



# Semantic Web

---

Seminar Internet-Technologie

Seminarleitung:  
Prof. Dr. Herbert Göttler,  
Thomas Gottron

Seminararbeit von Florian Altherr WS 2005/06



# Inhalt

---

- Begriffsklärung
- Die Idee des Semantic Web - Motivation und Vision
- Das Schichtenmodell
  - Basistechnologien: Unicode, URI
  - XML
  - RDF und RDFS
  - Ontologien, OWL und seine Vorgänger
  - Logic, Proof, Trust
- Ausblick / Anwendungsbeispiele

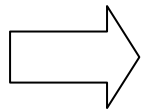


# Begriffe

---

**Semantik =**

Teilgebiet der Linguistik, das sich mit Sinn und Bedeutung von sprachlichen Zeichen beschäftigt



Im Semantic Web: **Maschinen** sollen **Bedeutung von Daten verstehen** durch Anreicherung der im Web vorhandenen Daten mit Metainformationen



# Begriffe

---

## **Semantisches Netz =**

- Formales Modell von Begriffen und Relationen
- Repräsentation von Wissen
- Darstellung durch Graphen
  - Knoten = Begriffe
  - Kanten = Relationen



# Die Idee

---

- **Semantic Web** = Web der 2. Generation
- Erweiterung des WWW um Dimension „Bedeutung“
- Heute: Web-Daten für Maschinen **lesbar**
- Vision: Web-Daten für Maschinen **verständlich**  
(nach T.Berners-Lee)
- Keine Spezifikation, sondern Philosophie



# Motivation: Grenzen des WWW (1)

---

- Informationsflut im WWW – Problem der Filterung relevanter Informationen
- Für menschliche Nutzer konzipiert
- basiert auf HTML
- bedarf der Interpretation - von Maschinen nicht leistbar/ Kontext aus Begriffen und Zusammenhängen fehlt



## Motivation: Grenzen des WWW (2)

---

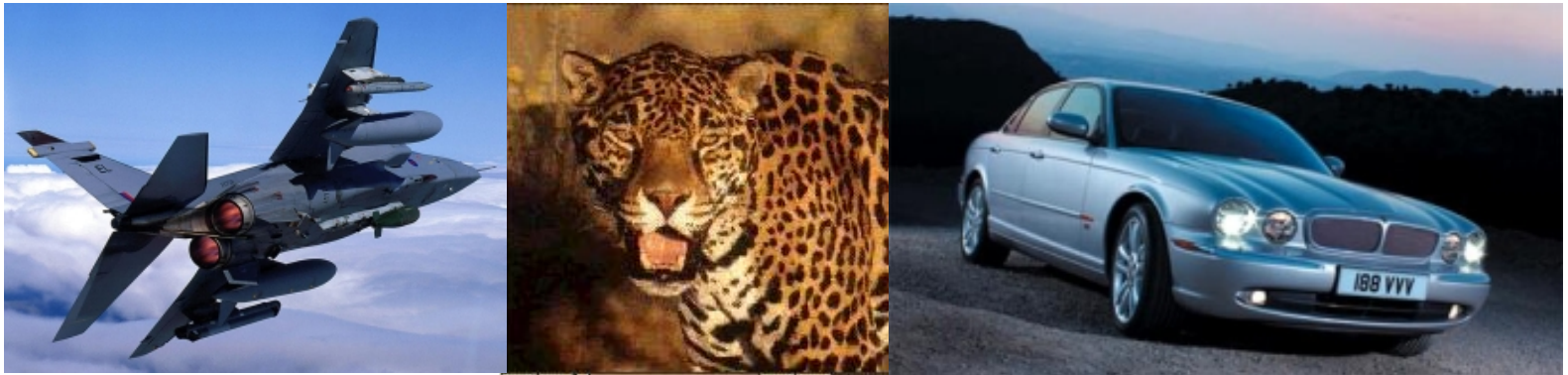
Für Maschinen ist :

- Teilinformation unbrauchbar
- Sinnerschließung schwierig
- Analogiebildung nicht möglich
- Keine Kombination von Information



"denn sie wissen nicht, was sie tun..."

- Beispiel: Jaguarkauf



- Suche nach Schlüsselwörtern etc., aber Problem bei Synonymen, Homonymen
- Kein Kontextverständnis

