

# Workshop „Intelligente Portale“

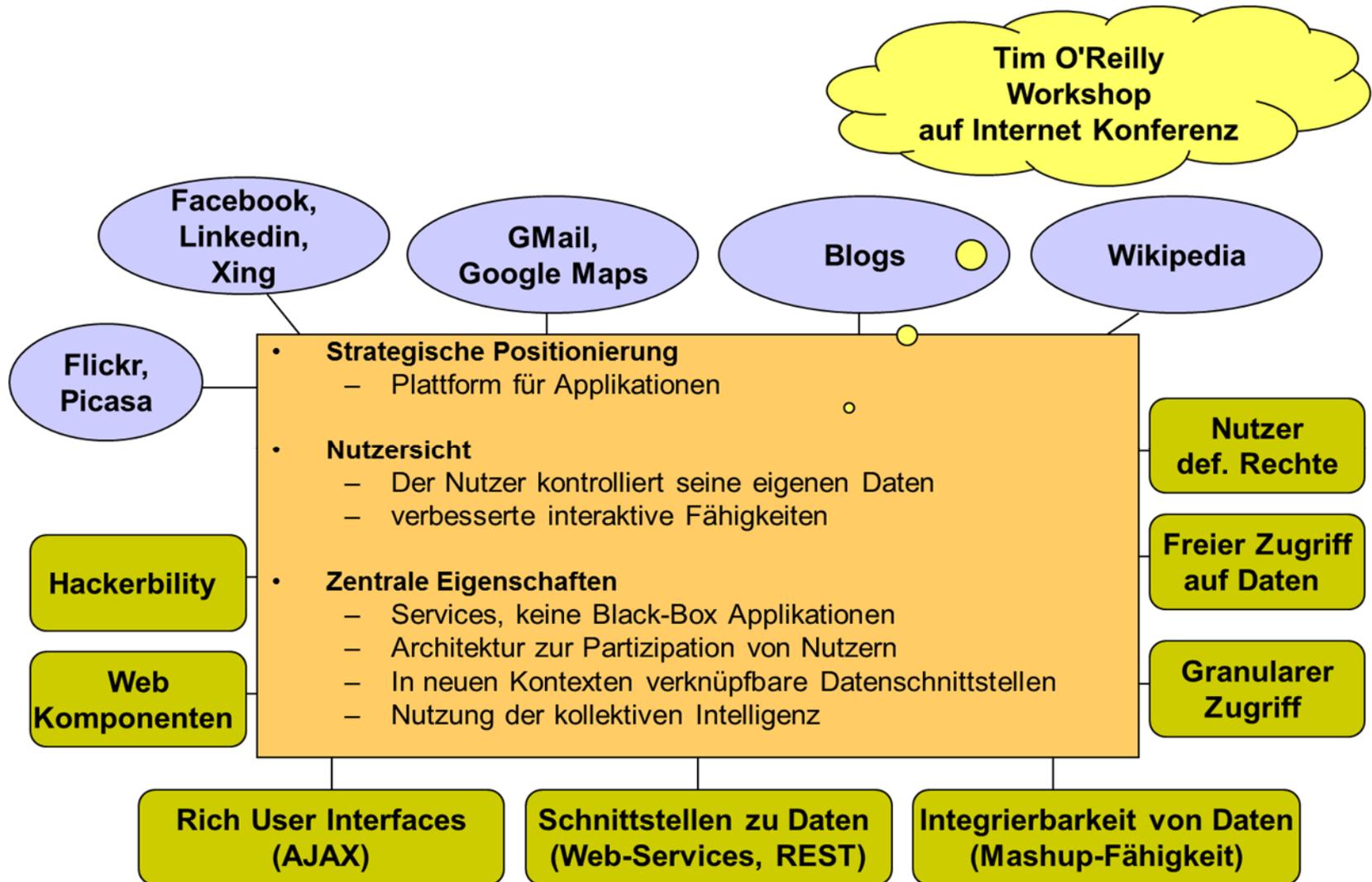
Ein dienstorientierter Zugang zu Informationsquellen

Institut für Angewandte Informatik (IAI)



