Ein neues Konzept für die semantische Suche in heterogenen Informationssystemen zu Fragestellungen aus Umwelt und Energie

Zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
der Fakultät Maschinenbau
Karlsruher Institut für Technologie (KIT)
genehmigte
Dissertation
von
Thorsten Schlachter

Tag der mündlichen Prüfung 09.10.2018

Hauptreferent: Prof. Dr.-Ing. Dr. h. c. mult. Georg Bretthauer
Korreferentin: Prof. Dr. Dr.-Ing. Dr. h. c. Jivka Ovtcharova
Kurzfassung

Das Ziel der vorliegenden Arbeit besteht darin, ein neues Konzept für die semantische Suche in heterogenen Informationssystemen zu Fragestellungen aus Umwelt und Energie zu entwickeln, d.h. die Konzeption und Entwicklung einer Suchfunktion für Webportale, die zwar für den Nutzer so einfach wie herkömmliche Internet-Suchmaschinen funktioniert, jedoch qualitativ bessere, ggf. mehr Ergebnisse liefert als eine konventionelle Volltextsuche.

Dazu werden, ausgehend von einer Grundarchitektur, vier Architekturvarianten entworfen, vorgestellt und in konkreten Umsetzungsbeispielen evaluiert.

Schlagwörter: Semantik, Suche, Suchmaschine, Systemarchitektur, Umwelt, Energie

Abstract

The aim of this work is to develop a new concept for semantic search in heterogeneous information systems on the domains of the environmental and energy, that is, the conceptual design and development of a search function for web portals, which works as simple as conventional internet search engines, but provides qualitatively better, possibly more results than a conventional full-text search.

To this end, four architectural variants are designed out of a basic architecture and are evaluated in concrete implementation examples.

Keywords: semantics, search, search engine, system architecture, environment, energy
# Inhaltsverzeichnis

Kurzfassung .........................................................................................2
Abstract .........................................................................................2
Inhaltsverzeichnis ...........................................................................3
Abbildungsverzeichnis ....................................................................6
Abkürzungsverzeichnis ....................................................................8
Vorwort ............................................................................................11

1 Einleitung ......................................................................................13
  1.1 Bedeutung der semantischen Suche in heterogenen Informationssystemen .... 13
  1.2 Darstellung des Entwicklungsstandes von heterogenen Informationssystemen und der semantischen Suche ....................................................... 20
  1.3 Ziele und Aufgaben ................................................................... 24
    1.3.1 Harmonisierung von Semantik ................................................ 25
    1.3.2 Umgang mit unterschiedlichen Datentypen ................................ 26
    1.3.3 Datenquellen, Datenfluss, Konsistenz ...................................... 26
    1.3.4 Nutzung von Standards ............................................................ 26
    1.3.5 Freie Softwarekomponenten und Nutzung von Open Source ............ 27
    1.3.6 Abgrenzung ........................................................................... 27
    1.3.7 Was ist neu an der vorliegenden Arbeit? ..................................... 28
  1.4 Übersicht über die Arbeit ............................................................ 30

2 Ein neues Konzept für die semantische Suche ..................................32
  2.1 Grundidee und Übersicht ............................................................ 32
  2.2 Zielsysteme .............................................................................. 34
    2.2.1 Definition ............................................................................ 34
    2.2.2 Zielsysteme mit un- bzw. schwach strukturierten Inhalten ................ 34
    2.2.3 Semantik von Zielsystemen .................................................... 35
    2.2.4 Generische Datentypen ........................................................... 35
  2.3 Vorverarbeitung der Suchanfrage ................................................ 36
  2.4 Abbildung und Harmonisierung von Vokabularen ......................... 37
  2.5 Integrierte Ergebnisdarstellung (Mashup) ....................................... 37
  2.6 Verbindende Schicht zur Beschreibung und Realisierung von Anwendungen .......................................................... 38

3 Architekturvarianten .................................................................40
  3.1 Übersicht ................................................................................. 40
  3.2 Grundlagen ............................................................................ 40
3.2.1 Server-Zentrierung .......................................................... 40
3.2.2 Client-Zentrierung ............................................................ 41
3.2.3 Hybrider Ansatz ............................................................... 42
3.2.4 Nutzung von Web-Widgets .................................................. 43
3.2.5 Kopplung von Web-Widgets per Eventbus ................................ 43
3.3 Erste Architekturvariante: Semantische Erweiterung von Suchanfragen und Nutzung externer Datenquellen durch die Volltextsuchmaschine .......... 44
3.3.1 Semantische Erweiterung von Suchanfragen .......................... 46
3.3.2 OneBoxes zur Einbindung externer Datenquellen ...................... 46
3.3.3 Bewertung ..................................................................... 48
3.4 Zweite Architekturvariante: Serverseitige Verarbeitung der Suchanfrage, SearchBroker und Ontologiesystem .................................................. 49
3.4.1 SearchBroker als zentrale Komponente der Suche ...................... 51
3.4.2 Spezialisierte Plugins zur semantischen Vorverarbeitung der Suchanfrage ................................................................. 53
3.4.3 Auflösung thematischer Bezüge durch die Nutzung von Ontologien ............................... 53
3.4.4 Ontologiesystem ................................................................ 56
3.4.5 Zielsysteme und Zielsystembeschreibungen ............................. 58
3.4.6 Anfragen ........................................................................ 60
3.4.7 Mashup-Steuerung und Ergebnisdarstellung ............................. 61
3.4.8 Bewertung ..................................................................... 62
3.5 Dritte Architekturvariante: Serviceorientierung, „Webcache“, clientseitige Verarbeitung ................................................................. 64
3.5.1 Webcache ..................................................................... 64
3.5.2 Generische Services ............................................................ 66
3.5.3 Generische Frontend-Komponenten ....................................... 67
3.5.4 Zusammenspiel von Frontend-Komponenten ............................ 69
3.5.5 Verknüpfung semantischer Objekte und Klassen ....................... 72
3.5.6 Bewertung ..................................................................... 74
3.6 Vierte Architekturvariante: Ausbau zu semantischen Diensten / Linked Data ..................................................... 76
3.6.1 Identität von Objekten .......................................................... 78
3.6.2 Nutzung bzw. Generierung von Verknüpfungen ......................... 79
3.6.3 Beziehungsdienst ............................................................... 82
3.6.4 Metadatendienst ................................................................ 84
3.7 Gegenüberstellung der vier Architekturvarianten ......................... 85

4 Umsetzungsbeispiele ...................................................................... 87
4.1 Energieportal Baden-Württemberg ............................................. 87
4.2 Semantische Suche nach Umweltinformationen (SUI) .................... 89
4.3 Energieatlas 2015 .................................................................. 95
4.3.1 Ziele und Zielgruppen des Energieatlas ............................... 95
4.3.2 Erscheinungsbild des Energieatlas ....................................... 97
4.3.3 Systemarchitektur ............................................................... 99
4.3.4 Weiterentwicklung und Flexibilisierung der Liferay-Portlets ........ 100
4.4 LUPO-Portale .................................................................... 101
<table>
<thead>
<tr>
<th>Inhaltsverzeichnis</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.5 Mobile Apps</td>
<td>104</td>
</tr>
<tr>
<td>4.5.1 App „Meine Umwelt“</td>
<td>104</td>
</tr>
<tr>
<td>4.5.2 LHP-App „Meine Pegel“</td>
<td>108</td>
</tr>
<tr>
<td>4.5.3 Technischer Rahmen zur App-Entwicklung</td>
<td>111</td>
</tr>
<tr>
<td>5 Evaluation und Diskussion</td>
<td>116</td>
</tr>
<tr>
<td>5.1 Anwendungsszenarien (Use-Cases)</td>
<td>116</td>
</tr>
<tr>
<td>5.1.1 Szenario 1 „Politiker“</td>
<td>116</td>
</tr>
<tr>
<td>5.1.2 Szenario 2 „Bauen“</td>
<td>116</td>
</tr>
<tr>
<td>5.1.3 Szenario 3 „Öko-Urlaub“</td>
<td>117</td>
</tr>
<tr>
<td>5.1.4 Szenario 4 „Solardächer“</td>
<td>118</td>
</tr>
<tr>
<td>5.1.5 Szenario 5 „Ökostrom“</td>
<td>118</td>
</tr>
<tr>
<td>5.2 Evaluation und Bewertung der ersten Architekturvariante</td>
<td>118</td>
</tr>
<tr>
<td>5.3 Evaluation und Bewertung der zweiten Architekturvariante</td>
<td>122</td>
</tr>
<tr>
<td>5.4 Evaluation und Bewertung der dritten Architekturvariante</td>
<td>127</td>
</tr>
<tr>
<td>5.5 Diskussion der vierten Architekturvariante</td>
<td>131</td>
</tr>
<tr>
<td>6 Zusammenfassung</td>
<td>133</td>
</tr>
</tbody>
</table>

**Anhang: Grundlagen** ................................................................. 137

A1 Das Semantic Web ................................................................. 137
A1.1 Linked Data ................................................................. 137
A1.2 Vokabulare ............................................................................. 138
A1.3 Abfragen (Queries) ............................................................. 138
A1.4 Inferenzen (Schlussfolgerung) ............................................ 139
A2 Datentypen und der Strukturierungsgrad von Daten ............... 139
A2.1 Grundlagen für die maschinelle Verarbeitung von Daten ...... 141
A2.2 Semantische Interpretation von Daten ................................ 142
A3 Webportale ................................................................................. 142
A4 Serviceorientierte Architekturen .......................................... 143
A4.1 Microservices ......................................................................... 145
A4.2 Schnittstellen und Protokolle .............................................. 146
A5 Cloud-Dienste ........................................................................... 148

**Eidesstatliche Versicherung** .................................................... 150

**Literaturverzeichnis** ................................................................. 151
Abbildungsverzeichnis

Abbildung 1: Komponenten einer Architektur für eine semantische Suche .................. 18
Abbildung 2: Rahmen einer allgemeinen Architektur für die semantische Suche ........... 32
Abbildung 3: Umsetzung auf Basis einer vorhandenen Volltextsuchmaschine (neue
Entwicklungen und eigene Anteile in rot) ......................................................... 45
Abbildung 4: Serverseitige Umsetzung mit SearchBroker und Ontologiesystem
(neue Entwicklungen und eigene Anteile in rot) ............................................... 50
Abbildung 5: Übersicht über die Komponenten des Portals .................................. 51
Abbildung 6: Übersicht über den SearchBroker .................................................... 52
Abbildung 7: Semantische Treppe nach (Pellegrini und Blumauer 2006) ................. 54
Abbildung 8: Beispiel einer OpenSearch-Description (XML) ............................... 58
Abbildung 9: Umsetzung als serviceorientierte Architektur mit Aufbau eines
„Webcache“ als Sammlung generischer Dienste (neue Entwicklungen und
eigene Anteile in rot) .................................................................................. 65
Abbildung 10: Suchergebnisseite mit Karte, Layer-Auswahl, Volltext- und
Metadaten-Trefferlisten im Umweltinformationsnetz Sachsen-Anhalt
(Screenshot Umweltinformationsnetz Sachsen-Anhalt) .................................... 70
Abbildung 11: Verknüpfung von Windkraftanlagen und (Natur-)Schutzgebieten
durch die Suche nach „windrad schutzgebiet langenburg“ im Umweltportal
Baden-Württemberg (Screenshot) ..................................................................... 72
Abbildung 12: Umsetzung als serviceorientierte Architektur mit zusätzlichem Link-
Service (neue Entwicklungen und eigene Anteile in rot) .................................. 77
Abbildung 13: Menüstruktur (oben) und Tagcloud (unten) im Energieportal Baden-
Württemberg (Screenshots) ............................................................................ 88
Abbildung 14: Beispiel für Kartendarstellung im Energieportal Baden-Württemberg:
Eignung von Dachflächen für Photovoltaikanlagen auf Basis der solaren
Einstrahlung (Screenshot). Dieselbe Darstellung wird auch im Energieatlas
Baden-Württemberg verwendet .................................................................... 89
Abbildung 15: Ontologiesystem in SUI; aus: (Bügel et al. 2011b) ............................ 92
Abbildung 16: Ontology-Mapping im SUI-System; aus: (Bügel et al. 2011b) ............ 93
Abbildung 17: Daten zu bestehenden Windkraftanlagen im erweiterten Daten- und
Kartenangebot des Energieatlas (Screenshot) ............................................... 97
Abbildung 18: Komponente zur Anzeige von aktuellen Kennzahlen für die
Einspeisung von Wind- und Solarenergie (Ausschnitt Screenshot
Energieatlas Baden-Württemberg) ................................................................ 98
Abbildung 19: Informieren (links), Melden (mittig), Erleben (rechts) –
Kernfunktionen der „Meine Umwelt“-App (Screenshots der App „Meine
Umwelt“) .......................................................................................................... 105
Abbildung 20: Start-Bildschirm, Navigation und Auswahl des Bundeslandes
(Screenshots der App „Meine Umwelt“) ......................................................... 106
Abbildung 21: Bereich Informieren beinhaltet Karten mit Unterthemen (links),
Detailinformationen zu ausgewählten Objekten (mittig) sowie aktuelle
Messwerte (rechts) (Screenhots der App „Meine Umwelt“) .............................. 107
Abbildung 22: Verschiedene Meldethemen (links), Formular zum Erfassen von
Standort, Sachdaten (mittig) im Bereich „Melden“, sowie die Anzeige von
eingegangenen Meldungen im Bereich „Informieren“ (Screenshots der App
„Meine Umwelt“) ................................................................................................ 108
Abbildung 23: Webangebot des länderübergreifenden Hochwasserportals LHP, links der normalen Webansicht, rechts der mobilen Ansicht (Screenshots) ...109

Abbildung 24: Übersicht der Pegel als Karte (links), Pegeldetails mit Ganglinie (mittig) und Favoritenliste (rechts) (Screenshots der App „Meine Pegel“) ...............110

Abbildung 25: Einrichtung einer Pegelwarnung (links), Einrichtung von Abonnement (mittig) und Eingang von Mitteilungen (rechts) (Screenshots der App „Meine Pegel“) .................................................................110

Abbildung 26: LUPO-Baukasten als Fundament zur Erstellung von Umwelt-Apps ....112

Abbildung 27: Konzeptionelle Struktur der "Meine Umwelt" App.................................113

Abbildung 28: Buildpipeline der App „Meine Umwelt“ (nach Projektdokumentation „Meine Umwelt“, xdot GmbH) .................................................................114

Abbildung 29: Strukturierung versus Standardisierung der Datenschemata; nach (Holzinger 2014) ..........................................................................................140

Abbildung 30: Beispiel einer Microservice-basierten Architektur für Landesumweltportale .............................................................................................146