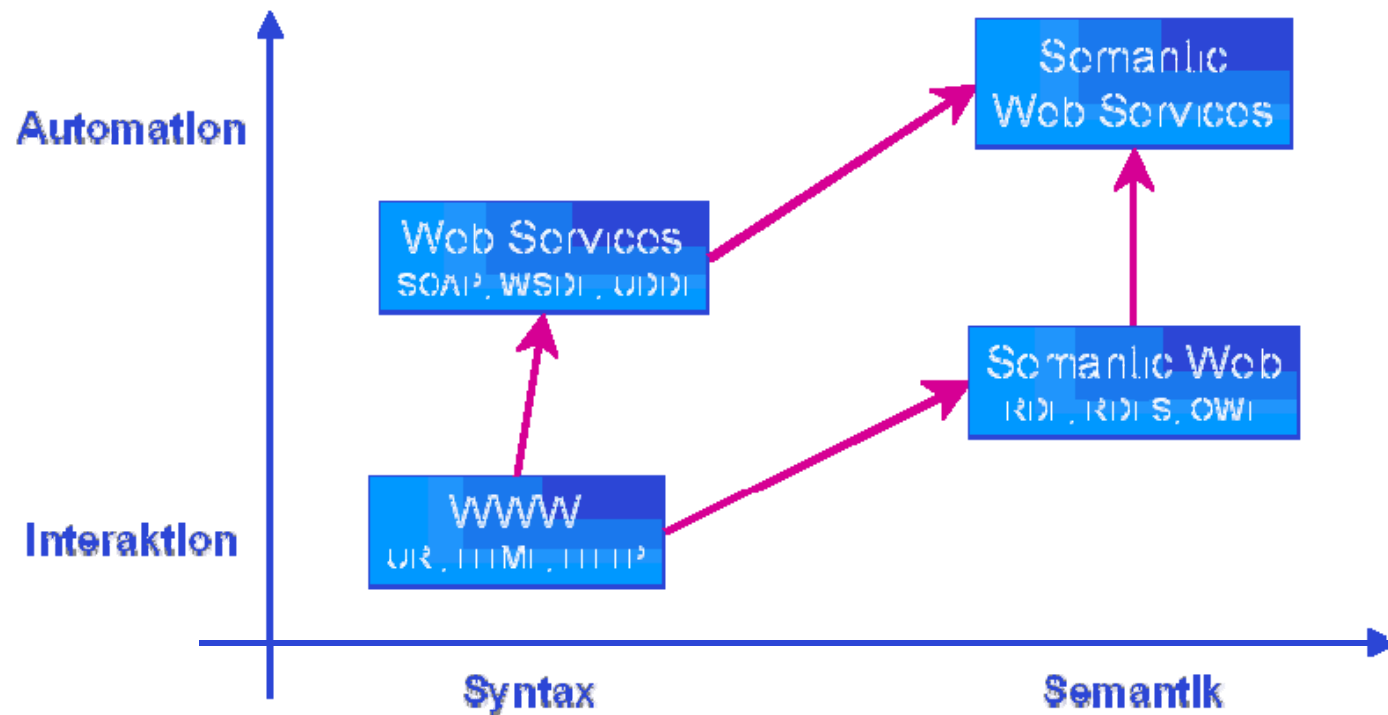
# Motivation und Vision

„The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation“



(“The Semantic Web lifts off” von Tim Berners-Lee und Eric Miller, W3C. ERCIM News No. 51, Oktober 2002)

# Motivation und Vision



Quelle: [DF]

Quelle: <http://www.jeckle.de/semanticWebServices/intro.html>



# Motivation und Vision

---

- Auftrag an persönlichen Agenten:  
“Buche eine 7-tägige Reise nach Nizza, suche nach einem Hotel, einem Flug und einem Mietauto, reisen kann ich zwischen dem 25.02. und dem 24.4. 2006, ich will möglichst wenig zahlen, aber per Auto oder Bus höchstens 10 Min. brauchen zum Strand“



# Motivation und Vision

---

Der „Agent“

- Kennt persönliche Vorlieben
- Bildet Wissen aus der Vergangenheit
- Verknüpft lokales Wissen mit entferntem (Web Services)



# Voraussetzungen

---

**Voraussetzungen** für Entwicklung des SW:  
(nach Berners-Lee)

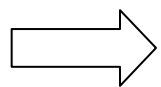
- gemeinsame Begriffswelt
- Informationslieferanten, die Web-Inhalte semantisch kennzeichnen
- Intelligente Softwareagenten, die darauf operieren



# Was ist nötig?

---

- Eindeutige Namen für Ressourcen (URIs)
  - Gemeinsames Datenmodell zur Beschreibung von Metadaten (RDF)
  - Gemeinsame Vokabularien
  - Inferenzregeln
- } Ontologien



SW ist metadatenbasierte Infrastruktur für Schlussfolgerungen im Web



# Semantic Web

---

- W3C legt Techniken fest und koordiniert Entwicklung
- Kerntechniken: URI, XML u. RDF unabhängig/früher entwickelt
- Verständnis von Beziehungen zwischen Datenressourcen
  - Nicht künstliche Intelligenz, sondern wohldefinierte Operationen auf wohldefinierten Daten
  - Aber Einsatz von Techniken aus AI (Logik, Ontologien, WBS)



# Warum nicht XML?

---

- Händlerdaten:

```
- <Weinladen>
  - <Laden>
    <Tel href="tel:+4961314711">06131/4711</Tel>
    <Oeffnungszeiten>Mo-Fr 19-16 Uhr</Oeffnungszeiten>
  </Laden>
  - <Lager>
    - <Italien>
      - <Chianti>
        <Weingut>...</Weingut>
        <Jahr>1995</Jahr>
        <PreisProFlasche>20</PreisProFlasche>
        <Lagermenge>25 Kisten</Lagermenge>
      </Chianti>
    </Italien>
    <Frankreich> </Frankreich>
  </Lager>
</Weinladen>
```



# Grenzen von XML

---

- Weinkonsument bestellt über seinen Agenten „eine Kiste Rotwein des Jahrgangs 1995“
- Agent sichtet Angebot, findet aber weder „Jahrgang“ noch „Rotwein“
- Erkennt nicht, dass Jahr und Jahrgang Synonyme sind und Chianti ein Rotwein ist



# Lösung (1)

---

- Agent weiß:

```
<owl:Class rdf:ID="Wein"/>
```

```
[- <owl:Class rdf:ID="Rotwein">
```

```
<rdfs:subClassOf rdf:resource="#Wein"/>
```

```
</owl:Class>
```

```
[- <owl:Class rdf:ID="Chianti">
```

```
<rdfs:subClassOf rdf:resource="#Rotwein"/>
```

```
</owl:Class>
```



