
# Externalisierte Autorisierung mit XACML

Die eXtensible Access Control Markup Language
Java Forum Stuttgart 2012

# über uns und mich...

Stefan Bohm

- Senior Consultant für Identity- und Accessmanagement

iC Consult

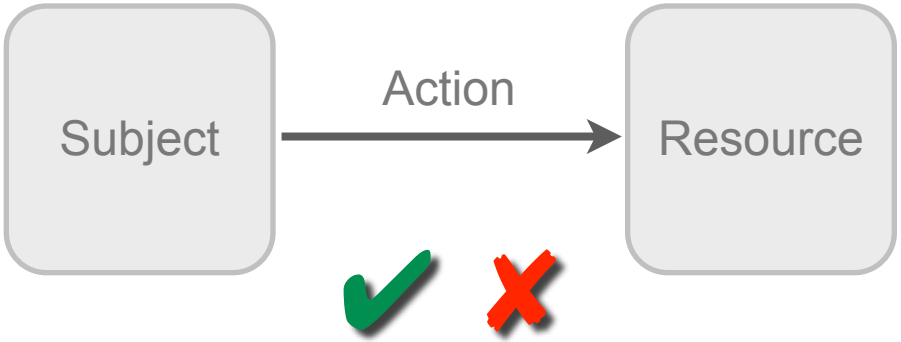
- Sitz: Oberhaching, München,
- Lokationen: Stuttgart, Frankfurt, Essen, Hamburg, Zürich
- 60 Angestellte, 10 Freelancer (Stand: 01.06.2012)

# Kommt Ihnen das bekannt vor?

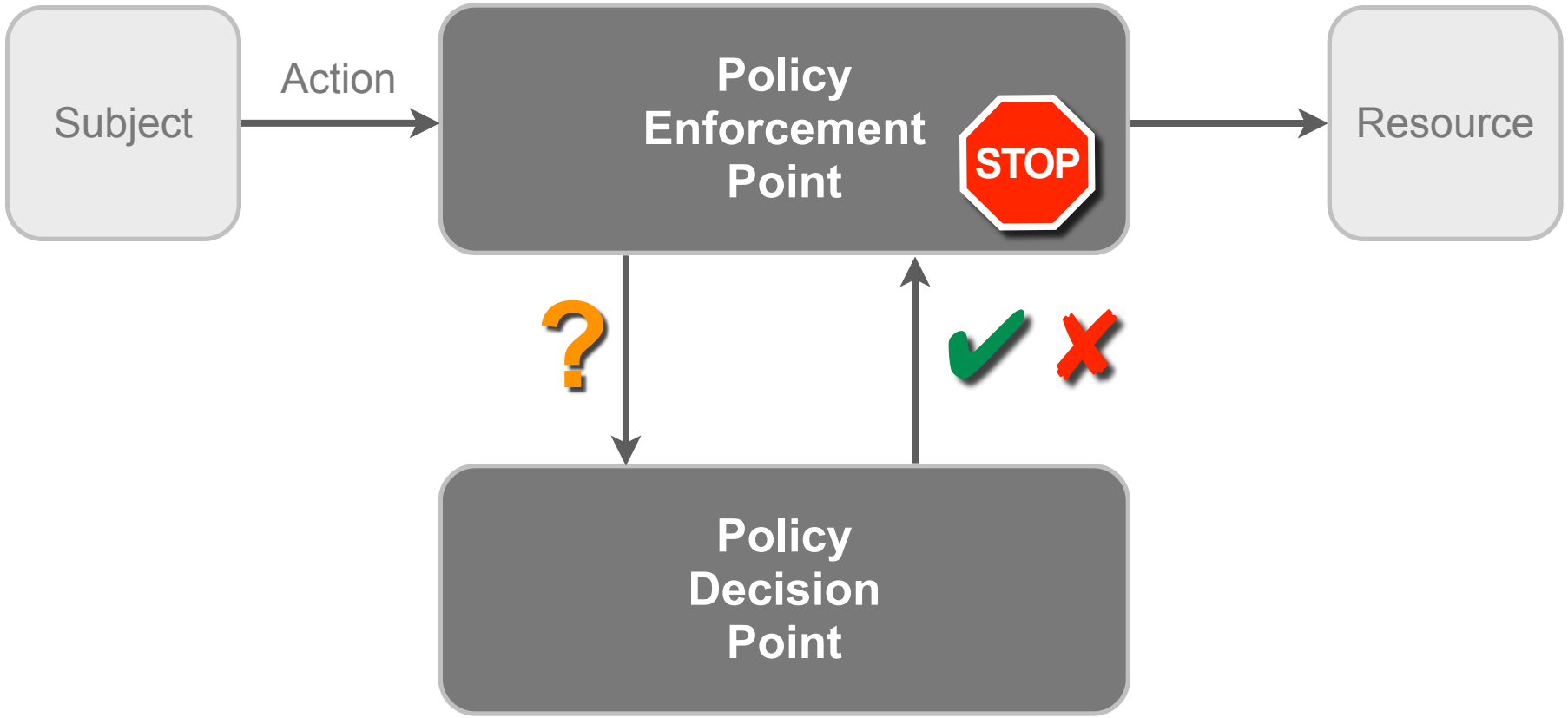
```
// Access to confidential documents requires client-cert authentication.
 if (document.getClassification() == Classification.CONFIDENTIAL) {
     if (req.getAuthType().equals(HttpServletRequest.CLIENT_CERT_AUTH)) {
         String userId = IdentityUtil.convertToUserId(req.getUserPrincipal());

         if (resource.getState() == State.DRAFT) {
             //Access control for document in state DRAFT
             String author = resource.getAuthorId();
             if (req.isUserInRole(AUTHOR) && author.equalsIgnoreCase(userId) &&
                (action == Action.READ || action == Action.WRITE)) {
                     // allow r/w access for author of resource
                     ...
                 }
             } else if (req.isUserInRole(MANAGER) &&
                     IdentityUtil.isSupervisorOf(userId, author) &&
                     action == Action.READ) {
                 // allow read access for manager of author
                 ...
             }
           ...
```

