# XACML Obligations expression and handling in distributed Grid applications

Draft version 0.3, August 9, 2007

## 1 Obligations in XACML policy and XACML Response message

From XACML 2.0 specification section “2.12. Actions performed in conjunction with enforcement” (lines 557-566):

In many applications, policies specify actions that MUST be performed, either instead of, or in addition to, actions that MAY be performed. This idea was described by Sloman [Sloman94]. XACML provides facilities to specify actions that MUST be performed in conjunction with policy evaluation in the `<Obligations>` element. This idea was described as a provisional action by Kudo [Kudo00]. There are no standard definitions for these actions in version 2.0 of XACML. Therefore, bilateral agreement between a *PAP* and the *PEP* that will enforce its *policies* is required for correct interpretation. *PEPs* that conform with v2.0 of XACML are required to deny *access* unless they understand and can discharge all of the `<Obligations>` elements associated with the *applicable policy*. `<Obligations>` elements are returned to the *PEP* for enforcement.

### 1.1 Obligations expression examples

a) Example <Obligations> from XACML2.0 specification:

```
<Obligations>
<Obligation ObligationId="urn:oasis:names:tc:xacml:example:obligation:email"
FulfillOn="Permit">
<AttributeAssignment
   AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:mailto"
   DataType="http://www.w3.org/2001/XMLSchema#string">
     &lt;AttributeSelector
     RequestContextPath="//md:/record/md:patient/md:patientContact/md:email"
     DataType="http://www.w3.org/2001/XMLSchema#string"/&gt ;
</AttributeAssignment>
<AttributeAssignment
   AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:text"
   DataType="http://www.w3.org/2001/XMLSchema#string">
     Your medical record has been accessed by:
</AttributeAssignment>
<AttributeAssignment
   AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:text"
   DataType="http://www.w3.org/2001/XMLSchema#string">
     &lt;SubjectAttributeDesignator
     AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
     DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
</AttributeAssignment>
</Obligation>
</Obligations>
</Policy>
```

b) Example <Obligations> from OGSA-AuthZ specification “Use of XACML Request Context to access a PDP” (edited by David Chadwick)

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<Response xmlns="urn:oasis:names:tc:xacml:2.0:context:schema:os"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:context:schema:os
http://docs.oasis-open.org/xacml/xacml-core-2.0-context-schema-os.xsd">
<Result ResourceId="12345">
  <Decision>Permit</Decision>
    <Status>
      <StatusCode Value="urn:oasis:names:tc:xacml:1.0:status:ok"/>
    </Status>
  <Obligations>
    <Obligation
       ObligationId="http://sec.cs.kent.ac.uk/GGF/XACML/obligation/example"
       FulfillOn="Permit">
    <AttributeAssignment
        AttributeId="http://sec.cs.kent.ac.uk/GGF/XACML/environment/balance"
        DataType="http://www.w3.org/2001/XMLSchema#integer">
&lt;Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:integer-add"&gt;
&lt;Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:integer-one-and-
only"&gt;
&lt;ActionAttributeDesignator
AttributeId="http://sec.cs.kent.ac.uk/GGF/XACML/MRAM.get.size"
DataType="http://www.w3.org/2001/XMLSchema#integer"/&gt;
&lt;/Apply&gt;
&lt;Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:integer-one-and-
only"&gt;
&lt;EnvironmentAttributeDesignator
AttributeId="http://sec.cs.kent.ac.uk/GGF/XACML/environment/balance"
DataType="http://www.w3.org/2001/XMLSchema#integer"/&gt;
&lt;/Apply&gt;
&lt;/Apply&gt;
    </AttributeAssignment>
  </Obligation>
</Obligations>
</Result>
</Response>
```

Note. Presented above way of expressing Obligations may have serious security concerns because of including executive commands.

c) Example with pool accounts mapping in Grid

Note. Below options are provide only illustration how Obligations can be expressed in the XACML2.0 compliant format. This is not a goal of this document and section to make suggestions about preferred format, however it can be suggested that although Option 3 can bring possible flexibility it is not recommended way of expressing and communicating Obligations because of security concerns.

Option 1.

```
<!-- Obligations format option 1 (UID, GID explicitly mentioned as separate XML
elements inside AttributeAssignment element) -->
<Obligations>
<Obligation
   ObligationId="http://glite.egee.org/JRA1/Authz/XACML/obligation/map.poolaccount"
   FulfillOn="Permit">

<!-- This is a common part that specify to what kind of attribute the next ‘map.to’
action is applied to -->
<AttributeAssignment
   AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute: requesting-subject"
   DataType="http://www.w3.org/2001/XMLSchema#string">
     &lt;SubjectAttributeDesignator
     AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
     DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
```

```
</AttributeAssignment>

<!-- This is actual account attribute/name to which it should be mapped -->
<AttributeAssignment
   AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:mapto"
   DataType="http://www.w3.org/2001/XMLSchema#string">
     &lt;UnixId
     DataType="http://www.w3.org/2001/XMLSchema#string"&gt;
          okoeroo&gt;UnixId&gt;
     &lt; GroupPrimary
     DataType="http://www.w3.org/2001/XMLSchema#string"&gt;
         computergroup&gt;GroupPrimary&gt;
     &lt;GroupSecondary
     DataType="http://www.w3.org/2001/XMLSchema#string"&gt;
         datagroup&gt;GroupSecondary&gt;
</AttributeAssignment>
</Obligation>
</Obligations>
```

Option 2.

