

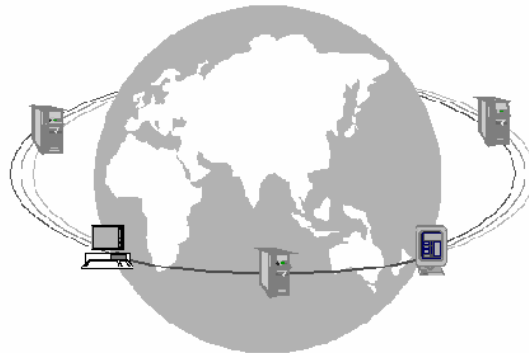
Erfahrungen bei der Implementierung mit Staging-Tabellen von HISSOS/SVA und dem Dokumentenmodell von DirXML



Jena

Bauhaus-Universität

Weimar



Erfurt



Ilmenau

Technische Universität Ilmenau
Universitätsrechenzentrum
Jörg Deutschmann

Gliederung

HIS-Schnittstelle::Staging-Tabellen

Dokumentenmodell – XML-basierte Integration

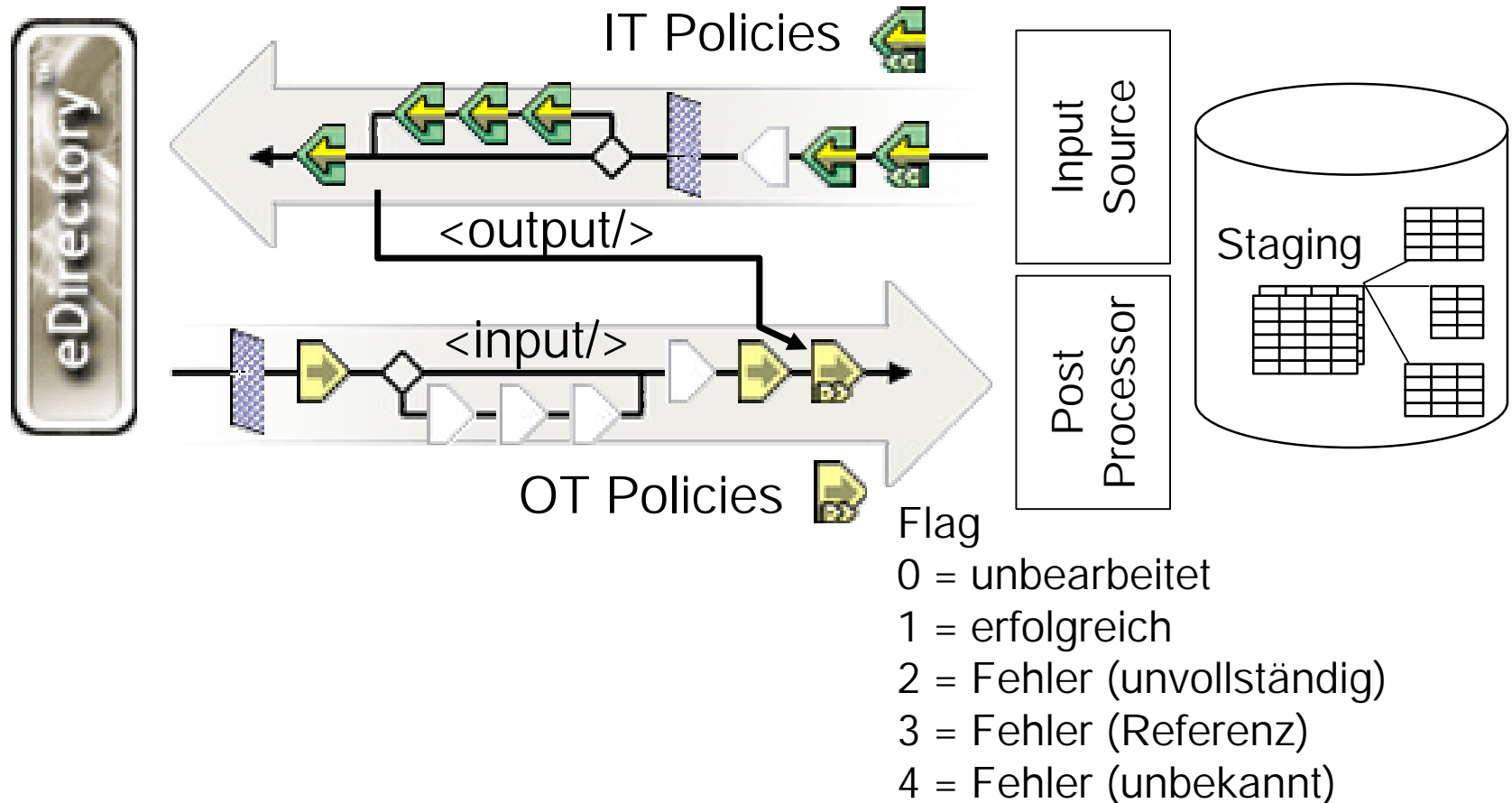
Erfahrungen von der Arbeit der Entwicklungsgruppe