# Was ist Autorisierung?



# Was ist Autorisierung?

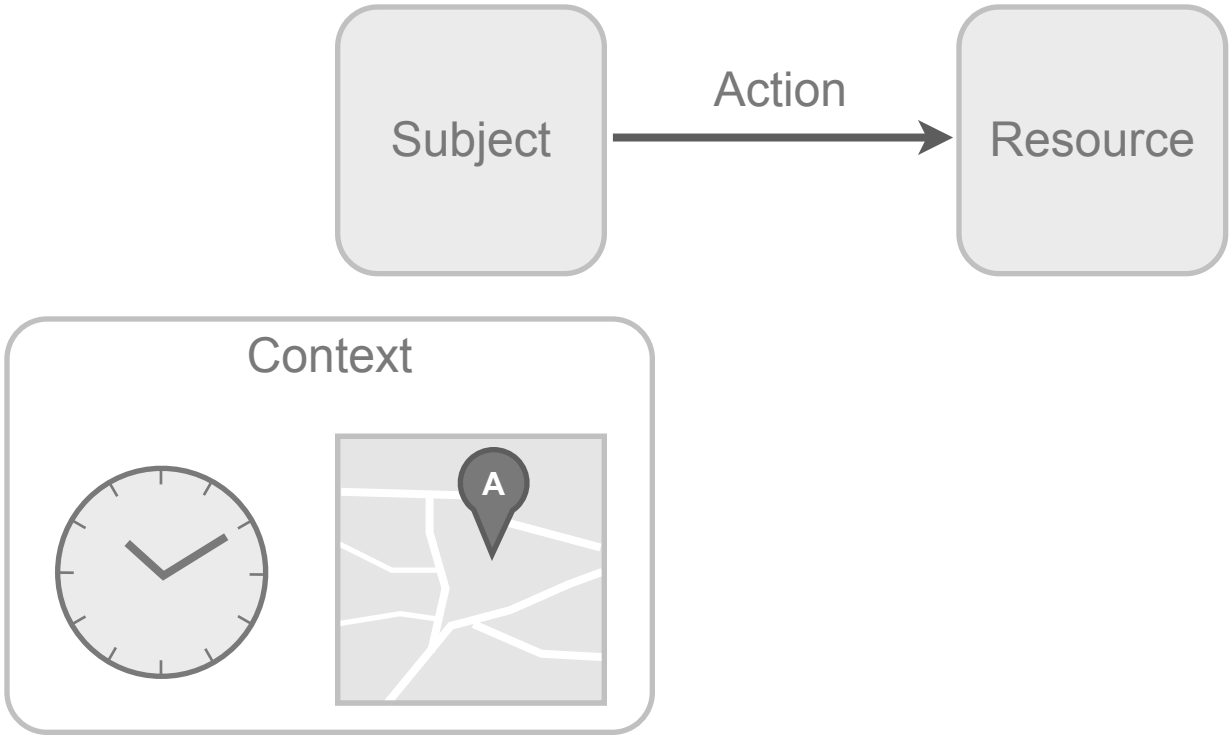


# Reality Check

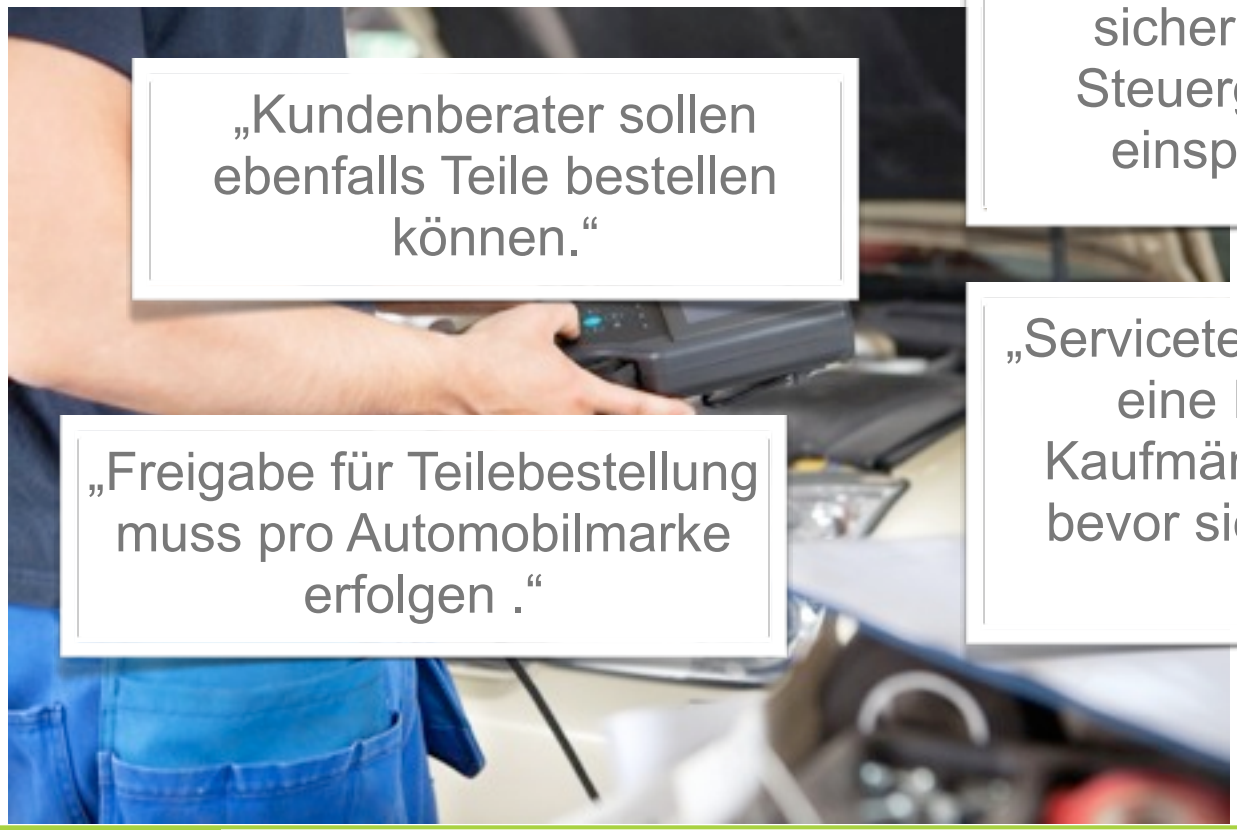
# Kontextbasierte Autorisierung

„Transaktionen dürfen nur zu Geschäftszeiten und aus dem Intranet getätigt werden.“

# Kontextbasierte Autorisierung



# Feingranulare, attributbasierte Autorisierung



„Kundenberater sollen ebenfalls Teile bestellen können.“

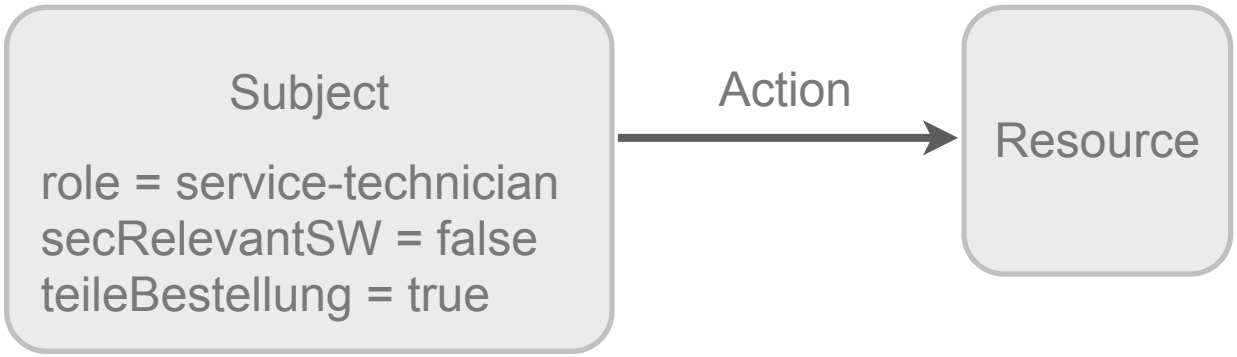
„Freigabe für Teilebestellung muss pro Automobilmarke erfolgen .“

„Servicetechniker brauchen eine Freigabe des Werkstattleiters bevor sie sicherheitsrelevante Steuergerätesoftware einspielen dürfen.“

