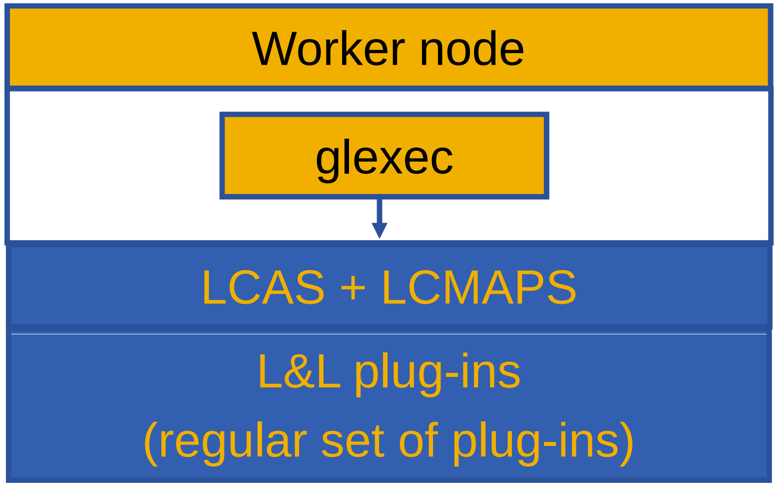
pre-WS GT4 gk, gridftp, opensshd

gt4-interface

LCAS + LCMAPS

L&L plug-ins

(regular set of plug-ins + GPbox)



Worker node

glexec

LCAS + LCMAPS

L&L plug-ins

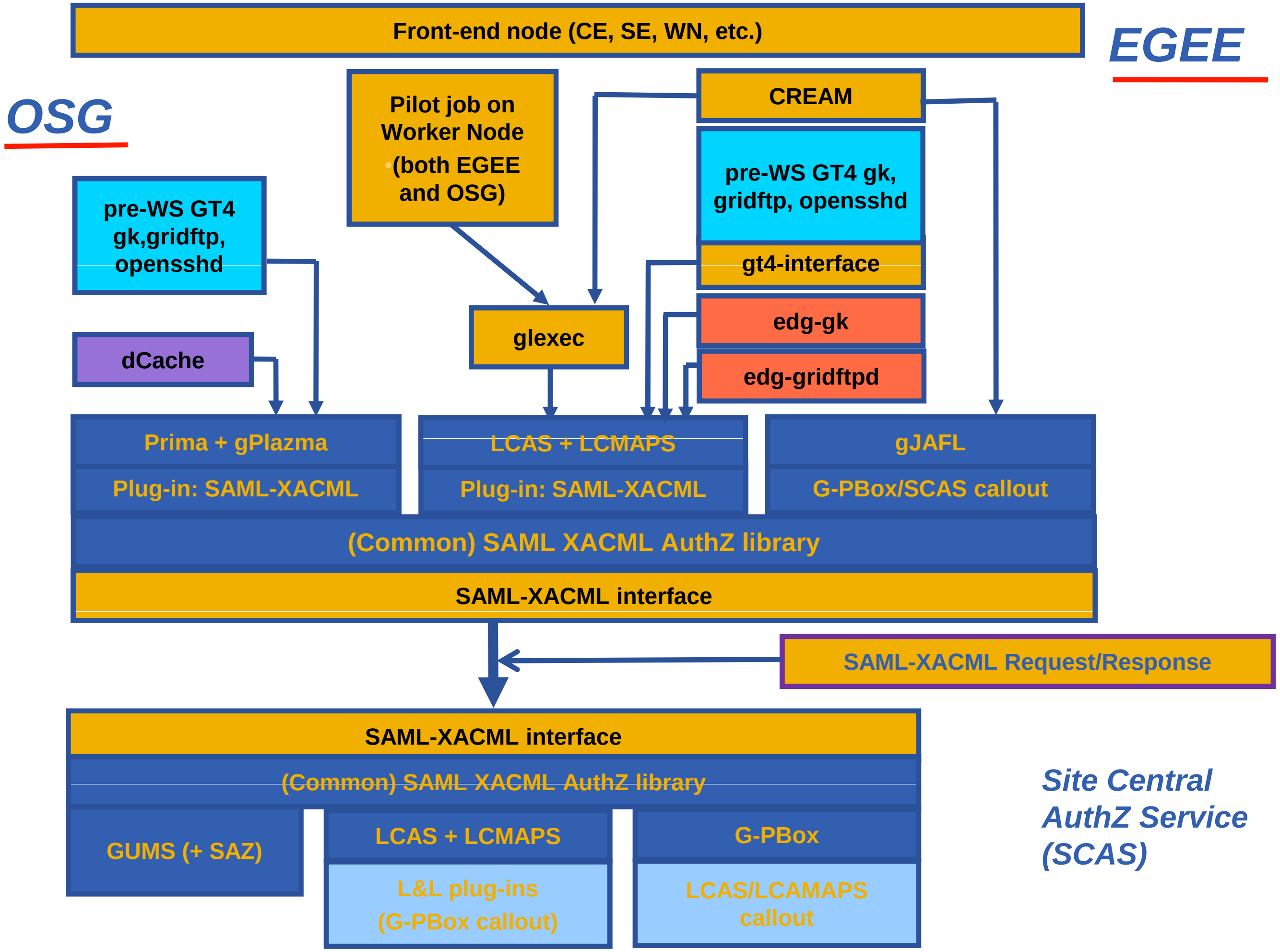
(regular set of plug-ins)

