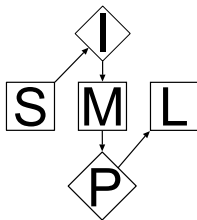


# Wochenbericht

Michael Schneidt

Studienprojekt A: SIMPL

17. Januar 2010



Geplante Aufgaben und Tätigkeiten

Erledigte Aufgaben und Tätigkeiten

Übersicht

Benötigte Arbeitszeit/Aufwände

Gewonnene Erkenntnisse

Neues für SIMPL

Verwendete Quellen/Literatur

Identifizierte Aufgaben für die kommende Woche

Zu klärende Fragen

# Geplante Aufgaben und Tätigkeiten

- ▶ Feinentwurf
- ▶ SIMPL Core Web Services
- ▶ Client für das Eclipse SIMPL Core Plug-In

# Erledigte Aufgaben und Tätigkeiten

## Übersicht

- ▶ Feinentwurf
- ▶ SIMPL Core Web Services
- ▶ Client für das Eclipse SIMPL Core Plug-In

# Erledigte Aufgaben und Tätigkeiten

## Investierte Arbeitszeit/Aufwände

- ▶ Feinentwurf / 12h
- ▶ SIMPL Core Web Services / 6h
- ▶ Client für das Eclipse SIMPL Core Plug-In / 10h

# Gewonnene Erkenntnisse

## Gewonnene Erkenntnisse

### ► SIMPL Core Web Services

- Fehler (Tomcat: stdout\_\*.log): org.apache.axis2.AxisFault: Service {http://webservices.core.simpl.org/}DataSourceService was not found in the WSDL
  - Die WSDL die mit ws-gen.exe aus der JAVA-Klasse generiert wird, ist wohl fehlerhaft und wird gar nicht verwendet. Die WSDL wird zwar erzeugt, aber die Schemata können nicht abgerufen werden und der Service somit nicht ohne Fehler genutzt werden.
  - Lösung: tools.jar aus dem JDK in den Classpath kopieren, dann wird alles korrekt zur Laufzeit erzeugt. Die WSDL muss auch nicht mehr mit ws-gen.exe generiert und in der Java-Klasse annotiert werden. :-)
- Komplexe Objekte wie z.B. HashMaps werden jetzt serialisiert, als String übertragen und beim Client deserialisiert
  - Vorteil: sehr einfach! ;-)
  - Nachteil: mehr Daten zu übertragen / Datentyp muss auf der Gegenseite bekannt sein / Kein Klartext der Daten
  - Damit keine Zeichensatzprobleme entstehen werden die Strings BASE64 enkodiert

# Neues für SIMPL

- ▶ simpl-core-deploy.zip
  - ▶ SIMPL Core Deployment Archive. Muss unter "Tomcat 6.0\webapps" entpackt werden und nistet sich in ODE ein. Enthält den SIMPL Core, die Web Services und alle abhängigen Libraries.

# Verwendete Quellen/Literatur



# Identifizierte Aufgaben für die kommende Woche

- ▶ Problem mit Libs lösen, evtl. mit einem ClassLoader
- ▶ DataSourceService Methoden: `getDataSourceTypes`, `getDataSourceSubTypes`, `getDataSourceLanguages`

# Zu klärende Fragen (1)

1. Macht es Sinn die SDOs auch serialisiert zu übertragen?  
Problem bei großen Datenmengen im SDO? Evtl. mal testen.
2. Wie soll der Security Service funktionieren, vor allem wie soll das SSO garantiert werden. Wo könnte eine Transformation von Informationen nötig sein? Evtl übertragen wir alle Sicherheitsinfos in SAML und XACML und müssen andersartige Infos erst dahin transformieren.
3. Wie werden die Verbindungen zu den Datenquellen verwaltet (Connection Pool), macht das dann der Security oder der Datasource Service? Oder neuer Service der deployt werden muss? Tomcat oder andere Service Bus können da evtl schon was. (Michael Reiter)
4. Wie werden die funktionalen Anforderungen (Interface, Treiber, WSDL, JDBC) an den Strategy Service übermittelt?