

# **SPML-basierte Provisionierung im Identity Management**

**Herbsttreffen des ZKI-Arbeitskreises  
Verzeichnisdienste, Jena, 4.-5.10.2011**

**Peter Gietz,  
DAASI International GmbH**

# DAASI International GmbH

- **Spezialisiert auf Verzeichnisdienste, digitale Signatur, (Federated) Identity Management, Grid-Computing und eHumanities (einschl. anwendernaher Programmierung)**
- **Spin-Off der Universität Tübingen**
  - **seit 2000 auf dem Markt**
  - **7 Mitarbeiter (Tendenz steigend)**
- **Hauptkundenzielgruppe: Hochschulen, Behörden, Forschungseinrichtungen, Bibliotheken und Verwaltungen**
- **Forschungsorientiert:**
  - **BMBF-Projekte zu Grid-Computing (IVOM, GapSLC) und eHumanities (TextGrid, DARIAH-DE)**
- **Konzentriert auf Open-Source, Aktiv in Standardisierung (IETF, OGF, TERENA, DFN, ...)**
- **Mehr unter: [www.daasi.de](http://www.daasi.de)**

# Agenda

1. Grundlegendes zu Identity Management und Provisionierung
2. Einführung in SPML
3. Beschreibung unseres SPML Lösungsansatzes
4. Praktische Erfahrungen