```
<!-- Obligations format option 2 (UID, GID explicitly mentioned as one string in
the AttributeAssignment elements) -->
<Obligations>
<Obligation
   ObligationId="http://glite.egee.org/JRA1/Authz/XACML/obligation/map.poolaccount"
   FulfillOn="Permit">

<!-- This is a common part that specify to what kind of attribute the next ‘map.to’
action is applied to -->
<AttributeAssignment
   AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:requesting-subject"
   DataType="http://www.w3.org/2001/XMLSchema#string">
     &lt;SubjectAttributeDesignator
     AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
     DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
</AttributeAssignment>

<!-- This is actual account attribute/name to which it should be mapped -->
<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”
UnixId="okoeroo" GroupPrimary="computergroup" GroupSecondary="datagroup"
DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
</AttributeAssignment>
</Obligation>
</Obligations>
```

Option 3.

```
<!-- Obligations format option 3 (UID, GID are referred to by their uri in external
gridmap file) -->
<Obligations>
<Obligation
   ObligationId="http://glite.egee.org/JRA1/Authz/XACML/obligation/map.poolaccount"
   FulfillOn="Permit">

<!-- This is a common part that specify to what kind of attribute the next ‘map.to’
action is applied to -->
<AttributeAssignment
   AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute: requesting-subject"
   DataType="http://www.w3.org/2001/XMLSchema#string">
```

```
     &lt;SubjectAttributeDesignator
     AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
     DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
</AttributeAssignment>

<!-- This is actual account attribute/name to which it should be mapped -->
<AttributeAssignment
   AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:text"
   DataType="http://www.w3.org/2001/XMLSchema#string">
     &lt;AttributeSelector
     GridMapPath="//gmap:/uid/gmap:primay/gmap:secondary"
     DataType="http://www.w3.org/2001/XMLSchema#string"/&gt ;
</AttributeAssignment>
</Obligation>
</Obligations>
```

### 1.2 Example of expressing and handling Obligations with PEP communicating its Obligations handling capability

This example models an idea to communicate PEP Obligations handling capability to the PDP in the Environment element. However, to make it possible to select the applicable policy based on returned Obligations, we need to put explicit values of the ObligationId's into the policy Environment matching expression.

a) Example policy for CE with obligations to map-to/enforce UID and GUD in pool accounts.

```
<?xml version="1.0" encoding="UTF-8"?>
<Policy
      xmlns="urn:oasis:names:tc:xacml:2.0:policy:schema:os"
      xmlns:xacml-context="urn:oasis:names:tc:xacml:2.0:context:schema:os"
      xmlns:md="http://www.medico.com/schemas/record"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:policy:schema:os
        access_control-xacml-2.0-policy-schema-os.xsd"
      PolicyId="urn:oasis:names:tc:xacml:2.0:scas-policy:example007:policy"
      RuleCombiningAlgId="urn:oasis:names:tc:xacml:1.0:rule-combining-
algorithm:deny-overrides">
  <Description>
  Example007 - Test for policy selection by supported obligations in Environment
element.
  </Description>
  <PolicyDefaults>
    <XPathVersion>http://www.w3.org/TR/1999/Rec-xpath-19991116</XPathVersion>
  </PolicyDefaults>
  <Target>
    <Subjects>
      <Subject>
        <SubjectMatch
           MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
        <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">
        VO-EGEE</AttributeValue>
        <SubjectAttributeDesignator
          SubjectCategory="urn:oasis:names:tc:xacml:1.0:subject:-category:access-
subject"
          AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-vo"
          DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </SubjectMatch>
      </Subject>
    </Subjects>
    <Actions>
      <AnyAction/>
    </Actions>
    <Resources>
```

```
      <Resource>
        <ResourceMatch
          MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
        <AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#string">http://nikhef.nl/VO-
EGEE/CE01
        </AttributeValue>
          <ResourceAttributeDesignator
            AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
            DataType="http://www.w3.org/2001/XMLSchema#URI"/>
M       </ResourceMatch>
      </Resource>
    </Resources>
    <Environments>
      <Environment>
        <EnvironmentMatch
            MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
          <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#string">
             obligation.UID
          </AttributeValue>
         <AttributeSelector
              RequestContextPath="./xacml-context:Environment/xacml-
context:Attributet/xacml-
context:AttributetValue/scas:PEPconfig/scas:SuportedObligations/scas:
ObligationId/@ObligationId"
              MustBePresent="true"
              DataType="http://www.w3.org/2001/XMLSchema#string"/>
         </EnvironmentMatch>
       </Environment>
       <Environment>
         <EnvironmentMatch
              MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
            <AttributeValue
                 DataType="http://www.w3.org/2001/XMLSchema#string">
              obligation.GID
            </AttributeValue>
            <AttributeSelector
                 RequestContextPath="./xacml-context:Environment/xacml-
context:Attributet/xacml-
context:AttributetValue/scas:PEPconfig/scas:SuportedObligations/scas:
ObligationId/@ObligationId"
                 MustBePresent="true"
                 DataType="http://www.w3.org/2001/XMLSchema#string"/>
         </EnvironmentMatch>
       </Environment>
     </Environments>
    </Target>
    <Rule
          RuleId="urn:oasis:names:tc:xacml:2.0:scas-policy:example007:rule"
          Effect="Permit">
      <Description>
        User with role "researcher"  from VO "EGEE" can access Resource "CE".
      </Description>
      <Target>
        <Subjects>
           <AnySubject/>
        </Subjects>
        <Resources>
          <AnyResource/>
        </Resources>
        <Actions>
          <Action>
            <ActionMatch
                MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
               <AttributeValue