„Servicetechniker brauchen eine Freigabe des Kaufmännischen Leiters bevor sie Teile bestellen dürfen.“

iC CONSULT

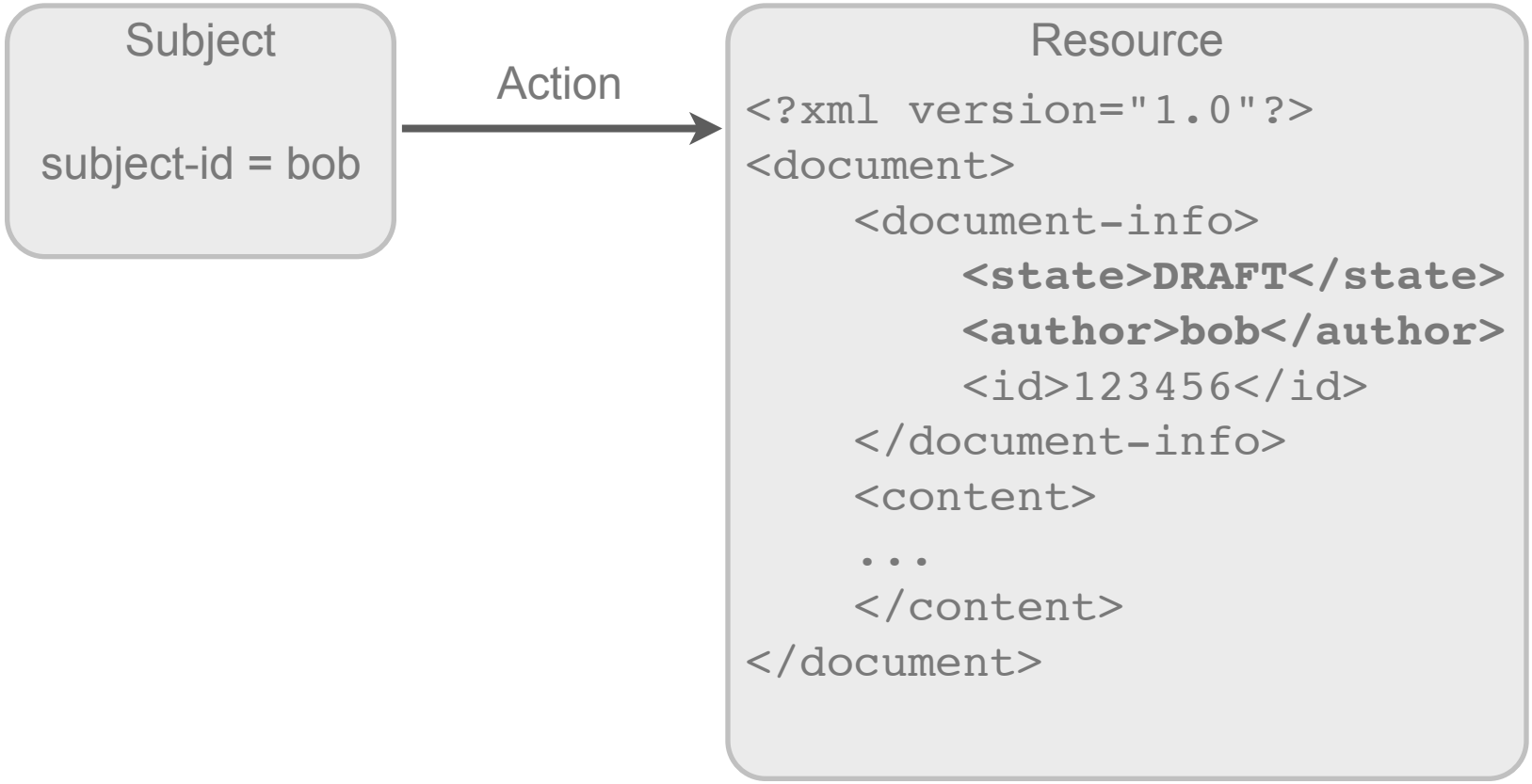
# Feingranulare, attributbasierte Autorisierung



# Inhaltsbasierte Autorisierung

„Ein Dokument im Zustand DRAFT darf nur für den Autor einsehbar und änderbar sein.“

# Inhaltsbasierte Autorisierung



# Segregation of Duties

# eXtensible Access Control Markup Language

- XML-basierte Sprache für Attribute-based Access Control
- OASIS-Standard
- Version 1.0: 2003
- Version 2.0: 2005
- Version 3.0: wird aktuell veröffentlicht