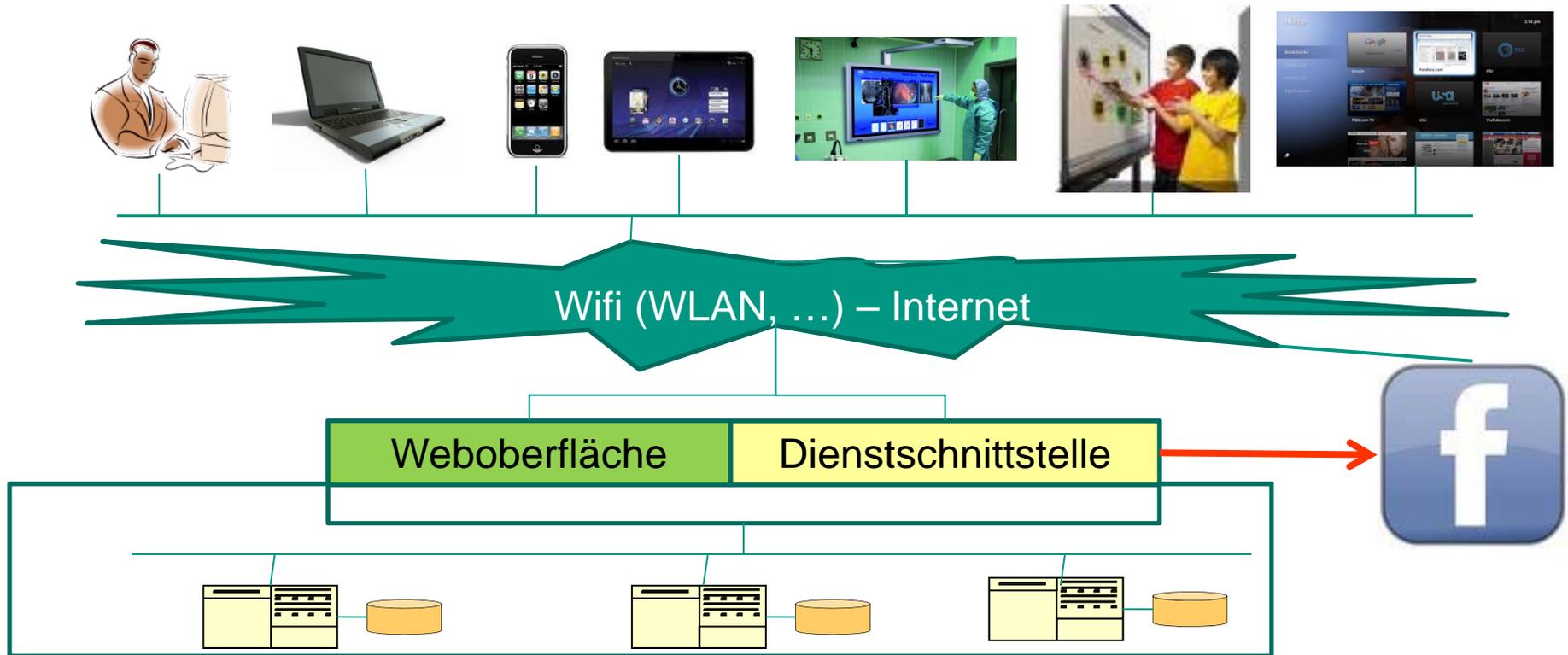
- Einführung / Motivation
- Ein Grundkonzept für moderne, intelligente Portale
- Dienste und Portale
- Zusammenfassung

# Ausgangspunkt - Vision des Web 2.0



- Klassische Web-basierte Systeme verlieren Nutzer!
  - Viele Benutzer – insbesondere die Jüngeren - besuchen nur noch selten kleinere, dedizierte Webangebote. Vielmehr halten Sie sich in sehr großen „Social Networks“ auf oder besuchen größere Dienstleistungsportale im Web
    - **Diese sozialen Netzwerke und Portale aggregieren als Mehrwert für den Nutzer Dienste oder Informationen aus verschiedenen Quellen und bringen Sie aktiv zu ihren Nutzern**
  - Andere Nutzer interagieren über mobile Plattformen mit nativen Apps mit solchen Portalen (nicht mehr über klassische Weboberflächen)
  - **Web-basierte Systeme müssen sich aktiv neue „Vertriebskanäle“ suchen, um die Verluste auszugleichen oder ihre Nutzerbasis auszubauen**