## Lösung (2)

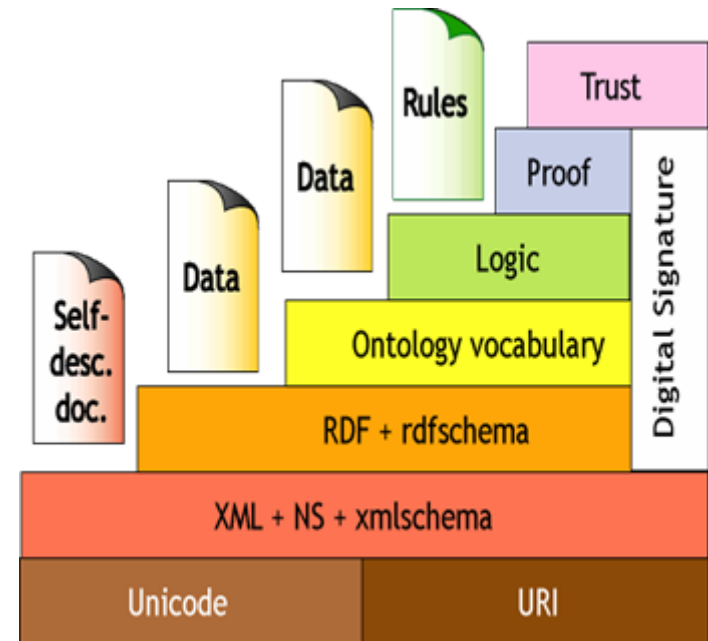
---

- Agent weiß:

```
[- <owl:DatatypeProperty rdf:ID="Jahrgang">  
<owl:equivalentProperty rdf:resource="#Jahr"/>  
<rdfs:domain rdf:resource="#VWein"/>  
<rdfs:range rdf:resource="&xsd;#integer"/>  
</owl:DatatypeProperty>
```

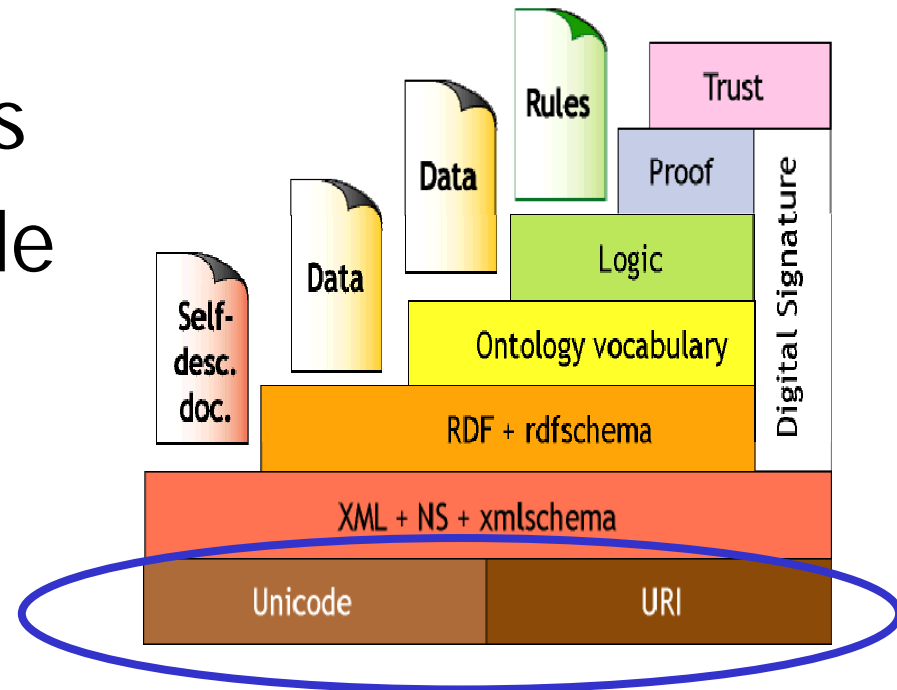
# Schichtenmodell des SW

- nach T. Berners-Lee
- 1998: „A roadmap to the Semantic Web“
- bisher nur untere 4 Schichten in Verwendung
- „Evolution, nicht Revolution“