DataType="http://www.w3.org/2001/XMLSchema#string">SubmitJob</AttributeValue>
               <ActionAttributeDesignator
AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
                   DataType="http://www.w3.org/2001/XMLSchema#string"/>
            </ActionMatch>
```

```
          </Action>
        </Actions>
      </Target>
      <Condition FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-at-least-
one-member-of">
        <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-bag">
           <AttributeValue
DataType="http://www.w3.org/2001/XMLSchema#string">researcher</AttributeValue>
        </Apply>
        <SubjectAttributeDesignator
             AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-role"
             DataType="http://www.w3.org/2001/XMLSchema#string"
             Issuer="EGEEAttributeIssuer"/>
      </Condition>
    </Rule>
    <Obligations>
      <Obligation
          ObligationId="urn:oasis:names:tc:xacml:2.0:scas-
policy:example007:policy:obligation.UID"
          FulfillOn="Permit">
        <AttributeAssignment
          AttributeId="urn:oasis:names:tc:xacml:1.0:example:attribute:access-
subject"
          DataType="http://www.w3.org/2001/XMLSchema#string">
             &lt;SubjectAttributeDesignator
               AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
               DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
          </AttributeAssignment>
<!-- This is actual account attribute/name to which it should be mapped -->
          <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>
   <Obligation
       ObligationId="urn:oasis:names:tc:xacml:2.0:scas-
policy:example007:policy:obligation.GID"
       FulfillOn="Permit">
     <AttributeAssignment
          AttributeId="urn:oasis:names:tc:xacml:1.0:scas-policy:subject:subject-
group"
         DataType="http://www.w3.org/2001/XMLSchema#string">
      GID-researchers
     </AttributeAssignment>
   </Obligation>
  </Obligations>
</Policy>
```

b) Example Request message with PEP Obligations capability in the Environment element

```
<?xml version="1.0" encoding="UTF-8"?>
<Request
      xmlns="urn:oasis:names:tc:xacml:2.0:context:schema:os"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:context:schema:os
        access_control-xacml-2.0-context-schema-os.xsd">
  <Subject>
    <Attribute
          AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
          DataType="http://www.w3.org/2001/XMLSchema#string">
       <AttributeValue>Wim Huizinga</AttributeValue>
    </Attribute>
```

```
    <Attribute
          AttributeId="urn:oasis:names:tc:xacml:2.0:subject:subject-vo"
          DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>VO-EGEE</AttributeValue>
    </Attribute>
    <Attribute
          AttributeId="urn:oasis:names:tc:xacml:2.0:subject:subject-role"
          DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>researcher</AttributeValue>
    </Attribute>
  </Subject>
  <Resource>
    <Attribute
         AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
         DataType="http://www.w3.org/2001/XMLSchema#anyURI">
      <AttributeValue>http://nikhef.nl/VO-EGEE/CE01</AttributeValue>
    </Attribute>
  </Resource>
  <Action>
    <Attribute
         AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
         DataType="http://www.w3.org/2001/XMLSchema#string">
       <AttributeValue>SubmitJob</AttributeValue>
    </Attribute>
  </Action>
  <Environment>
    <Attribute
          AttributeId="urn:oasis:names:tc:xacml:2.0:scas-policy:pep-
config:obligations"
          DataType="http://www.w3.org/2001/XMLSchema#xml">
      <AttributeValue>
        <scas:PEPconfig>
          <scas:SuportedObligations>
            <scas:Obligation ObligationId="obligation.UID"/>
            <scas:Obligation ObligationId="obligation.GID"/>
          <scas:SuportedObligations>
        <scas:PEPconfig>
      </AttributeValue>
    </Attribute>
  </Environment>
</Request>
```

Policy example using simple conventions that require minimum PEP functionality.

```
<?xml version="1.0" encoding="UTF-8"?>
<Policy
      xmlns="urn:oasis:names:tc:xacml:2.0:policy:schema:os"
      xmlns:xacml-context="urn:oasis:names:tc:xacml:2.0:context:schema:os"
      xmlns:md="http://www.medico.com/schemas/record"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:policy:schema:os
        access_control-xacml-2.0-policy-schema-os.xsd"
      PolicyId="urn:oasis:names:tc:xacml:2.0:scas-policy:example007:policy"
      RuleCombiningAlgId="urn:oasis:names:tc:xacml:1.0:rule-combining-
algorithm:deny-overrides">
  <Description>
  Example007 - Test for policy selection by PEP type.
  </Description>
  <PolicyDefaults>
    <XPathVersion>http://www.w3.org/TR/1999/Rec-xpath-19991116</XPathVersion>
  </PolicyDefaults>
  <Target>
    <Subjects>
      <Subject>
        <SubjectMatch
           MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
        <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">
        VO-EGEE</AttributeValue>
        <SubjectAttributeDesignator
          SubjectCategory="urn:oasis:names:tc:xacml:1.0:subject:-category:access-
subject"
```

```
          AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-vo"
          DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </SubjectMatch>
      </Subject>
    </Subjects>
    <Actions>
      <AnyAction/>
    </Actions>
    <Resources>
      <Resource>
        <ResourceMatch
          MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
        <AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#string">http://nikhef.nl/VO-
EGEE/CE01
        </AttributeValue>
          <ResourceAttributeDesignator
            AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
            DataType="http://www.w3.org/2001/XMLSchema#URI"/>
       </ResourceMatch>
       <ResourceMatch
          MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
          <AttributeValue
              DataType= "http://www.w3.org/2001/XMLSchema#string">GT4-CE
          </AttributeValue>
          <ResourceAttributeDesignator
              AttributeId="urn:some:path:peptype"
              DataType="http://www.w3.org/2001/XMLSchema#String"/>
        </ResourceMatch>
      </Resource>
    </Resources>
    </Target>
    <Rule
          RuleId="urn:oasis:names:tc:xacml:2.0:scas-policy:example007:rule"
          Effect="Permit">
      <Description>
        User with role "researcher"  from VO "EGEE" can access Resource "CE".
      </Description>
      <Target>
        <Subjects>
           <AnySubject/>
        </Subjects>
        <Resources>
          <AnyResource/>
        </Resources>
        <Actions>
          <Action>
            <ActionMatch
                MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
              <AttributeValue