Abbildung 31: Cloud-Pyramide (nach http://skalicloud.com/v4/the-cloud-pyramid/) ...149
<table>
<thead>
<tr>
<th>Abkürzung</th>
<th>Beschreibung</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ajax</td>
<td>Asynchronous JavaScript and XML</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface (Programmierschnittstelle)</td>
</tr>
<tr>
<td>APK</td>
<td>Android Package</td>
</tr>
<tr>
<td>APPX</td>
<td>Microsoft App Package</td>
</tr>
<tr>
<td>CMS</td>
<td>Content Management System</td>
</tr>
<tr>
<td>CRUD</td>
<td>Create, Read, Update, Delete</td>
</tr>
<tr>
<td>CSS</td>
<td>Cascading Style Sheets</td>
</tr>
<tr>
<td>DOC</td>
<td>Microsoft Word-Datei</td>
</tr>
<tr>
<td>DTD</td>
<td>Document Type Definition (Dokumenttypdefinition)</td>
</tr>
<tr>
<td>EnEG</td>
<td>Energieeinsparungsgesetz</td>
</tr>
<tr>
<td>EnEV</td>
<td>Energieeinsparverordnung</td>
</tr>
<tr>
<td>FADO</td>
<td>Fachdokumente Online</td>
</tr>
<tr>
<td>FTP</td>
<td>File Transfer Protocol</td>
</tr>
<tr>
<td>GEMET</td>
<td>General Multilingual Environmental Thesaurus</td>
</tr>
<tr>
<td>GSA</td>
<td>Google Search Appliance</td>
</tr>
<tr>
<td>HATEOAS</td>
<td>Hypermedia as the Engine of Application State</td>
</tr>
<tr>
<td>HTML</td>
<td>Hypertext Markup Language (Hypertext-Auszeichnungssprache)</td>
</tr>
<tr>
<td>HTML5</td>
<td>Hypertext Markup Language Version 5</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>IaaS</td>
<td>Infrastructure as a Service</td>
</tr>
<tr>
<td>IAI</td>
<td>Institut für Angewandte Informatik</td>
</tr>
<tr>
<td>ID</td>
<td>Identifier</td>
</tr>
<tr>
<td>IOSB</td>
<td>Fraunhofer-Institut für Optronik, Systemtechnik und Bildauswertung</td>
</tr>
<tr>
<td>IoT</td>
<td>Internet of Things</td>
</tr>
<tr>
<td>IPA / .ipa</td>
<td>iOS application archive</td>
</tr>
<tr>
<td>ISS</td>
<td>International Space Station</td>
</tr>
<tr>
<td>JCP</td>
<td>Java Community Process</td>
</tr>
<tr>
<td>JSON</td>
<td>JavaScript Object Notation</td>
</tr>
<tr>
<td>Abkürzung</td>
<td>Deutscher Begriff</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------</td>
</tr>
<tr>
<td>JSR</td>
<td>Java Specification Request (im Rahmen eines Java Community Pro- cess)</td>
</tr>
<tr>
<td>JSR-286</td>
<td>Java Portlet Specification 2.0</td>
</tr>
<tr>
<td>KIT</td>
<td>Karlsruher Institut für Technologie</td>
</tr>
<tr>
<td>LUBW</td>
<td>Landesanstalt für Umwelt Baden-Württemberg, bis 2017 Landesanstalt für Umwelt, Messungen und Naturschutz Baden-Württemberg</td>
</tr>
<tr>
<td>MAT</td>
<td>Mensch/Aufgabe/Technik</td>
</tr>
<tr>
<td>OGC</td>
<td>Open Geospatial Consortium</td>
</tr>
<tr>
<td>OWL</td>
<td>Ontology Web Language</td>
</tr>
<tr>
<td>PaaS</td>
<td>Platform as a Service</td>
</tr>
<tr>
<td>PDF</td>
<td>Portable Document Format</td>
</tr>
<tr>
<td>PM10</td>
<td>Particulate Matter 10µ</td>
</tr>
<tr>
<td>RDF</td>
<td>Resource Description Framework</td>
</tr>
<tr>
<td>RDFa</td>
<td>RDF in Attributes</td>
</tr>
<tr>
<td>RDF(S)</td>
<td>Resource Description Framework Schema (RDF Schema)</td>
</tr>
<tr>
<td>REST</td>
<td>Representational State Transfer</td>
</tr>
<tr>
<td>RSS</td>
<td>Really Simple Syndication</td>
</tr>
<tr>
<td>SaaS</td>
<td>Software as a Service</td>
</tr>
<tr>
<td>SASS</td>
<td>Syntactically Awesome Style Sheets</td>
</tr>
<tr>
<td>SKOS</td>
<td>Simple Knowledge Organization System</td>
</tr>
<tr>
<td>SNS</td>
<td>Semantic Network Service</td>
</tr>
<tr>
<td>SOA</td>
<td>Serviceorientierte Architektur</td>
</tr>
<tr>
<td>SOAP</td>
<td>Simple Object Access Protocol</td>
</tr>
<tr>
<td>SOS</td>
<td>Sensor Observation Service</td>
</tr>
<tr>
<td>SMTP</td>
<td>Simple Mail Transfer Protocol</td>
</tr>
<tr>
<td>SPARQL</td>
<td>SPARQL Protocol and RDF Query Language</td>
</tr>
<tr>
<td>SUI</td>
<td>Semantische Suche nach Umweltinformationen</td>
</tr>
<tr>
<td>UDDI</td>
<td>Universal Description, Discovery and Integration</td>
</tr>
<tr>
<td>UM</td>
<td>Ministerium für Umwelt, Klima und Energiewirtschaft Baden- Württemberg</td>
</tr>
<tr>
<td>URI</td>
<td>Uniform Resource Identifier</td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator</td>
</tr>
<tr>
<td>Abkürzung</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>UUID</td>
<td>Universally Unique Identifier</td>
</tr>
<tr>
<td>W3C</td>
<td>World Wide Web Consortium</td>
</tr>
<tr>
<td>WMS</td>
<td>Web Map Service</td>
</tr>
<tr>
<td>WSDL</td>
<td>Web Service Definition Language</td>
</tr>
<tr>
<td>WWW</td>
<td>World Wide Web</td>
</tr>
<tr>
<td>XML</td>
<td>Extensible Markup Language</td>
</tr>
</tbody>
</table>
Vorwort


In abgeschlossenen Bereichen, z.B. den Datenbeständen einer Behörde, ist die Vielfalt der Daten geringer als im gesamten Internet, allerdings werden auch dort eine Vielzahl von Datentypen, Systemen, Datenformaten etc. verwendet. Ein Vorteil von abgeschlossenen Bereichen gegenüber dem Internet ist jedoch häufig, dass die Daten sich auch inhaltlich auf einen beschränkten Bereich (eine inhaltliche Domäne) beziehen. Das heißt, dass bei der Klassifizierung von Daten lediglich innerhalb der Domäne zugeordnet werden muss. In vielen Fällen wird sogar ein einheitliches Wortgut (Vokabular) zur Beschreibung der Daten verwendet, manchmal sind sogar die Schemas der Datensätze oder gemeinsame Schlüssellisten (einhreich) festgelegt. Das erleichtert das eindeutige Zuordnen von Daten zu Themen und bietet damit ein großes Potenzial für das präzise Auffinden von Daten zu einem bestimmten Thema oder die Darstellung von mit den Daten im Zusammenhang stehenden weiteren Daten.

In der vorliegenden Arbeit geht es um das Auffinden von Daten innerhalb abgegrenzter Domänen, insbesondere aus den Bereichen „Umwelt“ und „Energie“. Hier soll für einen begrenzten, dennoch großen Datenbestand das Potenzial einer semantischen Suche geprüft und in verschiedenen Umsetzungsvarianten realisiert werden. Dazu sollen Daten aus dem Bereich der Umweltverwaltung herangezogen werden, auch da er in sei-

Anspruch der vorliegenden Arbeit ist es, die entwickelten informationstechnischen Architekturen und Verfahren auch in der Praxis zu erproben, d.h. in konkreten Projekten umzusetzen. Das bedeutet, dass sie nicht nur grundsätzlich oder im Ansatz funktionieren, sondern auch einen Praxistest bestehen müssen.

Die vorliegende Arbeit wäre nicht möglich gewesen ohne die Unterstützung, Aufmunterung, Geduld und Hartnäckigkeit verschiedener Personen, insbesondere meines Betreuers Prof. Dr.-Ing. Dr. h. c. mult. Georg Bretthauer, meiner Chefs und Kollegen Dr. Clemens Düpmeyer, Dr. Werner Geiger, Rainer Weidemann, Eric Braun, Claudia Greceanu, Christina Grieß, Christian Schmitt, Gerd Zilly und Prof. Dr. Veit Hagenmeyer.

In vielen Projekten, die in die vorliegende Arbeit eingeflossen sind, durfte ich mit Kollegen anderer Institutionen an spannenden Themen zusammenarbeiten. Stellvertretend für viele weitere danke ich Wolfgang Schillinger (LUBW), Martina Tauber (LUBW), Renate Ebel (LUBW, IM BW), Roland Mayer-Föll (UM BW), Kurt Weissenbach (UM BW), Fernando Chaves Salamanca (IOSB), Ulrich Bügel (IOSB), Andreas Abecker (FZI, disy), Thomas Sattler (DECON-network), Joachim Fock (Umweltbundesamt), Thomas Bandholtz (ehem. innoQ), Lars Koch (CONVOTIS AG) sowie den vielen Partnern aus der LUPO-Kooperation.

Meiner Frau Stephanie und meinem Sohn David bin ich zu Dank verpflichtet, da sie phasenweise wenig von ihrem Mann bzw. Papa hatten.

Zuletzt möchte ich auch meinen Eltern danken, die mir in vielen Phasen meines Studiums und meines Lebens den Rücken freigehalten haben.
1 Einleitung

1.1 Bedeutung der semantischen Suche in heterogenen Informationssystemen

Aus Sicht eines Nutzers steht und fällt der Nutzen von Webportalen (s. Anhang A3) und Informationssystemen mit der Frage, wie schnell und mit welchem Aufwand er oder sie an die gewünschte Information gelangt bzw. die gewünschte Aufgabe erledigen kann. Die Art und Weise der Internetnutzung und Online-Recherche hängt dabei stark von der betrachteten Nutzergruppe ab. Nach der Online-Studie von ARD und ZDF (Koch und Frees 2015) waren zumindest knapp 80% der deutschen Bevölkerung im Jahr 2015 zumindest gelegentlich online, der Anteil der täglichen Internet-Nutzer lag bei rund 63%. Während 76% der deutschen Internet-Nutzer angeben, das Internet zur Suche nach Informationen zu verwenden, verwenden sogar 82% Online-Suchmaschinen, um an die gewünschten Daten zu gelangen, was eindrucksvoll den Erfolg und Nutzen von Internet-Suchmaschinen zeigt. Nicht nur aus der Sicht der Nutzer stellen Suchmaschinen zentrale Einstiegspunkte ins Web dar, sondern auch Informationsanbieter müssen sich um ein hohes Ranking und eine entsprechend gute Platzierung auf den Trefferseiten der Suchmaschinen bemühen. Nach (Lunapark 2015) lag der weltweite Marktanteil der Suchmaschine google.com im Jahr 2015 bei über 90%, in Deutschland sogar bei über 93%.

Offenbar setzt Googles Suchmaschine also einige wesentliche Erfolgsfaktoren um, die ihre Nutzung für eine große Zahl von Internet-Nutzern attraktiv macht.


- die simple Nutzung und Bedienoberfläche („single search slot“, mobile enabled user interface, ...),
- die hohe Relevanz der Suchergebnisse,
- das Erkennen, was gesucht wird (Semantik der Anfrage),
- die Nutzung relevanter, hochwertiger Datenquellen,
- Kompetenz und Glaubwürdigkeit der Suchmaschine,
- gute Ranking-Faktoren,
- die Integration von Einzeldiensten (Volltextsuche, Nachrichten, Medien, Karten, Zeiträumen, ...),

1 Dabei spielt die Nutzung mobiler Endgeräte eine zunehmend größere Rolle. 23% der Bevölkerung nutzten das Internet im Jahr 2015 täglich von einem Mobilgerät aus, 55% zumindest gelegentlich. Mobile Endgeräte werden dabei vornehmlich durch jüngere Personen im Alter unter 50 Jahren genutzt.
Einleitung

- Data Mining (das Erschließen großer Datenbestände; inklusive dem Sammeln von Wissen über den Informationsbedarf des einzelnen Nutzers),
- Offenheit (Openness, das Teilen von Inhalten),
- der Zugang zu verwandten Themen und
die (für den Nutzer) kostenfreie Nutzung.


Gründe für die Nicht-Indexierbarkeit gibt es viele, z.B. kontextabhängige Inhalte, dynamische Inhalte, limitierten Zugriff (Passwortschutz, Ausschluss von Suchmaschinen), spezifische/proprietäre Datenformate oder Softwareprodukte, Nutzung von Scripten, fehlende Links etc.


Auch wenn für einige der aufgeführten Probleme, z.B. die Ausführung von im Inhalt enthaltenem Programmcode, z.B. JavaScript, während der Indexierung, durchaus Lösungsansätze oder sogar Lösungen existieren, so gibt es für den größten Teil des Deep Web keine Zugangslösungen und die enthalten Inhalte werden vermutlich schon aufgrund ihrer Masse auch in absehbarer Zeit nicht von Internet-Suchmaschinen erreicht werden - durch die automatisierte Erzeugung von Daten („Internet of Things“ (IoT), Sensornetze) werden die Anzahl der Datenquellen und die Menge der erzeugten Daten sogar noch rasanter wachsen als bisher (Siemens 2014).

Die beschriebene Problematik für große Internet-Suchmaschinen bietet jedoch Chancen und Nischen für kleine, spezialisierte Suchmaschinen, die nicht den Anspruch haben, eine Recherche über das gesamte Web anzubieten, sondern sich auf einen spezifischen, thematisch (eng) eingegrenzten Bereich („Domäne“) spezialisieren. Für sie kann es sich lohnen, Aufwand in die Implementierung spezifischer Schnittstellen zu relevanten Informationen zu stecken, um organisatorisch/rechtlich Zugang zu bestimmten Daten zu bekommen oder um Daten aus verschiedenen Quellen miteinander zu verknüpfen.

In einer abgegrenzten, bekannten Domäne ist es zusätzlich erheblich leichter, auch die Semantik von Daten und damit auch Zusammenhänge zu erfassen (Gliozzo und Strapparava 2009). Während die meisten klassischen Internet-Suchmaschinen zu großen Teilen auf dem Vergleich von Zeichenketten basieren (Lewandowski 2015), kann eine
spezialisierte Suchmaschine zusätzlich sowohl die Semantik der indexierten Inhalte als auch die Semantik von Suchanfragen erfassen. Hilfreich kann hier auch die Erkennung bestimmter Muster in den Suchanfragen sein, z.B. die Verknüpfung eines thematischen Suchbegriffes mit einem Orts- oder Zeitbezug. So kann zum Beispiel das Erkennen des Schlagwortes “Bahn” im Zusammenhang mit zwei erkannten Städtenamen und einer Zeitangabe zur Suche nach Bahnverbindungen zwischen den beiden Orten genutzt werden - sofern die Suchmaschine über die entsprechenden Daten oder eine Schnittstelle zu einem entsprechenden Hintergrundsystem (Bahn-Fahrplan) verfügt.


Für die Nutzer aus der Zielgruppe „Öffentlichkeit“ muss die Suche so einfach und transparent nutzbar sein, wie sie es von klassischen Volltextsuchmaschinen gewohnt sind, und komplexe Technik innerhalb der Suchmaschine darf sich nicht in einer komplexen Nutzerschnittstelle widerspiegeln.