# Unicode + Uniform Resource Identifier

- technologische Basis
- Kodierung in Unicode
- Adressierung durch URIs





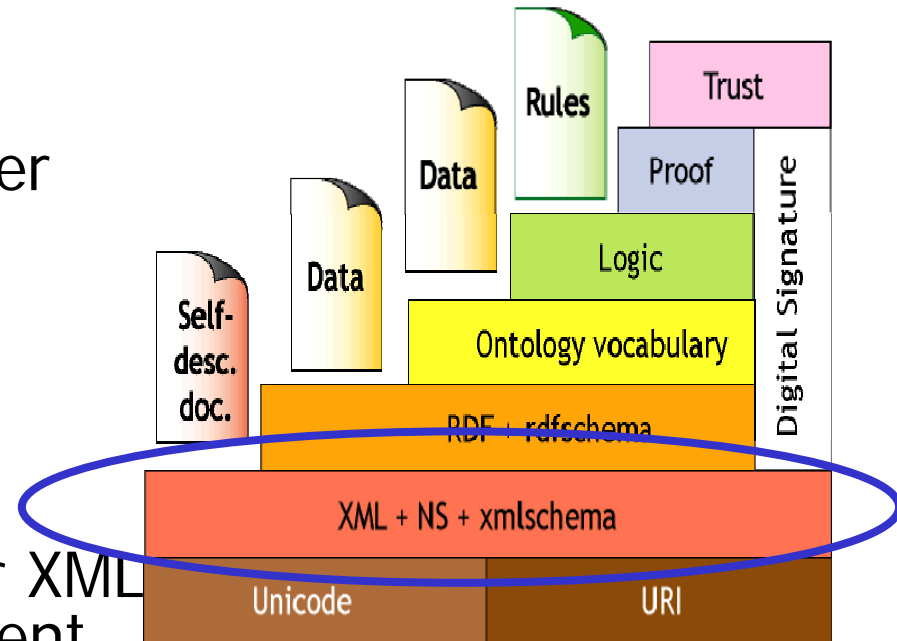
# Uniform Resource Identifier

---

- Adressierung von Ressourcen
- auch erster Schritt zur Beschreibung von Bedeutung
- Z.B. Stadt Paris über <http://www.paris.fr>
- Aber: dort evtl. keine explizite Beschreibung der Bedeutung

# XML, XML Schema + NS

- Bereitstellen von Inhalt
- Kodierung/Serialisierung der höheren Schichten
- Vorteil:
  - maschinelle Verarbeitung
  - Einbettung in HTML
  - leichte Transformation
- NS: Verwendung mehrerer XML Sprachen in einem Dokument ohne Namenskonflikte





# XML Schema

---

- Modellierungssprache, die XML um Datentypen, Strukturierungs- und Wiederverwendungsmöglichkeiten erweitert
- XML Schema oder DTD
  - definiert Vokabular
  - rein syntaktische Festlegung, keine Semantik



# Namespaces

---

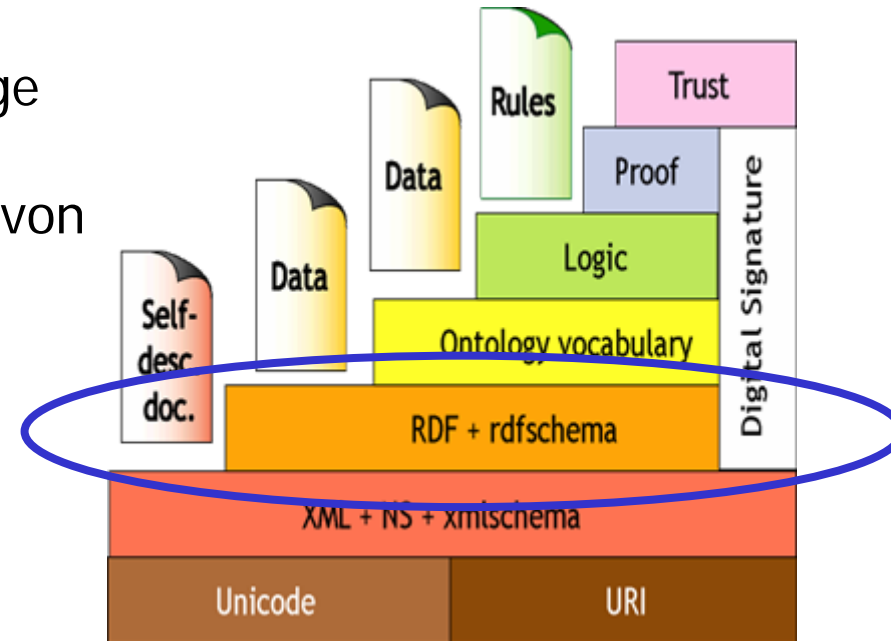
xmlns:rdf="<http://www.w3.org/1999/02/22-rdf-syntax-ns#>"

xmlns:rdfs="<http://www.w3.org/2000/01/rdf-schema#>"

xmlns:owl="<http://www.w3.org/2002/07/owl#>"

# Resource Description Framework

- erster „neuer“ Standard / Grundlage
- recht offenes Datenmodell
- Ziel: standardisierte Beschreibung von Semantik (maschinenlesbar)
- Aussagen in Form von RDF-Tripel:  
Ressource (Subjekt)  
Eigenschaft (Prädikat)  
Wert (Objekt)  
=> Subjekt hat Relation zu Objekt
- Darstellungsformen:  
Text, Graph, N3, RDF/XML





---

- Aussage / Information

„http://altherr.name/familie/index.html	Subjekt
---	---------

ist erstellt von	Prädikat
------------------	----------

Florian Altherr“	Objekt
------------------	--------



# RDF/XML

---

```
<?xml version="1.0" encoding="UTF-8" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description rdf:about="http://altherr.name/familie/index.html">
    <dc:title>Private Webseite der Familie Altherr</dc:title>
    <dc:creator>Florian Altherr</dc:creator>
  </rdf:Description>
</rdf:RDF>
```

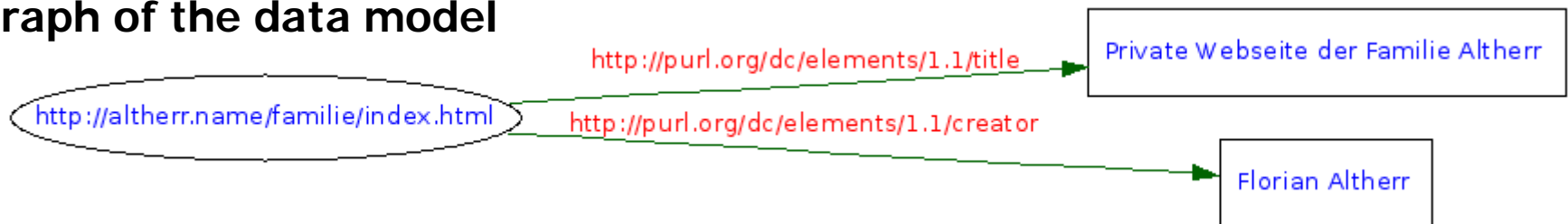
Vorteil: XML-Dokument, gut in andere Techniken (des W3Cs) integrierbar (WS)