DataType="http://www.w3.org/2001/XMLSchema#string">SubmitJob</AttributeValue>
              <ActionAttributeDesignator
AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
                  DataType="http://www.w3.org/2001/XMLSchema#string"/>
            </ActionMatch>
          </Action>
        </Actions>
      </Target>
      <Condition FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-at-least-
one-member-of">
        <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-bag">
           <AttributeValue
DataType="http://www.w3.org/2001/XMLSchema#string">researcher</AttributeValue>
        </Apply>
        <SubjectAttributeDesignator
            AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-role"
            DataType="http://www.w3.org/2001/XMLSchema#string"
            Issuer="EGEEAttributeIssuer"/>
      </Condition>
    </Rule>
    <Obligations>
```

```
      <Obligation
          ObligationId="urn:oasis:names:tc:xacml:2.0:scas-
policy:example007:policy:obligation.UID"
          FulfillOn="Permit">
        <AttributeAssignment
          AttributeId="urn:oasis:names:tc:xacml:1.0:example:attribute:access-
subject"
          DataType="http://www.w3.org/2001/XMLSchema#string">
             &lt;SubjectAttributeDesignator
               AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
               DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
          </AttributeAssignment>
<!-- This is actual account attribute/name to which it should be mapped -->
          <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>
   <Obligation
       ObligationId="urn:oasis:names:tc:xacml:2.0:scas-
policy:example007:policy:obligation.GID"
       FulfillOn="Permit">
     <AttributeAssignment
          AttributeId="urn:oasis:names:tc:xacml:1.0:scas-policy:subject:subject-
group"
         DataType="http://www.w3.org/2001/XMLSchema#string">
      GID-researchers
     </AttributeAssignment>
   </Obligation>
  </Obligations>
</Policy>
```

With the following request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request
      xmlns="urn:oasis:names:tc:xacml:2.0:context:schema:os"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:context:schema:os
        access_control-xacml-2.0-context-schema-os.xsd">
  <Subject>
    <Attribute
          AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
          DataType="http://www.w3.org/2001/XMLSchema#string">
       <AttributeValue>Wim Huizinga</AttributeValue>
    </Attribute>
    <Attribute
          AttributeId="urn:oasis:names:tc:xacml:2.0:subject:subject-vo"
          DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>VO-EGEE</AttributeValue>
    </Attribute>
    <Attribute
          AttributeId="urn:oasis:names:tc:xacml:2.0:subject:subject-role"
          DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>researcher</AttributeValue>
    </Attribute>
  </Subject>
  <Resource>
    <Attribute
         AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
         DataType="http://www.w3.org/2001/XMLSchema#anyURI">
      <AttributeValue>http://nikhef.nl/VO-EGEE/CE01</AttributeValue>
```

```
    </Attribute>
    <Attribute
         AttributeId="urn:some:path:peptype"
         DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>CE-GT4</AttributeValue>
    </Attribute>
  </Resource>
  <Action>
    <Attribute
         AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
         DataType="http://www.w3.org/2001/XMLSchema#string">
       <AttributeValue>SubmitJob</AttributeValue>
    </Attribute>
  </Action>
</Request>
```

c) Example Response message with returned Obligations containing obligations to assign/map to UID and GID

```
<?xml version="1.0" encoding="UTF-8"?>
<Response xmlns="urn:oasis:names:tc:xacml:2.0:context:schema:os"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:context:schema:os access_control-
xacml-2.0-context-schema-os.xsd">
  <Result ResourceId=" http://nikhef.nl/VO-EGEE/CE01">
    <Status>
      <StatusCode Value="OK"/>
      <StatusDetail>DecisionID</StatusDetail>
      <StatusMessage>Request is successful</StatusMessage>
    </Status>
    <Decision>Permit</Decision>
    <Obligations>
      <Obligation
          ObligationId="urn:oasis:names:tc:xacml:2.0:scas-
policy:example007:policy:obligation.UID"
          FulfillOn="Permit">
        <AttributeAssignment
          AttributeId="urn:oasis:names:tc:xacml:1.0:example:attribute:access-
subject"
          DataType="http://www.w3.org/2001/XMLSchema#string">
             &lt;SubjectAttributeDesignator
               AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
               DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
          </AttributeAssignment>
<!-- This is actual account attribute/name to which it should be mapped -->
          <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>
   <Obligation
       ObligationId="urn:oasis:names:tc:xacml:2.0:scas-
policy:example007:policy:obligation.GID"
       FulfillOn="Permit">
     <AttributeAssignment
          AttributeId="urn:oasis:names:tc:xacml:1.0:scas-policy:subject:subject-
group"
         DataType="http://www.w3.org/2001/XMLSchema#string">
```

```
      GID-researchers
     </AttributeAssignment>
   </Obligation>
  </Obligations>
</Response>
```