Viele Ziele der Energiewende haben einen direkten Bezug zur Umwelt und viele Auswirkungen der Energiewende werden sich daher auch im Umweltbereich zeigen bzw. Kriterien für die Zielerreichung lassen sich durch die Beobachtung der Umwelt bestimmen und/oder messen, z.B. die Begrenzung der globalen Temperatur oder die Verbesserung der Luftqualität. Daneben stehen viele Entscheidungen im Bereich der erneuerbaren Energien in einem direkten oder indirekten Zusammenhang mit Umweltfragen, z.B. beim Flächenverbrauch, Bauen in Schutzgebieten, Wasserverbrauch für Kühlwasser, Stauung von Fließgewässern, Nutzung von Biomasse zur Energierzeugung (statt als Dünger) etc. Umwelt und Energie stellen also zwei eng miteinander verwobene Domänen dar, die auch jede für sich betrachtet werden kann.

Die Verknüpfung bzw. die Verknüpfbarkeit von Energie- und Umweltinformationen ist für die Klärung vieler Fragen zur Wirksamkeit des Einsatzes erneuerbarer Energien bzw. zur Energiewende von essentieller Bedeutung.


Es fehlt dem UIS BW jedoch bislang eine Suchfunktion, die Daten aus der Vielzahl von Einzelsystemen des UIS BW zusammentragen und in übergreifender Art und Weise darstellen kann.

Die zentrale Forschungsfrage beschäftigt sich daher mit der Verbesserung der Suchergebnisse innerhalb der Domäne „Umwelt“ und „Energie“:

„Wie kann ein Nutzer eines Webportals in einer heterogenen Landschaft von existierenden Web- und Informationssystemen durch simple Suchfragen („single search slot-Philosophie“) zu Fragestellungen aus den Bereichen „Umwelt“ und „Energie“ umfassende (semantisch stimmige) integrierte Antworten erhalten?“

Unter der Formulierung „von existierenden Web- und Informationssystemen“ werden Systeme verstanden, die „so sind wie sie sind“, d.h. der Betreiber des Webportals inklusive der Suchmaschine hat keinen Einfluss auf die Inhalte und Form der von den Systemen bereitgestellten Daten und ggf. Metadaten. Die Daten werden in einer „heterogenen Landschaft“, d.h. über unterschiedliche Protokolle, Repräsentationen, technische Formate und Codierungen sowie verschiedene (implizite oder explizite) Semantiken verfügbar gemacht. Sie werden durch die Suchmaschine erfasst und im Portal in stimmigen Trefferansichten dargestellt, d.h. jedem Datentyp werden eine oder mehrere passende Darstellungsformen (Liste, Tabelle, Diagramm, Karte etc.) zugeordnet.

Das Gesamtziel der vorliegenden Arbeit ist also die Konzeption und Entwicklung einer Architektur für die Suchfunktion von Webportalen, die zwar für den Nutzer so einfach wie herkömmliche Internet-Suchmaschinen funktioniert, jedoch bessere, ggf. mehr Ergebnisse liefert als eine konventionelle Volltextsuche. Dabei sollten folgende wissenschaftlichen Teilziele erreicht werden (Abbildung 1):

- Erkennen der Semantik einer Suchanfrage innerhalb einer gegebenen Domäne
- Beschreibung der Semantik von Daten gegebener Informationssysteme bezüglich einer vorgegeben Domäne, die sich ggf. aus mehreren Vokabularen (s. Anhang A1.2) zusammensetzt:
  - Mapping der Vokabulare untereinander (Artikulation) bzw. Harmonisierung
  - Abbildung der Daten auf Vokabulare und ggf. Nutzung zugehöriger Schemata
  - Nutzung von Zusatzwissen zu Orts- und Zeitbezug, auch als generische Zusammenhänge zwischen Daten
  - Harmonisierung der Darstellung von Daten aus verschiedenen Informationssystemen
  - Nutzung der in den Vokabularen enthaltenen Semantik, z.B. zur Weiternavigation, Gruppierung, Facettierung etc.
- Beschreibung und Realisierung des technischen Zugriffs auf heterogene Informationssysteme (Datentypen, Schnittstellen und Formate)
- Nutzung generischer Komponenten zur Präsentation von Daten innerhalb eines Webportals
- Präsentation/Darstellung der Suchergebnisse in einer integrierten Trefferansicht
  - Unterschiedliche Darstellungen (Karte, Diagramm, Tabelle, Liste etc.)
Einleitung

- Möglichkeit zur Kommunikation unter den Darstellungskomponenten.

Abbildung 1: Komponenten einer Architektur für eine semantische Suche

Abbildung 1 zeigt den Rahmen einer allgemeinen Architektur für die semantische Suche und damit bereits eine Art Grundarchitektur: Die Zielsysteme ZS_i (unten) enthalten die Daten, jeweils in einer spezifischen Darstellung und Semantik. Die Zielsysteme enthalten im Allgemeinen verschiedene Datentypen, bieten unterschiedliche Schnittstellen zum Zugriff auf die Daten und verwenden dabei eine ganze Reihe technischer Formate („Anbindung Zielsystem“).

Dem Nutzer soll über ein oder mehrere Webportale (rechts oben) eine Recherchemöglichkeit für die in den Zielsystemen enthaltenen Daten geboten werden. Die minimalen Bestandteile der Benutzerschnittstelle sind ein Eingabefeld („Suchschlitz“) für das Formulieren der Suchanfrage (oben links) sowie die Präsentation der Suchergebnisse („Ergebnisdarstellung“), z.B. innerhalb von Webportalen, ggf. in einer integrierten Ansicht, welche adäquate Darstellungen (Karte, Diagramm, Liste etc.) für die verschiedenen Datentypen bietet.

Der zwischen Zielsystemen und Webportalen befindliche rote Kasten („?“) repräsentiert die im Rahmen der vorliegenden Arbeit zu lösenden Aufgabenstellungen, zunächst den technischen Zugriff auf die Zielsysteme und anschließend die semantische Verknüpfung zwischen den Suchanfragen der Nutzer und den in den Zielsystemen enthaltenen Daten. Mehrere Vokabulare V_i (links unten) können zur semantischen Harmonisierung

Die Kriterien für das Erreichen von Zielen sind deshalb unter Berücksichtigung der verschiedenen Rollenverständnisse zu definieren (Abschnitt 1.3). Die Evaluation soll anhand ausgewählter Anwendungsszenarien erfolgen (Abschnitt 1.3.6). Der Spielraum möglicher Architekturvarianten wird durch die gegebenen Voraussetzungen (Abschnitt 1.2) und existierende Nebenbedingungen (Abschnitt 1.3.6) eingeschränkt.

Die oben aufgeführten Ziele stehen jedoch nicht allein, sondern müssen jeweils im Kontext verschiedener Rollen betrachtet werden:

- Nutzer von Webportalen (bzw. übergreifenden Informationssystemen)
- Betreiber solcher Portalsysteme
- Betreiber von Informationssystemen als originäre Datenquellen (Zielsysteme)
- Weitere (potenzielle) Nutzer der Daten / Übergreifende Interessen.


² Der Schluss auf die Nordhalbkugel beinhaltet eine Voreinstellung (Default-Wert), welche der typischen Verwendung der Suche (in deutschsprachigen) Webportalen entspricht. Ggf. könnte sich die Interpretation durch weitere Informationen, z.B. eine in der Suchanfrage enthaltene Ortsangabe oder weitere Kontextinformationen (Standort des Nutzers), ändern.
verfügbarer Informationen wird in der Regel auch ein Mapping der externen Daten auf ein im Portal verwendetes Datenmodell notwendig sein, das ebenfalls per Konfiguration möglich sein soll.


### 1.2 Darstellung des Entwicklungsstandes von heterogenen Informationssystemen und der semantischen Suche


---

\(^3\) Hier soll nicht auf die damit verbundenen rechtlichen Fragestellungen eingegangen werden.
Bei den zu betrachtenden Zielsystemen (Abbildung 1) handelt es sich um eine Menge heterogener Informationssysteme, die sich in ganz unterschiedlichen Aspekten unterscheiden können, z.B.

- Datentypen (Art von Daten, Strukturierung, Semantik...)
- Schnittstellen (für Maschinen/Menschen)
- Datenformate/Repräsentationen (z.B. HTML, PDF, JSON, XML, WMS...)
- Semantik (definiert je Zielsystem, aber nicht übergreifend/global)
- Erreichbarkeit (z.B. fehlende Verlinkung, Dark Web)
- Verfügbarkeit (z.B. Vorhandensein von Wartungsfenstern)
- Skalierbarkeit (z.B. beschränkte Anzahl paralleler Nutzeranfragen).

Neben der semantischen Verknüpfbarkeit von Daten ist eine weitere wesentliche Herausforderung der semantischen Suche, dem Nutzer eine möglichst einfache und intuitive Benutzerschnittstelle zu bieten, gerade weil viele existierende Ansätze keine besonders einfachen – wenn dafür auch mächtige – Benutzerschnittstellen bieten, und sich damit eher an Experten als an Laien bzw. die interessierte Öffentlichkeit wenden (Mangold 2007; Bizer et al. 2009).


Die Suchanfragen an eine semantische Suche sollen daher einfach, z.B. in Form eines „single search slot“ formuliert werden können, d.h. sie werden in Form einer einzigen Zeichenkette an die Suchfunktion übermittelt; natürlichsprachliche Eingaben lassen sich mithilfe von Spracherkennung auf solche Zeichenketten abbilden. Dabei sollte es um eine rein inhaltliche Formulierung der Suche gehen, d.h. möglichst nahe an der natürlichen Sprache und ohne formale Syntax, wie z.B. SPARQL (W3C 2008).

Solche einfachen Benutzerschnittstellen und Suchanfragen stehen im Gegensatz zu komplexeren Suchformularen, die mehrere Suchschlitze/Formularfelder anbieten, in denen der Nutzer semantisch vorklassifizierte Suchbegriffe, z.B. Orts- oder Zeitangaben, explizit angeben bzw. auswählen kann. Die Herausforderung für die semantische Suchmaschine besteht darin, die Semantik der Suchanfrage zu erkennen, ggf. unter Zuhilfenahme von Kontextinformationen, und sie auf die Semantik(en) der angeschlossenen Zielsysteme abzubilden.

Die Ergebnispräsentation soll in einem Webportal erfolgen. Dazu sollen grundsätzlich generische Komponenten zum Einsatz kommen. Der Anschluss neuer Zielsysteme soll also nicht automatisch die Entwicklung neuer Frontend-Komponenten bedeuten, vielmehr sollen vorhandene Komponenten automatisch oder maximal durch Konfiguration in der Lage sein, Daten aus dem neuen Zielsystem in der Ergebnispräsentation zur Anzeige zu bringen. Wenn die Ergebnispräsentation aus mehreren Komponenten be-
Einleitung

steht, muss sie dennoch eine konsistente Gesamtsicht auf die Suchergebnisse bieten. Daher müssen (hoch-)konfigurierbare Komponenten zur generischen Darstellung verschiedener Datentypen (Volltext, Messdaten, Objektdaten, Metadaten, Dokumente, Medien) bereitgestellt werden.

Im Idealfall stehen Beschreibungen der Semantik der darzustellenden Daten, z.B. in Form von formalen Datenschemata (Brickley und Guha 2014; schema.org 2016b; jsonschema-org 2018; W3C 2012b), zur Verfügung, welche die Konfiguration von Anzeigekomponenten automatisieren oder zumindest unterstützen können.

Die Orchestrierung der einzelnen Komponenten muss möglich sein, um eine Gesamtanwendung als Zusammenspiel mehrerer Komponenten erstellen zu können. Änderungen in einer Komponente, z.B. die Aktualisierung der Elemente (Inhalte) in einer Liste, muss den anderen Komponenten mitgeteilt werden, damit sie ggf. auf die Veränderung reagieren können.

Bei Nutzerinteraktionen, z.B. dem Verschieben eines Kartenausschnitts oder der Selektion eines Objektes, soll die Konsistenz der Ergebnispräsentation gewahrt bleiben, z.B.

- fehlende Daten nachgeladen werden
- mehrere unterschiedliche Repräsentationen des selektierten Objektes (z.B. in einer Liste, einer Karte und einer Detailansicht) ebenfalls als ausgewählt dargestellt werden
- Änderungen der Suchanfrage zur Aktualisierung der Ergebnispräsentation und/oder ihrer Bestandteile führen.


Bei einer domänenspezifischen Suchmaschine, die z.B. Daten aus dem Bereich „Umwelt“ bereitstellt, stellt sich zunächst die Frage nach den Grenzen der Domäne, die im
Einleitung


Auch die Präsentation der Suchergebnisse sollte spezifisch erfolgen – d.h. je nach Datentyp sollten die Daten passend präsentiert werden, z.B. Geoobjekte in einer Kartenansicht, Messwerte als Tabelle oder Diagramm, Volltexttreffer inklusive der Fundstelle (Snippets) etc. Dabei ist es wichtig, dass die besten Treffer nicht in einer Masse von „Rauschen“, d.h. für die Suchanfrage nicht oder wenig relevanten Informationen, untergehen. Bei wenig spezifischen Suchanfragen kann das eine Gratwanderung sein, einerseits sollen Informationen zu einem unspezifischen Thema wie „Bauen“ gefunden werden, wer sich so einen Überblick verschaffen möchte, darf sich nicht in zu vielen Details verlieren.

Zur Beurteilung der Qualität der Treffer zu einer gegebenen Suchanfrage ist nicht deren Anzahl, sondern in erster Linie die Relevanz der Ergebnisse für den Nutzer entscheidend. Da relevante Treffer im Allgemeinen aus verschiedenen Zielsystemen stammen können, sollten auch Treffer aus allen relevanten Systemen gefunden werden. (Griesbaum et al. 2002) verwenden die Begriffe Makroprecision und Mikroprecision zur Beschreibung der Effektivität einzelner Suchanfragen, im ersten Fall der gesamten Trefferliste, im zweiten Fall des einzelnen Ergebnisses.

---

Die Kriterien zur Bewertung der Relevanz von Suchergebnissen sind daher sowohl für die Trefferliste als Ganzes als auch für die einzelnen Treffer anzugeben, z.B.

- die Vollständigkeit der Treffer
- die Aktualität der Treffer
- die Relevanz bzw. Bewertung der Quelle, z.B. ob es sich um behördliche Informationen handelt
- der Bezug zu weiteren Ergebnissen (Links, Relationen).

(Lewandoski und Höchstötter 2007) führen für die Bewertung von Suchmaschinen vier Evaluationsbereiche an, die sich wiederum aus verschiedenen Evaluationsmaßen zusammensetzen:

- Qualität des Index (Vollständigkeit, Aktualität)
- Qualität der Suchresultate (im Vergleich zu anderen Suchmaschinen bzw. im Vergleich zur absolut zur Verfügung stehenden Information$^5$)
- Qualität der Suchfunktion (z.B. Möglichkeiten zur Filterung nach Metaattributen wie Sprache oder Dokumenttyp)
- Nutzerfreundlichkeit (Usability).

Für die vorliegende Arbeit sind aufgrund der Vorgaben (gegebene Menge von Zielsystemen, single search slot) vor allem die Qualität der Suchresultate und die Nutzerfreundlichkeit (insbesondere auch bei der Präsentation der Resultate) von Bedeutung.

Da die Bewertung der Suchresultate sehr stark von den Erwartungen des einzelnen Nutzers abhängt, soll die Evaluation anhand ausgewählter Anwendungsszenarien vorgenommen werden.