# Strategie größerer Internet-basierter Portale



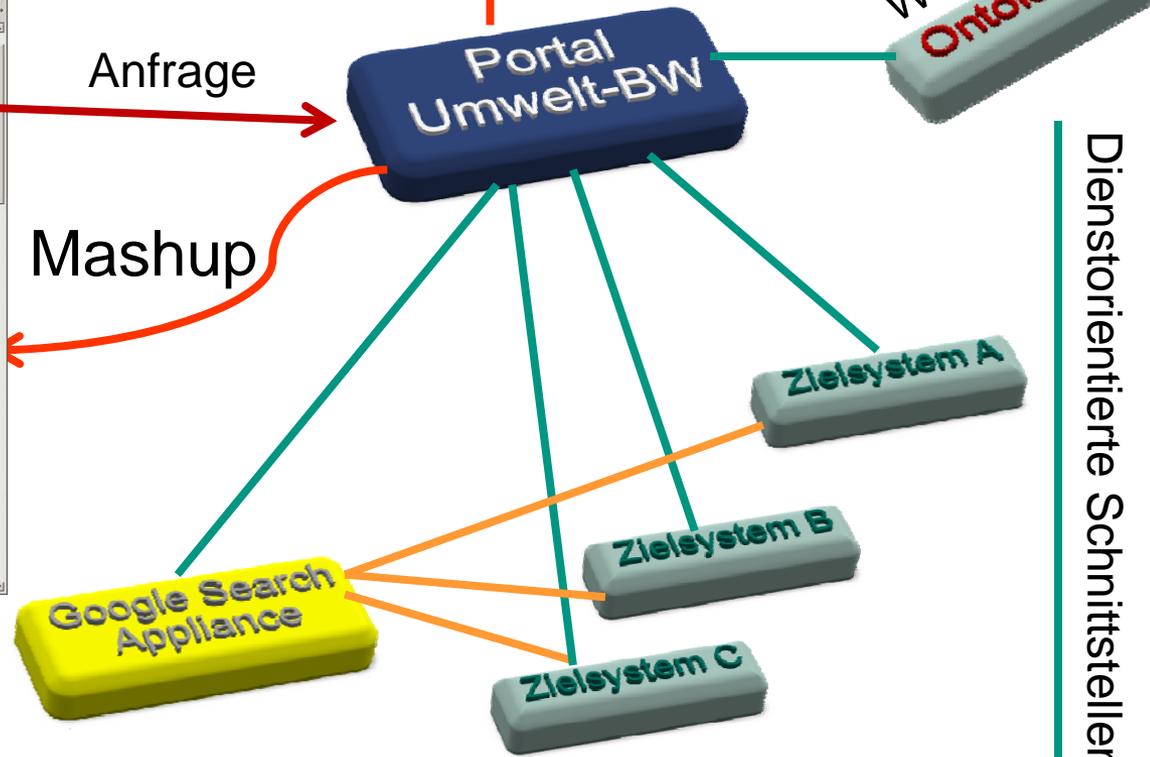
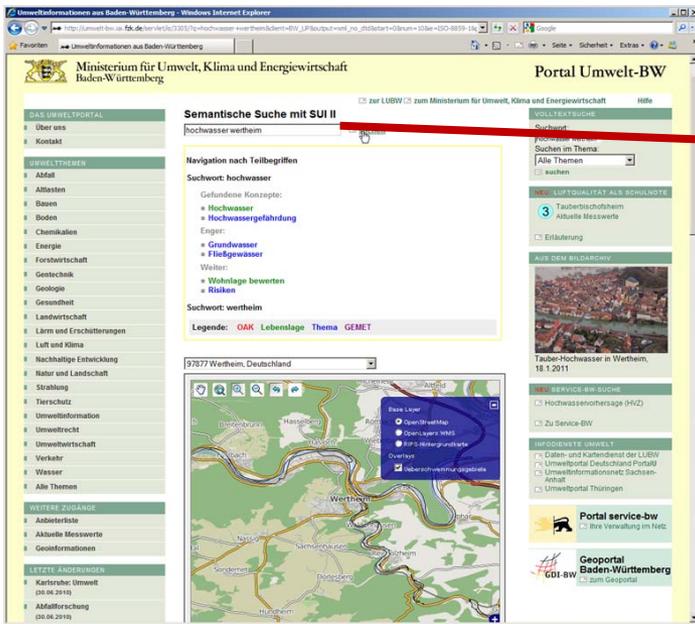
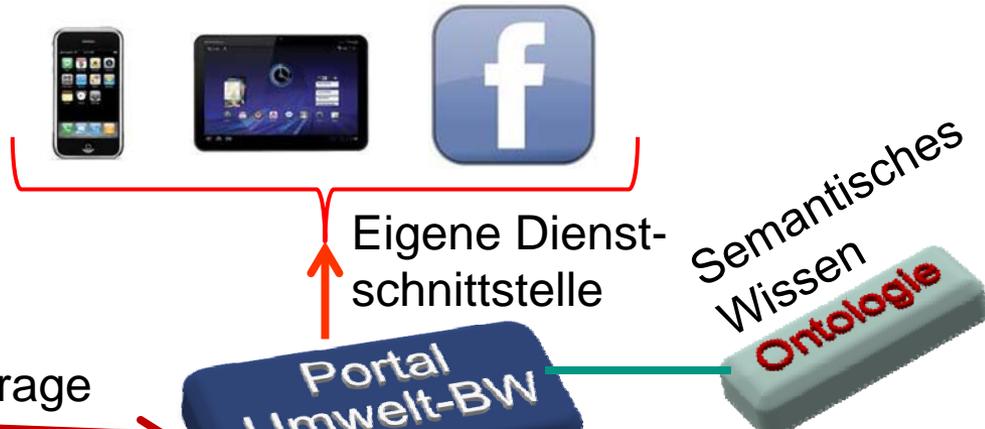
- Bereitstellung von Dienstleistung nicht nur über Weboberflächen sondern auch über dienstorientierte Schnittstellen
  - Erlaubt Zugriff sowohl über Web als auch über Apps und klassische Anwendungen
  - Dies ermöglicht auch die Integration mit anderen Systemen (z.B. Facebook)
- Zusammenfassen eigener Dienstleistungen in größeren Portalen

