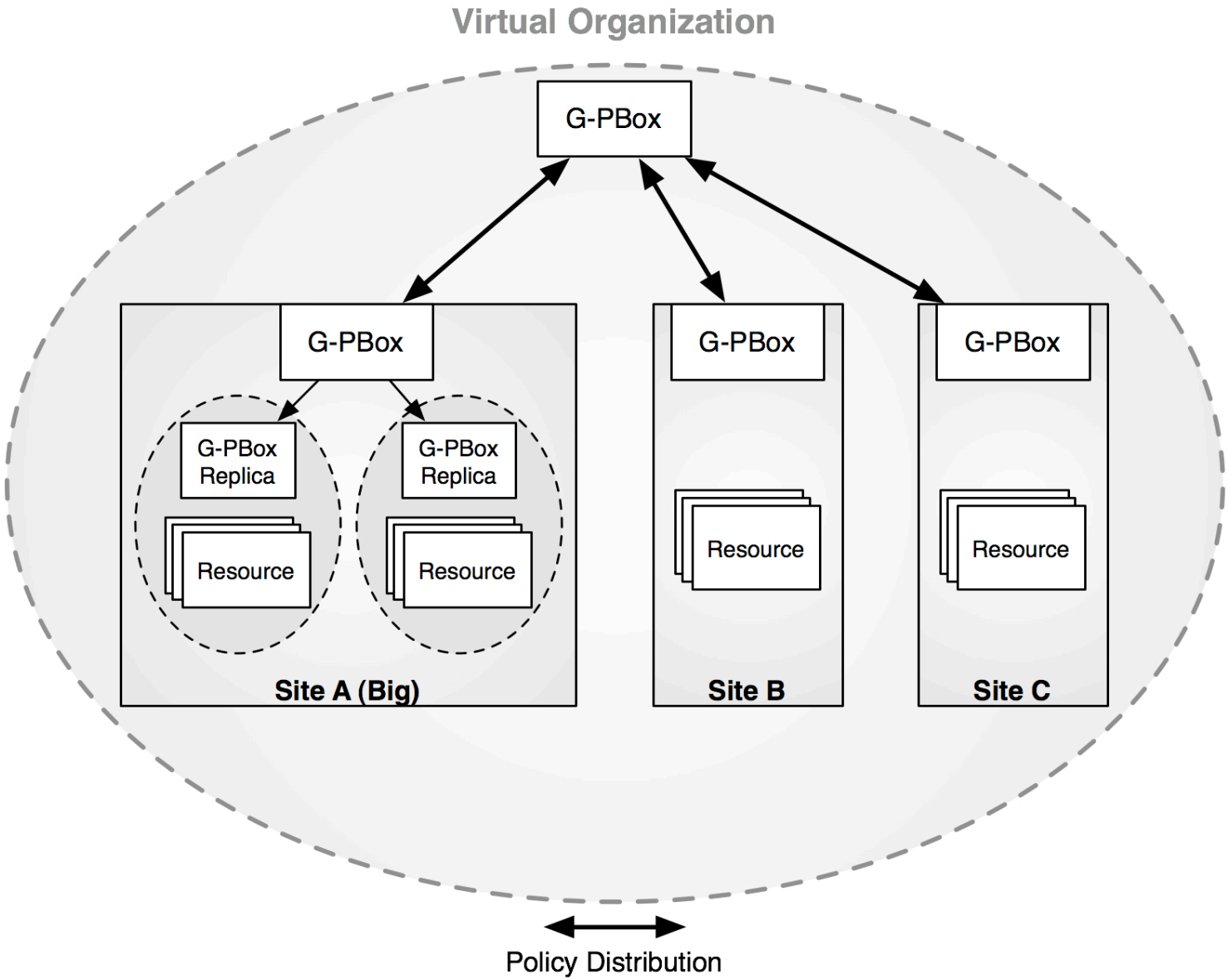
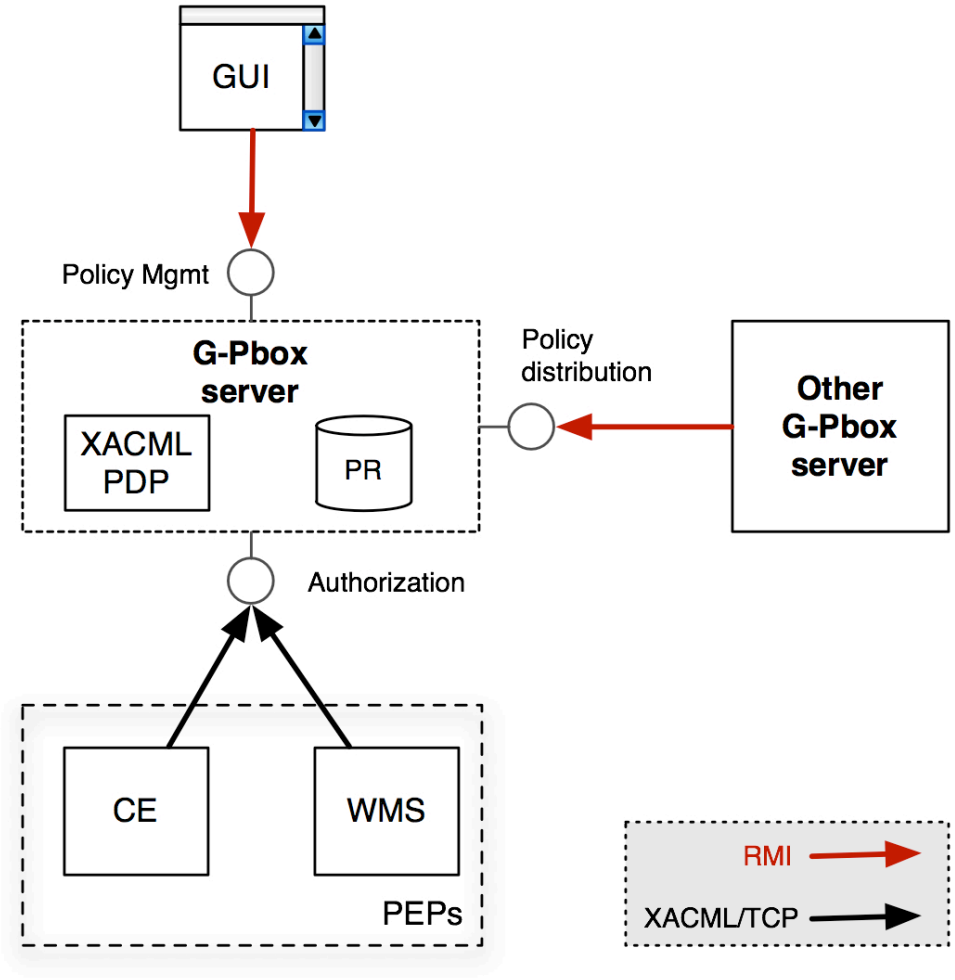
G-PBox: current status and future plans