### 1.3 Ziele und Aufgaben

Das Ziel der vorliegenden Arbeit besteht darin, ein neues Konzept für die semantische Suche in heterogenen Informationssystemen zu Fragestellungen der Umwelt und Energie zu entwickeln. Dazu sind die folgenden wissenschaftlichen Zielstellungen zu untersuchen:

   Die Grundarchitektur umfasst die Zielsysteme, in denen gesucht werden soll, Hintergrundwissen in Form von Vokabularen und Gazetteer-Dienste und

---

$^5$ Eher ein theoretischer Fall, denn dafür sind Metainformationen notwendig, deren Vollständigkeit kaum geprüft bzw. belegt werden kann.
Webportale zur Präsentation der Suchergebnisse. Der Raum dazwischen bietet viele Freiheiten für eine Grundarchitektur und Ausprägungen davon.


5. **Ableitung von Aussagen zur Leistungsfähigkeit** des neuen Konzeptes aus den Erkenntnissen der Erprobung.

Wie bereits in Abschnitt 1.1 erwähnt wurde, gibt es Randbedingungen, die das Erstellen eines (Domänen-) spezifischen Recherche- und Informationsportals gegenüber der globalen Internetsuche vereinfachen.

Die Anforderungen und Ziele entsprechen in vielen Punkten den Anforderungen an die Suche innerhalb geschlossener Organisationen, z.B. Firmen-Intranets, die häufig unter dem Begriff "Enterprise Search" (Lange 2009) subsumiert werden. Auch hier sind die Datenquellen in der Regel bekannt und gut beschrieben. Im Gegensatz zu vielen Intranet-Systemen sind die jetzt betrachteten Datenquellen jedoch heterogener, insbesondere werden (nicht harmonisierte) Datenquellen externer Anbieter herangezogen, so dass die Bandbreite von wohlbekannten und wohldenierten Datenquellen (Semantik und Syntax bekannt) bis hin zu „fremden“ Datenquellen (Semantik weitgehend unbekannt, Daten semi- oder unstrukturiert) reicht.

### 1.3.1 Harmonisierung von Semantik

Im Gegensatz zu Internet-Suchmaschinen stehen für viele Informationssysteme zusätzliche (Meta-)Informationen zur Verfügung. Viele Datenquellen sind bekannt, enthalten strukturierte oder semi-strukturierte Daten und anhand von Metadatenbeschreibungen ist häufig auch die Semantik der Daten (bzw. ihrer Schemata) bekannt. Auf den ersten Blick scheint die semantische Suche in bekannten Datenquellen also einfacher als bei einer allgemeinen Internet-Suche, jedoch besteht auch hier eine Herausforderung in der Harmonisierung der Daten aus verschiedenen Quellen, mit anderen Wor-

1.3.2 Umgang mit unterschiedlichen Datentypen


1.3.3 Datenquellen, Datenfluss, Konsistenz

Eine weitere Nebenbedingung, die sich aus der Rolle eines Recherche- und Informationsportals ergibt, ist, dass auf Informationen im Wesentlichen lesend zugegriffen wird. Der unidirektionale Informationsfluss von einem Informationssystem in Richtung des Portals vereinfacht grundsätzlich den Zugriff, dennoch sind für jede Datenquelle Konsistenzbedingungen festzulegen und zu implementieren, insbesondere wenn Daten zwischengespeichert, gecached oder weiterverarbeitet werden. Die Konsistenzbedingungen sollen sicherstellen, dass im Rechercheportal und der originären Datenquelle zu jedem Zeitpunkt konsistente Sichten auf die Daten verfügbar sind bzw. Abweichungen innerhalb eines akzeptablen Bereiches, z.B. eines definierten maximalen zeitlichen Versatzes (Latenz), bleiben.

1.3.4 Nutzung von Standards


1.3.5 Freie Softwarekomponenten und Nutzung von Open Source

In einem Softwareprojekt wird sich immer wieder die Frage nach der Nutzung vorhandener Lösungen (Komponenten, Bibliotheken) stellen. Häufig gestellte nichtfunktionale Anforderungen an solche Komponenten sind deren kostenfreie Verfügbarkeit bzw. deren Vorliegen als Open Source Software. Die häufig kolportierten Vorteile von Open Source Lösungen (Kostenfreiheit/Lizenzmodell, einfache Verfügbarkeit, Anpassbarkeit, Anbieterunabhängigkeit, Stabilität und Sicherheit) sind jedoch im Einzelfall zu prüfen und gegen eventuelle Nachteile (Fehlen von Service Level Agreements bzw. professionellem Support, Verbot der kommerziellen Nutzung, offene Haftungs- und Gewährleistungsfragen) abzuwägen.

In Rahmen der vorliegenden Arbeit sollen freie und Open Source-Komponenten bevorzugt werden.

1.3.6 Abgrenzung

In der vorliegenden Arbeit sollen auch Konzepte und Technologien aus dem Bereichen der Wissensrepräsentation und des Semantic Web genutzt werden, es geht jedoch explizit nicht darum einzelne Informationssysteme für das Semantic Web (Web 3.0) aufzurüsten.

Die Wissensrepräsentation (Knowledge Representation) setzt sich dabei im Wesentlichen aus den Bereichen Logik (Strukturen zur Bildung von Regeln, die zum Schließen genutzt werden können), Ontologien (Definition von Konzepten zur Repräsentation von Objekten und Beziehungen dazwischen) und der Berechenbarkeit zusammen. Im Gegensatz zum bestehenden WWW der Dokumente wird das Semantic Web also einzelne Objekte enthalten, die eindeutig semantisch beschrieben und adressierbar sind. Grundvoraussetzungen hierfür sind Linked Data, Vokabulare, Mechanismen zum Abfragen (Queries) (s. Anhang A1.3) und Inferenzen (s. Anhang A1.4).

Wie oben beschrieben, können die Ideen und Prinzipien des Semantic Web als Richtschnur für Konzepte und Entwicklungen dienen, insbesondere um auch für die be-
schriebene semantische Suche von den zu erwartenden Entwicklungen in Richtung eines Web 3.0 profitieren zu können.


Es geht auch nicht darum, massenhaft Daten(-sätze) aus existierenden Informations- systemen als Instanzen in bestehende Ontologien zu integrieren, sondern vielmehr Klassen von Daten mit Hilfe von Vokabularen zu beschreiben und so eine semantisch eindeutige Einordnung von Daten aus verschiedenen Datenbeständen zu erreichen.

Es existieren eine ganze Reihe semantischer Suchmaschinen\(^7\), die den Anspruch haben, dem Nutzer auf einer Ergebnisseite eine konkrete Antwort zu einer bestimmten Frage zu geben. Jede Antwort wird auf Basis des vorhandenen Datenbestands (in einer sehr großen Datenbank) berechnet, ggf. als mathematisch korrektes Ergebnis der Anfrage „Wo befindet sich gerade die ISS?“ bei der semantischen Suchmaschine „Wolfram|Alpha“ (Wolfram|Alpha 2017). In der vorliegenden Arbeit geht es nicht um die Entwicklung einer solchen semantischen Suchmaschine, dazu müssten die Daten wie oben beschrieben vollständig in eine entsprechende Datenbasis, z.B. Ontologie, überführt werden.

### 1.3.7 Was ist neu an der vorliegenden Arbeit?

Es gibt eine Vielzahl von (Internet-)Suchmaschinen, meistens textvergleichbasiert, aber auch einige mit semantischer Unterstützung oder vollständig semantischer Erfassung von Suchanfragen. Für Internet-Suchmaschinen ist eine semantische Suche

---


\(^7\) Beispiele für semantische Suchmaschinen:

- **AskWiki**: Semantische Suchmaschine für den Datenbestand der deutschsprachigen Wikipedia mit Eingabe der Anfrage per Sprache
  

- **GoPubMed**: Semantische Suchmaschine für die biomedizinische Domäne
  

- **Swoogle**: Semantische Suchmaschine, die Dokumente, Begriffe und Daten im semantischen Web suchen kann;
  
  [http://swoogle.umbc.edu/](http://swoogle.umbc.edu/)

- **WolframAlpha**: „Antwortmaschine“ des Mathematikers Stephen Wolfram mit Schwerpunkt auf den exakten Wissenschaften;
  
Einleitung

(Verstehen der Suchanfrage und liefern einer spezifischen Antwort) prinzipiell schwierig, da sie bei Suchanfragen mit der gesamten natürlichen Sprache und deren vollständigem Wortgut umgehen müssen. Zum Beispiel erschweren Mehrdeutigkeiten (Homonyme), Synonyme, Flexionen und Zusammensetzungen (insbesondere in der deutschen Sprache) das Erkennen der (möglichst eindeutigen) Bedeutung einer Suchanfrage – das kann ggf. durch Rückfragen und eine Auswahl durch den Nutzer aufgelöst werden. Für einige der beschriebenen Probleme existieren Lösungsansätze, z.B. hinterlegte Wörterbücher, Synonymlisten, Thesauri etc.

Eine deutliche Einschränkung von Internet-Suchmaschinen stellen jedoch die durch sie erschlossenen Inhalte dar: Die meisten Internet-Suchmaschinen bieten nur Zugang zu solchen Inhalten, die über eine WWW-Repräsentation (HTML-Seiten) verfügen oder per URL als Datei-Download (Dokument) verfügbar sind. Andere Systeme (wie Datenbanken) werden meist nicht durchsucht.


Im Gegensatz zu Internet-Suchmaschinen stehen für die Aufgabenstellung der vorliegenden Arbeit weitergehende Informationen zur Verfügung, die in einer Architektur für die übergreifende Suche in heterogenen Informationssystemen bzw. verschiedenen Ausprägungen einer solchen Architektur, zusammenzuführen sind:

- Die Domäne(n) der Suchanfragen ist/sind beschränkt und bekannt. Es handelt sich um die Domänen „Umwelt“ und „Energie“. Damit ist das Hintergrundwissen und die Anzahl der ihm zugrundeliegenden Vokabulare beschränkt. Sie lassen sich daher auch in praktischen Projekten für die Suche verwenden.

- Ebenso sind die angeschlossenen heterogenen Datenquellen bekannt und in ihrer Anzahl beschränkt. Sie enthalten strukturierte oder semi-strukturierte Daten und anhand von Metadatenbeschreibungen ist häufig auch die Semantik der einzelnen Daten (bzw. Schemata/Datenquellen) bekannt.

- Auf den ersten Blick scheint die Suche in solchen Datenquellen also einfacher, jedoch besteht die Herausforderung in der Harmonisierung der Daten aus den verschiedenen Quellen, mit anderen Worten: Der Übersetzung der Suchanfrage in die jeweilige Semantik (und in die technische Repräsentation der Anfrage) der einzelnen Zielsysteme, die Interpretation der Ergebnisse (nach der Semantik) und die Zusammenführung der Daten in einer einheitlichen Repräsentation der Suchergebnisse (Ergebnis-Mashup).

Einleitung


Das beschriebene Vorgehen ist neu für das Umweltinformationssystem Baden-Württemberg, in welchem bisher inselartige Suchlösungen (für einzelne Systeme) oder textvergleichbasierte Suchlösungen, z.B. im Umweltportal, verwendet werden, die jedoch im Wesentlichen auf die Suche in Websystemen und Dokumentenbeständen beschränkt sind.


Im Gegensatz zu vielen kleinen bis mittleren Firmen-Intranets sind die hier verwendeten Datenquellen jedoch heterogener, insbesondere werden (nicht harmonisierte) Datenquellen externer Anbieter herangezogen, so dass die Bandbreite zwischen wohlbekannten und wohldefinierten Datenquellen (Semantik und Syntax bekannt, strukturierte Daten) bis hin zu „fremden“ Datenquellen (Einstiegs punkt/Schnittstelle bekannt, Semantik weitgehend unbekannt, semi- oder unstrukturierte Daten) geht.

1.4 Übersicht über die Arbeit

Im folgenden Kapitel 2 werden grundlegende Prinzipien für die Architektur einer neuartigen semantischen Suche (für die Domänen Umwelt und Energie) entwickelt. Dabei werden bewusst Freiheitsgrade gegeben, die verschiedene Umsetzungen nach den entwickelten Prinzipien zulassen. Dadurch soll eine „Evolution“ der semantischen Suche ermöglicht werden, die eine Evaluierung der konkreten Umsetzungsvarianten in möglichst realen Umgebungen ermöglichen soll. Startpunkt der Evolution ist ein Suchportal mit einer klassischen Volltextsuche – im Gegensatz zu einer semantischen Suche.

Kapitel 4 illustriert die verschiedenen Evolutionsstufen anhand konkreter Umsetzungsbeispiele, die teilweise produktiv im Einsatz sind oder waren. Das soll den Anspruch der vorliegenden Arbeit betonen, eine praxistaugliche Architektur und tatsächlich nutzbare Systeme zu entwickeln.

Die Gegenüberstellung, Diskussion und Bewertung der beschriebenen Architekturvarianten ist in Kapitel 5 beschrieben. Kapitel 6 zieht ein abschließendes Fazit und bietet einen Ausblick für weitere Entwicklungen.
Ein neues Konzept für die semantische Suche

Im zweiten Kapitel soll ein neues, allgemeines Konzept für die semantische Suche in heterogenen Informationssystemen entwickelt werden. Das allgemeine Grundkonzept wird anschließend in verschiedenen Architekturvarianten umgesetzt werden, die sich im Einsatz verschiedener Technologien zur Realisierung architektonischer Grundaufgaben unterscheiden.

2.1 Grundidee und Übersicht

Abbildung 2 zeigt die wesentlichen Komponenten des neuen Konzeptes für die semantische Suche.


Abbildung 2: Rahmen einer allgemeinen Architektur für die semantische Suche
Alle weiteren Komponenten der Grundarchitektur werden zunächst als offen angenommen, d.h. grundsätzlich können beliebige Module, die einen Beitrag zur Erreichung der Ziele leisten können, hinzugefügt werden – selbstverständlich mit dem Ziel, den Gesamtaufwand, d.h. die Anzahl und die Komplexität der Module, möglichst gering zu halten.

Die Grundarchitektur besteht daher aus wenigen zentralen Grundaufgaben:

1. Erfassung der Semantik von Zielsystemen \( Z_S \)
2. Harmonisierung der Semantik verschiedener Zielsysteme (Vokabulare \( V_i \))
3. Realisierung des technischen Zugriffs auf die Zielsysteme \( Z_S \)
4. Klassifikation des Typs von Informationen anhand generischer Datentypen
5. Realisierung der Darstellung von Beziehungen zwischen Daten/Objekten
6. Erfassung der Semantik der Suchanfrage
7. Nutzung von Zusatzinformationen zur Anreicherung bzw. Explizierung der Suchanfrage (z.B. Koordinaten zu Ortsnamen) mit Hilfe der Gezetteer-Dienste \( G_i \)
8. Nutzung der Semantik der Suchanfrage zur Abfrage der Zielsysteme
9. Koordination der Ergebnisdarstellung (Generierung eines Mashup)

Neben den obligatorischen Grundaufgaben gibt es einige optionale Bestandteile, die zur Erfüllung weitergehender Aufgaben genutzt werden können, z.B. die Bereitstellung von Daten als Linked Data im Sinne des Semantic Web (Berners-Lee 2006):

11. Bereitstellung von Daten über eindeutige Bezeichner (URIs)
12. Darstellung von Beziehungen zwischen Daten (Objekten) mit Hilfe von URIs

Die erste Hauptaufgabe besteht in der Harmonisierung der Semantiken der verschiedenen Zielsysteme \( Z_S \), respektive der ihnen zugrundeliegenden Vokabulare \( V_i \). Das kann z.B. durch die Zusammenfassung zusammengehörender/identischer Konzepte aus den verschiedenen Vokabularen oder über die Abbildung auf ein (zu definierendes) übergreifendes Vokabular geschehen.