# RDF

## Triples of the Data Model

Number	Subject	Predicate	Object
1	<a href="http://altherr.name/familie/index.html">http://altherr.name/familie/index.html</a>	<a href="http://purl.org/dc/elements/1.1/title">http://purl.org/dc/elements/1.1/title</a>	"Private Webseite der Familie Altherr"
2	<a href="http://altherr.name/familie/index.html">http://altherr.name/familie/index.html</a>	<a href="http://purl.org/dc/elements/1.1/creator">http://purl.org/dc/elements/1.1/creator</a>	"Florian Altherr"

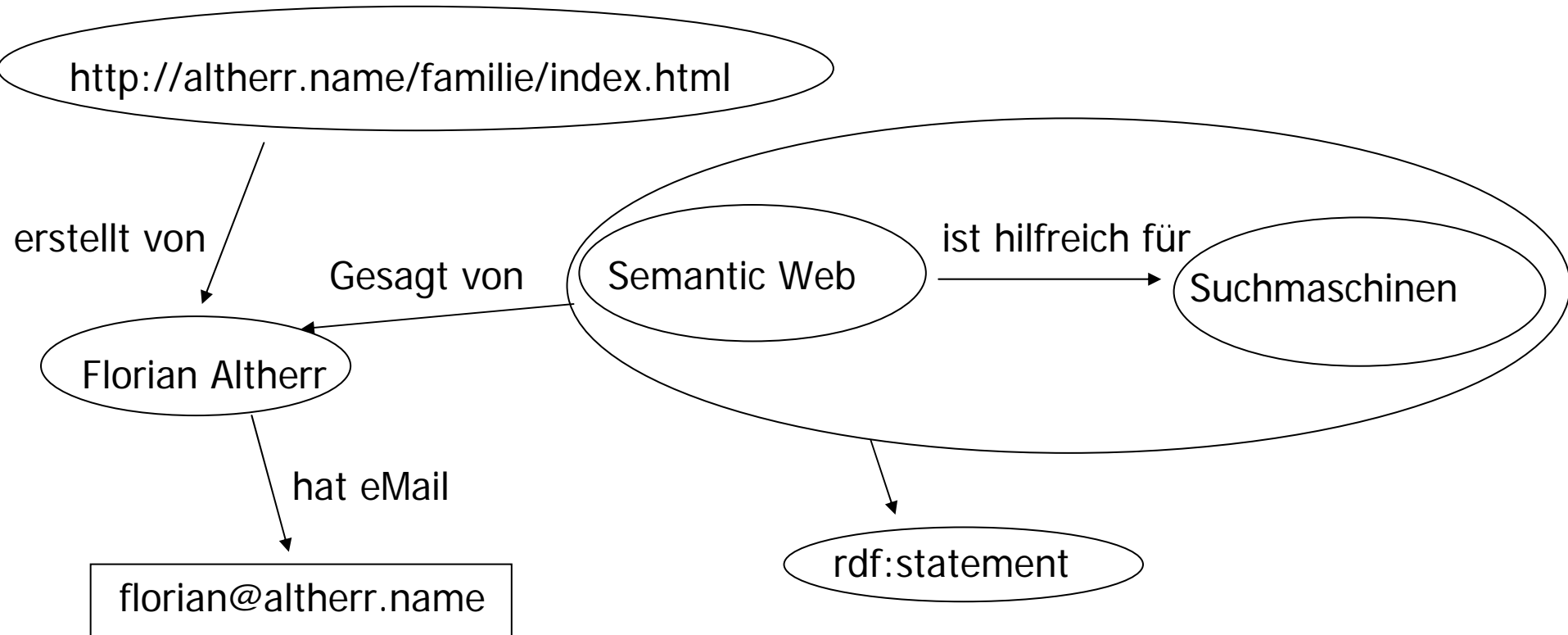
## Graph of the data model



Vorteil: bei komplexen Aussagen übersichtlicher

# RDF

## Aussagen über Aussagen:





# RDF

---

- beliebig viele Verknüpfungen
- URIs verwenden, außer für Literale
- „zitieren“ = Aussagen über Aussagen
- „blank nodes“
- gemeinsames Vokabular (z.B. Dublin Core)
- RDF versucht nicht alle möglichen Beziehungen inhaltlich zentral zu definieren



# RDF

---

## Beispiel:

```
<rdf: Description rdf:about="#Klaus" >  
  <vaterVon rdf:resource="#Chris" >  
</rdf: Description >
```

„Klaus ist der Vater von Chris“  
(einfaches semantisches Netz)



# RDF-Schema

---

- Beschreibung d. Eigenschaften von Ressource (RDF) reicht nicht
  - Bedeutung klarer durch Beschreibung ihres Verhältnisses zu anderen Ressourcen
    - => inhaltlich in Beziehung setzen / semantisch einordnen
  - RDFS klassifiziert Ressourcen und ihre Eigenschaften und verknüpft sie untereinander
    - Ressourcenklassen mit gemeinsamen Eigenschaften
    - Hierarchien (z.B. Subklassen)
    - Wertebereich einschränken
    - Domäne von Eigenschaften einschränken
- => Maschinen können Bedeutung von Ressourcen besser erkennen