Speaker *Andrea Ceccanti*
Location *CERN*
Date *25/10/2007*

G-PBox: a distributed AuthZ service



Current G-PBox status and arch

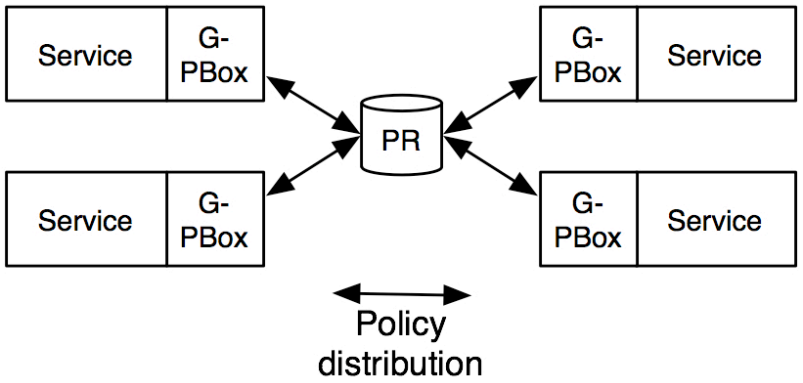


- **Policy Administration Point (PAP)**
  - Policy Repository (Exist XML DB)
  - Administrative interface (Java swing GUI)
  - Policy distribution
- **Policy Decision Point (PDP)**
  - Customized Sun XACML engine
    - XACML v. 1.1 supported
- **Policy Enforcement Point (PEP)**
  - LCAS/LCMAPS plugin
  - WMS
  - Java, C/C++ APIs