Die verschiedenen Bausteine des neuen Konzeptes werden im Folgenden beschrieben. Aus ihnen ergibt sich eine Gesamtarhitektur mit einer ganzen Reihe von Freiheitsgraden bei der Entwicklung, die zur Implementierung verschiedener konkreter Architekturvarianten führt, die in Kapitel 3 beschrieben werden.
2.2 Zielsysteme

Als Zielsysteme $ZS_i$ kommen alle Informationssysteme infrage, die Daten für die gegebenen Domänen zur Verfügung stellen können. Dabei ist es zunächst weitgehend irrelevant, in welcher Form die Daten vorliegen bzw. welche Schnittstellen zum Zugriff auf die Daten bereitgestellt werden. Dennoch müssen für die Zielsysteme einige Festlegungen getroffen werden.

2.2.1 Definition

Jedes Zielsystem $ZS_i$ stellt genau eine Klasse von Daten (Objekten) zur Verfügung, d.h. alle Daten gehören zu genau einem semantischen Konzept. Das bedeutet, dass Zielsysteme in erster Linie strukturierte Daten mit einer gegebenen Semantik zur Verfügung stellen.

Was für viele Informationssysteme, die im Allgemeinen Objekte verschiedener Konzepte sowie Beziehungen dazwischen enthalten, zunächst eine Einschränkung darstellt, lässt sich jedoch leicht auflösen, denn jedes Informationssystem, das verschiedene Konzepte verwaltet, lässt sich als Menge mehrerer Informationssysteme darstellen, die jeweils nur ein Konzept bereitstellen. Zielsysteme stellen Daten elektronisch per Netzwerk (Internet) erreichbare Schnittstellen zur Verfügung. Dabei ist die Nutzung standardisierter Schnittstellen und Formate zwar hilfreich, jedoch keine obligatorische Anforderung, da der Zugriff z.B. über Adapter (Gamma 2004) oder Fassaden (Gamma 2004) mit entsprechenden Transformationen realisiert werden kann. Der Zugriff auf Zielsysteme kann auch indirekt, z.B. über Indexe oder strukturierte Suchmaschinen stattfinden, wenn die Konsistenz der redundant gehaltenen Daten gewährleistet ist.

2.2.2 Zielsysteme mit un- bzw. schwach strukturierten Inhalten


Der Konsistenzbegriff ist für jedes Zielsystem zu definieren.
Beziehungen zu Konzepten bzw. Objekten haben. Dennoch lassen sich auch Dokumente klas
sifizieren, indem ihnen Beziehungen zu bekannten Konzepten zugordnet
werden. Dabei können sowohl der Inhalt als auch Metainformationen genutzt werden,
z.B.

- „Dokument X“ – „ist-ein“ – „Forschungspapier“, 

Es gibt eine ganze Reihe von Diensten, die eine automatisierte Klassifizierung von
Dokumenten für ein gegebenes Vokabular übernehmen können11.

Un- und schwach strukturierte Daten können alternativ, ohne Erfassung ihrer Seman-
tik, auch über „klassische“ Suchmaschinentechnologie, d.h. durch Verwendung der
originalen Suchanfrage, bereitgestellt werden.

2.2.3 Semantik von Zielsystemen

Jedes Zielsystem kann für die Darstellung seiner Daten (Objekte) eine eigene Sema-
tik enthalten, die sich z.B. in der Namensgebung, der Auswahl von Attributen, der Ver-
wendung bestimmter Formate und Einheiten, der Verwendung von Ober- und Unter-
konzepten etc. ausdrückt.

In vielen Fällen liegt die Semantik nicht explizit und in durch Maschinen prozessierba-
rer Form vor, sondern steckt direkt in den Daten oder in Metainformationen, z.B. in
beschreibenden Dokumenten (Holten 1999).

Um eine Harmonisierung der Semantik verschiedener Zielsysteme erreichen zu kön-
nen, muss die Semantik jedes einzelnen Zielsystems explizit erfasst werden, um auf
die Semantik der anderen Zielsysteme abgebildet werden zu können.

Die Einschränkung, dass Zielsysteme so aufgefasst werden, dass sie jeweils nur Daten
eines einzelnen Konzeptes enthalten dürfen, vereinfacht die Beschreibung ihrer Sem-
antik.

2.2.4 Generische Datentypen

Da Daten aus einer Vielzahl von Zielsystemen in einer integrierten Ergebnisdarstellung
präsentiert werden sollen, ist es notwendig, deren Darstellungen generisch zu halten.

Im Laufe der Arbeit wurden für die Bereiche Umwelt und Energie im Wesentlichen
sechs Klassen von Datentypen (s. Anhang A2 ) identifiziert, die sich generisch behan-
deln lassen:

- Objektinformationen

Ein neues Konzept für die semantische Suche

- Einzelne Objekte
- Listen von Objekten
  - Geodaten bzw. Daten mit explizitem Geobezug (beinhalten in der Regel Objektinformationen)
    - Einzelne Objekte
    - Mengen von Objekten (z.B. Kartenlayer)
  - Zeitreihen bzw. Messdaten (mit einem Zeitbezug einzelner Messwerte)
  - Tabellarische Informationen
  - Metadaten, in verschiedenen Granularitäten und Ausprägungen
  - Binäre Assets (inkl. zugehöriger Metadaten)
    - Durchsuchbare Assets (Dokumente)
    - Mediendateien (Bilder, Videos, Audios).

Die Daten eines einzelnen Zielsystems können dabei gleichzeitig zu mehreren Klassen gehören, d.h. es kann verschiedene Schnittstellen geben, die verschiedene Sichten auf Daten erlauben, z.B. Windkraftanlagen als Objekte, als Objekte mit Geobezug und als tabellarische Informationen.


### 2.3 Vorverarbeitung der Suchanfrage

Suchanfragen lassen sich wie Dokumente gegen ein vorhandenes Vokabular klassifizieren und so auf die darin enthaltenen Konzepte bzw. Deskriptoren abbilden.

Die so gewonnenen Informationen können verwendet werden, um die Suchanfrage anzureichern, z.B. der Suchanfrage weitere Attribut-Wert-Paare oder Strukturinformationen (z.B. Oberbegriffe, Synonyme) hinzuzufügen.

---

12 Semantisch gibt es hier Überschneidungen mit (Listen von) Objektinformationen, z.B. Resultate von DB-Abfragen.

13 Je nach Kontext können z.B. auch Suchergebnisse einer Volltextsuchmaschine als Metadaten betrachtet werden.

14 Inklusive Unterstützung der Anfrage-Schnittstelle.
Ein neues Konzept für die semantische Suche


Orts- und Zeitangaben nehmen hier eine Sonderstellung ein. Sehr viele Objekte haben einen Ortsbezug (Standort, ggf. zeitlich veränderlich\textsuperscript{15}). Messwerte und Zeitreihen haben einen Zeitbezug. Insofern stellen Ort und Zeit universelle Größen dar, bezüglich derer Objekte miteinander in Beziehung gebracht werden können („ist in der Nähe von“, „ist innerhalb von“, „ereignet sich zur selben Zeit“ etc.), ohne dass eine Beziehung zunächst explizit vorliegt. Gerade bei der Suche nach konkreten Objekten stellt der Ortsbezug einen gängigen Anwendungsfall dar.

2.4 Abbildung und Harmonisierung von Vokabularen


Um dennoch eine einheitliche semantische Suche realisieren zu können, muss eine übergreifende Nutzung von Vokabularen ermöglicht werden, z.B. durch Abbildung von gleichen/ähnlichen Konzepten aufeinander oder durch die Darstellung von Beziehungen zwischen Konzepten verschiedener Vokabulare („is-a“, „is-in-semantic-relation-with“).

Hinzu kommt die Notwendigkeit zur Beschreibung der Semantik von Systemen, bei denen die Semantik nicht explizit vorliegt, z.B. in Form von zusätzlichen Metadaten.

2.5 Integrierte Ergebnisdarstellung (Mashup)


Dabei sollte es für den Nutzer einerseits keine Rolle spielen, dass die dargestellten Daten aus unterschiedlichen Quellen kommen, andererseits sollte es ihm auch ermög-

\textsuperscript{15} Zum Beispiel wechseln mobile Messstationen regelmäßig ihren Standort.

\textsuperscript{16} In der vorliegenden Arbeit wird immer von einer Kombination von Konzepten und den zugehörigen Schemata ausgegangen.
licht werden, die dargestellten Daten fachlich zu selektieren bzw. zu filtern, z.B. die Darstellung aller Windkraftanlagen auf der Karte ein- bzw. auszuschalten.


2.6 Verbindende Schicht zur Beschreibung und Realisierung von Anwendungen

Zur Erzeugung integrierter Ergebnisdarstellungen (Mashups) bedarf es also der Koordinatien zwischen den einzelnen Darstellungskomponenten bzw. auch der Kommunikation der Komponenten untereinander.

Die Koordination muss berücksichtigen, welche Zielsysteme Daten in welcher Form liefern können, welche Klassen von Datentypen im (gesamten) Suchergebnis vorliegen und ggf. welche Arten der Darstellung überhaupt gewünscht sind, d.h. ob es z.B. personalisierte Einstellungen/Präferenzen des Nutzers gibt oder ob ein Redakteur für einen bestimmten Anwendungsfall bereits Voreinstellungen getroffen hat, z.B. dass eine bestimmte Zusammenstellung von Ansichten für einen Anwendungsfall bereits existiert.


Im folgenden Kapitel werden verschiedene konkrete Architekturvarianten präsentiert, welche jeweils einige, mehrere oder alle der vorgestellten Bausteine enthalten und die entsprechenden Aufgaben umsetzen. Die verschiedenen Architekturvarianten bauen dabei aufeinander auf, d.h. die in einer Architekturvariante gemachten Erfahrungen gehen in die Entwicklung der folgenden Architekturvarianten ein.

Das Vorgehen ist pragmatisch. Es setzt auf der in den Landesumweltportalen Baden-Württemberg, Sachsen-Anhalt und Thüringen vorhandenen Suchfunktionalität (einer kommerziellen Suchmaschine, Stand 2009) auf, und versucht sie sukzessive zu ver-
Ein neues Konzept für die semantische Suche

bessern, wobei die Komplexität des Gesamtsystems sowie die Menge der über die Suche verfügbar gemachten Daten stetig zunimmt.
3 Architekturvarianten

3.1 Übersicht

In Kapitel 3 werden vier Architekturvarianten in ihrer zeitlichen Entwicklung gegenübergestellt. Die Varianten entstanden in Form von Evolutionsstufen der in Kapitel 2 entwickelten neuen Architektur.


Die erste Variante, eine Art Bestandsaufnahme des Status Quo, stellt den Ausgangspunkt der Überlegungen dar und offenbart dabei gleich so viele Schwächen, dass eine größere Zahl von Anforderungen und Prinzipien daraus abgeleitet werden können.

3.2 Grundlagen

Zunächst wurde das Konzept für die semantische Suche mittels SearchBroker anhand einer rein serverseitigen Anwendung erstellt (Abecker et al. 2009a; Bügel et al. 2010). Im Laufe der Zeit ergab sich jedoch, vor allem getrieben durch Einführung von HTML5, ein Quantensprung in den clientseitigen Möglichkeiten von Webanwendungen. Viele Probleme, sowohl technische wie die Cross-Origin Policy (W3C 2009a) als auch „weiche“, wie die breite Nutzer-Akzeptanz von JavaScript, waren gelöst, und rund um HTML5 entstanden eine große Menge von Frameworks und Bibliotheken, die eine Verschiebung von Mechanismen vom Server auf den Client, d.h. in den Browser, möglich machten. Funktional reiche HTML5-Anwendungen waren möglich, ebenso wie die Nutzung bzw. der Aufbau serviceorientierter Architekturen (SOA) (s. Anhang „Serviceorientierte Architekturen“) in Informationssystemen, die ebenfalls durch das Aufkommen leichtgewichtiger Technologien wie REST (REpresentational State Transfer, s. Anhang A4.2) oder JSON gefördert wurden (Zustandslose Client-Server-Kommunikation mit identifizierbaren Ressourcen als zentrale Dekompositionseinheit, auf Basis uniformer Schnittstellen, die sich nahtlos in das Konzept „Hypermedia“ einfügt. „Leichtgewichtig“ steht dabei z.B. im Gegensatz zu klassischen Webservices, z.B. „SOAP“).

3.2.1 Server-Zentrierung

Die Konzentration auf eine rein serverseitige Lösung barg also plötzlich erhebliche Nachteile, insbesondere was die Möglichkeiten zum Aufbau von ergonomischen, funk-
Architekturvarianten

Die wesentlichen Nachteile einer serverzentrierten Architektur sind (Wang et al. 2015):

- Bei jedem Laden der Seite entsteht eine ganze Kette von Zugriffen auf Hintergrund-Dienste wie Datenbanken oder externe Dienste (Wang et al. 2015)ste, selbst wenn dabei dieselben Abfragen erneut ausgeführt werden. Caching-Mechanismen können zwar zur Reduzierung beitragen, das Grundproblem besteht jedoch grundsätzlich weiter.

Die Serverzentrierung bietet jedoch auch Vorteile (Wang et al. 2015; Wang et al. 2014; Wang et al. 2015; Guo et al. 2009):

- Die Anwendung kann für eine konkrete Laufzeitumgebung entwickelt und optimiert werden. Dabei können Technologien optimal aufeinander abgestimmt werden.
- Die gesamte Infrastruktur kann für den Zugriff auf Hintergrundsysteme optimiert werden.

3.2.2 Client-Zentrierung

Die im Client (Browser) laufende Anwendung macht es nun möglich, direkt und ohne Umweg über den Server (Neuladen der Seite) auf Interaktionen und Ereignisse zu reagieren, insbesondere die Darstellung zu verändern (z.B. Sortierung in einer Tabelle) oder bei Bedarf weitere Inhalte nachzuladen.

Hierzu ist der Zugriff auf entsprechende Datendienste notwendig, die auf dem Heimatserver der Anwendung, jedoch auch auf davon unabhängigen Servern, bereitgestellt werden können.

Viele Probleme der rein serverseitigen Anwendungen können so gelöst werden:

- Flaschenhals Server-Anwendung
- Vermeidung des Nachladens von ganzen HTML-Seiten
- Hoch-interaktive und ergonomische Benutzeroberflächen.


Manche Teile einer Anwendung lassen sich ggf. nicht ohne Weiteres komplett auf den Client übertragen, insbesondere wenn damit der Zugriff auf bzw. die Verarbeitung von großen Datenmengen verbunden ist. Hier ist es meist sinnvoll, die betroffenen Teile der Anwendung in einen vom Client nutzbaren Service umzubauen, so dass die eigentliche Verarbeitung auf einem leistungsfähigen Server abläuft (auf dem ggf. auch die Daten gehalten werden und direkt zur Verfügung stehen), und im Client nur ein Stellvertreter (Proxy, Stub) des serverseitigen Dienstes vorhanden ist, der sich um den Aufruf des Dienstes und das Durchreichen der Ergebnisse kümmert. Der aus obigen Überlegungen resultierende hybride Ansatz wird im folgenden Abschnitt beschrieben.