This slide was borrowed from O.Koeroo’s presentation at MWSG/EGEE07

Issues with this setup:

- share/distribute the **gridmapdir** for mapping consistency
- share/distribute the **configurations** for the nodes
- share/distribute **authorization** files, like **grid/groupmapfiles** and a **blacklisting** file
- **Scaling** issues; lots of node will probably **overload** an NFS server

SAML-XACML interface based interoperability picture



Front-end node (CE, SE, WN, etc.)

EGEE

OSG

Pilot job on Worker Node (both EGEE and OSG)

CREAM

pre-WS GT4 gk, gridftp, opensshd

gt4-interface

edg-gk

edg-gridftpd

pre-WS GT4 gk,gridftp, opensshd

dCache

glexec

Prima + gPlazma

Plug-in: SAML-XACML

LCAS + LCMAPS

Plug-in: SAML-XACML

gJAFL

G-PBox/SCAS callout

(Common) SAML XACML AuthZ library

SAML-XACML interface

SAML-XACML Request/Response

SAML-XACML interface

(Common) SAML XACML AuthZ library

GUMS (+ SAZ)

LCAS + LCMAPS

L&L plug-ins (G-PBox callout)

G-PBox

LCAS/LCAMAPS callout

Site Central AuthZ Service (SCAS)

Policy enforcement mechanisms and Obligations

- **Policy Obligation is one of the policy enforcement mechanisms**
  - ***Obligations*** are a set of operations that must be performed by the ***PEP*** in conjunction with an ***authorization decision*** [XACML2.0]
- **Obligations enforcement scenarios**
  - Obligations are enforced by PEP at the time of receiving obligated AuthZ decision from PDP
  - Obligations are enforced at later time when the requestor accesses the resource or service
    - Require use of AuthZ assertions/tickets/(restricted proxy?)
  - Obligations are enforced before or after the resource or service accessed/delivered/consumed
    - Not discussed in current study/document – refer to OGSA AUTHZ-WG discussions

Obligations use cases

- Account mapping
- Priority/queue
- Resource/Storage path/location
- Quota assignment
- Service combination with implied conditions (e.g., computing and storage resources)
- Usable resources/quota

