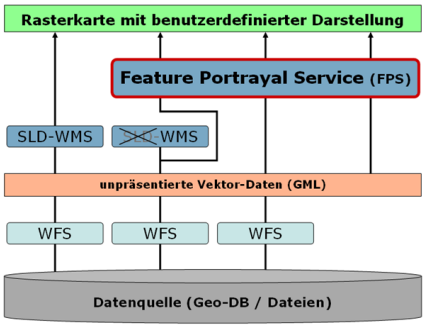
Hochschule für Technik und Wirtschaft Dresden

Fakultät Geoinformation

Implementierung eines Feature Portrayal Service

Fundamentale Voraussetzung für alle professionellen Kartendienste ist die Möglichkeit zur Kontrolle der grafischen Präsentation. Aus diesem Grund besteht ein starker Bedarf an der visuellen Darstellung von Geodaten. Diese transformiert rohe Informationen in wichtige und entscheidungstragende Werkzeuge. Der **Feature Portrayal Service (FPS)** hat die Aufgabe, diese Anforderungen umzusetzen und damit wesentlich zur **kartographischen Interoperabilität** in Geodateninfrastrukturen beizutragen.

Der Dienst entstammt dem internationalen Standard **SLD 1.1.0**. Es handelt sich um einen **Darstellungsdienst**, der die interoperable Nutzung von **Vektor-Daten** ermöglicht. Dieser **spezielle Web Map Service** hat die Aufgabe, Geodaten von **beliebigen Web Feature Services** sowie rohe **GML-Daten** in einer gewünschten grafischen Ausprägung zu präsentieren. Dabei ist der jeweilige Nutzer in der Lage, eigene Darstellungen zu definieren und auf einen beliebigen Datenbestand anzuwenden.




```
<UserDefinedSymbolization
SupportSLD="1"
UserLayer="1"
UserStyle="1"
RemoteWFS="1"
InlineFeatureData="1"
RemoteWCS="0"/>
```

Ob ein Web Map Service grundsätzlich die Funktionalitäten eines Feature Portrayal Service zur Verfügung stellt, ist den **Capabilities** des Dienstes zu entnehmen. SLD 1.1.0 erweitert die üblichen WMS-Eigenschaften um das Element **UserDefinedSymbolization**, dessen optionale Attribute Aufschluss über die Art des SLD-WMS liefern.

Der Dienst

Um eine Auswahl bezüglich eines geeigneten Open Source Frameworks zur Implementierung des Dienstes zu treffen, waren intensive **Software-Untersuchungen** notwendig.

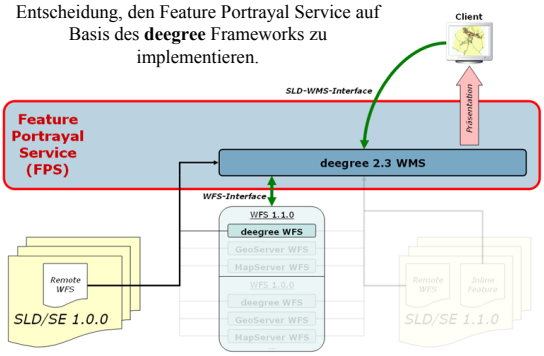
Ein Feature Portrayal Service muss in allen denkbaren Kombination arbeiten können. Daher muss der Nutzer per **SLD** eigene Darstellungen (**UserStyle**) auf beliebige fremde Web Feature Services (**RemoteWFS**) anwenden können. Diese Datenquellen werden in benutzerdefinierten Ebenen (**UserLayer**) angegeben.

	deegree 2.3	MapServer 5.6.1	GeoServer 2.0.1
SLD 1.0.0	✓	✓	✓
SLD 1.1.0	✗	✓	✗
UserStyle	✓	✓	✓
UserLayer	✓	✗	✓
WFS 1.0.0	✗	✗	✓*
WFS 1.1.0	✓*	✗	

* wird nur eingeschränkt unterstützt

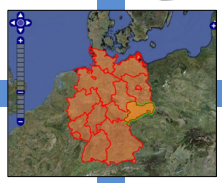
Während MapServer aufgrund der Resultate als Feature Portrayal Service nicht in Frage kam, lagen deegree und GeoServer beim Stand der Implementierung recht nah beieinander. Entscheidender Nachteil von GeoServer war ein derzeit eingebauter Timeout, der jeden Darstellungsprozess vorzeitig abbricht, weshalb die Funktionsweise als FPS nicht wirklich getestet werden konnte.

Als Ergebnis der Untersuchungen fiel die Entscheidung, den Feature Portrayal Service auf Basis des **deegree** Frameworks zu implementieren.



Schlussendlich stand die Erkenntnis, dass die Implementierung auf Open Source Basis zum gegenwärtigen Zeitpunkt nicht ohne **Defizite** durchgeführt werden konnte. Der als Feature Portrayal Service eingesetzte deegree WMS kann lediglich deegree WFS der Version 1.1.0 darstellen.