# Was bedeutet das für Portale, wie Umwelt-BW?

- Vor Web 2.0
  - Portale = Linksammlungen (Kataloge) von anderen Webangeboten
  - Ergänzt durch Metadaten (Metadatenkataloge)
  - Mit Volltextsuche als Index-basierte Erschließung von einzelnen Webseiten der Fremdangebote
  
- Mit Nutzung von Web 2.0 und Semantic Web
  - Rückwärtige Datensysteme sollten über dienstorientierte Schnittstellen an das Portal angebunden werden
  - Verknüpfung der Daten im Portal als „**Mashups**“ unter Nutzung von Rich Client Nutzerschnittstellen (Google Maps oder Legato)
  - Eine semantische Beschreibungen der Daten der einzelnen Datensysteme erlaubt dabei eine flexible Verknüpfung von Daten im Portal
  - Portale sollten alle verfügbaren Daten wieder über dienstorientierte Schnittstellen für Dritte bereitstellen
    - Um andere Vertriebswege für die eigenen Informationen zu öffnen

# Mögliches Konzept für ein „Intelligentes Portal“

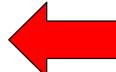
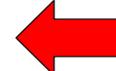
Anfrage:  
Hochwasser Wertheim



Ergebnis einer Suchanfrage

s. Vortrag  
T. Schlachter,  
Demonstration

# Wichtige Teilaspekte des intelligenten Portals

- Vorverarbeitung eingehender Suchbegriffe
  - Präzisere Erfassung des „thematischen Kontexts“  **Hochwasser**
    - Die zentrale Aufgabe der Ontologie
    - Auflösung schwammiger oder synonymmer Begriffe zu stark definierten fachlichen Konzepten (Ontologie)
    - Hierarchische Einordnung des „thematischen Kontexts“ und Rückgabe der Beziehungen zu anderen Konzepten
  - Auflösung des räumlichen und zeitlichen Kontexts  **Wertheim** einer Suchangabe
- Abbildung von (parametrisierten) Aufrufen der Dienstschnittstellen angeschlossener Systeme auf (Raum, Zeit, Themenkontext) einer Anfrage
- Dienstorientierte Anbindung anderer Informationssysteme
- Mashup-Bildung

# Wo werden welche Teilaspekte erläutert?

- Thorsten Schlachter behandelt in seinem Vortrag die Gesamtarchitektur des „Umwelt-BW 3.0“ Portals
  - Er erläutert dabei
    - Aspekte der Vorverarbeitung
    - Formale Beschreibung und Aufruf der Zielsysteme
    - Sowie die Mashup-Bildung
- Ulrich Bügel erläutert in seinem Vortrag Aufgabe und Funktionsweise des Ontologiesystems
- Dieser Vortrag geht im Folgenden auf generelle Aspekte der Dienstschnittstellen näher ein