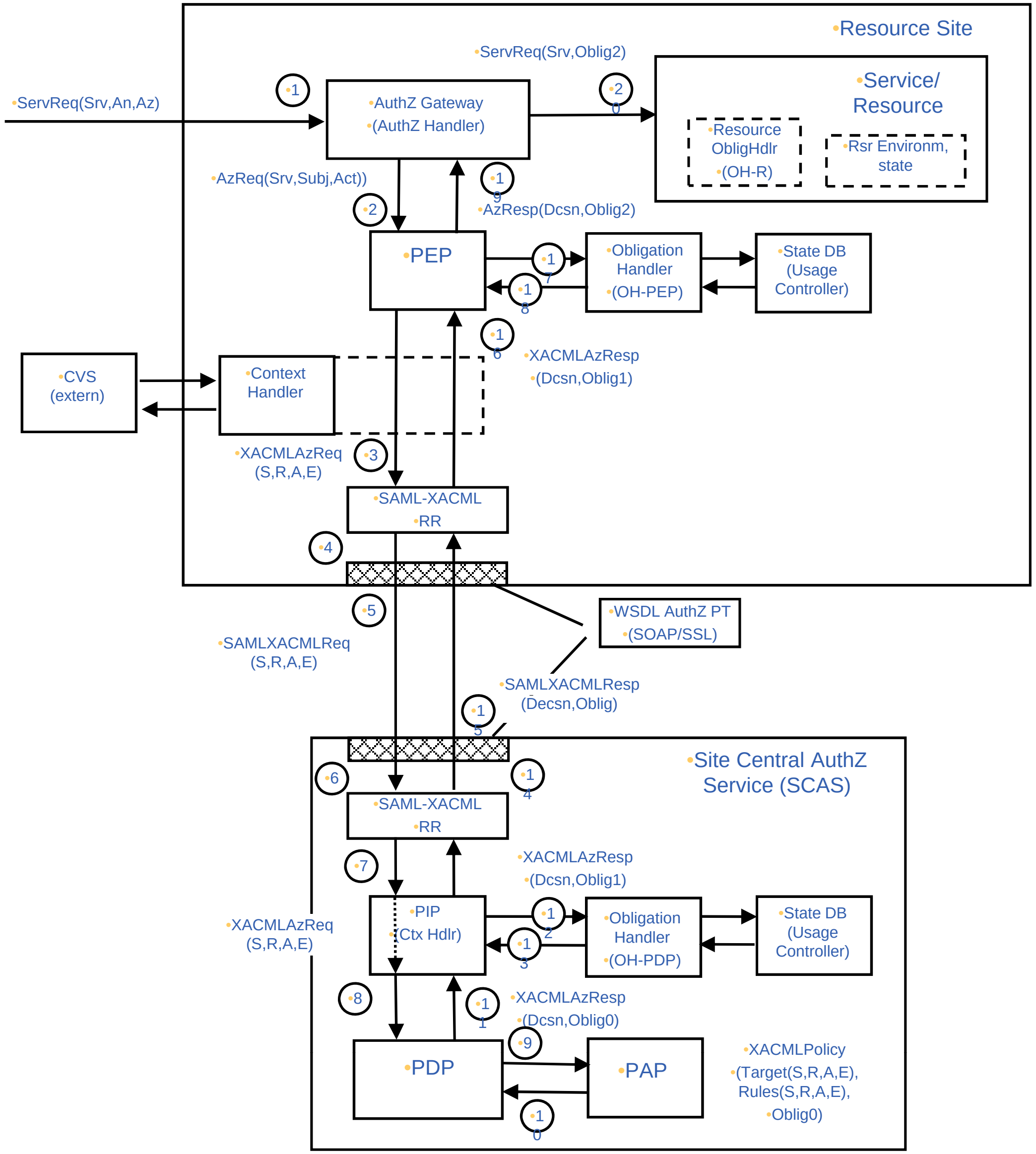
Current OSG-EGEE Obligations List

- **[T] [S] UID + GID**
- **[T] [S] Multiple secondary GIDs**
  - Requires UID+GID
- **[T/E] [R] AFS token (type string)**
  - Requires UID+GID
- **[E] [S] Username (for CE)**
- **[T/E] [R] Path restriction**
  - Requires UID+GID or Username
- **[A] [S] Storage priorities (gPlazma)**
  - Requires UID+GID or Username
- **[E] [R] File system privilege mask**

Legend:

- [T] – policy may use template Obligation
- [A] - policy may use explicit Obligation
- [S], [R], [A] – Obligation applied to AuthZ Subject, Resource, Action