### 3.2.3 Hybrider Ansatz


Es bietet sich an, hierfür entsprechende Dienste zu implementieren, die zum Beispiel über eine REST-Schnittstelle aufgerufen werden können. Andere Teile des Search-Brokers lassen sich ebenfalls als solche Dienste realisieren, tatsächlich wurden Dienste zum Ermitteln des Ortsbezugs einer Suchanfrage auch im serverzentrierten Ansatz
bereits über externe Gazetteer-Dienste implementiert, die auch direkt in der Client-Seite genutzt werden könnten.

3.2.4 Nutzung von Web-Widgets


Die Parametrisierung der Widgets erfolgt sowohl durch entsprechende Konfigurationsdateien als auch durch die clientseitige Kommunikation der Widgets untereinander, bei der beispielsweise Informationen zur Änderung des Ortsbezugs ausgetauscht werden. Jedes Widget kann individuell auf die Ereignisse reagieren und ggf. Daten nachladen oder die Darstellung ändern, ohne dafür die ganze Seite neu laden zu müssen.


3.2.5 Kopplung von Web-Widgets per Eventbus

Das Konzept kleiner unabhängiger Softwarekomponenten innerhalb einer HTML-Seite impliziert, wie oben beschrieben, die Notwendigkeit des Datenaustauschs zwischen Komponenten. Wird z.B. der Darstellungsbereich einer Kartenkomponente durch das Verschieben des Ausschnitts oder durch Zoomen verändert, sollen andere Komponenten, die Inhalte bezüglich des angezeigten Ortes filtern, auf die Veränderung hingewiesen werden und ihre Darstellung entsprechend anpassen. Die Ergänzung eines Begriffs im Suchschlitz soll das Neuladen von Suchergebnissen triggern, jedoch können auch andere Komponenten an den Suchbegriffen interessiert sein, z.B. sollen passen-


### 3.3 Erste Architekturvariante: Semantische Erweiterung von Suchanfragen und Nutzung externer Datenquellen durch die Volltextsuchmaschine

Die LUPO Landesumweltportale (Schlachter et al. 2008; Schlachter et al. 2014b) als fachliche Rechercheportale verwenden seit 2008 kommerzielle Suchmaschinentechnologie, hauptsächlich zur Indizierung von un- und semistrukturierten Daten, insbesondere von Websites und Dokumentenbeständen, aber auch von ausgewählten strukturier-ten Datenbeständen, z.B. aus relationalen Datenbanken. Die bis heute im Einsatz befindliche Suchmaschine Google Search Appliance (GSA) wurde unter anderem wegen der hohen Relevanz ihrer Suchergebnisse ausgewählt, jedoch bereits bei der Erstellung von Kriterien zur Auswahl einer Suchmaschine und bei den ersten Konzepten zu deren Einbindung in die Landesumweltportale spielten strategische Überlegungen bezüglich der semantischen Erweiterung von Suchanfragen sowie die Einbeziehung wei-
Architekturvarianten

In Architekturvarianten (außerhalb des Volltextindex) eine zentrale Rolle (Schlaicher et al. 2008). Die strategischen Überlegungen zielen auf das Verfügbarmachen von Umweltinformationen als Linked Data für das Semantic Web.

Die entsprechende Umsetzung ist in Abbildung 3 dargestellt. Die Suchmaschine Google Search Appliance (GSA) greift direkt auf die Zielsysteme \( ZS_i \) zu. Alle gelesenen Daten, dabei handelt es sich im Wesentlichen um unstrukturierte bzw. semistrukturierte Daten wie HTML-Seiten oder PDF-Dokumente, werden im Index der Suchmaschine abgelegt. Die Suche erfolgt im Kern textvergleichbasiert mit vielfältigen Relevanzkriterien und kann verschiedene Wörterbücher, z.B. der deutschen Sprache (Wortstämme), nutzen.

Als mögliches gemeinsames Vokabular steht der Umweltthesaurus UMTES des Umweltbundesamtes (Domäne „Umwelt“) zur Verfügung, der im Rahmen der Semantic Network Services (SNS) des Umweltbundesamtes (Umweltbundesamt 2016) elektronisch bereitgestellt wird. Das Vokabular kann die Suche ergänzen, indem Informationen aus dem Umweltthesaurus, z.B. Synonymketten (Benennung gleichartiger Objekte anhand verschiedener Suchbegriffe), in die Suche integriert werden.

Abbildung 3: Umsetzung auf Basis einer vorhandenen Volltextsuchmaschine (neue Entwicklungen und eigene Anteile in rot)
Weitere, bisher nicht indexierte Datenquellen bzw. Zielsysteme (im Bild $ZS_2$ und $ZS_4$) lassen sich, falls notwendig über Adapter (im Bild $Adapter_2$ bzw. $Adapter_4$), über die OneBox-Schnittstelle der GSA parallel zur Index-basierten Suche abfragen und zusammen mit der klassischen Volltext-Trefferliste präsentieren.

### 3.3.1 Semantische Erweiterung von Suchanfragen

Die wesentliche Motivation für die semantische Erweiterung der Suchanfragen war die Beobachtung, dass die von Anwendern verwendeten Suchbegriffe häufig nicht der in Fachdokumenten gebräuchlichen Fachterminologie entsprachen und die auf dem Vergleich von Zeichenketten basierende Suchmaschine so häufig keine oder zu wenige Treffer lieferte. Durch die Integration eines Fachthesaurus (GEMET/Semantic Network Service (SNS)) (Bandholtz; Umweltbundesamt 2016; Angrick et al. 2002; Rüther und Bandholtz 2008), der in seinen Synonymketten auch umgangssprachliche Begriffe enthielt, als Suchmaschinen-Vokabular $V_S$ konnte mit relativ geringem Aufwand eine deutliche Verbesserung erreicht werden, eine Suche nach „Müll“ enthielt beispielsweise auch Treffer zum behördlichen Begriff „Abfall“ und dem Unterbegriff „Schrott“.

Neben der Erweiterung der Suchanfragen durch Wörterbücher und Thesauri ist mit der GSA auch die manuelle Pflege von besonders relevanten Ergebnissen für bestimmte Suchanfragen (Key-Matches) (Google Inc. 2014a) möglich. So kann zum Beispiel bei der Eingabe eines Begriffes wie „Feinstaub“ auf eine passende Webseite hingewiesen werden.

Zusätzlich lassen sich in der Ergebnisliste Hinweise auf verwandte Suchbegriffe (Google Inc. 2014b) einblenden, z.B. um Benutzer beim Aufkommen aktueller Umweltfragen wie „Feinstaub“ oder „Gammelfleisch“ noch vor der Aufnahme in einen Fachthesaurus auf die entsprechenden Fachtermini („PM10“ bzw. „Lebensmittelhygiene“) hinzuweisen.

### 3.3.2 OneBoxes zur Einbindung externer Datenquellen

Die Notwendigkeit einer Möglichkeit zur Einbeziehung weiterer Informationssysteme (außerhalb des Suchindex) in eine Suchanfrage hat im Wesentlichen drei Gründe:

1. Nicht alle Quellsysteme lassen sich mit den üblichen Mechanismen (Web-Crawler, Verfügbare Konnektoren der Suchmaschine) indexieren
2. Eine Indizierung vieler Datentypen durch eine Suchmaschine ist nicht sinnvoll (z.B. häufig aktualisierte Messdaten)

Die GSA bietet eine flexible Möglichkeit zur Erweiterung der Ergebnisliste, sogenannte OneBoxes (Google Inc. 2015). Zum Beispiel ausgelöst durch die Verwendung bestimmter Suchbegriffe können neben der eigentlichen Ergebnisliste weitere Suchergebnisse eingebunden werden. Solche Suchergebnisse können entweder aus einer
parallelen Suche auf einem Teilbereich des Suchindex kommen oder durch die Online-
Abfrage weiterer externer Informationssysteme gewonnen werden.

Die externen Informationssysteme müssen hierzu die (proprietären) OneBox-
Schnittstelle implementieren, die als Eingabe im Wesentlichen die verwendeten Such-
begriffe in Form eines URL-Parameters bekommt. Für die Ausgabe und Datenübermit-
lung an die GSA ist ein (ebenfalls proprietärem) XML-Dialekt festgelegt, der für jeden
Treffer die Angabe eines Titels sowie einer URL (als Sprungziel) verlangt. Darüber
hinaus lässt das Format für jeden Treffer eine generische Liste weiterer Schlüssel-
Wert-Paare zu.

Obwohl zum Zeitpunkt der Einführung der GSA für die Landesumweltportale keine ents-
sprechenden OneBox-kompatiblen Dienste existierten, konnten dank des simplen Me-
chanismus innerhalb weniger Wochen eine ganze Reihe externer Datenquellen an die
GSA angeschlossen und damit innerhalb der Landesumweltportale verfügbar gemacht
werden. Dazu wurden meist kleine Proxy-Dienste als Adapter-Software implementiert,
d.h. kleine Softwaremodule, die auf der einen Seite die OneBox-Schnittstelle imple-
mentierten und auf der anderen Seite an eine existierende Schnittstelle oder Daten-
quell angebunden waren. Inhaltlicher Schwerpunkt waren aktuelle Messdaten wie
Pegelwerte oder Luftqualitätsdaten sowie die Suchschnittstellen weiterer fachlich rele-
vanter Portale (Statistische Landesamt, Verwaltungsportal service-bw).

Auch wenn die Simplizität des OneBox-Mechanismus auf der einen Seite zum schnel-
len Anschluss weiterer Datenquellen beigetragen hat, offenbarte er auf der anderen
Seite sehr schnell Grenzen. Insbesondere die Beschränkung der "Eingabe" auf die bei
der Suchanfrage verwendeten Suchbegriffe stellte eine deutliche Einschränkung dar:

- OneBoxes profitieren nicht von der semantischen Erweiterung der Suchanfrage
durch Wörterbücher und Thesauri.
- OneBoxes müssen Daten letztlich auf Basis von einem oder wenigen Begriffen
liefern, ihnen stehen keine weiteren Informationen und kein weiterer Kontext zur
Verfügung.
- OneBoxes agieren autark, d.h. ohne Informationen darüber, ob parallel weitere
Informationssysteme angefragt wurden bzw. welche Daten von dort geliefert
wurden.

Konsequenz daraus ist, dass z.B. Pegel (zur Bestimmung des Wasserstandes in Ge-
wässern) nur auf Basis des Namens (meist ein Ortsbegriff wie "Maxau") oder des Ge-
wässernamens ("Rhein") gefunden werden. Wird ein davon abweichender Ortsbegriff
wie "Karlsruhe" verwendet, der in den Metadaten keines Pegels enthalten ist, liefert der
OneBox-Dienst keine Treffer. Um die Funktionalität einer Ortssuche mit Umkreissuche
als OneBox realisieren zu können, muss der OneBox-Dienst die Explizierung der ent-
haltenen Ortsbegriffe sowie die Umkreissuche selbst implementieren, z.B. durch Nut-
zung eines Gazetteer-Services, vorausgesetzt die Metadaten der Pegel enthalten eine
explizite Ortsinformation (Geokoordinaten). Hat ein OneBox-Dienst die Ortssuche tat-
sächlich realisiert, steht die gewonnene Information jedoch weiteren OneBoxen nicht zur Verfügung.

Ein weiterer Nachteil des OneBox-Mechanismus besteht in der Art und Weise wie OneBox-Dienste durch die GSA aufgerufen und ihre Ergebnisse verarbeitet werden. Der Anspruch, die Ergebnisse von OneBox-Diensten gemeinsam mit dem Suchergebnis auszuliefern, setzt der möglichen Verarbeitungsdauer inklusive Latenzen eine (konfigurierbare) Obergrenze. Die Qualitätsanforderungen an die Google-Volltextsuche zwingen alle angefragten OneBoxen innerhalb eines bestimmten Timeouts zu antworten. Liegt der GSA bis zum konfigurierten Zeitpunkt keine Antwort einer OneBox vor, so werden die Suchergebnisse ohne die möglichen Zusatzinformationen der betreffenden OneBox ausgeliefert und stehen somit in der Trefferansicht nicht zur Verfügung.

Umgekehrt wird aber auch die GSA-Suche durch lange laufende OneBoxes regelmäßig bis zum eingestellten Timeout (z.B. 1000 ms) ausgebremst, so dass sich die Suche für den Nutzer spürbar verlangsamen kann.

Das Ausliefern eines monolithischen Ergebnis-Dokuments (XML) mit allen Teilergebnissen führt dazu, dass zur Anzeige der Suchergebnisse eine Abbildung von Teilergebnissen auf die passenden Anzeigekomponenten notwendig ist, d.h. sie muss in irgendeiner Form konfiguriert oder programmiert werden, im Falle der WebGenesis-basierten Umweltportalen die Aufgabe eines Dispatchers, der den Daten eine Reihe von Templates zuordnen konnte.

In den Webgenesis-basierten Landesumweltportalen (Schlacher et al. 2008) werden die Trefferansichten komplett serverseitig prozessiert. Durch die Verwendung von Templates wird bereits im Server fertiger HTML-Code erzeugt, der so an den Client (Webbrowser) übertragen wird, was letztlich zu starren und statisch wirkenden Anzeigen führt.

Eine direkte Auswirkung der serverseitigen Erzeugung von kompletten Trefferansichten ist, dass Interaktionen des Nutzers jeweils zu Änderungen der Suchanfrage führen, und damit jeweils ein Neuladen des gesamten Suchergebnisses (inklusive aller OneBoxen) nötig ist, d.h. Teilergebnisse können nicht unabhängig von anderen nachgeladen werden.

3.3.3 Bewertung
Auch wenn der OneBox-Mechanismus durchaus Schwächen hat, so hat er sich für einige Anwendungsfälle bewährt, z.B. für die Suche in Metadaten zu Medien oder die kaskadierende Suche in weiteren Suchmaschinen (Statistisches Landesamt, service-bw), und wurde im Bereich der LUPO Landesumweltportale jahrelang intensiv genutzt.

Es ergeben sich jedoch daraus einige direkte Folgerungen:

- Die „semantische Suche“ in der vorgestellten ersten Architekturvariante bezieht sich lediglich auf die Verwendung von Synonymketten für die Erweiterung der textvergleichbasierten Volltextsuche. Sie führt zu einer echten Verbesserung

- (Semantische) Erweiterungen bzw. Explizierungen der Suchanfrage sollen auch bei der Anfrage aller möglichen Zielsysteme bzw. Datenquellen (hier: auch bei OneBox-Systemen) zur Verfügung stehen.
- Abfrageschnittstellen (APIs) der Zielsysteme ZS, müssen den Anforderungen der jeweiligen Anwendungsfälle gerecht werden (z.B. Umkreissuche mit Mittelpunkt und Radius).
- Suchergebnisse bzw. Daten aus mehreren/verschiedenen Zielsystemen/Datenquellen sollen unabhängig voneinander (asynchron) angefragt werden können.
- Zielsysteme sollen einheitliche Schnittstellen und Datenformate verwenden, um die Möglichkeit zur Wiederverwendung von (generischen) Komponenten zu verbessern und somit den Aufwand für deren Entwicklung zu reduzieren.
- Die Bereitstellung von Daten in separaten Diensten (Services) ist sinnvoll, insbesondere wenn sie nicht über eine Suchmaschine indexiert werden können oder wenn die Bereitstellung in einem separaten Dienst über eine spezifische Abfrageschnittstelle (API) bzw. in einem spezifischen Format einen Mehrwert für die Anwendung bietet.
- Der “Zwang” zur Nutzung von spezifischen (ggf. nicht standardisierten oder proprietären) Schnittstellen führt dazu, dass existierende Systeme, welche die Schnittstellen nicht zur Verfügung stellen können, über Adapter/Proxy angebunden werden müssen.