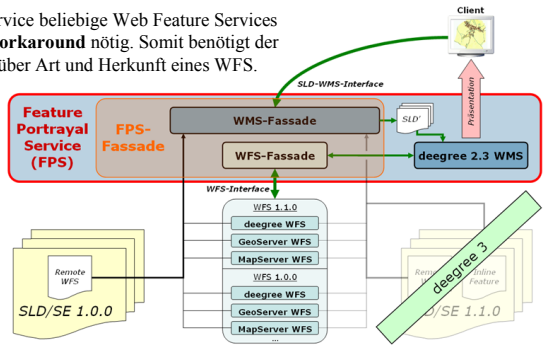
Implementierung: deegree FPS

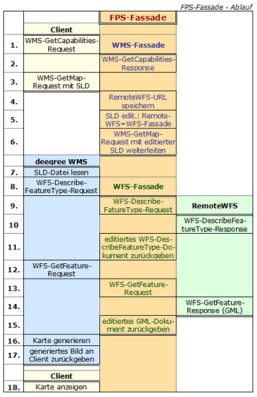


Workaround: FPS-Fassade

Da der Feature Portrayal Service beliebige Web Feature Services präsentieren soll, war ein **Workaround** nötig. Somit benötigt der Nutzer keine Kenntnisse über Art und Herkunft eines WFS.

Die **Fassade** ist ein Entwurfsmuster aus der objektorientierten Softwareentwicklung und verfolgt den Zweck, spezialisierte Anforderungen an Schnittstellen zu bedienen.





Im Workaround kommuniziert der deegree FPS mit einer WFS-Fassade, welche die notwendigen Schnittstellen anbietet. Diese Architektur hat den Vorteil, dass dieser Teil der FPS-Fassade einen ganz normalen WFS darstellt, wobei intern zunächst die Daten der eigentlichen RemoteWFS besorgt und dahingehend verändert werden, dass auch der deegree FPS mit diesen umgehen kann.

Der fehlende Support von SLD 1.1.0 resultiert aus dem derzeitigen Entwicklungsstand von deegree. Da zu erwarten ist, dass sich dies in kommenden Veröffentlichungen ändern wird, wäre ein komplexer Workaround für diesen Sachverhalt mit einem unverhältnismäßig großen Aufwand verbunden. Auf die Unterstützung von InlineFeature (Einbindung von GML) kann verzichtet werden, da es sich um ein optionales Element handelt.

Die FPS-Fassade kann also **jeden WFS** symbolisieren.

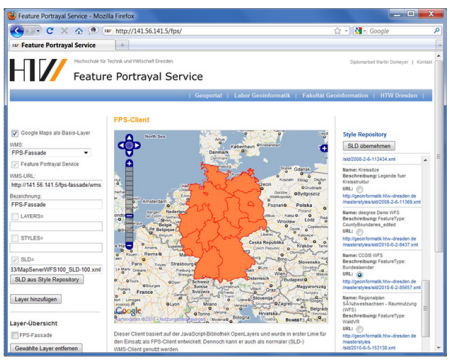
FPS-Client

```
http://141.56.141.5/fps-fassade/wms?
SERVICE=WMS&VERSION=1.1.1&REQUEST=GetMap
&SRS=EPSG:4326&BBOX=5.8,47.2,15.1,55.1
&FORMAT=image/png&WIDTH=500&HEIGHT=425
&SLD=http://www.htw-dresden.de/~s58783/
MapServerWFS100_SLD-100.xml
```

Um den Feature Portrayal Service optimal nutzen zu können, ist ein entsprechender Client sinnvoll. Dieser generiert den entsprechenden **GetMap-Request** und verringert damit den Aufwand für den Nutzer.

Der entwickelte Client basiert auf der JavaScript-Bibliothek **OpenLayers**. Zur Auswahl stehen sowohl der **deegree FPS** als auch die **FPS-Fassade**. Da die korrekte Funktionsweise als Feature Portrayal Service nur mit der FPS-Fassade gegeben ist, sollte diese bevorzugt eingesetzt werden.

Es besteht die Möglichkeit, SLD-Dokumente aus dem **Style Repository** der HTW Dresden einzufügen. Der Nutzer kann dort eigene SLD-Files hochladen. Jeder Stil erhält eine URL, die anschließend den jeweiligen Diensten mitgeteilt werden kann. Der Vorteil des Ablegens der Dokumente im Style Repository liegt darin, dass diese dann einer breiten Masse an Benutzern zugänglich gemacht werden. Außerdem verfügt nicht jeder Anwender über einen öffentlich zugänglichen Webserver, um seine eigenen Darstellungsdefinitionen abzulegen.



Diplomarbeit: Martin Domeyer
vorgelegt: Juli 2010

1. Gutachter: Prof. Dr.-Ing. F. Schwarzbach
2. Gutachter: Frau Prof. Dr.-Ing. M. Müller