Obligations Handling Reference Model



Generic AuthZ service model

PEP – Policy Enforcement Point

PDP – Policy Decision Point

PAP – Policy Authority Point

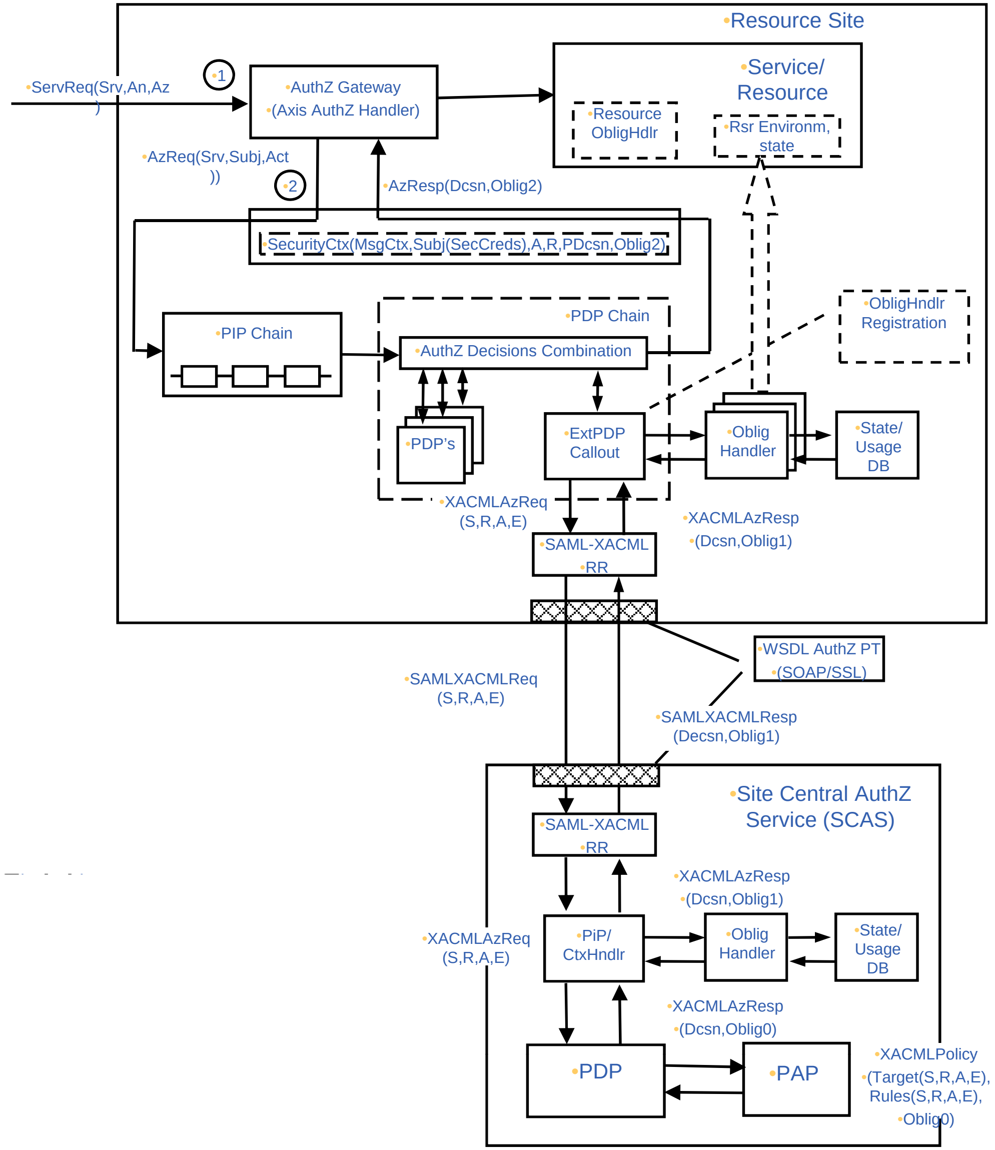
OH – Obligation Handler

CtxHandler – Context Handler

(S, R, A, E) – components of the AuthZ request (Subject, Resource, Action, Environment)

Obligations Handling in gJAF

gJAF Obligations Handling Dataflow



Obligations Handling Stages

Obligation0 = tObligation => Obligation1 (“OK?”, (Attributes1 v Environments1))
=> Obligation2 (“OK?”, (Attributes2 v Environments2))
=> Obligation3 (Attributes3 v Environments3)

- **Obligation0 – (stateless or template)**
  Obligations are returned by the PDP in a form as they are written in the policy. These obligations can be also considered as a kind of templates or instructions, tObligation.