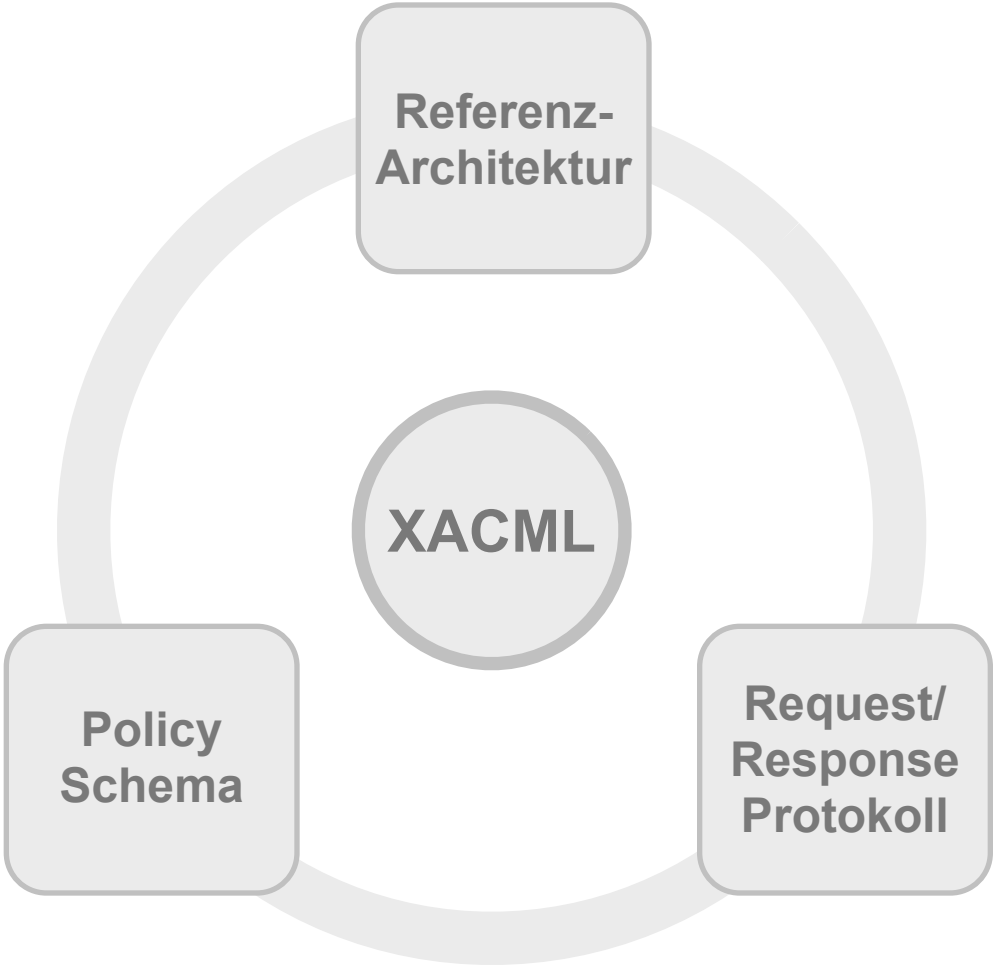
iC CONSULT

# eXtensible Access Control Markup Language

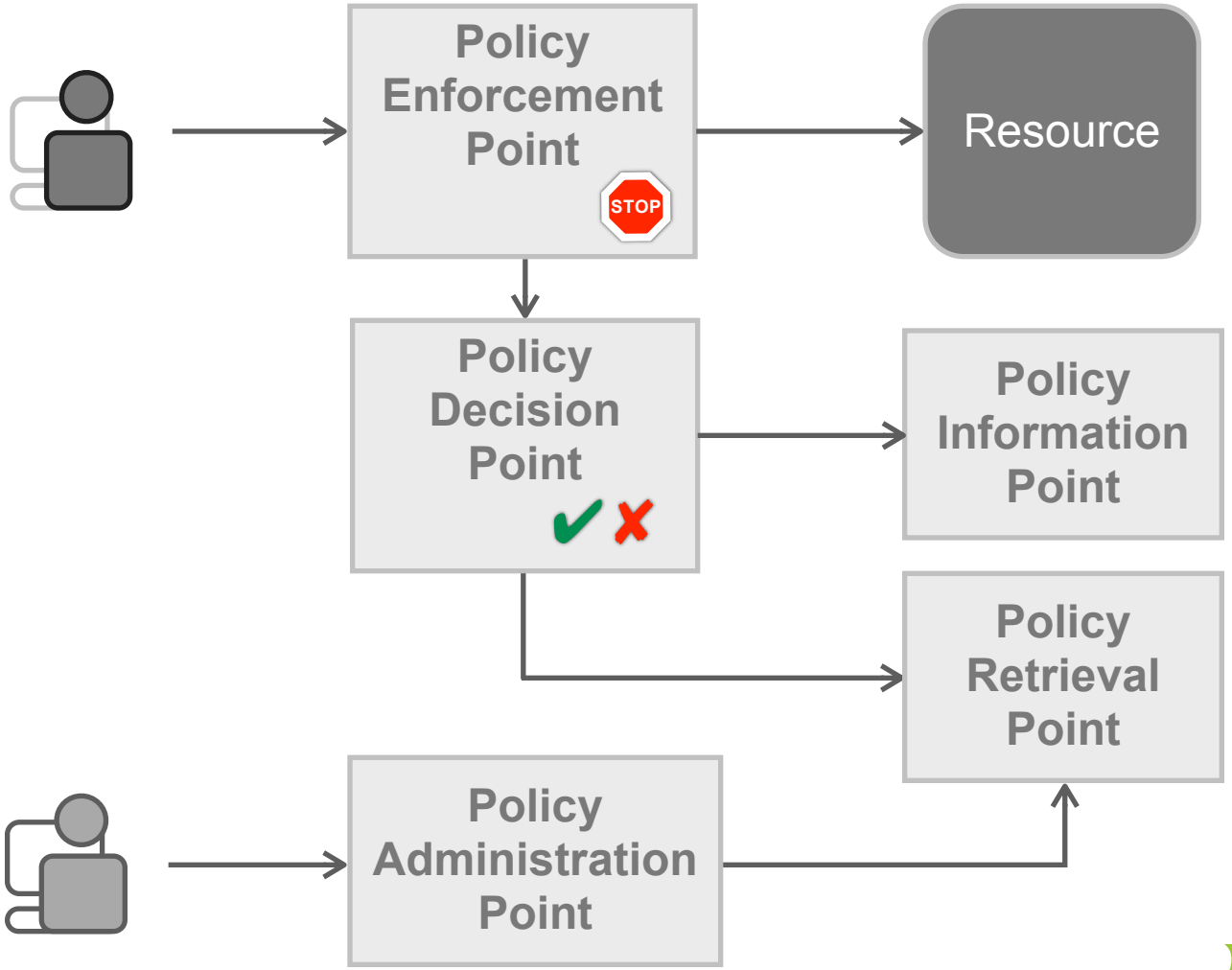
- Alle verfügbaren Informationen können verwendet werden:
  - Subject-Attribute
  - Resource-Attribute
  - Action-Attribute
  - Environment-Attribute
- Unterstützt alle wichtigen Autorisierungsmodelle:
  - Access Control Lists (ACLs)
  - Role-Based Access Control (RBAC)
  - Attribute-Based Access Control (ABAC)
  - Risk-Adaptive Access Control (RAdAC)
  - Mandatory Access Control (MAC)
  - Whitelists/Blacklists
  - Context-Based Authorization
  - Content-Based Authorization
  - ...

iC CONSULT

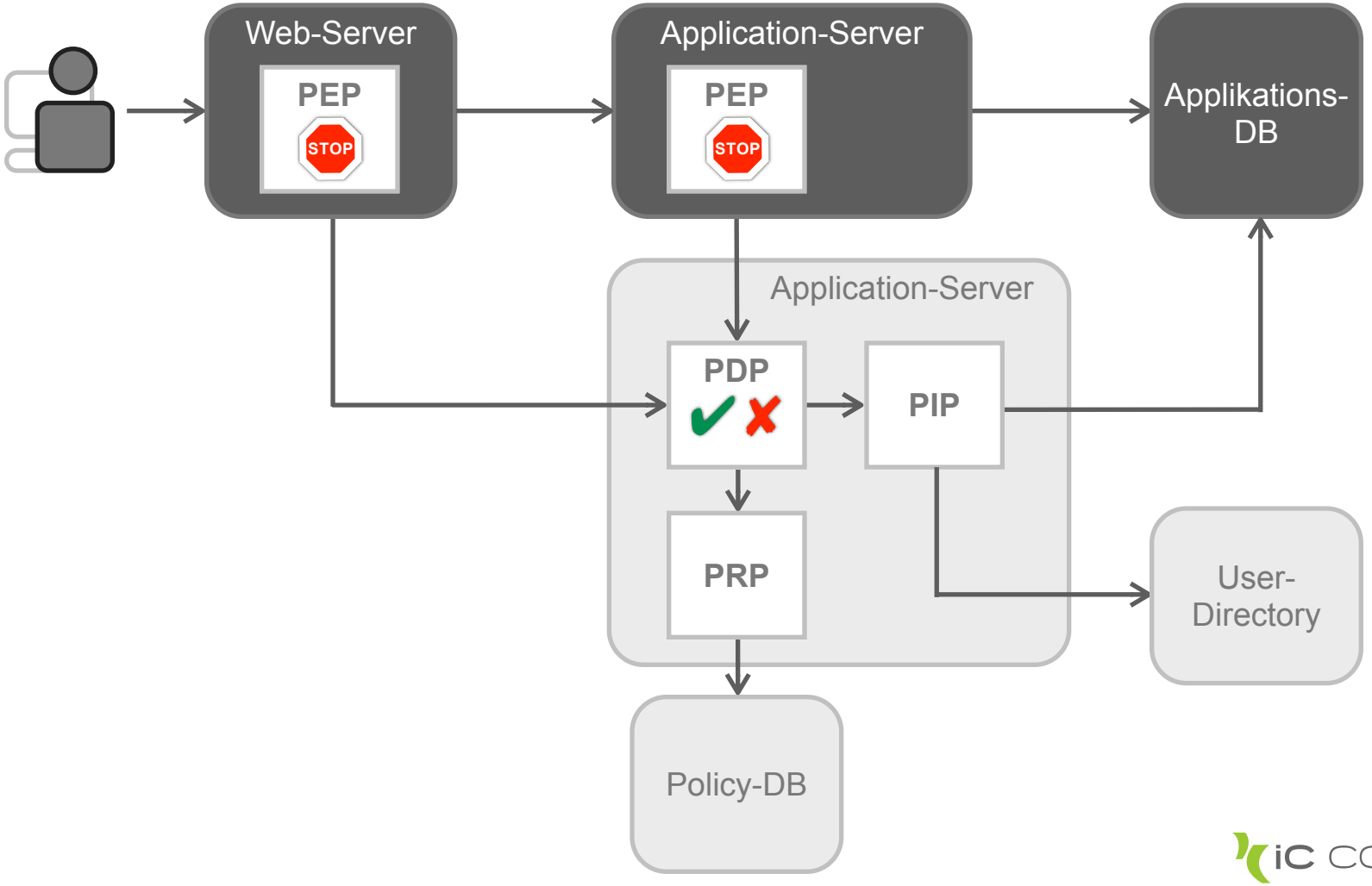
# XACML definiert...