The PDP engine

- **The "core" is the Sun's XACML implementation**
  - Only open source XACML implementation available when G-PBox was "conceived"

- **Enhancements:**
  - support for policies persistent storage (i.e., interaction with the Policy Repository)
  - communication layer (sun's pdp is a library)

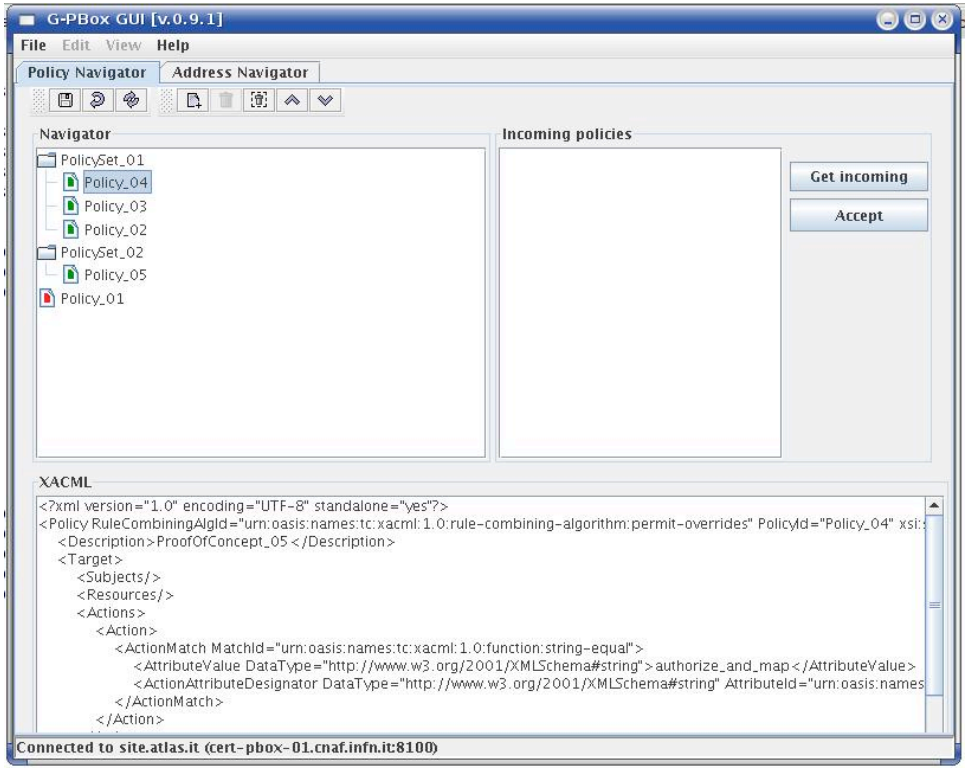


- **Architectural separation of the PDP and the PR makes easier to implement a PDP that is local to the services**
  - No network overhead for each AuthZ request

- **Policy distribution mechanism ensure consistent AuthZ across the G-PBox distributed instances**

The GUI

- **Used by VO/Site Administrators to manage policies**



- **Features:**
  - "Off-line" policy management
  - Policy/Policyset editor
    - to ease creation of XACML policies
  - Policy distribution management
- **Interacts:**
  - with the G-PBox server via RMI
  - with VOMS-Admin via WS
- **Proof of concept**
  - No formal requirements yet from Site/VO Admins
  - RMI interface since interoperability is not an issue in this phase