- **Obligation1 and Obligation 2**
  Obligations have been handled by Obligation handler at the SCAS/PDP side or at the PEP side, depending on implementation. Templates or instructions of the Obligation0 are replaced with the real attributes in Obligation1, e.g. in a form of “name-value” pair.
  - The result of Obligations processing/enforcement is returned in a form of modified AuthzResponce (Obligation1) or global Resource environment changes
  - Obligation handler should return notification about fulfilled obligated actions, e.g. in a form of Boolean value “False” or “True”, which will be taken into account by PEP or other processing module to finally permit or deny service request by PEP.
  - Note. Obligation1 handling at the SCAS or PDP side allows stateful PDP/SCAS.

- **Obligation3**
  Final stage when an Obligation actually takes effect (Obligations “termination”). This is done by the Resource itself or by services managed/controlled by the Resource.

Obligation expression in XACML

- **General Obligation term**

Obligation = Apply (TargetAttribute, Operation (Variables))

Obligation = Apply (TargetAttribute, Operation (Variables), Chronicle)

Ref: Chronicle attribute was proposed by OGSA AUTHZ-WG

```
<Obligation ObligationId="urn:oasis:names:tc:xacml:2.0:scas-
    policy:example007:policy:obligation.UID“ FulfillOn="Permit">
  <AttributeAssignment DataType=http://www.w3.org/2001/XMLSchema#string
      AttributeId="urn:oasis:names:tc:xacml:1.0:example:attribute:access-subject">
        &lt;SubjectAttributeDesignator
          AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
          DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
  </AttributeAssignment>
  <AttributeAssignment
          AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:poolaccount"
          DataType="http://www.w3.org/2001/XMLSchema#string">
      &lt;PoolAccountDesignator
          AttributeId="http://glite.egee.org/JRA1/Authz/XACML/obligation/poolaccount”
          DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
         egee-pool-next-available
     </AttributeValue>
  </AttributeAssignment>
</Obligation>
```

ObligationId format and semantics

- **ObligationId format**
  - should use OASIS SAML/XACML prefix
  - agreed namespace identifier for the target project or use cases
  - may use either URN or URI form
- **Suggested namespace identifiers**
  - glite:security:authz:(policy | policy:obligation)
  - http://glite.org/security/authorisation/
- **Suggested sub-trees for management and deployment purposes**
  - orgname/projname or servicename
  - example
  - test
- **Adding suffices for versioning and staging**
  - version0.1
  - stage0
  - template

Examples ObligationId

- **Examples using SAML/XACML URN style**

  urn:oasis:names:tc:xacml:2.0:glite:security:authz:policy:obligation:obligation.UID

  urn:oasis:names:tc:xacml:2.0:glite:security:authz:example007:policy:obligation:obligation.UID

  urn:oasis:names:tc:xacml:2.0:glite:security:authz:EGEE:policy:obligation:obligation.UID

- **Examples using general URI style**

  http://glite.org/security/authorisation/policy/obligation/obligation.UID

  http://glite.org/security/authorisation/CNAF/policy/obligation/obligation.UID

  http://glite.org/security/authorisation/CREAM/policy/obligation/obligation.UID/a=3&@#$&z=y*x

  - Note: Consider URI security issues

- **Examples adding versioning/staging suffix**

  urn:oasis:names:tc:xacml:2.0:glite:security:authz:policy:obligation:obligation.UID:version0.1

Globus, G-PBox, gJAF implementations and interoperability tests

- **Globus SAML-XACML Library**
  - C and Java based SAML-XACML library
  - Axis2 generated + supported classes
  - No native XACML PDP
- **G-PBox**
  - SAML-XACML library generated from schema
  - Native XACML PDP and XACML policies
- **gJAF**
  - OpenSAML2.0 extensions for SAML-XACML profile
  - SunXACML based native XACML PDP

- **Tests done so far**
  - Globus alpha test setup – OK, however problems to integrate XACML PDP
  - G-PBox library (with gJAF) - OK
  - Calling Globus with G-PBox libraries - Fail

Issues to discuss

- **Reference model for Obligations handling (OHRM)**
  - AuthZ ticket/assertion for the Obligated AuthZ decision integrity
- **Obligation expression format**
- **ObligationId and namespace(s)**
- **ObligationHandler API**
- **Interoperability and conformance test suite**