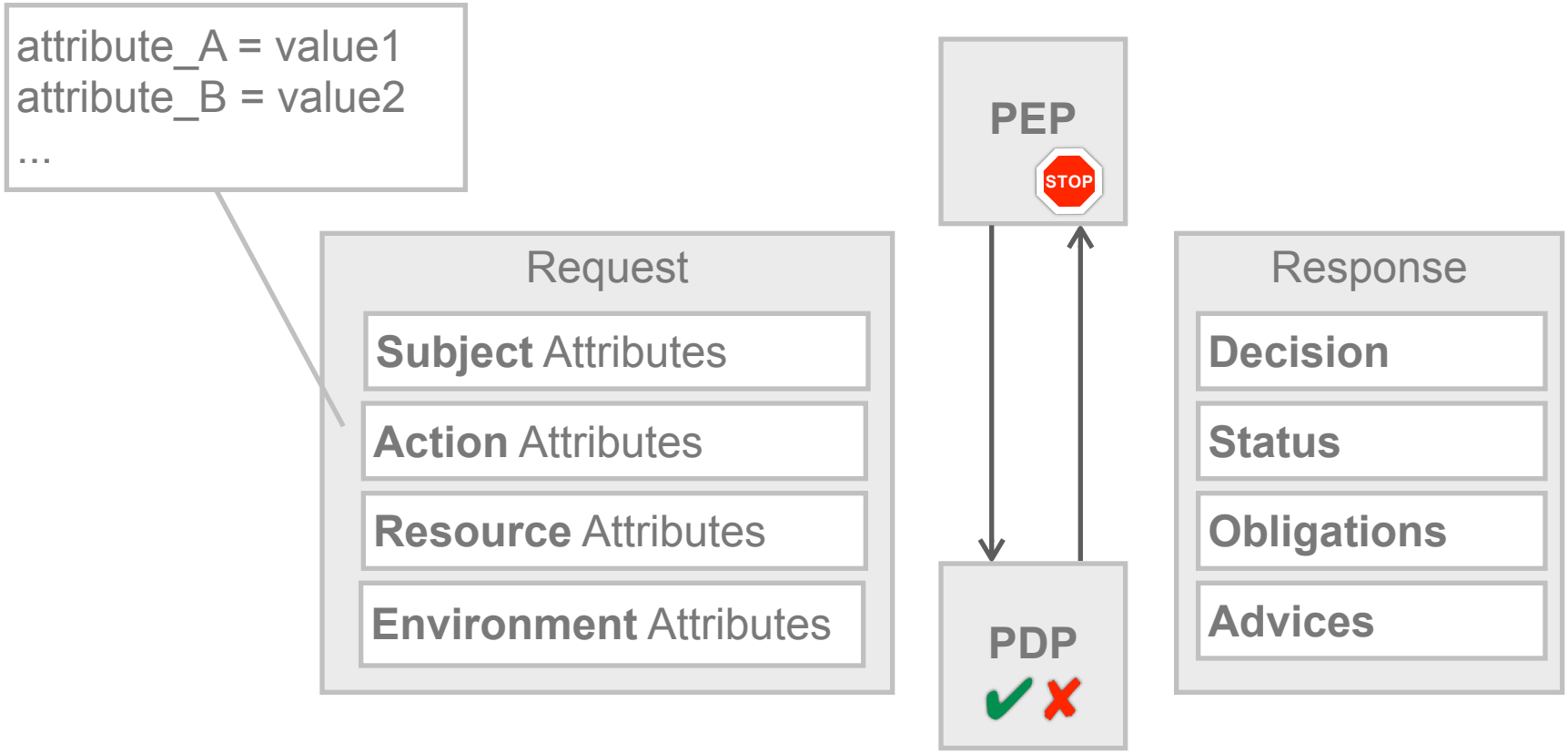
# Referenzarchitektur



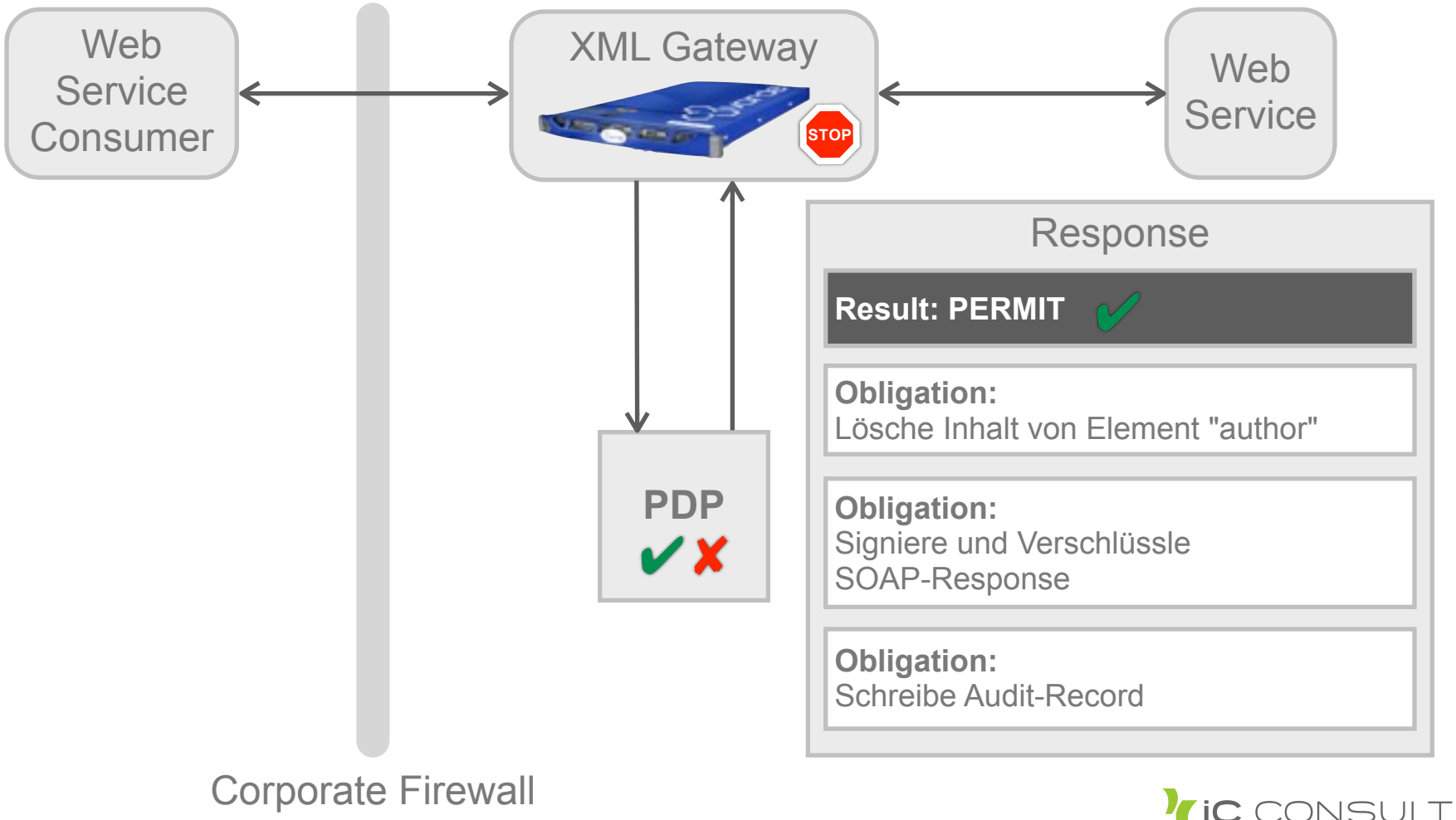
# Beispielhafte, konkrete Architektur