d) Response message after transforming by stateful pool accounts manager or UID Obligation handler – UID attribute value changed from declarative “egee-pool-next-available” to a specific pool account “egee-pool01”.

```
<?xml version="1.0" encoding="UTF-8"?>
<Response xmlns="urn:oasis:names:tc:xacml:2.0:context:schema:os"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:context:schema:os access_control-
xacml-2.0-context-schema-os.xsd">
  <Result ResourceId=" http://nikhef.nl/VO-EGEE/CE01">
    <Status>
      <StatusCode Value="OK"/>
      <StatusDetail>DecisionID</StatusDetail>
      <StatusMessage>Request is successful</StatusMessage>
    </Status>
    <Decision>Permit</Decision>
    <Obligations>
      <Obligation
          ObligationId="urn:oasis:names:tc:xacml:2.0:scas-
policy:example007:policy:obligation.UID"
          FulfillOn="Permit">
        <AttributeAssignment
          AttributeId="urn:oasis:names:tc:xacml:1.0:example:attribute:access-
subject"
          DataType="http://www.w3.org/2001/XMLSchema#string">
             &lt;SubjectAttributeDesignator
               AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
               DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
          </AttributeAssignment>
<!-- This is actual account attribute/name to which it should be mapped -->
          <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>
   <Obligation
       ObligationId="urn:oasis:names:tc:xacml:2.0:scas-
policy:example007:policy:obligation.GID"
       FulfillOn="Permit">
     <AttributeAssignment
          AttributeId="urn:oasis:names:tc:xacml:1.0:scas-policy:subject:subject-
group"
         DataType="http://www.w3.org/2001/XMLSchema#string">
      GID-researchers
     </AttributeAssignment>
   </Obligation>
  </Obligations>
</Response>
```

## 1.3 XACML 1.1 and XACML 2.0 schema definitions

### 1.3.1 XACML 2.0 Obligations definition – Specification Overview

[XACML 5.45.] Element <Obligation>

The `<Obligation>` element SHALL contain an identifier for the *obligation* and a set of *attributes* that form arguments of the action defined by the *obligation*. The `FulfillOn` attribute SHALL indicate the *effect* for which this *obligation* must be fulfilled by the *PEP*.

```
<xs:element name="Obligation" type="xacml:ObligationType"/>
<xs:complexType name="ObligationType">
<xs:sequence>
<xs:element ref="xacml:AttributeAssignment" minOccurs=”0” maxOccurs="unbounded"/>
</xs:sequence>
<xs:attribute name="ObligationId" type="xs:anyURI" use="required"/>
<xs:attribute name="FulfillOn" type="xacml:EffectType" use="required"/>
</xs:complexType>
```

The `<Obligation>` element is of **ObligationType** complexType. See Section 7.14 for a description of how the set of ***obligations*** to be returned by the ***PDP*** is determined. The `<Obligation>` element contains the following elements and attributes:
`ObligationId` [Required]
***Obligation*** identifier. The value of the ***obligation*** identifier SHALL be interpreted by the ***PEP***.
`FulfillOn` [Required]
The ***effect*** for which this ***obligation*** must be fulfilled by the ***PEP***.
`<AttributeAssignment>` [Optional]
***Obligation*** arguments assignment. The values of the ***obligation*** arguments SHALL be interpreted by the ***PEP***.

[XACML-5.46.] Element <AttributeAssignment>

The `<AttributeAssignment>` element is used for including arguments in *obligations*. It SHALL contain an `AttributeId` and the corresponding *attribute* value, by extending the AttributeValueType type definition. The `<AttributeAssignment>` element MAY be used in any way that is consistent with the schema syntax, which is a sequence of `<xs:any>` elements. The value specified SHALL be understood by the *PEP*, but it is not further specified by XACML. See Section 7.14. Section 4.2.4.3 provides a number of examples of arguments included in *obligations*.

```
<xs:element name="AttributeAssignment" type="xacml:AttributeAssignmentType"/>
<xs:complexType name="AttributeAssignmentType" mixed="true">
<xs:complexContent>
<xs:extension base="xacml:AttributeValueType">
<xs:attribute name="AttributeId" type="xs:anyURI" use="required"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
```

The `<AttributeAssignment>` element is of **AttributeAssignmentType** complex type. The `<AttributeAssignment>` element contains the following attributes:
`AttributeId` [Required]
The ***attribute*** Identifier.

[XACML-7.14. Obligations]

A *policy* or *policy set* may contain one or more *obligations*. When such a *policy* or *policy set* is evaluated, an *obligation* SHALL be passed up to the next level of evaluation (the enclosing or referencing *policy*, *policy set* or *authorization decision*) only if the *effect* of the *policy* or *policy set* being evaluated matches the value of the `FulfillOn` attribute of the *obligation*.

As a consequence of this procedure, no *obligations* SHALL be returned to the *PEP* if the *policies* or *policy sets* from which they are drawn are not evaluated, or if their evaluated result is "Indeterminate" or "NotApplicable", or if the *decision* resulting from evaluating the *policy* or *policy set* does not match the *decision* resulting from evaluating an enclosing *policy set*.

If the *PDP's* evaluation is viewed as a tree of *policy sets* and *policies*, each of which returns "Permit" or "Deny", then the set of *obligations* returned by the *PDP* to the *PEP* will include only the *obligations* associated with those paths where the *effect* at each level of evaluation is the same as the *effect* being returned by the *PDP*. In situations where any lack of determinism is unacceptable, a deterministic combining algorithm, such as ordered-deny-overrides, should be used.