Weitere Bemerkungen und Erkenntnisse

- In der OneBox-Architektur tritt explizite Semantik bestenfalls bei der Erweiterung von Suchanfragen (durch Thesauri und Wörterbücher) zutage. Sie steckt jedoch auch hier implizit in den verwendeten Thesauri und kommt bei den Anfragen an die OneBoxes nicht zur Anwendung.

Wenn die obigen Verbesserungen in die semantische Suche in heterogenen Informationsystemen eingebracht werden sollen, ist eine Modifikation der hier vorgestellten Architekturen erforderlich. Damit beschäftigen sich die folgenden Abschnitte.

3.4 Zweite Architekturvariante: Serverseitige Verarbeitung der Suchanfrage, SearchBroker und Ontologiesystem

Den Kern der zweiten Architektur-Alternative (Abbildung 4) bildet ein Portal, das möglichst alle verfügbaren Informationen einer Domäne für den Nutzer (Fachnutzer bzw.
die allgemeine Öffentlichkeit) zugänglich machen soll. Die Informationen liegen dabei nicht im Portal selbst, sondern in dedizierten Systemen, die lediglich über Schnittstellen in das Portal integriert und dort recherchierbar gemacht werden.

Abbildung 4: Serverseitige Umsetzung mit SearchBroker und Ontologiesystem (neue Entwicklungen und eigene Anteile in rot)

Das (Such-)Portal selbst besteht aus verschiedenen Komponenten, die dabei jeweils weitgehend unabhängig von anderen Komponenten, spezialisierte Aufgaben (z.B. Vorverarbeitung der Suchanfrage, Abfrage von Zielsystemen, die Visualisierung bestimmter Datentypen etc.) ausführen.

Der modulare Ansatz bietet weitreichende Vorteile, insbesondere die Wiederverwendbarkeit von Modulen, die Möglichkeit zum Zusammenstellen von Anwendungen als Kombination von Modulen (Baukasten) und die Möglichkeit zum Austausch von Modulen gegen neue, bessere, leistungsfähigere oder alternative Module.

Architekturvarianten

Abbildung 5: Übersicht über die Komponenten des Portals

Im unteren, grünen Teil befinden sich dabei die Komponenten des Portals selbst, der obere, gelbe Teil beinhaltet die Komponenten der semantischen Ebene (die semantische Suche).

Im Portal sind die Generierung der eigentlichen Nutzeroberfläche (Portal Frontend), die Applikationslogik (Portal Applikation), eine Daten-Persistenzschicht zur Speicherung von (Meta-)Daten (Portal Persistenz) innerhalb des Portals sowie eine spezialisierte Komponente zur Aufbereitung der Ergebnisansichten für Suchanfragen (Mashup Generator) enthalten. Nutzer und Administratoren kommunizieren dabei ausschließlich mit dem Portal Frontend. Das Frontend kann prinzipiell (modularer Ansatz) gegen andere Frontend-Komponenten ausgetauscht werden, die zum Beispiel alternative Ansichten, etwa für die Nutzung der Anwendung auf mobilen Geräten, implementieren können.


3.4.1 SearchBroker als zentrale Komponente der Suche

Der SearchBroker stellt die Kernkomponente der semantischen Ebene dar (Abbildung 6).
Architekturvarianten

Er verfügt über Informationen zu allen angeschlossenen Zielsystemen in Form von Zielsystembeschreibungen. Jede Zielsystembeschreibung enthält einerseits Informationen zum Zugriff auf das jeweilige Zielsystem (technische Schnittstellen), andererseits aber auch Informationen zur Art und Bedeutung (Semantik) der im Zielsystem abrufbaren Informationen. Die Verfügbarkeit der Semantik hebt den SearchBroker von konventionellen Volltextsuchmaschinen ab, die meist nicht über Metainformationen zur Semantik der indexierten Daten verfügen und die Einordnung (Gewichtung) von Inhalten z.B. aus deren Struktur (etwa Auszeichnung in Form von HTML-Tags) vornehmen müssen. Wenn die Interpretation einer HTML-Tabelle, die in jeder Zeile aktuelle Ozon-Messwerte zu einer Messstation enthält, nur auf einer strukturellen Ebene (in der ersten Spalte steht eine Zeichenkette, in der zweiten Spalte steht eine Zahl), jedoch nicht auf einer semantischen Ebene (in der zweiten Spalte steht der aktuelle Ozon-Messwert in µg/m³ zu dem in der ersten Spalte genannten Ort) geschieht, gehen die Möglichkeit zur Nutzung der Informationen innerhalb des Suchergebnisses weitgehend verloren.


### 3.4.2 Spezialisierte Plugins zur semantischen Vorverarbeitung der Suchanfrage

Um dem SearchBroker bei der semantischen Auflösung der Elemente in der Suchanfrage zu helfen, stehen ihm zur (Vor-)Verarbeitung der Suchanfrage eine Reihe von spezialisierten Komponenten (Plugins) zur Verfügung, die sich jeweils um die Erkennung von bestimmten Elementen der Suchanfrage kümmern und andere, unbekannte Elemente ignorieren. So ist das Plugin „Gazetteer“ z.B. auf die Erkennung von deutschen Ortsnamen spezialisiert, kann dabei auch räumliche Gliederungen wie Regierungsbezirke oder Landkreis verarbeiten und deren Beziehungen zueinander (Stadt Karlsruhe liegt im Regierungsbezirk Karlsruhe, liegt im Land Baden-Württemberg) auflösen. Eine andere Komponente („Time“) kann z.B. auf die Erkennung von zeitlichen Begriffen spezialisiert sein, z.B. aus „im sommer 2013“ die entsprechenden Anfangs- und Enddaten generieren, ggf. auch alternative Interpretationen (meteorologischer Sommer, astronomischer Sommer) liefern.


Die Ermittlung von speziellen Elementen ist auf Basis von vorgegebenen Wertelisten (Liste aller Ortsnamen Deutschlands) oder auf Basis von syntaktischen Mustern/Regeln wie (hier für die Jahreszeiten eines bestimmten Jahres)

```
(frühling|sommer|herbst|winter)\s+\d{4}
```

relativ einfach.

### 3.4.3 Auflösung thematischer Bezüge durch die Nutzung von Ontologien

Schwieriger ist dagegen das Erkennen eines Themenbezugs, z.B. aus dem Begriff „ozonmesswert“ einen inhaltlichen Bezug zu den Begriffen „Ozon“ und „Messwert“ und den dahinter stehenden Konzepten herzustellen. Selbst wenn man Teilbegriffe erkennen kann (z.B. durch Nutzung eines Wörterbuches auf die Zusammensetzung des Be-


Innerhalb einer Domäne bilden die verwendeten Begriffe daher meist ein kontrolliertes Vokabular und es existieren in vielen Fällen Fachthesauri, Taxonomien oder andere verfügbare Systematiken, die sich in ihrer Darstellung, Komplexität und ihrer semantischen Reichhaltigkeit unterscheiden, s. Abbildung 7.

Abbildung 7: Semantische Treppe nach (Pellegrini und Blumauer 2006)

Taxonomien erweitern das Glossar um eine hierarchische Struktur, d.h. die Beziehungen zwischen Begriffen sind in Form einer Ordnung (Oberbegriff, Unterbegriff) dargestellt. Weitergehende Beziehungen zwischen Begriffen lassen sich jedoch in einer Taxonomie nicht darstellen.

In einem Thesaurus können Beziehungen zwischen Begriffen/Objekten durch fest definierte Relationen, z.B. Synonym, Oberbegriff, Assoziation etc., ausgedrückt werden. Durch die Verwendung unterschiedlicher Relationen können verschiedene Arten von Beziehungen ausgedrückt werden, die statische Hierarchie einer Taxonomie wird aufgelöst.


Ist das Wortgut (Vokabular) einer Domäne eindeutig beschrieben, z.B. in Form einer Taxonomie, eines Thesaurus, einer Topic Map oder einer Ontologie, so lässt sich der thematischen Bezug einer Suchanfrage innerhalb der Domäne mit Hilfe des Wortguts im Allgemeinen präziser beantworten als allgemeine Fragen, wie sie z.B. von Internet-Suchmaschinen beantwortet werden.


Deskriptoren eignen sich sowohl für die Beschreibung von thematischen Bezügen in Suchanfragen als auch in Zielsystemen. Alle anderen Begriffe aus der Domäne lassen sich auf die zugehörige Deskriptoren abbilden.
3.4.4 Ontologiesystem

Zur Darstellung der verschiedenen Aspekte von Informationsquellen (thematische, örtliche Zuordnung, Lebenslage) werden auf inhaltlicher Ebene verschiedene, miteinander vernetzte Ontologien verwendet:

- **Domänen-Ontologie/Themen-Ontologie**: Auf Basis eines vorhandenen Vokabulars (z.B. einer Taxonomie, eines Thesaurus oder einer vollständigen Ontologie) modelliert die Domänen-Ontologie die thematischen Konzepte und Zusammenhänge in der spezifischen Domäne, z.B. „Umwelt“ oder „Energie“ eines Informationssystems. Ggf. kann es sich auch um mehrere Domänen-Ontologien handeln, die miteinander zu vernetzen sind (s.u. „Artikulationsontologie“).


- **Informationsontologie** (optional): Informationen über Informationsquellen (Zielsysteme) können ebenfalls in einer Ontologie abgelegt werden (Zielsystembeschreibungen) und z.B. thematische Zuordnungen (Bezug zur Domänen-Ontologie) sowie Informationen zum Zugriff auf das Zielsystem enthalten. Auf deren Basis kann die Anfrage eines Nutzers in Anfragen an die angeschlossenen Zielsysteme transformiert werden.

- **Artikulationsontologie**: Die oben aufgeführten Ontologien stellen noch disjunkte Teilontologien dar, d.h. es gibt keine Bezüge zwischen den Konzepten und Instanzen der Teilontologien. Zur Verbindung der Teilontologien wird daher eine weitere Ontologie, eine sogenannte Artikulationsontologie (Nikolai 2002)
hinzugefügt, die quasi den Klebstoff zwischen den einzelnen Teilen darstellt, sie kann darüber hinaus auch innerhalb einer Teilontologie Beziehungen darstellen, falls sie noch nicht enthalten sind, z.B. Relationen zwischen Ober- und Unterbegriffen.


Da die Ontologieentwicklung als evolutionärer Prozess verstanden werden muss, sieht die Architektur des Ontologiesystems die Nutzung eines integrierten Ontologienmanagements vor. Neben der persistenten Speicherung der Ontologie, z.B. in einer relationalen Datenbank, umfasst das Ontologiesystem Funktionen für die Weiterentwicklung, das Mapping unabhängig voneinander entstandener Ontologieteile, die Anbindung von Inferenzmaschinen und Visualisierungskomponenten, die Population von Instanzdaten, die Entwicklung von Abfragen sowie die Verwaltung von Nutzer- und Provenancedaten (d.h. Angaben zum Datenursprung).


Suchanfragen können mit Hilfe des Ontologiesystems semantisch erfasst, d.h. einem oder mehreren Deskriptoren aus den Bereichen der Domäne, ggf. einer oder mehreren Lebenslagen, Orten und weiteren per Ontologie verknüpften Inhalten zugeordnet wer-
den. Die so gewonnenen Informationen können dann für möglichst konkrete Anfragen an die angeschlossenen Zielsysteme verwendet werden.

3.4.5 Zielsysteme und Zielsystembeschreibungen


Jedes Zielsystem wird in der Zielsystembeschreibung benannt (Name) und seine Schnittstelle(n) und die möglichen Rückgabeformate beschrieben. Der Name und ggf. eine eindeutige zugeordnete ID gewährleisten die Unterscheidung von Zielsystemen, wenn z.B. unterschiedliche Daten aus derselben Datenquelle abgerufen werden sollen.

Die Schnittstellenbeschreibung liegt in Form einer OpenSearch-Description (Google Inc. 2015; opensearch.org 2013) vor. Schnittstellen werden als URL-Muster mit Platzhaltern für Parameter beschrieben. Die Parameter sind durch Inhalte zu füllen, die aus der Suchanfrage selbst sowie aus der Vorverarbeitung der Suchanfrage gewonnen werde. Dazu müssen die Parameter syntaktisch und semantisch beschrieben werden, z.B. in welcher Form Geokoordinaten übergeben werden, etwa als Paar von Latitude (Breite) und Longitude (Länge), die jeweils in Dezimalgrad anzugeben sind, oder in Form einer Location-ID (z.B. eines Gemeindeschlüssels), falls Geodaten im Zielsystem vorhanden sind und so referenziert werden können. OpenSearch-Description bieten die Möglichkeit mehrere unterstützte Rückgabeformate aufzuzählen und für jedes mögliche Rückgabeformat eine eigene Schnittstellenbeschreibung anzugeben. Das lässt dem konsumierenden System (hier: dem Search Broker) die Wahl, in welcher Form die gewünschten Daten abgerufen werden sollen, z.B. wenn es mehrere mögliche Visualisierungsformen (z.B. Karte, Liste, Diagramm, Tabelle) für die Daten gibt.

Über weitere Metadatenfelder innerhalb der OpenSearch-Description können z.B. Kurzbeschreibung, Organisation etc. angegeben werden.

Abbildung 8: Beispiel einer OpenSearch-Description (XML)

Die Wahl des OpenSearch-Description-Formates schränkt die Auswahl möglicher Schnittstellen auf solche ein, denen eine URL (für einen GET-Request) zugrunde liegt. Das stellt zwar eine technische Einschränkung dar, nicht jedoch der Mächtigkeit der Schnittstelle, da weitere mögliche Schnittstellen, z.B. zum direkten Abfragen einer Datenbank per SQL, sich leicht über Adapter auf Basis von REST-Diensten, die eben über URLs adressierbar sind, realisieren lassen. Es fällt hier jeweils nur der zusätzliche Aufwand für die Implementierung des Adapters an.

Der Search Broker überführt die Suchanfrage anhand der Zielsystembeschreibung in von den Zielsystemen verstandene parametrisierte Anfragen (Abbildung 6). Er nutzt hierfür sowohl externe Dienste, wie einen Geonamensdienst (Gazetteer Service), als auch das Ontologiesystem als Quellen, um z.B. den Sachbezug von Suchbegriffen aufzulösen. Das Ontologiesystem muss nicht die gesamte Interpretationsarbeit der Suchparameter leisten, sondern nur noch die semantische Interpretation der Konzeptbezogenen Suchparameter (Abecker et al. 2009a).

### 3.4.6 Anfragen


Zum Beispiel kann das Gazetteer-Plugin einen Ortsnamen wie „Karlsruhe“ erkennen, und ihm weitere Eigenschaften zuordnen, wie:

- `geo:commune:name = Karlsruhe`
- `geo:commune:id = 08212000`

Letztere kann nun z.B. als Parameter für die Adressierung eines Zielsystems dienen, das eben solche (standardisierten) Gemeindekennziffern verarbeiten kann.