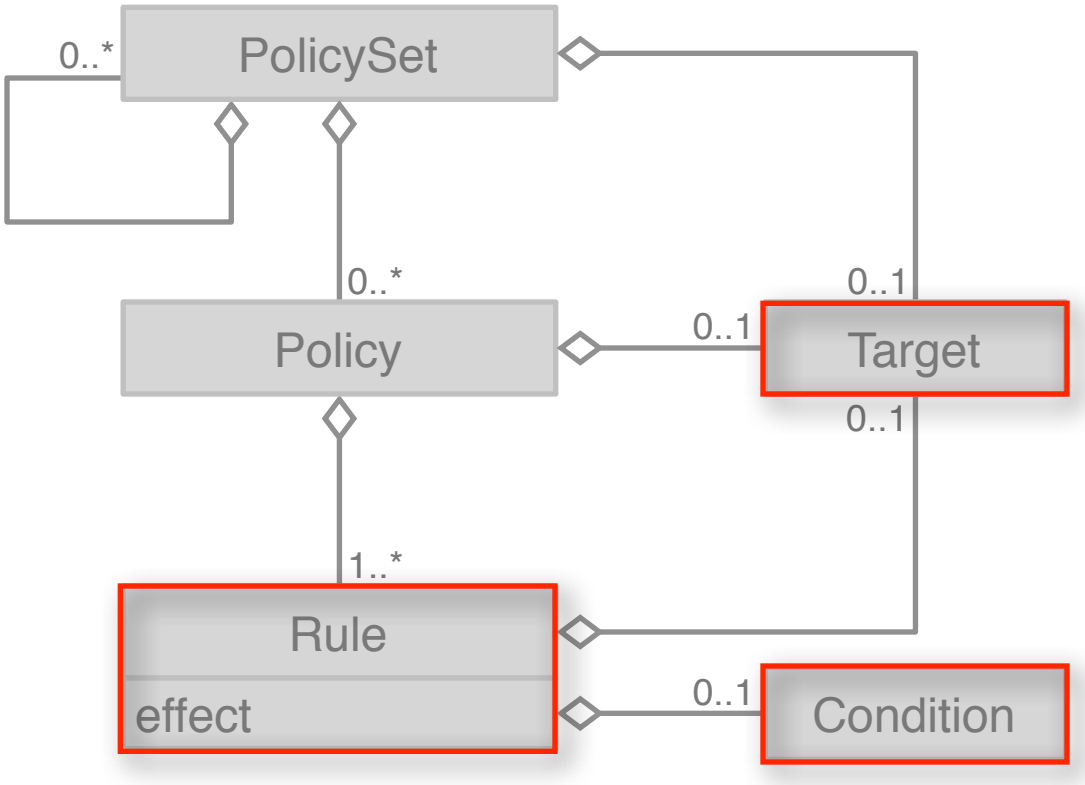
# Request/Response Protokoll



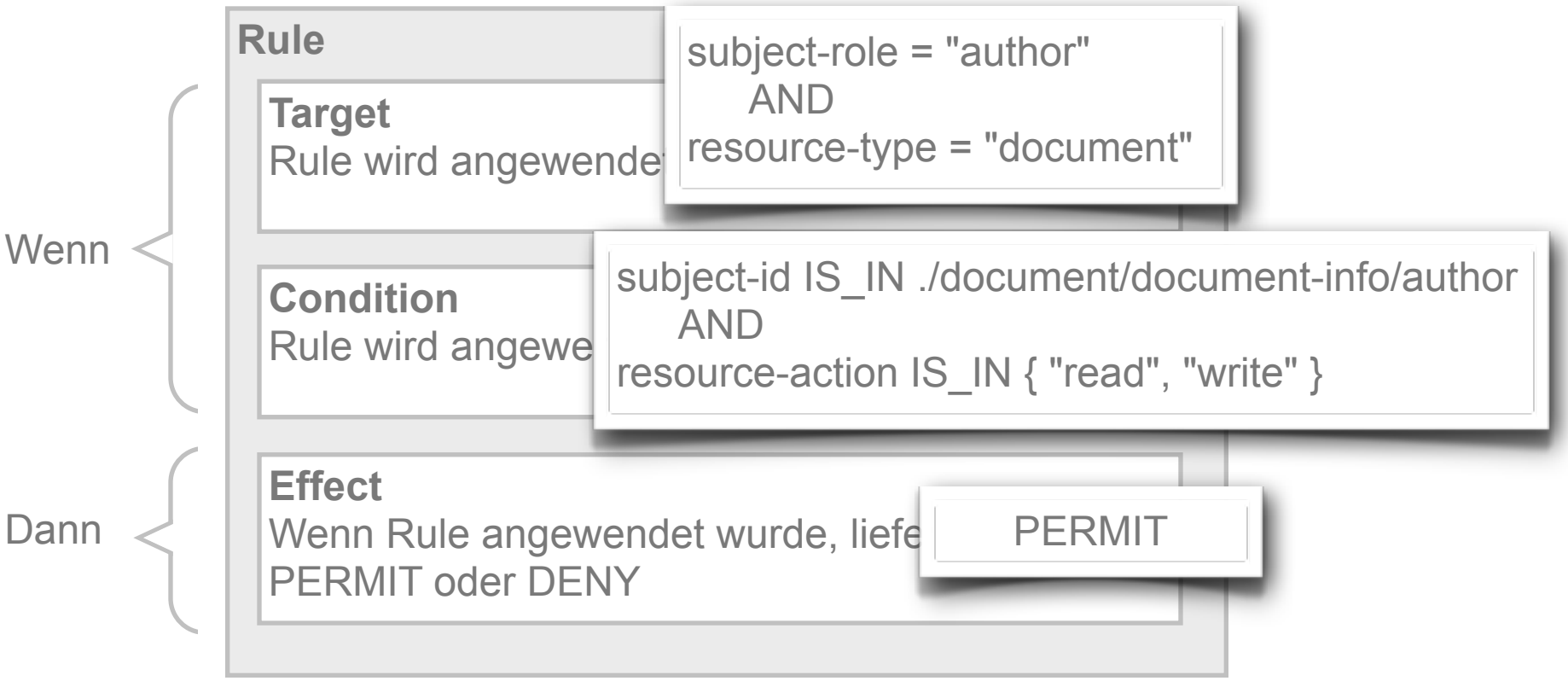
# Beispiel für den Einsatz von Obligations