<<To be extended>>

# 2 Obligations processing model and implementation

## 2.1 Generic Authorisation Request processing flow and Obligations handling model

Figure 2.1 below illustrates generic model for processing obligations in Site-central AuthZ service (SCAS). Site-central AuthZ service means that all site located resources and services use central AuthZ service that maintains a common set of policies for this site. Described here processing model is compliant to the model used in XACML (refer to XACML2.0 standard or see Appendix) but adds Web services and AuthZ callout protocol details and specifically focuses on Obligations handling flow.

A number of assumptions are made to reflect possible options in AuthZ service infrastructure implementation and different type of Obligations both stateful and stateless that are concerned with assigning pool accounts, enforcing quotas, controlling usable resource (e.g., number of resource access, purchased video/music listening time, etc.), logging and accounting.

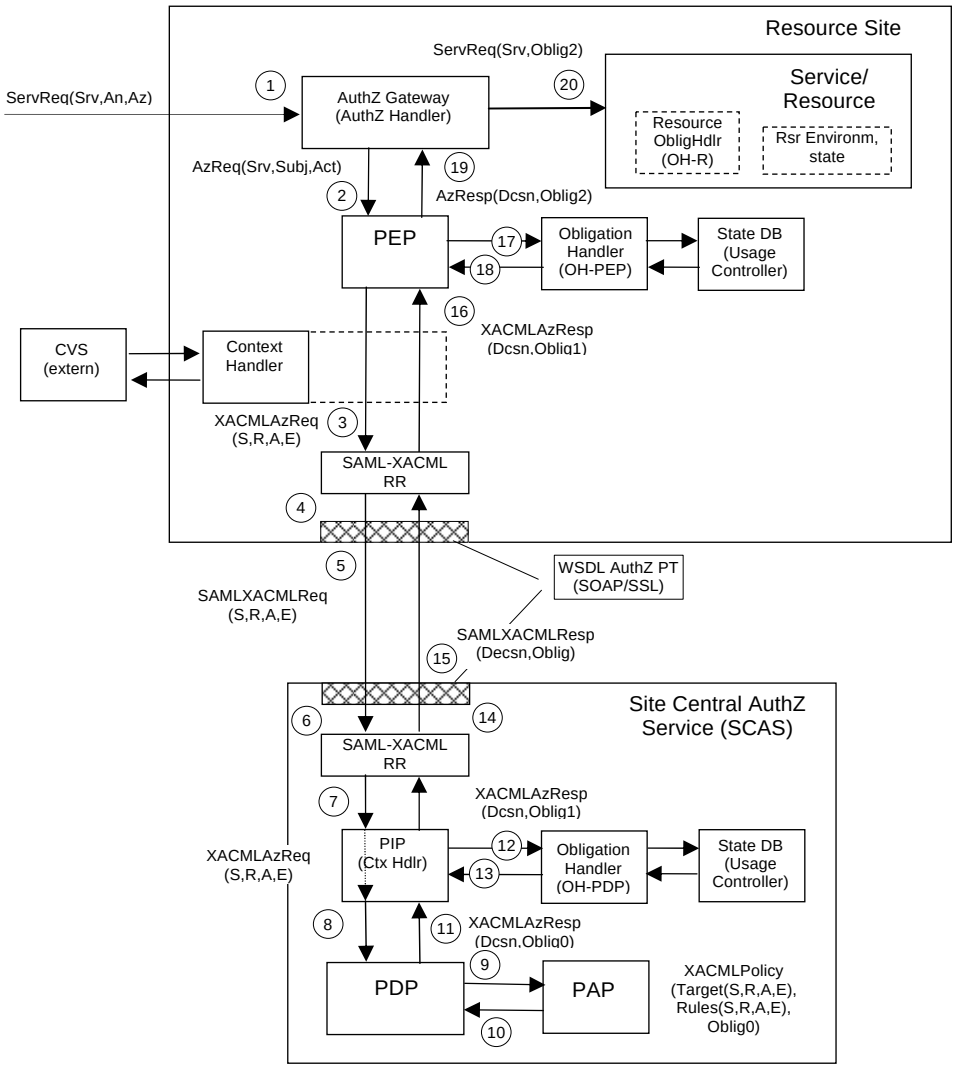


Figure 2.1. Generic Authorisation dataflow and Obligations handling in distributed AuthZ service.

It is important to notice that Obligations are the integral part of the policy and typically included into the policy at the stage of its creation by the policy administrator or resource owner. For the manageability purpose, policy is considered stateless and statefulness of Obligations is achieved by the Obligation handlers. The obligations enforcement process can be resulted either in modifying ServiceRequest (e.g., map from Subject to account name/type) or changing Resource/system sate or environment.

For the general (stateful) Obligations handling process there can be distinguished the following stages (note: not all stages are necessary to be implemented in one use case by they may exist in different cases):

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

1) Obligation0 – (stateless) 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. (Important to mention that due to security reason Obligations format and semantics should not use executable code or reference to locally executed commands).

2) Obligation1 or Obligation 2 – Obligations have been handled by Obligation handler at the SCAS/PDP side or at the PEP side, depending on implementation. In this case templates or instructions of the Obligation0 are replaced with the real attributes in Obligation1, e.g. in a form of “name-value” pair. During this stage, the Obligation handler can actually enforce Obligations or modify Obligations and send them further for enforcement by the Resource. The result of Obligations processing/enforcement, can be returned in a form of modified AuthzResponce (Obligation1) or in a form of global Resource environment changes that will be taken into account at the time when the requested service/resource are provided or delivered. In both cases (and specifically in the last case) 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. Option with Obligation1 handling at the SCAS or PDP side is introduced to illustration a case when we need to implement a stateful PDP/SCAS. This should not be considered as XACML specification violation as distinguishing between PEP and PDP functions in the generic Obligations handling model is based on what module actually makes policy based request evaluation.