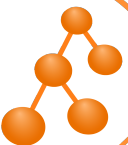


# Grundlegendes zu Identity Management

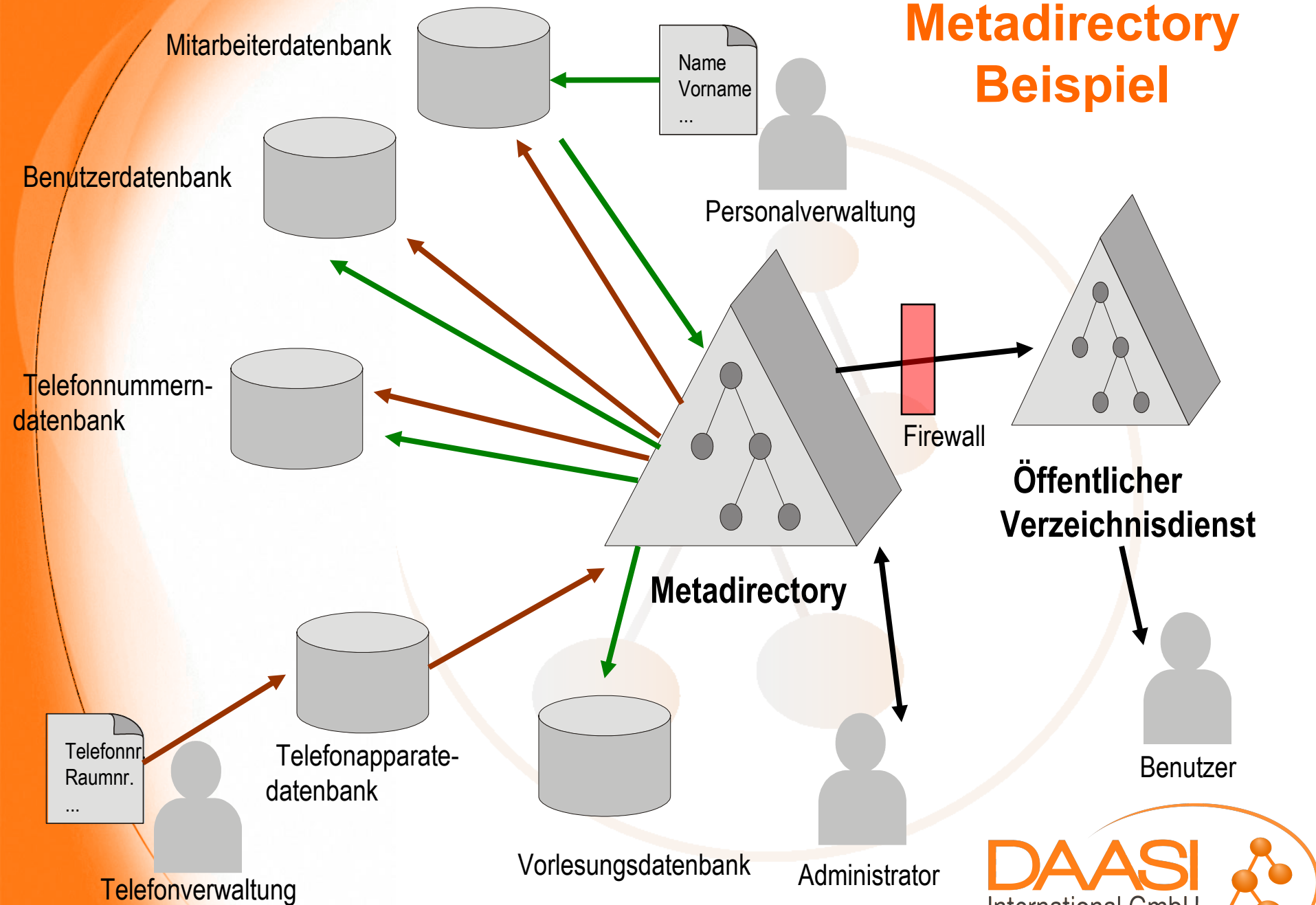


# Wichtigste Komponenten von Identity Management Systemen

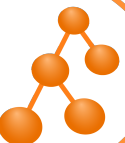
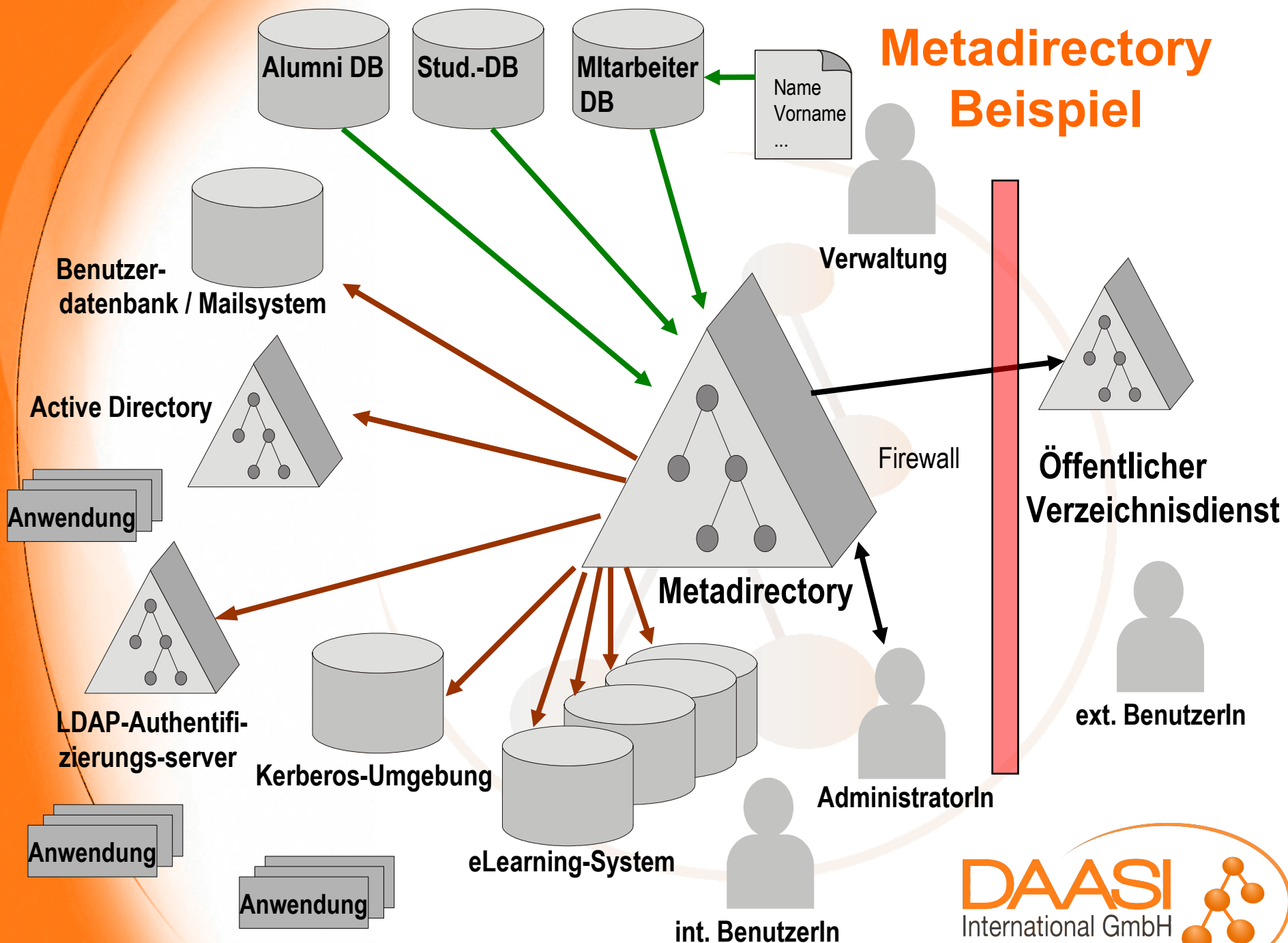
- **Quelldatenbanken**
  - autoritative Datenquellen und Anwendungen
- **Zielsysteme**
  - Konsumenten dieser autoritativen Daten
- **Verzeichnisdienste sind zentrale Bestandteile**
  - speichern Identitätsinformation, Passwörtern, Zertifikate, Rollen und Berechtigungen, Policy
  - Standards: X.500, LDAP
  - Implementierungen: OpenLDAP, Novell eDirectory, MS Active Directory
- **Metadirectories dienen zur**
  - Synchronisierung verschiedener Datenspeicher
  - Vermeidung von Inkonsistenzen
  - Passwort-Verwaltung und –Synchronisierung
- **Konnektoren verbinden**
  - Datenquellen mit Metadirectory
  - Metadirectory mit Zielsystemen (= Provisioning)



# Metadirectory Beispiel



# Metadirectory Beispiel





# Was haben wir gelernt?

- Die Anzahl der Quellsysteme ist begrenzt
  - Mitarbeiter-DB, StudierendenDB, AlumniDB
  - Vielleicht noch eine Gäste-Verwaltung für alle Fälle, die nicht in diesen 3 Datenbanken gepflegt werden
  - Oft sind die Datenbanken homogen
    - (z.B.: HISSOS, HISSVA)
- Die Anzahl der Zielsysteme ist nicht eingrenzbar
  - Neue Authentifizierungsverfahren
    - Passworthash-Problem!
  - Neue Anwendungen, die mehr als authentifizieren wollen
- Eine Lösung für letztere wäre ein SSO-System, welches Attribute überträgt (SAML / Shibboleth)
- Eine andere wäre generischeres Provisionieren