Performance Test results

- **G-PBox performance has been tested in the EGEE preview testbed**
- **Results presented at the Helsinki's AH Meeting**
  - No measurable overhead on the CE & WMS side
  - Small performance improvements in
    - AuthZ & Mapping on the CE
    - Resource selection on the WMS

Detailed test results

	AuthZ & Mapping mean execution time
Without G-PBox	**6.453 ± 0.007 sec**
With G-PBox	**6.522 ± 0.003 sec**

Site G-PBox

- Loaded with 3500 “fake” policies
- 1000 Globus-job-run

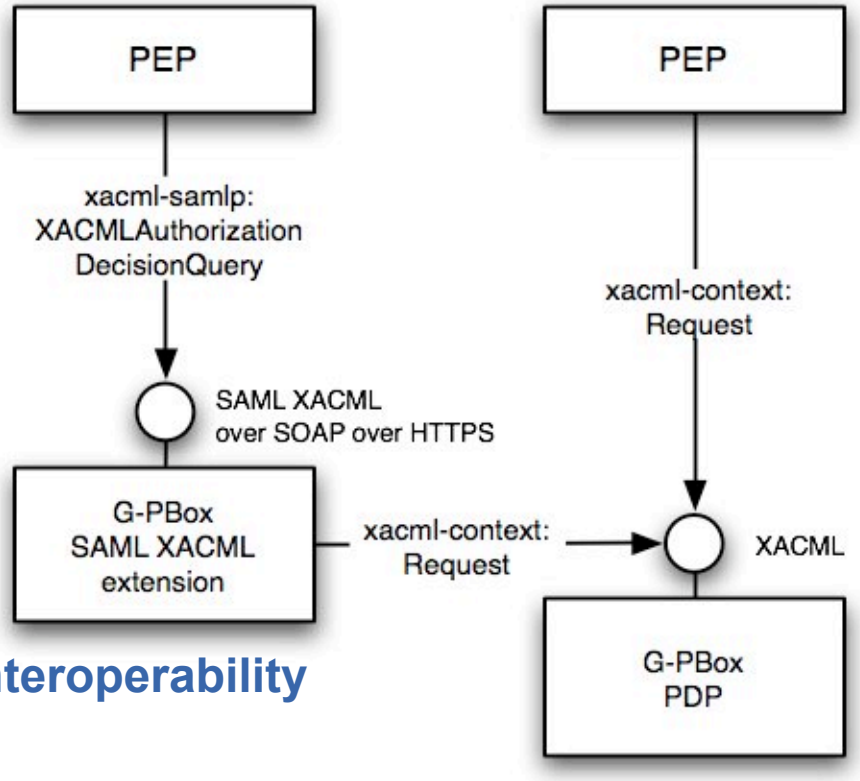
	Average time per list-match
No G-PBox	**7 ± 1 sec**
G-PBox queried with a proxy matching only a subset of the queues	**5 ± 0.6 sec**
G-PBox queried with a proxy matching all the queues	**7 ± 1 sec**

VO G-PBox:

- 1000 consecutive list-matches

Current interoperability efforts

- **Proof of concept implementation**
  - "Wrapped" G-PBox XACML PDP
  - Tomcat webapp
  - XACML over TCP interface still available



- **EGEE OSG GT joint effort on AuthZ Interoperability**
  - Service interface definition
  - Definition of a set of common obligations

- **We are helping evaluating the GT's solution**
  - Not standard compliant (the first three alpha versions contained patched schema files, now they're fixing this)
  - Fragile. When tested against other client/services it crashes

Short-term plans

- **Full XACML v. 2.0 compliance**
  - extension of current engine?
    - Sun's implementation not actively developed
  - evaluation of other implementations?
    - SICS' SPOT assertion server (http://www.sics.se/spot/assertion_server.html)
      - *BSD license*
      - *XACML v. 2.0 compliant*
      - *also based on Sun's implementation (negligible transition effort)*
- **GUI evolution and improvements**
  - according to feedback/requirements coming from users
- **Better WMS integration**