# Policy-Struktur



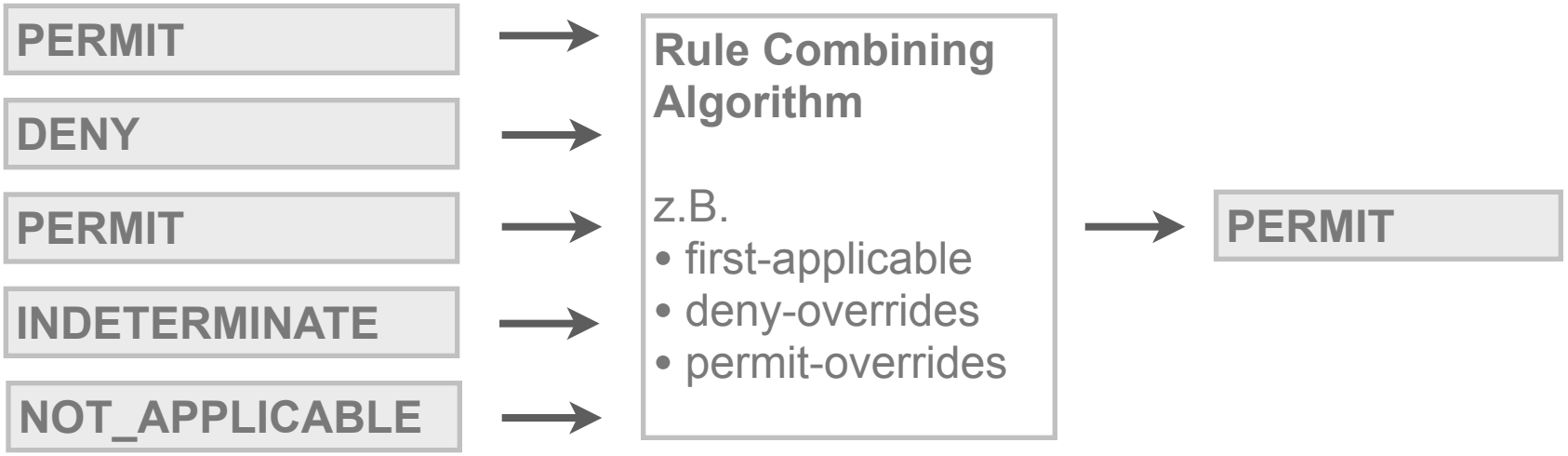
# Rule-Struktur



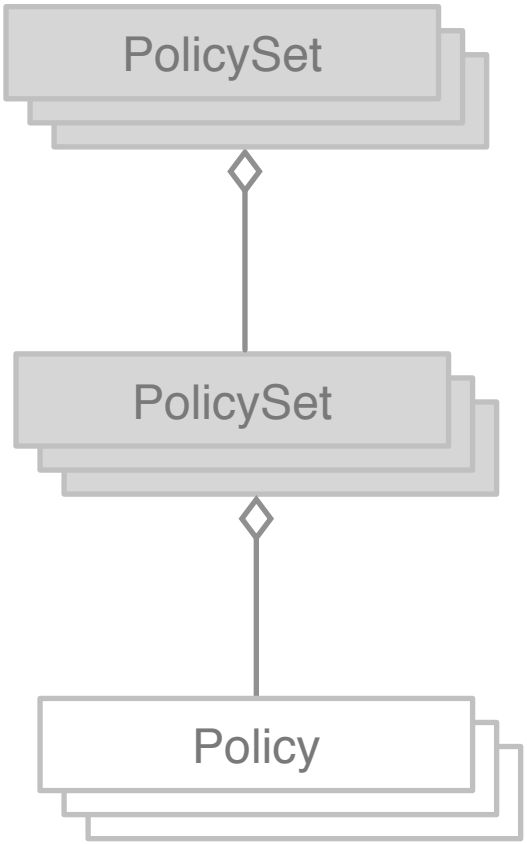
# Combining Algorithms

**Effekte der einzelnen Rules**

**Effekt der Policy**



# Verschachtelung von PolicySets




Produktgruppen

Produkte

Komponenten

# Entwickeln wir eine Beispielpolicy...

**Benutzer**

dürfen auf

**Ressourcen**,

die als **Confidential klassifiziert** sind,

nur dann **zugreifen**,

wenn sie sich über ein

**starkes Authentisierungsverfahren angemeldet** haben.

# Erst brauchen wir Attribute und Definitionen...

| Subject |
| --- |
| auth-strength: Integer |

Benutzer ist über starkes Authentisierungsverfahren angemeldet, wenn
auth-strength >= 2

| Resource |
| --- |
| classification: String |

classification $\in$ { "PUBLIC", "INTERNAL", "CONFIDENTIAL", "SECRET" }

# Rumpf der Policy

```
<?xml version="1.0" encoding="UTF-8"?>
<Policy RuleCombiningAlgId="deny-overrides">
    <Target>
        Target: Ressourcen mit classification=CONFIDENTIAL
    </Target>
    <Rule Effect="Permit" RuleId="AuthStrenghtHighPermit">
        Target: Subjects mit auth-strength >= 2
    </Rule>
    <Rule Effect="Deny" RuleId="AuthStrengthLowDeny">
        Target: Subjects mit auth-strength < 2
        Advice: required-auth-strenght=2
    </Rule>
</Policy>
```

# Target der Policy

```
<Target>
   <AnyOf>
      <AllOf>
         <Match MatchId="string-equal">
            <AttributeValue DataType="...#string">
               CONFIDENTIAL
            </AttributeValue>
            <AttributeDesignator Category="resource" DataType="...#string"
               AttributeId="classification"/>
         </Match>
      </AllOf>
   </AnyOf>
</Target>
```

# Target der Permit-Rule

```
<Target>
   <AnyOf>
      <AllOf>
         <Match MatchId="integer-less-than-or-equal">
            <AttributeValue DataType="...#integer">
               2
            </AttributeValue>
            <AttributeDesignator AttributeId="auth-strength"
               Category="subject-category:access-subject"
               DataType="...#integer"/>
         </Match>
      </AllOf>
   </AnyOf>
</Target>
```

# Target der Deny-Rule

```
<Target>
   <AnyOf>
      <AllOf>
         <Match MatchId="integer-greater-than">
            <AttributeValue DataType="...#integer">
               2
            </AttributeValue>
            <AttributeDesignator AttributeId="auth-strength"
               Category="subject-category:access-subject"
               DataType="...#integer"/>
         </Match>
      </AllOf>
   </AnyOf>
</Target>
```

# Advice

```
<AdviceExpressions>
   <AdviceExpression AdviceId="AuthStrengthTooLow" AppliesTo="Deny">
      <AttributeAssignmentExpression
         AttributeId="required-auth-strength">
         <AttributeValue DataType="...#integer">
            2
         </AttributeValue>
      </AttributeAssignmentExpression>
   </AdviceExpression>
</AdviceExpressions>
```