# Beispiel: Prinzip von AD-Konnektoren

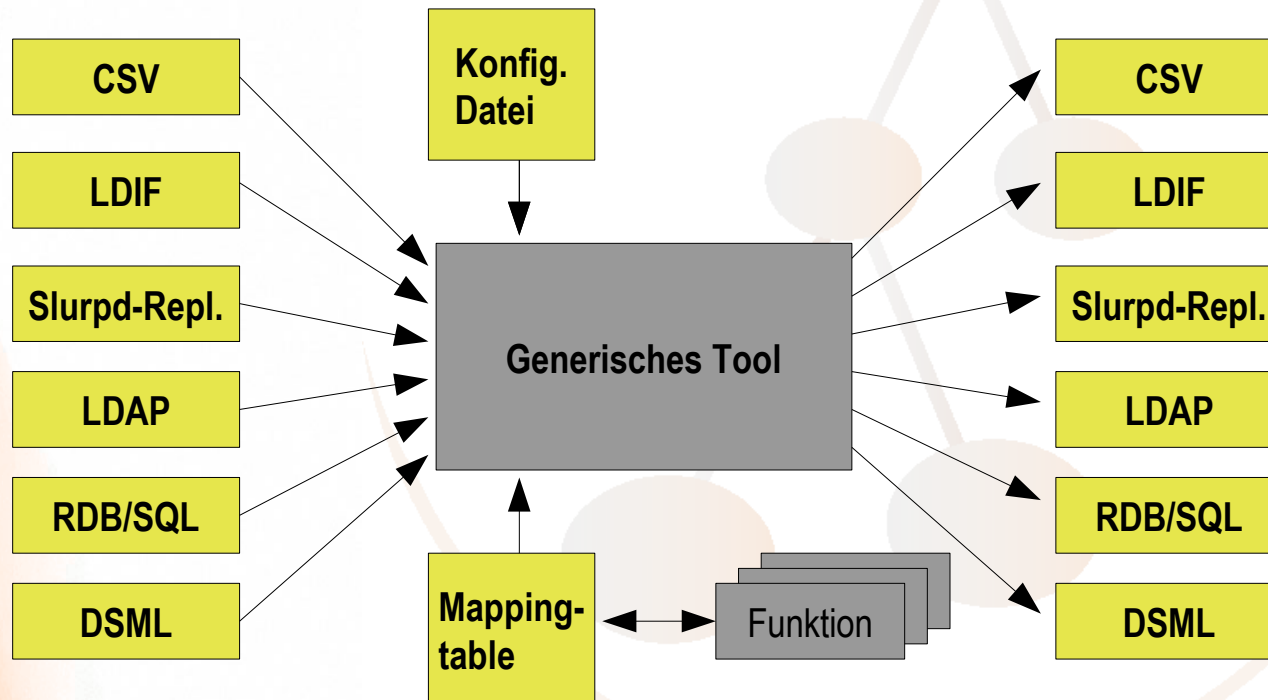
- Grundsätzlich kann ein AD über Standard-LDAP-Befehle angesprochen werden
  - Allerdings nicht 100%ge Unterstützung des LDAP-Standards ...
  - Nur LDAPS kein START\_TLS
- Zusätzlich gibt es das proprietäre Protokoll ADSI
- Bei Provisionierung von AD müssen einige Besonderheiten berücksichtigt werden:
  - SID Generierung
  - Kompliziertes Anlegen eines neuen Eintrags:
    - 1.) Eintrag anlegen mit den Daten und als gesperrt markieren
    - 2.) Passwort anlegen
    - 3.) Eintrag entsperren

## Andere Beispiele

- Leider hat sich kein Standard für LDAP-Replikation etabliert
  - Das nächste, was wir haben ist syncrepl von OpenLDAP
  - Deshalb müssen öfters auch LDAP-LDAP-Synchronisierung „handgestrickt“ werden
- Kerberos-DB (falls nicht LDAP) kann nur über Kerberos-spezifische Mittel befüllt werden
- Jede Anwendung, die nicht nur authentifizieren will, benötigt, eigene Provisionierungskonnektoren
- Auch wenn Identity Management Produkte Konnektoren-Entwicklungs-Tools haben, wird jedes neue Zielsystem zu einem mindestens kleinem Projekt
- Oder sie stricken wieder eine neue Lösung

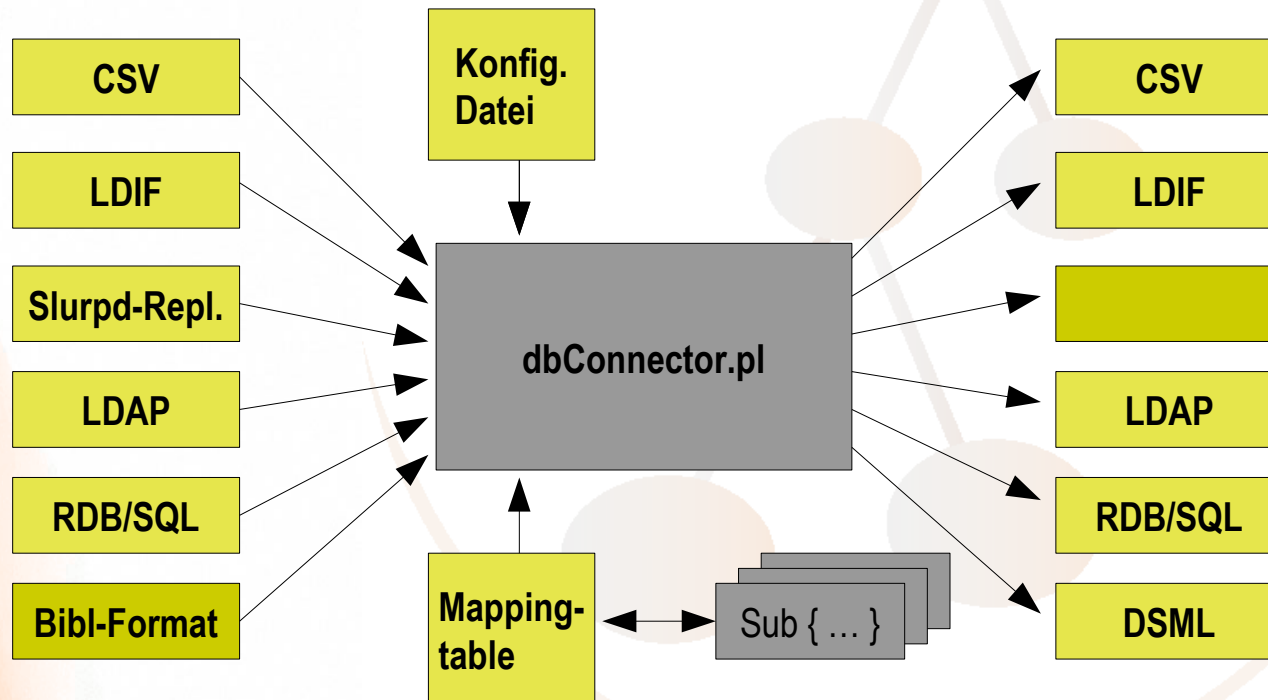
# Wie wäre es besser?