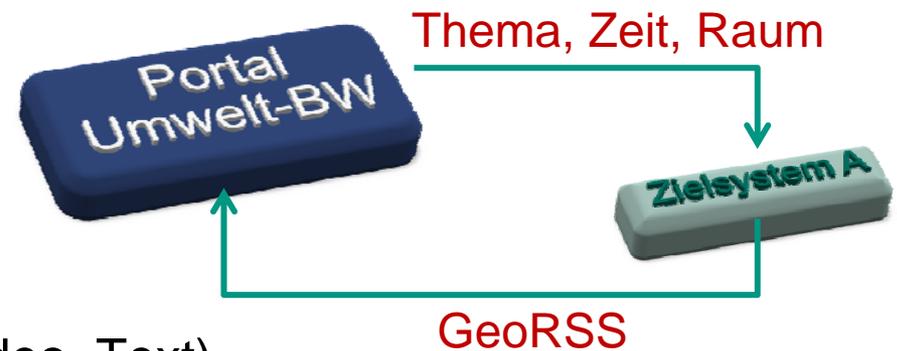
s. Vortrag  
T. Schlachter



s. Vortrag  
Ulrich Bügel

# Anbindung rückwärtiger Informationssysteme

- Idealerweise über dienstorientierte Schnittstellen
- Die Mashups in verschiedenen Kontexten ermöglichen
  - Nach Thema
  - Zeitlich
  - Räumlich
  - Nach Medienart (Bild, Grafik, Video, Text)
- Hierzu müssen Aufrufschnittstellen nach Möglichkeit gemäß diesen Angaben parametrisierbar sein
- Und die zurückgelieferten Ergebnisse Mashup-fähig sein
  - Dedizierte, Fachsemantik beschreibende XML-Formate
  - Oder standardisierte Formate (wie RSS, GeoRSS, etc.)
  - Notfalls HTML-Schnipsel ohne Rahmendesign
  - An Stelle von fertigen HTML-Seiten



- Die Erfahrung zeigt, dass die Entwicklung dedizierter dienstorientierter Schnittstellen aufwendig ist
- Deshalb: 80 / 20 Regel
  - 80 % der Systeme über Standardschnittstellen anbinden
    - Viele Systeme unterstützen bereits einfache Schnittstellen, z.B. OpenSearch-konforme Suchschnittstellen mit Rückgabeformaten, wie
      - RSS, GeoRSS oder MediaRSS
    - Wenn nicht, sind solche Schnittstellen einfach zu programmieren
    - Oder man lebt mit URL-basierten Schnittstellen, die HTML selbst zurückgeben und schneidet gewollte Teile aus
  - Entwicklung dedizierter Schnittstellen nur bei Systemen, wo es sich lohnt ( 20 %)
    - Beispiel: UIS-Sachdatensystem (Cadenza) in Baden-Württemberg

# Welche Art von Implementierung wählen?

- Klassische Web-Services oder eher auf dem REST-Prinzip basierte Dienste verwenden?
  - Mobile Geräteplattformen unterstützen derzeit nur elementare XML-Schnittstellen (Parser)
  - Web-Services erzeugen hier beim Parsen für Apps einen hohen Aufwand (bzgl. Performanz + Speicherbedarf)
  - REST-Schnittstellen unterstützen oft mehrere Rückgabeformate
    - Neben XML z.B. auch JSON
    - JSON ist ein sehr App-freundliches Format und eignet sich auch sehr gut für AJAX-Schnittstellen von Webanwendungen
    - REST-basierte Dienste haben weitere Vorteile
      - Können z.B. existierende HTTP-Infrastrukturen für Caching, Proxy- und Lastverteilung nutzen
      - Verwenden normale HTTP-Autorisierungsmechanismen
- Überall da, wo verschiedenartige Clients angebunden werden müssen, besser REST-basierte Dienstschnittstellen verwenden