# XACML ist etwas textlastig...

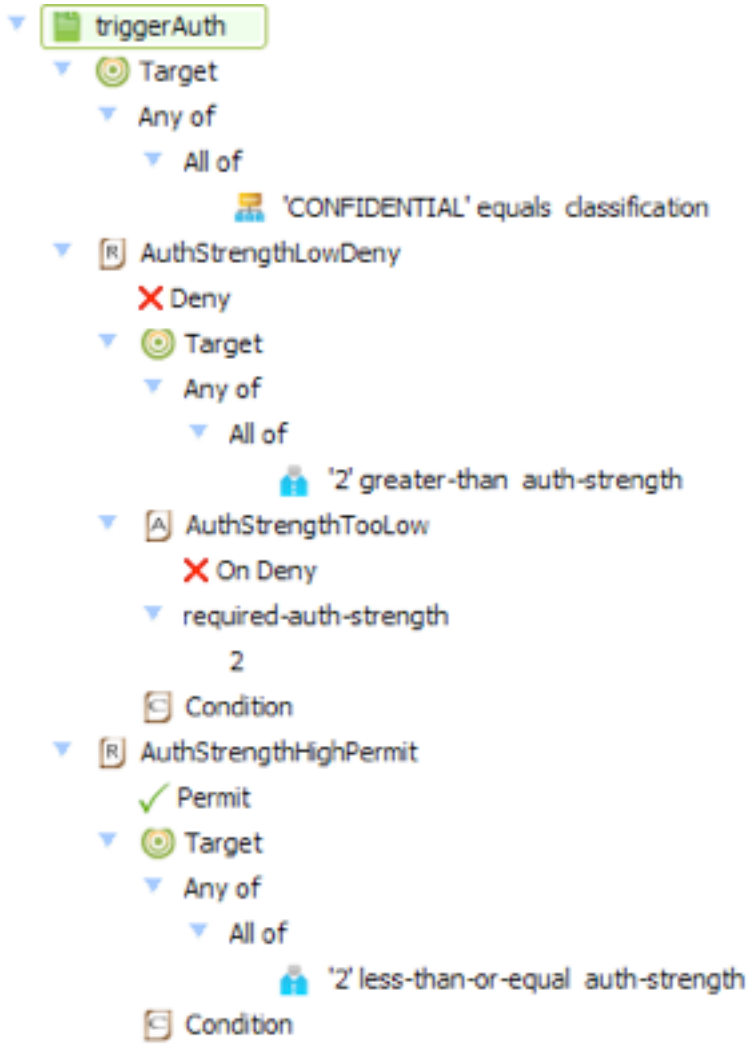
```
<?xml version="1.0" encoding="UTF-8"?>
<xacml3:Policy xmlns:xacml3="urn:oasis:names:tc:xacml:3.0:core:schema:wd-17" PolicyId="triggerAuth"
RuleCombiningAlgId="urn:oasis:names:tc:xacml:3.0:rule-combining-algorithm:deny-overrides" Version="1.0">
   <xacml3:Description>Policy demonstrating the use of Advices to trigger stronger user authentication</xacml3:Description>
   <xacml3:PolicyDefaults><xacml3:XPathVersion>http://www.w3.org/TR/1999/REC-xpath-19991116</xacml3:XPathVersion></xacml3:PolicyDefaults>
   <xacml3:Target>
      <xacml3:AnyOf>
         <xacml3:AllOf>
            <xacml3:Match MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
               <xacml3:AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">CONFIDENTIAL</xacml3:AttributeValue>
               <xacml3:AttributeDesignator AttributeId="classification" Category="urn:oasis:names:tc:xacml:3.0:attribute-category:resource"
DataType="http://www.w3.org/2001/XMLSchema#string" MustBePresent="false"/>
            </xacml3:Match>
         </xacml3:AllOf>
      </xacml3:AnyOf>
   </xacml3:Target>
   <xacml3:Rule Effect="Deny" RuleId="AuthStrengthLowDeny">
      <xacml3:Description/>
      <xacml3:Target>
         <xacml3:AnyOf>
            <xacml3:AllOf>
               <xacml3:Match MatchId="urn:oasis:names:tc:xacml:1.0:function:integer-greater-than">
                  <xacml3:AttributeValue DataType="http://www.w3.org/2001/XMLSchema#integer">2</xacml3:AttributeValue>
                  <xacml3:AttributeDesignator AttributeId="auth-strength" Category="urn:oasis:names:tc:xacml:1.0:subject-category:access-subject"
#DataType="http://www.w3.org/2001/XMLSchema#integer" MustBePresent="false"/>
               </xacml3:Match>
            </xacml3:AllOf>
         </xacml3:AnyOf>
      </xacml3:Target>
```

Es geht weiter...

# XACML ist etwas textlastig...

```
...
      <xacml3:AdviceExpressions>
        <xacml3:AdviceExpression AdviceId="AuthStrengthTooLow" AppliesTo="Deny">
          <xacml3:AttributeAssignmentExpression AttributeId="required-auth-strength"
Category="urn:oasis:names:tc:xacml:3.0:attribute-category:resource" Issuer="">
            <xacml3:AttributeValue DataType="http://www.w3.org/2001/XMLSchema#integer">2</xacml3:AttributeValue>
          </xacml3:AttributeAssignmentExpression>
        </xacml3:AdviceExpression>
      </xacml3:AdviceExpressions>
    </xacml3:Rule>
    <xacml3:Rule Effect="Permit" RuleId="AuthStrengthHighPermit">
      <xacml3:Description/>
      <xacml3:Target>
        <xacml3:AnyOf>
          <xacml3:AllOf>
            <xacml3:Match MatchId="urn:oasis:names:tc:xacml:1.0:function:integer-less-than-or-equal">
              <xacml3:AttributeValue DataType="http://www.w3.org/2001/XMLSchema#integer">2</xacml3:AttributeValue>
              <xacml3:AttributeDesignator AttributeId="auth-strength" Category="urn:oasis:names:tc:xacml:1.0:subject-category:access-subject"
DataType="http://www.w3.org/2001/XMLSchema#integer" MustBePresent="false"/>
            </xacml3:Match>
          </xacml3:AllOf>
        </xacml3:AnyOf>
      </xacml3:Target>
    </xacml3:Rule>
</xacml3:Policy>
```

# Mit kommerziellem PAP-UI



# XACML Erweiterbarkeit

- Datentypen
- Funktionen
- Combining Algorithms
- Technische Profile
  - Definieren zusätzliche Funktionalität
  - z.B. Multiple-Decision Profile
- Nichttechnische Profile
  - Beschreiben Best-Practices
  - z.B.
    - RBAC Profile
    - Hierarchical Resource Profile
    - Administration and Delegation Profile

iC CONSULT

# APIs und SPIs

APIs zur Integration des PEPs

- Open Source
  - OpenAz
- Herstellerspezifisch

SPIs zur Integration von PIP, PRP und PAP

- Herstellerspezifisch

SPIs zur Erweiterung von XACML

- Herstellerspezifisch

# OpenAZ-Beispiel

```
PepRequest req = pep.newPepRequest(
                    subject, action, resource, environment);
PepResponse resp = req.decide();
boolean permit = resp.allowed();
Map<String, Obligation> obligations = resp.getObligations();
```

# Produkte

- Open Source
  - JBoss PicketBox
  - SunXACML
- Kommerziell
  - Axiomatics
  - Oracle
  - Nextlabs
  - Quest

# Wohin geht die Reise?

# XACML ermöglicht

- Attributbasierte, feingranulare, kontextbasierte und inhaltsbasierte Autorisierung
- Die Umsetzung der wichtigsten Autorisierungsmodelle
- Applikations- und technologieunabhängige Autorisierung
- Konsistente Autorisierungsregeln über Applikationen und Plattformen hinweg
- Management von Autorisierungsregeln ohne Quellcodeänderung

iC CONSULT

# Aktueller Stand

XACML 3.0 ist ein ausgereifter Standard

Aber:

- Es gibt keine XACML-Produktsuites im Open-Source-Bereich
- Knappe Auswahl im kommerziellen Bereich
- „Was ist erlaubt?“-Queries nicht im Standard behandelt
- Attributmanagement als neue Disziplin („privilege-giving attributes“)

iC CONSULT

# Unsere Beobachtungen

- XACML wird in einigen Unternehmen in großem Maßstab eingesetzt.
- Bedarf nach standardbasierter, externalisierter Autorisierung steigt durch:
  - SOA und cloud-basierte Architekturen
  - Geschäftsmodelle mit starker Endkunden-Interaktion
  - Anforderungen aus GRC (Governance, Risk & Compliance)

iC CONSULT


Auch im realen Leben macht grobgranulare Autorisierung oft wenig Sinn.

# Vielen Dank für Ihre Aufmerksamkeit.


**KONTAKT**

Stefan Bohm
stefan.bohm@ic-consult.de
www.ic-consult.com

iC Consult GmbH
Keltenring 14
82041 Oberhaching