3) Obligation3 – This is the final stage when Obligations actually take effect, which can be defined as Obligations “termination”. This is done by the Resource itself or by services managed/controlled by the Resource.

Note. In this general discussion we are not considering possible logical or time wise sequence of enforcing Obligations, but this is a topic of recent discussion at “xacml-dev” and “ogsa-authz” mailing lists.

## 2.2 Authorisation Request processing flow and Obligations handling model in gJAF (and GT-AuthZ)

Figure 2.2 below illustrates possible Authorisation dataflow and Obligations handling when using external AuthZ callout from gJAF or GT-AuthZ framework to Site-central AuthZ service (SCAS).

All discussions from the previous section are applicable to this model. The figure also illustrate an option of flexible Obligation handlers registration at the ExtPDP Callout module which in the proposed module can be treated as a part of virtual PEP module. It is also considered that Obligations enforcement can be resulted either in changing SecurityContext or global Resource environments.

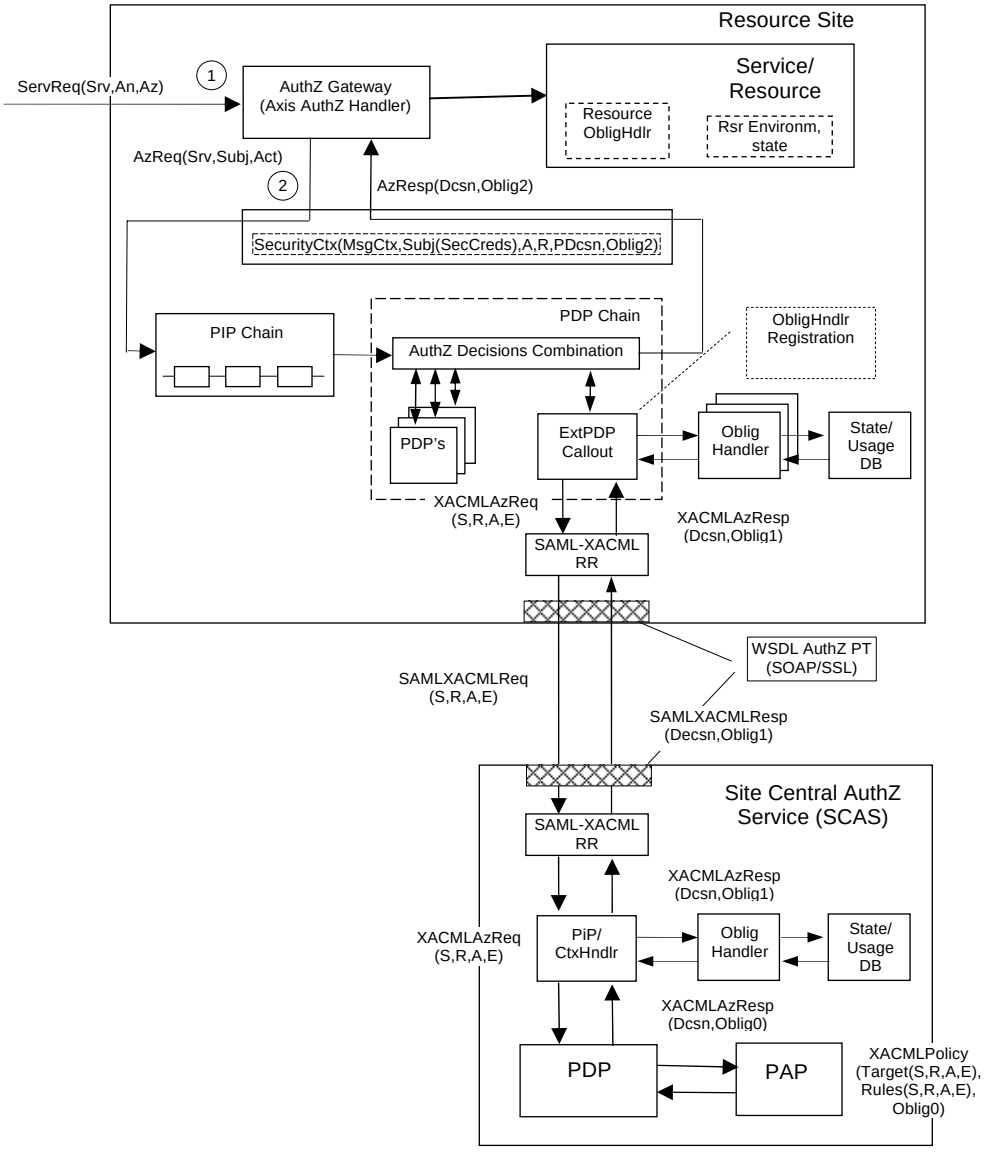


Figure 2.2. Authorisation dataflow and Obligations handling when using external AuthZ callout from gJAF or GT-AuthZ framework.

## Appendix A - Data-flow model

Source: eXtensible Access Control Markup Language (XACML) Version 2.0. OASIS Standard, 1 Feb 2005. [Online]: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf

The major actors in the XACML domain are shown in the data-flow diagram of Figure 1.