# RDFS

---

## Beispiel: subClassOf

```
<rdfs: Class rdf:ID="#Vater" >
```

```
<rdfs: subClassOf rdfs:resource="#Mann"/>
```

```
</rdfs: Class >
```

Jede Instanz von Vater = Instanz von Mann



# RDFS

---

Beispiel: subProperty, range, domain

```
<rdf:Property rdf:ID="#vaterVon">
```

```
<rdfs:subPropertyOf rdf:resource="elterVon"/>
```

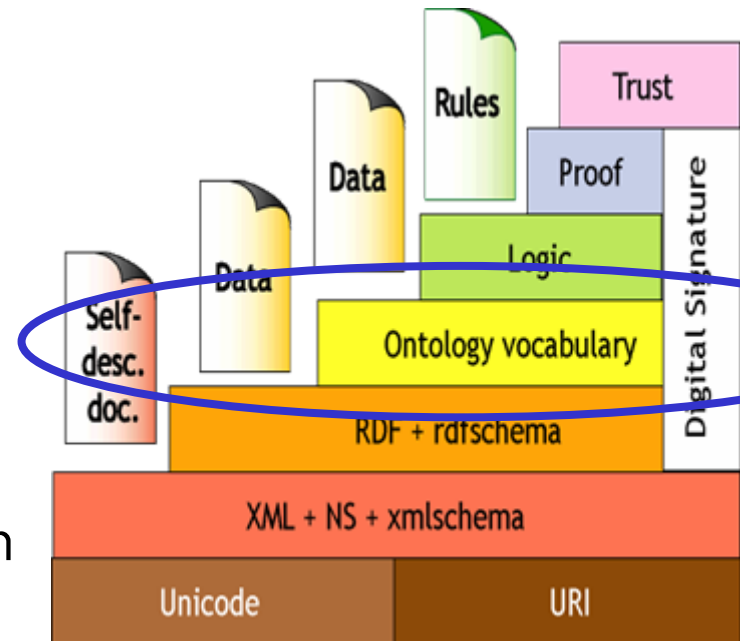
```
<rdfs:domain rdf:resource="#Vater"/>
```

```
<rdfs:range rfd:resource="#Mensch"/>
```

```
</rdf:Property>
```

# Ontologien

- Durch Zusammenfassen von Klassen mit festgelegten Eigenschaften ergibt sich Modell/Bild eines Wissensgebiets
  - SW: Modell, das Klassen von RDF-Ressourcen mit Eigenschaften u. Beziehungen definiert
  - Taxonomie als abgeschwächte Form der Ontologie (mit RDFS möglich)
- ⇒ Ontologiesprachen beschreiben noch detaillierter





# Ontologiesprachen

---

- Beschreibungssprache zum Erstellen und Publizieren von Ontologien
  - Aussagen, die Beziehungen zwischen Konzepten und logische Schlussregeln für sie festlegen
  - Frühe Sprachen:
    - SHOE
    - DAML (DARPA)
    - OIL (EU)
- } DAML+OIL ⇒ OWL (W3C)



# Web Ontology Language

---

- Klassen, Unterklassen, Eigenschaften, Untereigenschaften, Instanzen, Wertebereiche
- erweiterte Beschreibungsmöglichkeiten:
  - Eigenschaftsrestriktionen
  - Schnittmenge, Vereinigung, Komplement
  - Kardinalitäten (min,max,exakt)
  - Äquivalenzen
  - Charakter: symmetrisch, transitiv, funktional



# Web Ontology Language

---

- unterschiedliche Mächtigkeiten/Komplexitäten:
  - OWL-Lite: bietet kaum mehr als RDFS
  - OWL-DL: für praktischen Einsatz am geeignetesten (Verhältnis Komplexität - Entscheidbarkeit gewahrt)
  - OWL-Full: zahlreiche Sprachelemente, die Verhältnisse zwischen Klassen und ihren Eigenschaften sehr differenziert beschreiben



# Web Ontology Language

---

- Klassen betreffend
  - `<owl:class>`
  - `<owl:unionOf>`
  - `<owl:intersectionOf>`
  - `<owl:equivalentOf>`
- Eigenschaften betreffend
  - `<owl:Restriction>`
  - `<owl:allValuesFrom>`
  - `<owl:someValuesFrom>`



# Beispiel in OWL

---

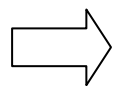
- [Buergeramt.owl](#)



# Erstes Fazit

---

- SW=basiert nicht auf „abgehobener“ künstlicher Intelligenz, sondern auf Einhaltung von Metadatenstandards
  - RDF, RDF Schema, OWL
- RDF=Datenmodell, triple-Prinzip, Notation in XML zur maschinellen Verarbeitung
- RDFS u. OWL erlauben Ontologiebildung (Konzepte versch.Ressourcentypen u.Beziehungen)
- Dezentral und offen