Zusammenfassung und Ausblick

Aktuelles Szenario

Meta Directory – Codex

Beispiel: Codex-HIS



Input Source

- Implementierung des **InputSource-Interface**
 - Beim Start des Treibers aufgerufen: **init()**-Methode mit den Parametern
 - Parameter-String zur Angabe des Datenbanktreibers, der Datenbank-URL, der Authentisierungsinformationen, der Tabellennamen etc.
 - Tracer-Objekt für die Ausgabe von Logging-Informationen
 - Beim Vorliegen von Output-Daten aufgerufen: **getInputFiles()**-Methode mit dem Rückgabewert
 - Array von Input-Files
- Implementierung des **Datenbankzugriffs**, inklusive der SQL-Statements mit Hilfe von Java Database Connectivity (JDBC)
- **Erzeugung** der XML-Dateien als XML-Zwischenformat „von Hand“ oder mit Hilfe des Document Object Model (DOM)

Input Source

- **XML-Zwischenformat:** Die InputSource-Erweiterung generiert aus der Abfrage der Staging-Tabellen ein XML-Dokument, das an die Input Transformation Policies gegeben wird.

```
<?xml version="1.0" encoding="UTF-8"?>
<delimited-text>
  <record>
    <add class-name="person" event-id="1" timestamp=...>
      <add-attr attr-name="id">
        <value>1</value>
      </add-attr>
      <add-attr attr-name="surname">
        <value>Bach</value>
      </add-attr>
      <add-attr attr-name="givenname">
        <value>Otto</value>
      </add-attr>
    </add>
    ...
  </record>
</delimited-text>
```

PostProcessing

- Implementierung des **PostProcessor-Interface**
 - Beim Start des Treibers aufgerufen: **init()**-Methode mit den Parametern
 - Parameter-String zur Angabe des Datenbanktreibers, der Datenbank-URL, der Authentisierungsinformationen, der Tabellennamen etc.
 - Tracer-Objekt für die Ausgabe von Logging-Informationen
 - Beim Vorliegen von Output-Daten aufgerufen: **nextOutputFile()**-Methode mit den Parametern
 - Output-File
- **Parsing** der XML-Datei mit Hilfe von Simple API for XML (SAX) bzw. Document Object Model (DOM)
- Implementierung des **Datenbankzugriffs**, inklusive der Generierung der SQL-Statements mit Hilfe von Java Database Connectivity (JDBC)