- Es gäbe ein Tool, das alle Datenbanken miteinander verbinden kann, das flexibel konfigurierbar und beliebig erweiterbar wäre



# Wie wäre es besser?

- Es *gibt* ein Tool, das alle Datenbanken miteinander verbinden kann, das flexibel konfigurierbar und beliebig erweiterbar wäre



- Funktioniert, ist aber komplex

# Wie wäre es noch besser?

- Alle Hersteller von Datenbanken und von Anwendungen, die Daten extern beziehen, einigen sich auf einen Standard, wie Provisionierungsinformation übertragen werden soll.
- Ein solcher Standard sollte:
  - Mit standardtools bewältigbar sein (z.B.: XSLT)
  - Offen für Erweiterungen sein
  - Über verschiedene Protokolle übertragbar sein (HTTP, SOAP über HTTP, etc.)
- Wenn neue Anwendungen/Datenbanken SPML unterstützen, ist die Integration denkbar einfach
- Es gibt einen solchen Standard, der zunehmend Beachtung findet: SPML

# Kurze Einführung in SPML

**Glossar:** DSML = Directory Service Markup Language.  
XML-Format zur Abbildung von LDAP-Daten und -Operationen



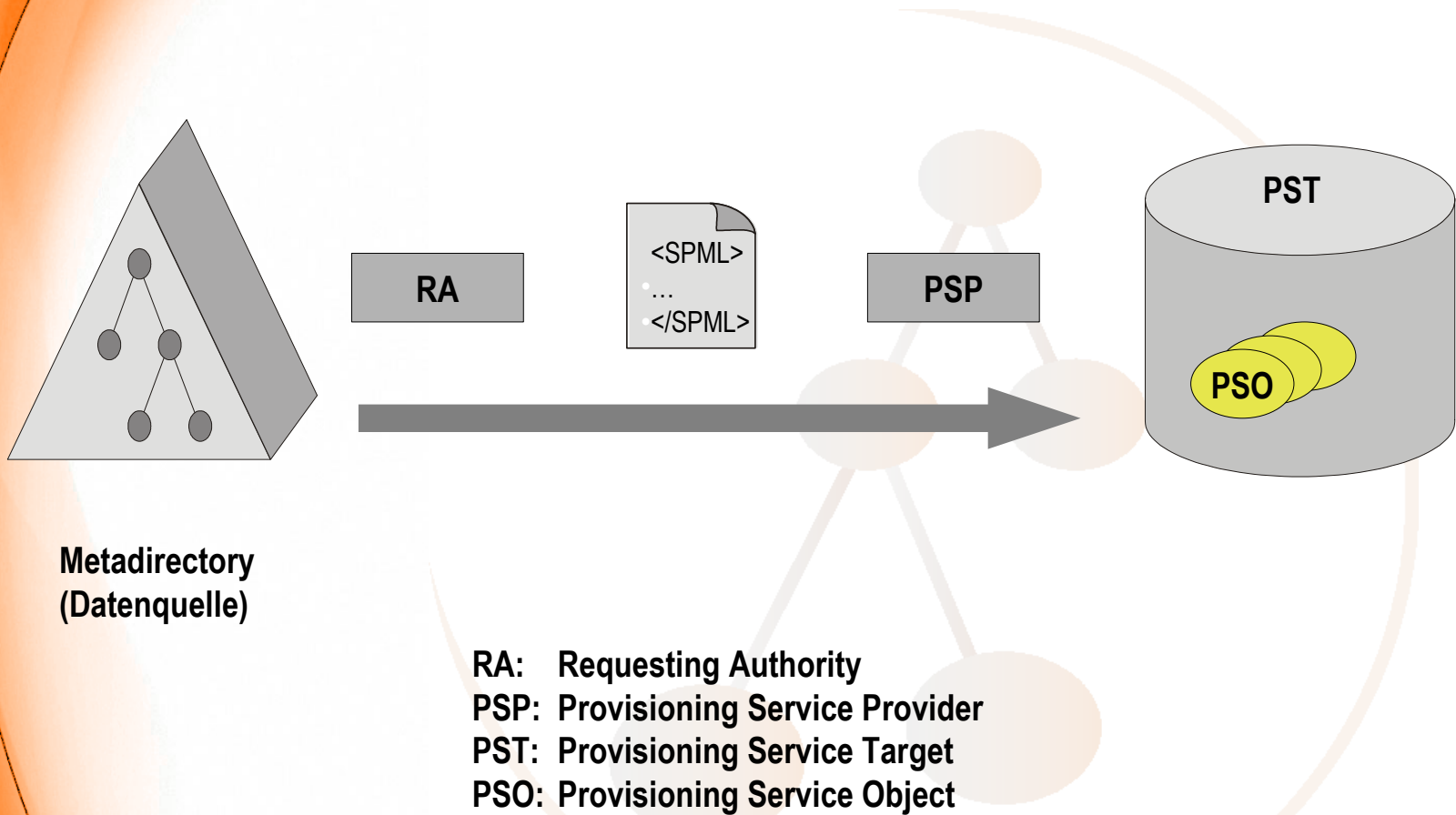
# Der SPML-Gedanke

- **SPML v2 (Service Provisioning Markup Language) ist OASIS Standard vom April 2006**
- **SPML spezifiziert ein XML-Format zur Provisionierung**
- **Soll unabhängig von der Art des Quell- und der Zielsysteme für Provisionierung verwendet werden können.**
- **Definiert Grundoperationen „add“, „modify“, „delete“ und „lookup“, sowie Erweiterungen wie z.B. „search“.**
  - **Jede Operation besteht aus einem Request und einem Response**
  - **Operationsmodell ist flexibel erweiterbar**
- **Offen für verschiedene Datenformate (SPML „Envelope“, flexible „payload“)**
- **Vordefiniertes Datenschema z.B. „SPMLv2 - DSMLv2 Profile“:**
  - **Datenänderungsanweisungen im DSMLv2-Format**
  - **Transportiert im SPML-Dokument**

# SPML-Komponenten

- **Provisioning System Object (PSO):**
  - Einzelnes Datenobjekt in einem Daten-Container, also z.B. ein AD-Eintrag
- **Provisioning Service Target (PST):**
  - Ist Container (Zielsystem) für Objekte (z.B. AD)
- **Provisioning Service Provider (PSP):**
  - Nimmt SPML-Dokumente für ein oder mehrere PSTs entgegen
  - Führt Änderungen auf PSOs des PSTs durch (also: ändert Einträge im AD)
- **Requesting Authority (RA):**
  - Erzeugt SPML-Dokumente die der PSP konsumiert
  - Wird am Quellsystem angeschlossen

# SPML-Komponenten



# Vor- und Nachteile

## ➤ Vorteile:

- SPML kann leicht von XML-Parsern eingelesen werden
- Erweiterbares Format
- Es werden nur Änderungen provisioniert (im Gegensatz zu einem Gesamtabgleich), diese finden zeitnah, z.B. jede Minute statt
- Klar strukturiertes generisch einsetzbares Provisionierungsmodell
- Standardisierte Möglichkeit, auch Gruppeninformationen zu provisionieren

## ➤ Nachteile:

- SPML geht davon aus, dass Datenänderungen am Zielsystem nur über die Provisionierung erfolgt
- Recovery eines Zielsystems über SPML nicht durch Standardoperationen möglich
- Typischer XML-Overhead