„schlaue“ Dokumente, deren Semantik erfasst werden kann



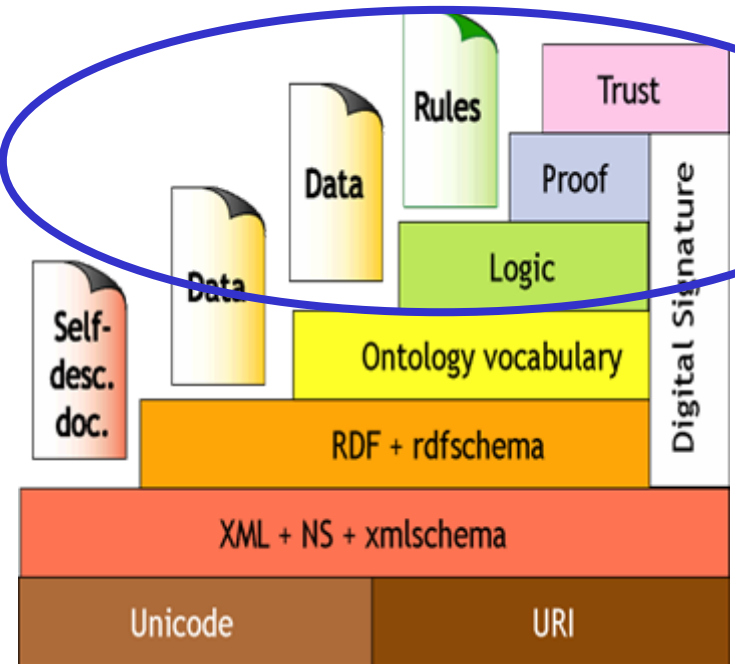
# Abfragesprachen

---

- SPARQL (W3C Empfehlung)
  - RDQL
  - Betrachtet RDF als RDB
  - SELECT, WHERE, ...
  - Ermöglicht Filterungen
  - Jedoch abhängig von Datenqualität
  - Nutzlos sobald andere Bezeichner gewählt
- => Auf Daten einheitlicher Ontologien angewiesen
- Erkennen keine Parallelitäten, Unterschiede
    - „wo:hat\_gekeltert“ und „ao:gekeltert\_von“

# Logic

- Datenstandards sorgen für wohldefinierte Bedeutung
- Um Schlussfolgern/neues Wissen bilden zu können nötig:
  - Logische Regeln auf RDF-Daten  
„wenn... dann...“ - Prinzip  
(Nicht nur von RDF/OWL-Konstrukten abhängig (subClassOf))
  - Programme die daraus schließen können („inference engines“)  
2 Zwecke:
    - Instance checking (Validating RDF Parser)
    - Schema /ontology mapping (RACER)





# Proof, Trust

---

- Beweisebene (proof layer):
  - nach den gleichen logischen Regeln logische Wahrheit bestimmter RDF-Aussagen prüfen (aus anderen Aussagen herleitbar)
- Vertrauensebene (trust layer):
  - Gefahr inhaltlich falscher Aussagen
  - RDF-Datenmodell offen u. dezentral, jeder kann beliebige Aussagen über beliebige Ressourcen treffen
  - nur vertrauenswürdige Quellen berücksichtigen
  - Verschlüsselung, Signaturen (XML Signatures), trust-network



# Stand der Dinge (1)

---

- “Semantic Web Activity Phase 1” vollendet
  - Kerntechnologien u. Infrastruktur implementiert
  - RDF/OWL spezifiziert
  - teilweise Ontologien geschaffen
  - ⇒ SW existiert also in kleinen Ansätzen
- Phase 2:
  - Verbreitung und Anwendung
  - Mehr (einfache) Werkzeuge
  - Eingliederung anderer Techniken  
(=> Semantic Web Services)



# Stand der Dinge (2)

---

- Allgemeine Akzeptanz, dass SW existieren soll
- Wege zu einem Semantic Web
  - Entdeckung der „Killer Application“
  - Bottom-Up Entwicklung getrieben von Anwendern, (Von Firmennetzwerken bzw. speziellen Anwendungsdomänen zu allgemeiner Verfügbarkeit)
- Ausblick:
  - Verschiedenste, teils noch unvorstellbare Nutzungsmöglichkeiten
  - Zusätzlich Anbindung an andere Datenquellen wie RDB, die schon Semantik enthalten
  - Multimedia im Semantic Web



# Risiken

---

- Sinkende Ergebnisbreite  
Automatische Nutzung semantischer Techniken bei Suche eliminiert Zufallsfunde (nicht gesucht und doch interessant)
- Sinkende Vertraulichkeit  
Durch eindeutige Referenzierung leidet Anonymität
- Sinkende Verlässlichkeit  
Verlagerung semantischer Filterungsprozesse an Maschinen verführt zu falschen Beschreibungen