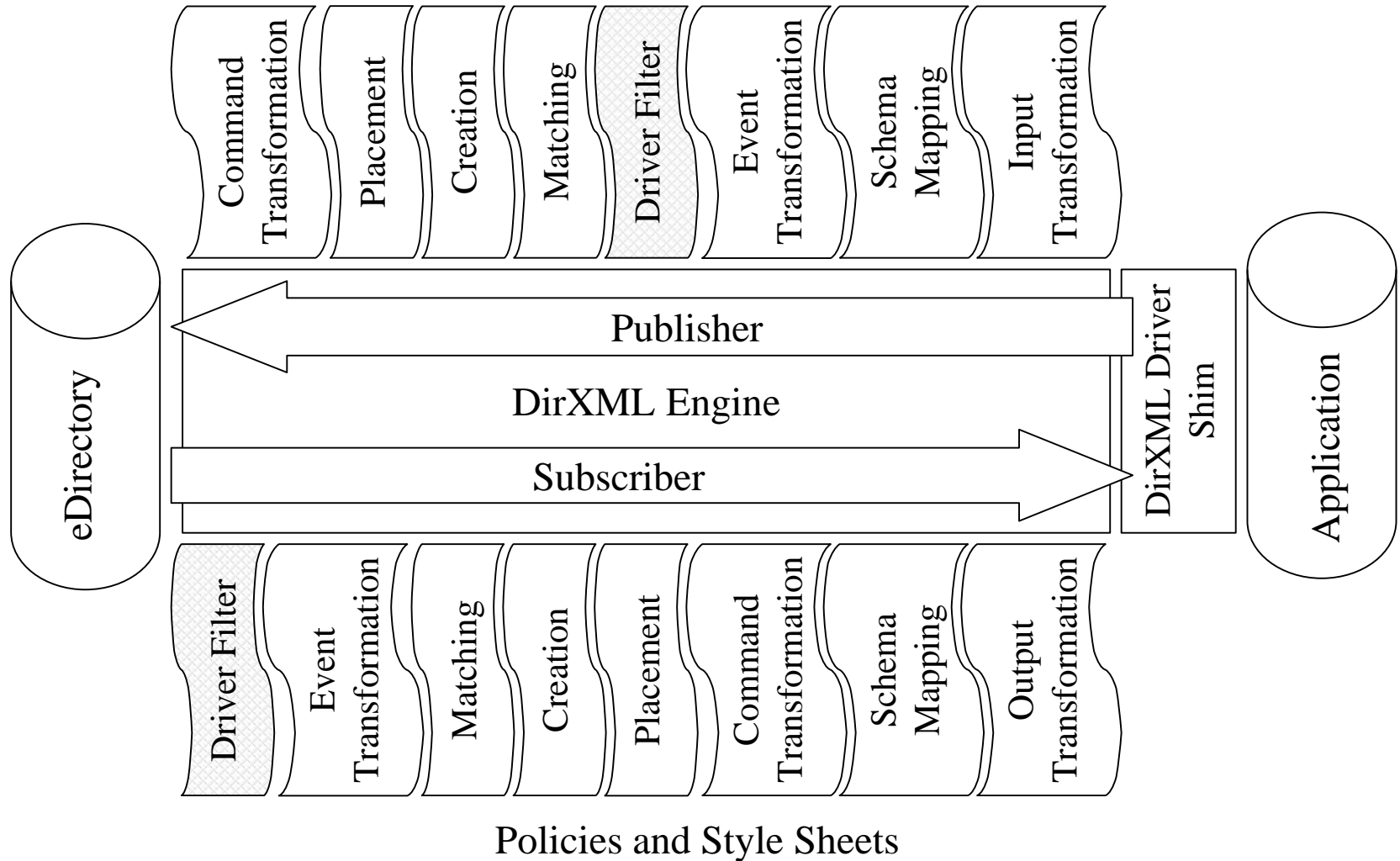
PostProcessing

- **XML-Zwischenformat:** Das Output-Dokument des Publisher-Kanals wird in den Output Transformation Policies zum Input-Dokument des Subscriber-Kanals.

```
<?xml version="1.0" encoding="UTF-8"?>
<nds dtdversion="2.0" ndsversion="8.x">
  <input>
    <modify class-name="status-person" dest-entry-id="5">
      <modify-attr attr-name="flag">
        <remove-all-values/>
        <add-value>
          <value>1</value>
        </add-value>
      </modify-attr>
    </modify>
  </input>
</nds>
```

DirXML Architektur

Policies and Style Sheets



Technologien im Umfeld des Projekts

- Verzeichnisse
 - X.500, Lightweight Directory Access Protocol (LDAP), eDirectory
- Relationale Datenbanken
 - Informix, MS Access, später evtl. PostGres
- Java APIs
 - Java Database Connectivity (JDBC), Simple API for XML (SAX), Document Object Model (DOM), XSLT API
- eXtensible Stylesheet Language Transformations (XSLT) + XPath, XLink
- eXtensibel Markup Language (XML)
- Document Type Definition (DTD)
 - NDS.DTD
- Nsure Identity Manager 2 (DirXML)
 - Policy Builder

Entwicklungsgruppe Codex

- 2 Mitarbeiter je Einrichtung „frei gestellt“
- Initialer Workshop und Coaching-Modell
 - 2-3 Wochen eigenständige Arbeit, 2-3 Tage Coaching
- Setup in den ersten zwei Wochen
 - Entwicklungsumgebung
 - XMLSPY Professional Edition version 2004 rel. 3
 - VMware Workstation 4.5.1
 - Image mit Win2000 Pro, eDirectory 8.7.3, Nsure Identity Manager 2 (DirXML), iManager 2.0.2
 - Concurrent Versions System (CVS) + Tortoise 1.6.1
- Implementierung gemäß Aufgabenverteilung
 - Erfurt: Codex-Auth
 - Weimar/Ilmenau: Codex-HIS
 - Jena: Codex-PICA, Codex-Self/Event

Entwicklungsgruppe Codex: Probleme und Erfolge

- Großes Detailwissen gleich zu Beginn der Entwicklung notwendig
- Auftauchen aus den Tiefen der Details
- Concurrent Versions System (CVS) ersetzt nicht die Kommunikation erleichtert aber den Austausch von Dokumenten wesentlich
- Abstände zwischen den Coaching-Terminen zum Teil etwas knapp
- Modell initialer Workshop und Coaching hat sich bewährt
- Bedürfnis für ein mal pro Woche Treffen/Videokonferenz
- Beförderung der Kommunikation zwischen den Einrichtungen
- Bisheriges Implementierungsergebnis: Prototyp mit unausgereifter Fehler- und Sonderfallbehandlung
- Feldtest für die Staging-Tabellen steht noch aus
 - Problem: Fachlichkeit für Stellen- und Studierendenverwaltung
 - Erarbeitung von Testfällen mit den Dezernaten für Personalangelegenheiten und Akademischen Angelegenheiten

Zusammenfassung und Ausblick