# Beschreibung der SPML-Implementierung von DAASI



**ALOIS-Folien von Frau Schmaus,  
Rechenzentrum Universität Augsburg**

**Folien 17 und 18**

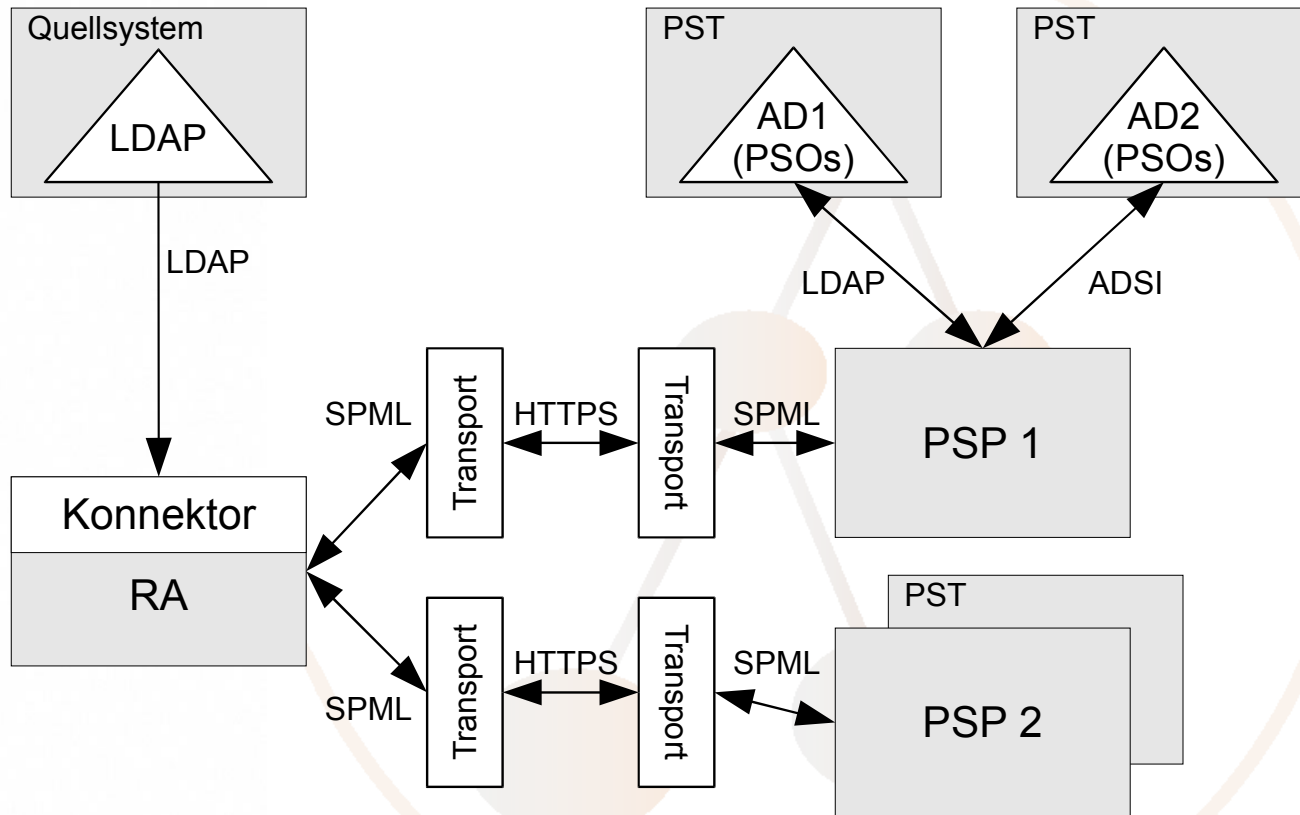




# Beteiligte Komponenten

- **Quellsystem OpenLDAP**
  - mit Overlay „accesslog“ durch das alle Änderungs-Operationen in einem Teilbaum des Servers gespeichert werden können
- **RA**
  - mit Konnektor für „accesslog“, der sehr zeitnah Änderungen wahrnimmt.
  - Transport über REST (HTTPS)
  - verwaltet eine Queue für jeden PSP
- **Active Directory als eins der möglichen Zielsysteme**
- **PSP für Active Directory**
  - Änderungen der Objekte (PSOs) über LDAP
  - oder über ADSI

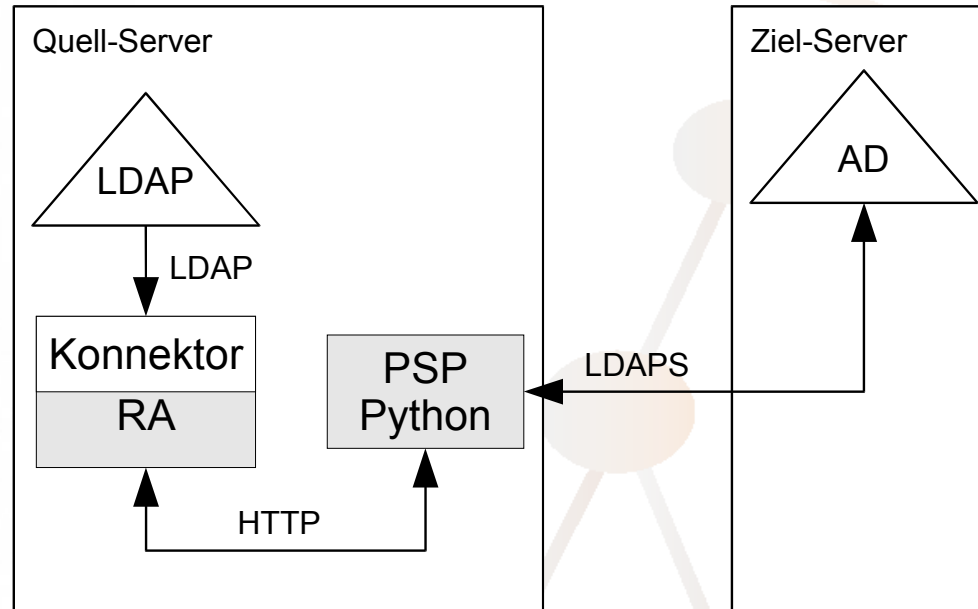
# Aufbau einer SPML-Umgebung



# Ablauf

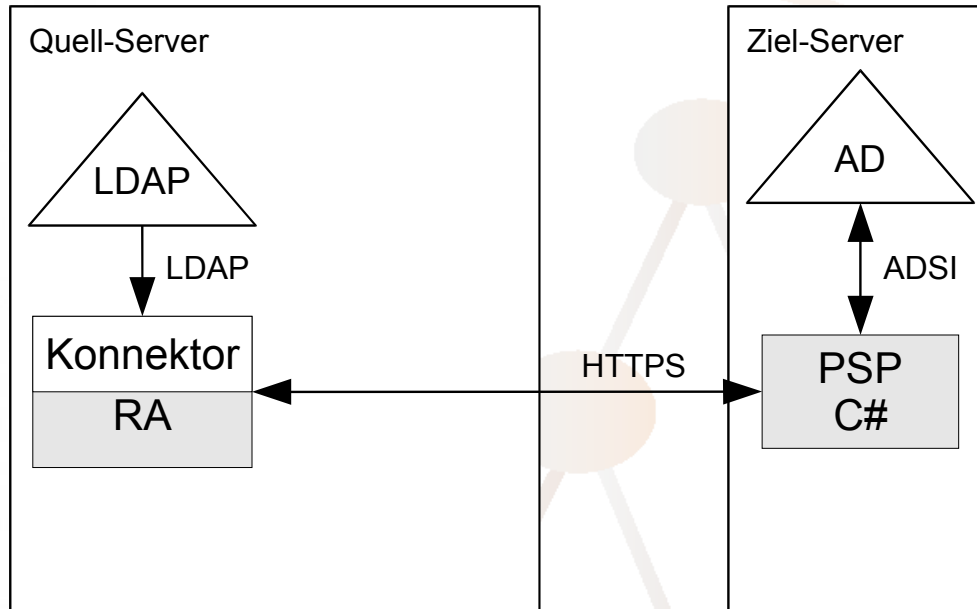
- Änderung im OpenLDAP wird über Overlay protokolliert
- Periodische Abfrage der protokollierten Änderungen werden ausgelesen
- Ausgelesene Änderungen werden in DSMLv2 transformiert
- DSMLv2-Dokumente werden für jeden PSP in SPML-Dokumente „verpackt“
- SPML-Request wird an PSP gesendet, dort:
  - SPML-Request wird „ausgepackt“ -> DSMLv2-Request
  - DSMLv2 wird in LDAP- oder ADSI-Anweisungen für PST transformiert
  - Ergebnis der Anweisung wird in DSMLv2 transformiert
  - DSMLv2-Dokumente werden für RA in SPML-Dokumente „verpackt“
- SPML-Response wird an RA gesendet
  - wenn Fehler auftritt bleibt das Dokument in der Queue