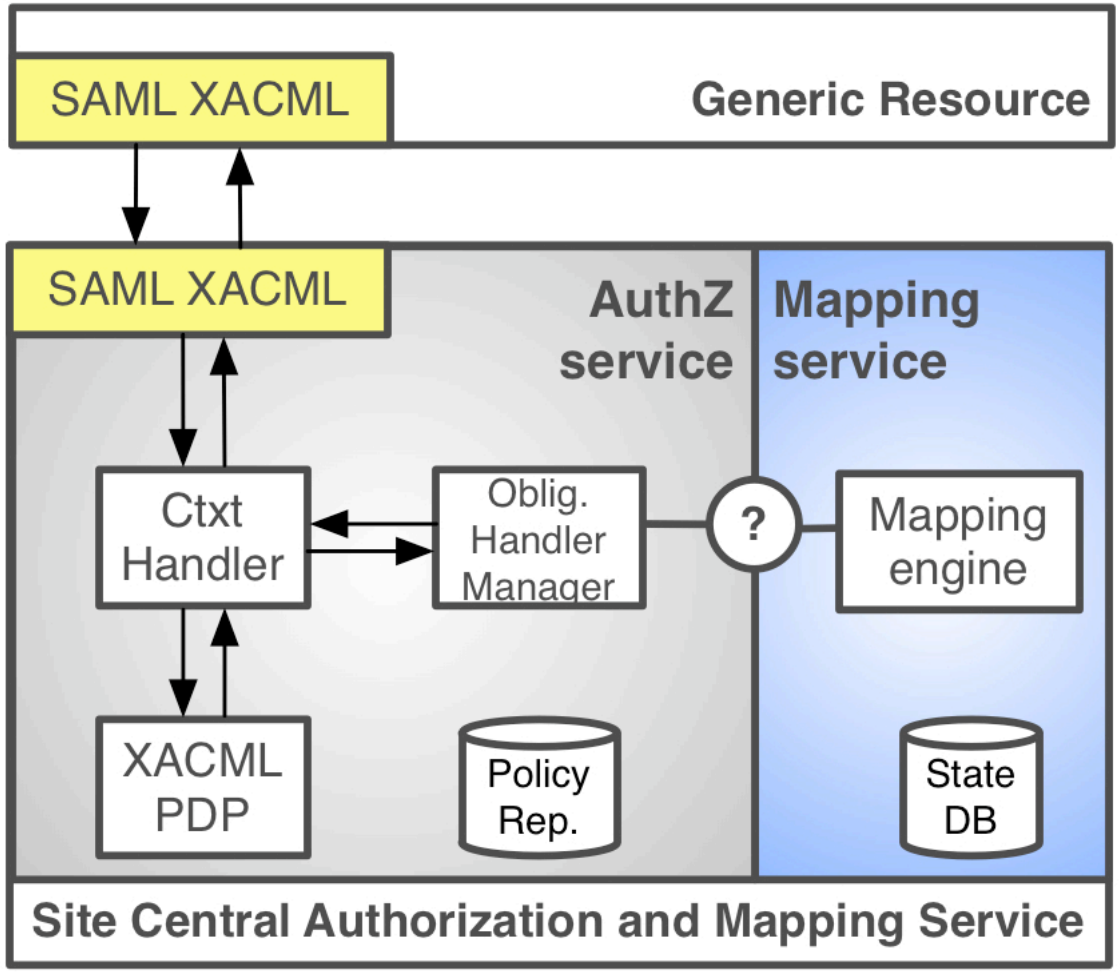
AuthZ & Mapping in the long term

- **The need for a site central mapping function has been the main motivation behind the site central authorization service**

- **AuthZ vs Mapping**
  - Conceptually different things
  - AuthZ: allowing/denying access to a resource
  - Mapping: an obligation that follows and depends on an AuthZ decision

- **A uniform XACML AuthZ service**
  - sits behind the SAML-XACML AuthZ interface
  - leverages an XACML compliant PDP engine
  - implements mapping leveraging XACML obligations
    - possibly using a callout to an “external” Mapping Service

Our long-term vision



Some components are already there

- you name them :)

We need agreement on the Obligation Handler Manager --> Mapping engine interface