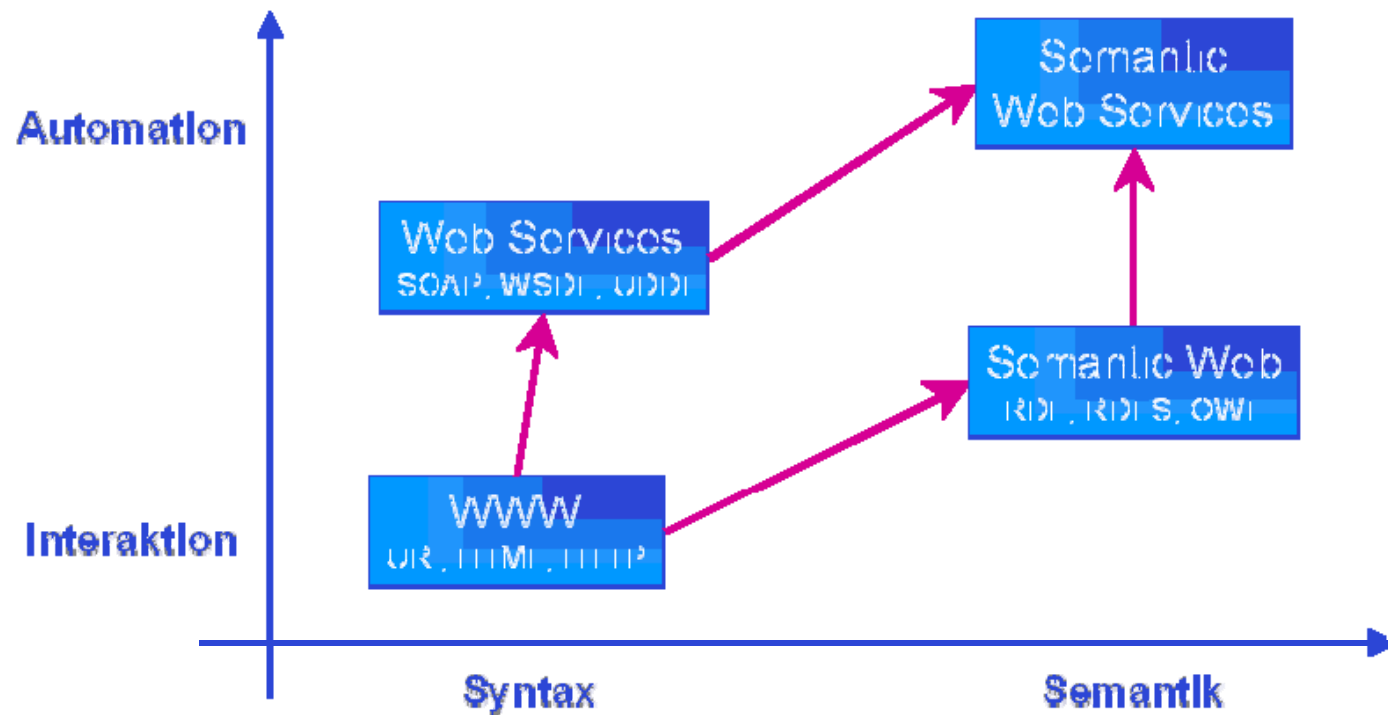


# Anwendungen

---

- Friend of a friend ([www.foaf-project.org/](http://www.foaf-project.org/))
- [Piggy Bank](#)
- RSS Version 1.0
- Mozilla intern (.conf, bookmarks)
- XMP ([www.adobe.com/products/xmp](http://www.adobe.com/products/xmp))
- Ontologieeditoren ( z.B. Protégé, Semantic Works 2006 )
- Ontologiesammlungen:
  - [www.geneontology.org](http://www.geneontology.org)
  - [www.schemaweb.info](http://www.schemaweb.info)
- [OntoWeb \(www.ontoweb.org\)](http://www.ontoweb.org)
- Intrawebs, knowledge bases
- OKAR (Fujitsu u. Ricoh, [www.labs.fujitsu.com/en/techinfo/okar/](http://www.labs.fujitsu.com/en/techinfo/okar/))  
“finde Person mit.... Fähigkeiten”
- MPEG-7: Annotierung von Multimedia
- WSDL, OWL-S

# Semantic Web Services



Quelle: [DF]

Quelle: <http://www.jeckle.de/semanticWebServices/intro.html>



# Semantic Web Services

---

- Erweiterung des WS-Konzepts durch Techniken des SW
- Agenten sollen mit Hilfe von Service-Ontologien SWS auffinden, aufrufen, zusammenstellen, überwachen
- Komplexe Verhaltensweisen durch Komposition



# OWL-S

---

- Ontologie auf Basis von OWL
- Auffinden, Ausführen, Komposition von WS
- Sehr allgemein, viel Raum für weitergehende Konzepte
- 3 Grundbestandteile:
  - Profile (Leistung)
  - Prozessmodelle (Ausführung)
  - Groundings (Zugriff)
- Vorteil: eindeutige, formal spezifizierte Semantik von WS-Parametern durch Ontologien
- einfaches Integrieren anderer Datenstrukturen



# WSDL (Web Service Description Language)

---

- Dokument enthält funktionelle Angaben:
  - Über Interface
  - Zugangsprotokoll und Details zum Deployment
  - notwendige Informationen für Zugriff auf den Service in maschinenlesbarem Format
- Nicht enthalten:
  - Quality of Service-Informationen
  - Taxonomien / Ontologien um Service semantisch einzuordnen

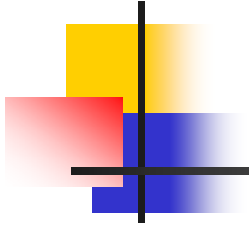


# Kritik

---

- erst wenige praxistaugliche Anwendungen
- großer Aufwand für Annotation
- abstrakte Begriffe ("Freiheit")
- Akzeptanz: Motivation zur Mitarbeit
  
- "I'd rather make progress by having computers understand what humans write, than by forcing humans to write in ways computers can understand."

Google-Mitbegründer Sergey Brin



Fragen???



# Linkempfehlungen

---

- [Tim Berners-Lee, James Hendle, Ora Lassila: The Semantic Web \(Scientific American, 5-2001 \)](#)
- [Webseite der W3C Semantic Web Activity](#)
- [Mario Jeckle: Semantic Web Services](#)
- [Jochen Notholt: Die Standards des Semantic Web](#)
- [Deutsche Übersetzung der W3C-Empfehlung zu OWL](#)
- [Wolfgang Dostal, Mario Jeckle u.a: Semantic Web](#)