# Aufteilung der Komponenten (1/2)



- **Keine Einwirkungen auf das AD-System**
- **Production ready**

## Aufteilung der Komponenten (2/2)



- Mehr Möglichkeiten durch ADSI (Group-Policies, etc.)
- PSP muss auf dem AD installiert werden
- Auf unserer Roadmap

# Erfahrungen





# Lessons learnt

- **Vorteil gegenüber täglichem Gesamtabgleich, da Änderungen zeitnah provisioniert werden**
- **Für Recovery und Synchronisierung wurden zwei zusätzliche SPML-Capabilities von DAASI spezifiziert und entwickelt (für Selbstheilung von Fehlern):**
  - **Identify-Capability:**
    - Versucht ein Objekt anhand dessen Daten zu identifizieren
    - z.B. für den Fall, dass im AD manuell ein Eintrag gelöscht und wieder angelegt wurde, also sich die ObjectID geändert hat
  - **Sync-Capability:**
    - Erhält Daten von RA und aktualisiert Daten des PST auf gleichen Stand
    - Also ein Gesamtabgleich

# Lessons learnt

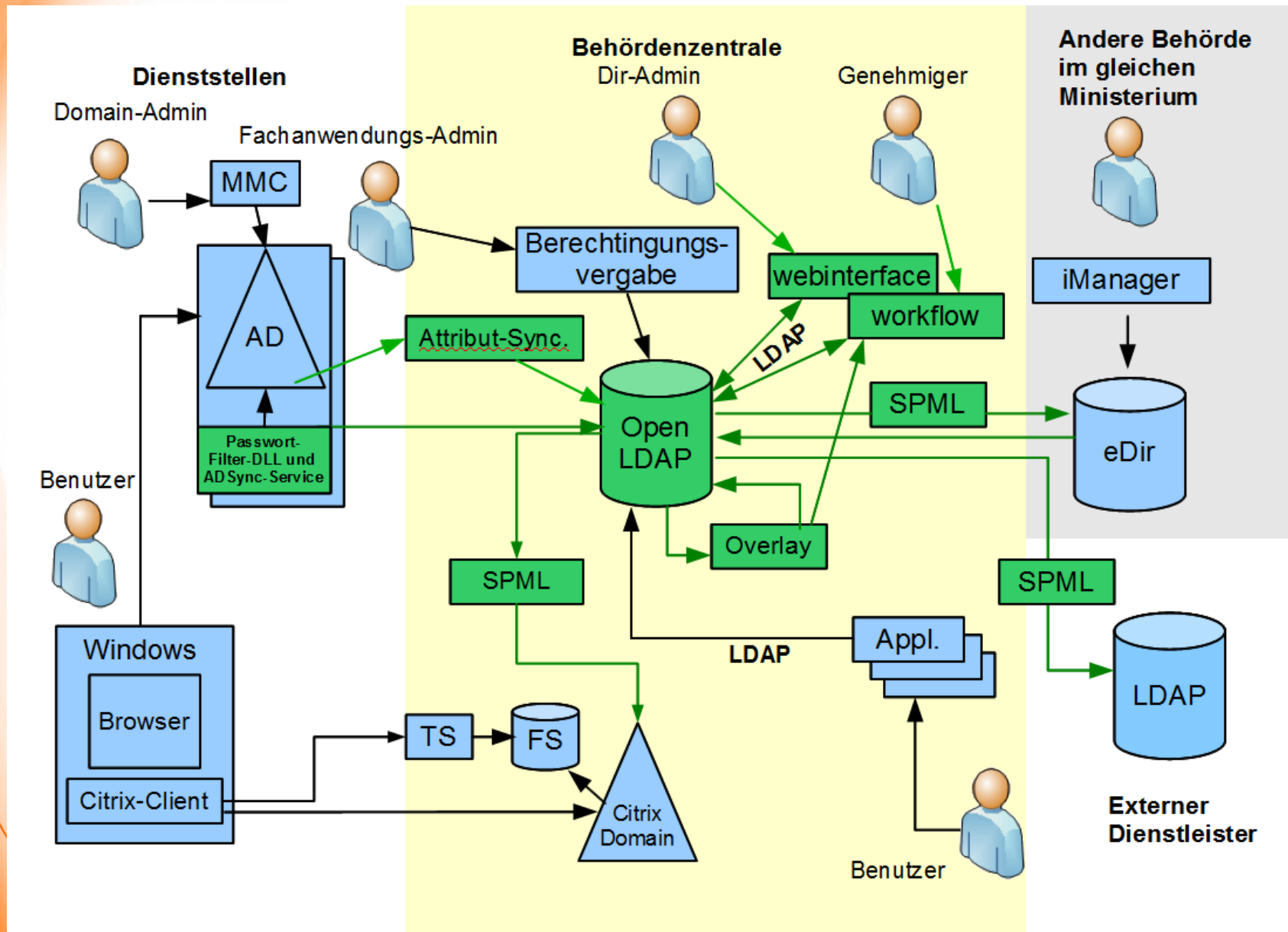
- **XSL-Transformationen vor Senden der Daten bei RA und vor Empfang an PSP haben sich bewährt z.B. für**
  - **Attribut-Mappings**
  - **Ignorieren spezieller Einträge**
  - **Hinzufügen von konstanten Daten**