# Welche Art von Dienstschnittstelle wählen?

- Für Portal-eigene Schnittstellen daher eher REST-basierte Dienstschnittstellen anbieten
- Für Verbindung mit rückwärtigen Datensystemen
  - Bei 80 % der Systeme
    - Sind dedizierte Schnittstellen zu aufwändig
    - Hier ist das Ziel generisch vorhandene, oder einfach zu implementierende URL-basierte Schnittstellen zu nutzen (oft REST-ähnlich)
    - Beispiele sind OpenSearch-konforme Suchschnittstellen mit Rückgabeformaten, wie
      - RSS, GeoRSS, etc. oder einfach nur HTML
  - Bei dedizierten Systemen (20 %)
    - Sind REST-basierte Dienstschnittstellen, oder
    - Aber auch klassische Web-Services – wenn bereits vorhanden – geeignet
    - Vorhandene Schnittstellen (egal welcher Art) kann man natürlich immer verwenden

# Zugriff auf Zielsysteme nach 80 % Regel

- Der größte Teil der Zielsysteme sollte ansprechbar sein, ohne sie in größeren Umfang ändern zu müssen

=>

Dazu muss für das Portal die Semantik der „Suchschnittstelle“ des Zielsystems beschrieben werden (und nicht umgekehrt)

- Formale Beschreibung der Schnittstelle des Zielsystems
  - Dies kann durch Zielsystem selbst oder z.B. im Portal erfolgen
  - Es muss eine Abbildung zwischen
    - Suchparameter des Zielsystems und
    - "semantische Suchinformationen", die Vorverarbeitung präzisiert, geben
  - Das Rückgabeformat muss auch beschrieben werden

# Beschreibung der Aufrufschnittstelle von Zielsystemen bei der 80 % Lösung

- Zur Beschreibung der (Such-)Aufrufschnittstelle eines Zielsystems benötigt man also

- Beschreibung möglicher Parameter der Aufrufschnittstelle

- Ortsbezug (Punkt, BoundingBox, Kreis, Namen von Verwaltungseinheiten)
- Zeitbezug (Zeitpunkt oder Zeitraum)
- Thematischer Bezug
  - Z.B. konkrete Einschränkung auf "Kategorie" von Informationen
  - Freie "Suchbegriffe", die weitere Einschränkungen bedeuten können, z.B. Name eines konkreten Objektes, etc.



- Spezifikation des Rückgabeformates

- Liste von Suchtreffern z.B. im RSS- oder GeoRSS-Format
- Tabelle mit Daten
- Karte oder Kartenlayer, die auf einer Karte angezeigt werden können
- Medientyp (wie Bild, Video, etc.)

# OpenSearch-basierte Schnittstellen

- URL-basierte, REST-ähnliche generische Suchschnittstellen
- Definieren parametrisierte URLs, über die eine Suche in einem Websystem angestoßen werden kann
- Das Rückgabeformat kann flexibel definiert werden
  - Der OpenSearch-Standard erlaubt es, dass Systeme nahezu beliebige Formate zurückgeben können
    - Gängig sind Feed-basierte XML-Formate, wie RSS oder das Atom-Format, die auch zu GeoRSS, etc. angereichert werden können
    - Formate erlauben erlauben Erweiterung um weitere XML-Namespaces
    - Erweiterung durch OpenSearch-Namespace für Metainformationen, wie Anzahl zurückgegebener und vorhandener Ergebnisse, etc.
- Das OpenSearch-Description Format erlaubt es, OpenSearch-Schnittstellen maschinenlesbar zu beschreiben



s. Vortrag  
T. Schlachter

**OpenSearch ist daher eine sehr gute generische Lösung für die 80 % Anbindung von rückwärtigen Informationssystemen**

- Organisationen, wie Umweltbehörden, müssen ihre Internet-Präsenz in größeren Portalen **bündeln und um dienstorientierte Schnittstellen erweitern**
  - Um für Nutzer interessant zu sein
  - Und neue potentielle Nutzungsszenarien über bereitgestellte Dienstschnittstellen zu erschließen
- Beim Anschluss von anderen Fachsystemen (Zielsystemen) sollte man nach **der 80 / 20 Regel** verfahren
  - Die meisten Systeme sollte man über einfache, möglichst generische Schnittstellen, **z.B. OpenSearch**, anbinden
  - Die **Entwicklung dedizierter Serviceschnittstellen** lohnt sich nur für einen **kleinen Teil Semantik-reicher Systeme**
- Als dienstorientierte Schnittstellen **eignen sich besonders REST-basierte Schnittstellen**