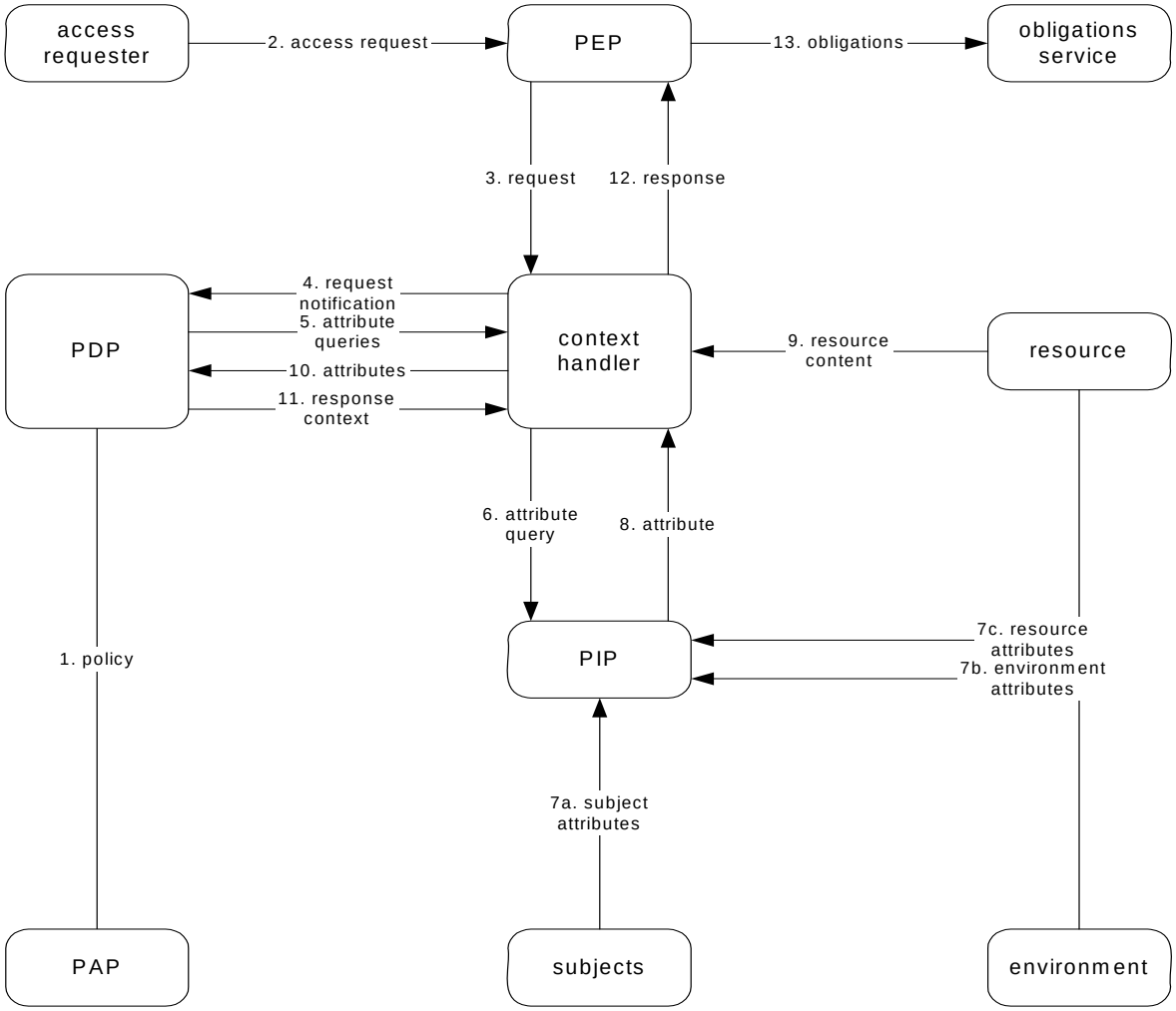


Figure 1 - Data-flow diagram

Note: some of the data-flows shown in the diagram may be facilitated by a repository. For instance, the communications between the ***context*** handler and the ***PIP*** or the communications between the ***PDP*** and the ***PAP*** may be facilitated by a repository. The XACML specification is not intended to place restrictions on the location of any such repository, or indeed to prescribe a particular communication protocol for any of the data-flows.

The model operates by the following steps.

1. ***PAP***s write ***policies*** and ***policy sets*** and make them available to the ***PDP***. These ***policies*** or ***policy sets*** represent the complete policy for a specified ***target***.
2. The access requester sends a request for access to the ***PEP***.
3. The ***PEP*** sends the request for ***access*** to the ***context handler*** in its native request format, optionally including ***attributes*** of the ***subjects***, ***resource, action*** and ***environment***.

4. The ***context handler*** constructs an XACML request ***context*** and sends it to the ***PDP***.
5. The ***PDP*** requests any additional ***subject***, ***resource, action*** and ***environment attributes*** from the ***context handler***.
6. The context handler requests the attributes from a ***PIP***.
7. The ***PIP*** obtains the requested ***attributes***.
8. The ***PIP*** returns the requested ***attributes*** to the ***context handler***.
9. Optionally, the ***context handler*** includes the ***resource*** in the ***context***.
10. The ***context handler*** sends the requested ***attributes*** and (optionally) the ***resource*** to the ***PDP***. The ***PDP*** evaluates the ***policy***.
11. The ***PDP*** returns the response ***context*** (including the ***authorization decision***) to the ***context handler***.
12. The ***context handler*** translates the response ***context*** to the native response format of the ***PEP***. The ***context handler*** returns the response to the ***PEP***.
13. The ***PEP*** fulfills the ***obligations***.
14. (Not shown) If ***access*** is permitted, then the ***PEP*** permits ***access*** to the ***resource;*** otherwise, it denies ***access***.