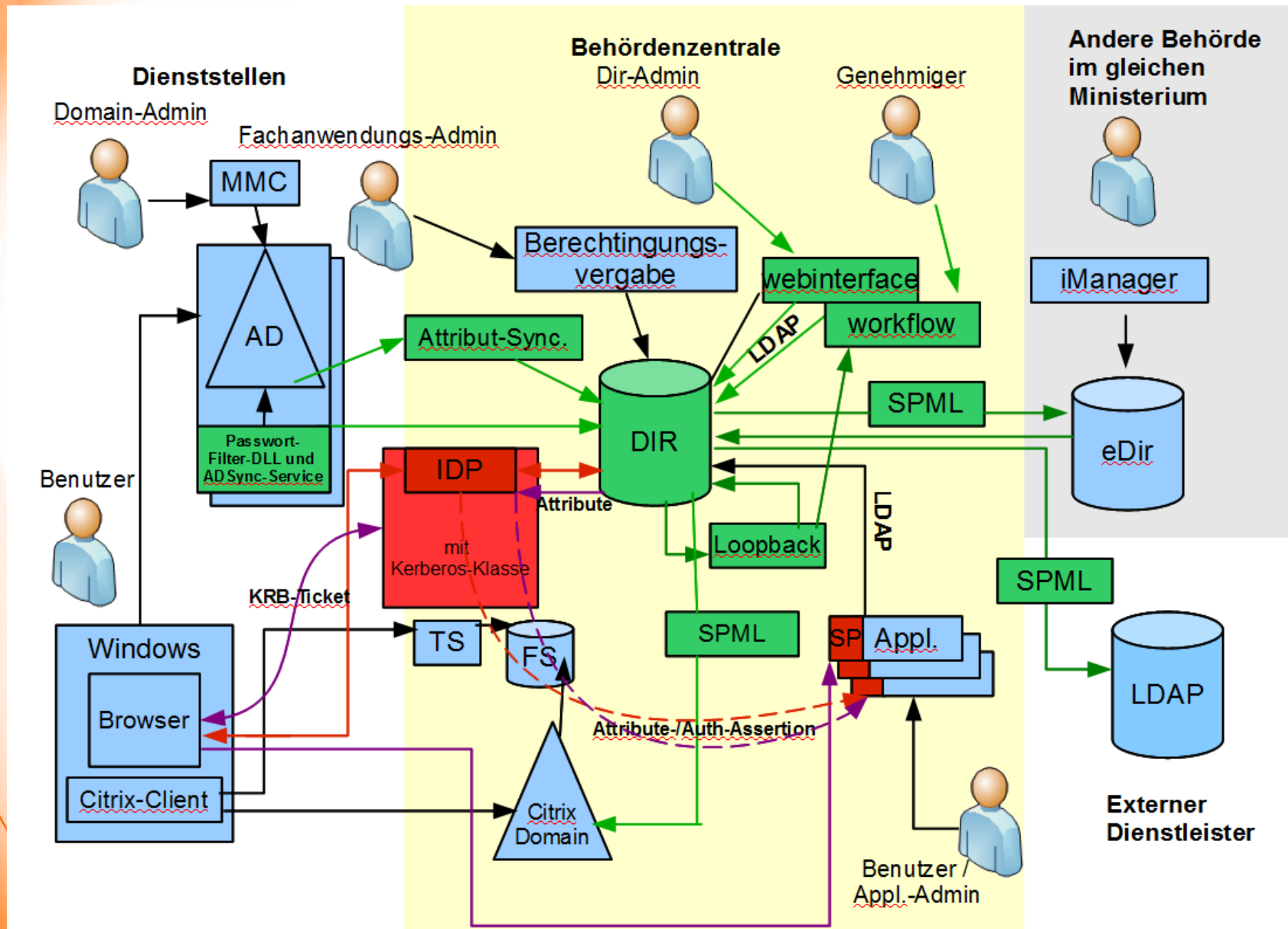
# Projektbeispiel

- Eine existierende auf proprietäre Software basierende Identity-Management-Lösung sollte mit Open-Source-Software nachgebaut werden
  - komplexe Synchronisierungsmechanismen
  - komplexe Berechtigungsattributvergabe
- Zusätzlich sollte WebSSO mithilfe von Shibboleth realisiert werden
  - ein IdP, der an den zentralen Verzeichnisdienst angeschlossen wird
  - mehrere SPs, die verschiedene zentrale Fachanwendungen schützen
- Schließlich sollte durch Integration der Windows-Kerberos-Authentifizierung die Notwendigkeit der Synchronisierung von Passwörtern entfallen

# Projektbeispiel



# Projektbeispiel



# Vielen Dank für Ihre Aufmerksamkeit!

## ➤ Fragen ?

### ➤ Kontakt und weitere Informationen:

**DAASI International GmbH**

**Europaplatz 3**

**D-72072 Tübingen**

**Web: <http://www.daasi.de>**

**Mail: [info@daasi.de](mailto:info@daasi.de)**

### ➤ Meet LDAP-Experts @ LDAPCon 2011

**October 10-11, 2011 in Heidelberg**

**[www.ldapcon.org](http://www.ldapcon.org)**