Ein anderes Gazetteer-Plugin kann zum selben Ortsnamen „Karlsruhe“ weitere Informationen liefern:

- `geo:lon = 8.4037563`
- `geo:lat = 49.0080848`
- `geo:bbox = 8.2756969,48.9494975 8.5318157,49.0666033`

Auf ähnliche Weise können durch ein weiteres Plugin auch zeitliche Begriffe expliziert werden, z.B. einer Suche mit dem Bestandteil „Sommer 2010“ die expliziten Start- und Enddaten zugeordnet werden:

- `datetime:calendar:day:first = 2010/06/21`
- `datetime:calendar:day:last = 2010/09/22`

Die Auflösung eines Themenbezugs stellt die wohl größte Herausforderung bei der Vorverarbeitung dar. Ziel der Auflösung ist die Abbildung des Themenbezugs auf eines oder mehrere Elemente (Konzepte, Deskriptoren) eines wohldefinierten Wortgutes, wie es in den angeschlossenen Zielsystemen verwendet wird. Die thematische Auflösung ist über eine Suche im oben beschriebenen Ontologiesystem realisiert. Die Suche kann insbesondere durch einen Parameter gesteuert werden, der die Größe der Umgebung...
(= den maximalen Abstand von Konzepten und Individuen) zu den gefundenen Suchbegriffen vorgibt (Bügel et al. 2011c).


### 3.4.7 Mashup-Steuerung und Ergebnisdarstellung

Nach Abschluss der Vorverarbeitung kann der SearchBroker auf Basis der Zielsystembeschreibungen entscheiden, für welche Zielsysteme die notwendigen Informationen vorliegen und wie sie angefragt werden können. Die Abfrage von Daten kann nun der SearchBroker selbst vornehmen oder die entsprechenden vollständigen Adressen an das Suchportal zurückliefern. Derzeit geht die Implementierung den zweiten Weg.

Im Umweltportal ist eine Mashup-Komponente für die Darstellung der Suchergebnisse verantwortlich. Sie kann abhängig von den gelieferten Ergebnissen entscheiden, in welcher Form sie dargestellt werden sollen. Für die Darstellungen liegen verschiedene HTML-Schablonen (Vorlagen) vor, die bei Bedarf mit Daten konkreter Datensätzen verknüpft und in die Ergebnisseite eingebaut werden.

Im Wesentlichen wird dabei zwischen folgenden Zielformaten unterschieden:

- Geodaten, z.B. darstellbar in einem Web Map-Client, z.B. Google Maps API
- Listen von Objekten mit Verweisen auf deren Darstellung in einem Fachsystem, z.B. in Form von Linklisten
- Tabellarische Daten und Diagramme, die ggf. nach HTML konvertiert werden
- Multimedia-Inhalte, z.B. in Form einer Galerie-Ansicht
- Text-Nachrichten (Formate Atom oder RSS), z.B. in Form von Übersichtslisten
- HTML-Seiten, HTML-Fragmente und Mikroformate, die an bestimmten Stellen im Layout eingeblendet werden
- Ergebnisse der Volltextsuche, z.B. in Form von Trefferlisten.


Darüber hinaus bieten die während der Vorverarbeitung gefundenen Informationen die Möglichkeit, dem Nutzer im Umweltportal weitere Navigationsschritte anzubieten. Dazu gehören zum Beispiel die Einschränkung bzw. Erweiterung des Suchraums auf Basis von Unter- respektive Oberbegriffen, aber auch die Auflösung von Mehrdeutigkeiten, die sich z.B. aus der Vorverarbeitung von Ortsnamen ergeben.
3.4.8 Bewertung


Die Harmonisierung der Vokabulare, die alle als Ontologien dargestellt werden, geschieht über eine Artikulationsontologie, die Relationen zwischen passenden Konzepten der anderen Ontologien vorhält.

Erkenntnisse aus der Implementierung und dem probeweisen Betrieb der zweiten Architekturvariante sind:

- Die semantische Einordnung (Annotation) und Erweiterung der Suchanfrage (bzw. der enthaltenen Suchbegriffe) mit Hilfe von Gazetteer-Diensten ist sinnvoll und notwendig.
- Insbesondere auch, um mögliche Parameter für spezifische Anfragen an spezifische Zielsysteme gewinnen zu können
- Die verwendeten Vokabulare müssen die Zieldomäne (z.B. des Portals) möglichst vollständig abdecken.
- Derzeit existierende Ontologiesysteme sind in der Regel nicht für die Verarbeitung großer Mengen von Objektinstanzen geeignet. Bereits wenige 100.000 In-
Architekturvarianten

stanzen führen in der Praxis zu nicht akzeptablen, langen Antwortzeiten. Hier ist z.B. eine Vorverarbeitung (Indexierung) notwendig, oder es wird auf die Speicherung von Objektinstanzen innerhalb des Ontologiesystems verzichtet.


- Die rein serverseitige Aufbereitung von Ansichten (Mashups) führt zu unflexiblen und statisch wirkenden Anwendungen, insbesondere weil Interaktionen (mangels Möglichkeit zu deren Verarbeitung im Client) in der Regel zum kompletten Neuladen einer gesamten Trefferansicht führen. Ein Suchergebnis wird erst dann an den Client ausgeliefert, wenn alle Verarbeitungsschritte abgeschlossen sind, d.h. das gesamte Suchergebnis vorliegt, was zu spürbaren Wartezeiten für den Nutzer führt.

- Die rein serverseitige Verarbeitung der Suche stellt darüber hinaus einen Flaschenhals der gesamten Suchanwendung dar. Es werden erst Ergebnisse ausgeliefert, wenn alle Anfragen abgearbeitet sind, schlimmstenfalls nach einem Timeout. Speziell die einzelnen Zielsysteme könnten stattdessen auch asynchron angefragt und die Ergebnisse dynamisch nachgeladen und dargestellt werden. Entsprechende dynamische Mechanismen im Client sind eine notwendige Voraussetzung hierfür.


- Da einzelne Komponenten zu Anzeige von Suchergebnissen in der Regel ohnehin auf die Verarbeitung bestimmter Datentypen und -formate spezialisiert sind (z.B. eine Diagrammkomponente nur Zeitreihendaten) ist es lediglich notwendig, jeder Komponenten ausschließlich die für sie bestimmten Daten zur Verfügung zu stellen - nicht den gesamten Datenbestand.

- Um Anzeigekomponenten effizient einsetzen zu können, sollen sie möglichst generisch sein, d.h. hochkonfigurierbar und ausgestattet mit möglichst flexiblen und standardisierten Schnittstellen.

- Für die verwendeten Szenarien ist eine Unterscheidung von Thema, Orts- und Zeitbezug sinnvoll. Ein Thema könnte zusätzlich in ein Thema und einen As-


### 3.5 Dritte Architekturvariante: Serviceorientierung, „Webcache“, clientseitige Verarbeitung


#### 3.5.1 Webcache

Wenn Informationen über zentrale Einstiegspunkte wie Webportale und mobile Anwendungen zur Verfügung gestellt werden sollen, müssen die Inhalte, die auf solchen Anwendungen basieren, meist auf Basisdaten zurückgreifen, die zu anderen Zwecken und in anderen Kontexten entstanden sind bzw. erhoben wurden. Der primäre Zweck der (ursprünglichen) Daten widerspricht häufig dem Zweck der Nachnutzung, z.B. der Präsentation in einem Portal. Zum Beispiel enthalten Informationen personenbezogene Daten, sind Lizenzen unterworfen, bestehen aus großen Informationsmengen, sind nur mit den entsprechenden Benutzerrechten zugänglich, werden in speziellen Datenformaten gespeichert, sind nicht über das Internet zugänglich, sind für Laien unverständi-
Architekturvarianten

lich, sind nicht rund um die Uhr verfügbar etc. Das bedeutet, dass die Originaldaten transformiert (z.B. gefiltert, aggregiert, anonymisiert etc.) werden müssen, ehe sie über ein Portal bereitgestellt werden können. Für die Zwecke der semantischen Suche kann die Transformation auch zur Anreicherung der Daten um semantische Informationen verwendet werden.

Die Grundidee des Webcache in der vorlegenden Arbeit ist die Bereitstellung von "internetfähigen" Kopien der Originaldaten (Abbildung 9). Die Informationen werden automatisch aus den ursprünglichen Systemen (Zielsystemen) extrahiert, z.B. Datenbanken und Fachanwendungen, die dann verarbeitet (Transformation) und in redundanten Systemen (dem Webcache) über standardisierte Schnittstellen (APIs) zur Verfügung gestellt werden. Die Vermeidung des direkten Durchgriffs auf die originalen Datenquellen (Zielsysteme) ermöglicht eine bessere Verfügbarkeit und nutzungsbasierter Skalierung von Diensten (Datendienste und Datenmanagement) und bietet Sicherheitsvorteile durch strikte Trennung von internem und externem/öffentlichem Zugang.

Abbildung 9: Umsetzung als serviceorientierte Architektur mit Aufbau eines "Webcache" als Sammlung generischer Dienste (neue Entwicklungen und eigene Anteile in rot)

Der Datenfluss ist zunächst unidirektional vom Zielsystem hin zum Webcache konzipiert. So stellt der Webcache eine schreibgeschützte Kopie der Daten dar. Konsistenz- oder Kohärenzbedingungen werden für jeden Datentyp und jedem Zielsystem festgelegt, die die Art und Häufigkeit der Synchronisation zwischen Zielsystem und Webcache beeinflussen. Begrenzt auf unidirektionale Datenflüsse stellt der Webcache
Architekturvarianten

66

Eine vereinfachte Anwendung dar. Im Allgemeinen hat er jedoch die volle Funktionalität des Datenmanagements, d.h. auch Funktionen zum Hinzufügen, Aktualisieren und Löschen von Daten stehen zur Verfügung, einschließlich Mechanismen für Authentifizierung und Autorisierung (Czernik 2016).

3.5.2 Generische Services

Um den Aufwand zur Einrichtung und zum Betrieb des Webcache auf einem akzeptablen Niveau zu halten, besteht ein wesentliches Ziel darin, die gesamte Information durch eine begrenzte Anzahl von generischen Diensten bereitzustellen, in Abbildung 9 angedeutet durch deren Schnittstellen API\textsubscript{i} (i = 1...n). Dazu müssen die notwendigen Daten und Funktionalitäten definiert werden. Je nach Anwendung ist neben speziell entwickelten, generischen Diensten auch der Einsatz von Cloud-Services (s. Anhang A5) möglich (Schlachter et al. 2014a).

Zunächst muss die Transformation der Originaldaten in ihre generische Darstellung definiert werden. Das kann eine weitere Verarbeitung wie Auswahl, Filterung, Aggregation etc. (Datenaufnahme, eng. Data Ingestion) umfassen. Der gewählte Ansatz hat den Vorteil, dass nur zulässige "internetfähige" Daten den Webcache erreichen können, z.B. ohne personenbezogene Daten, unter Verwendung aggregierter Werte, Filterung etc.

Für die Anforderungen im Umwelt- und Energiebereich wurden insgesamt acht generische Dienste identifiziert:

- Stammdatendienst
- Schemadienst
- Zeitreihen-Dienst
- Mediendienst
- (Volltext) Suchdienst
- Geodatendienst
- Metadatendienst
- Verbindungsdienst.

Die acht Kerndienste werden durch zwei weitere Dienste ergänzt, die konsumierende Anwendungen unterstützen:

- Anwendungskonfigurationsdienst
- Discovery-Dienst.

Alle Dienste sind weitgehend voneinander unabhängig, eine zentrale Anforderung an den für Microservices verwendeten Komponentenbegriff (Fowler und Lewis 2015). Daher ist eine Implementierung unter Verwendung von Microservices offensichtlich. In Laufzeitcontainern wie Docker (Docker 2016; Mouat 2016) verpackt, können sie ohne zusätzlichen Aufwand an einer Vielzahl von möglichen Infrastrukturen wie dedizierten Servern, Clustern oder in der Cloud betrieben werden. Unter Einsatz von Laufzeit-Infrastrukturen wie Kubernetes (Kubernetes 2016) sind operative Aspekte wie Rolling-
Updates, Monitoring, Skalierbarkeit und Load-Balancing nur eine Frage der Konfiguration - eine geeignete Computerinfrastruktur und Software-Design vorausgesetzt.

Alle Dienste nutzen geeignete Backend-Systeme, die insbesondere die Persistenz der Daten sicherstellen. Auch hier wird die Architektur von konkreten Systemen abstrahiert, so dass die Backend-Systeme problemlos ausgetauscht oder gleichzeitig unterschiedliche Backend-Systeme genutzt werden können. Die Auswahl geeigneter Backend-Systeme, z.B. verschiedene NoSQL-Technologien (Edlich 2011; Sullivan 2015), sichert auch dynamische Eigenschaften wie Lastverteilung, Skalierbarkeit etc.

Alle Services bieten ihre Funktionalität durch versionierte RESTful-Interfaces (Fredrich 2013) über Content-Negotiation (W3C 1999), d.h. der Client kann bei der Anfrage das gewünschte Rückgabeformat angeben, bzw. eine Prioritätenliste, sofern mehrere Rückgabeformate verfügbar sind. Das erleichtert die Entwicklung, Wartung und den Austausch einzelner Dienste.


### 3.5.3 Generische Frontend-Komponenten

Für die Präsentation von Daten und Inhalten in Nutzer-Frontends, z.B. Webportalen, Websites oder mobilen Anwendungen sind zur Darstellung verschiedener Datenformate entsprechende Frontend-Komponenten notwendig, z.B. zur Anzeige von

- Objektdaten (z.B. als Einzelansicht, als Liste oder als Tabelle),
- Diagrammen (z.B. Zeitreihen, Tortengrafiken),
- Geoinformationen auf einer Karte,
- Auswahllisten (z.B. zur Nachfilterung anhand von Facetten),
- Formularen.

Insbesondere innerhalb von Portalen, die per se Daten aus verschiedenen Quellen darstellen sollen, ist es sinnvoll, die Komponenten generisch, d.h. konfigurierbar und damit nutzbar für verschiedene Anwendungsfälle, auszulegen. Das bedeutet, dass z.B. die Datenquelle (Schnittstelle, Format), das Ausgabeformat (z.B. in Form von Templates), spezifische Styling-Formate (z.B. Cascading Stylesheets, CSS) und weitere Einstellungen per Konfiguration für eine spezifische Instanz der Komponente festgelegt werden können.


Zur komfortablen Nutzung innerhalb von Liferay werden sämtliche Widgets in sogenannten Wrapper-Portlets verpackt, d.h. Portlets, die lediglich einen Rahmen für die enthaltenen Web Widgets darstellen, um die Web Widgets als Portlets nutzen und konfigurieren zu können. Die Wrapper-Portlets können durch Autoren entsprechend einem gewählten Layout bequem per Drag&Drop neben weiteren Portlets in Seiten platziert


### 3.5.4 Zusammenspiel von Frontend-Komponenten

Aus Sicht eines Nutzers ergibt sich eine Gesamtanwendung meist aus dem Zusammenspiel mehrerer Einzelteile. Im Falle der Landesumweltportale besteht die Gesamtanwendung aus der Orchestrierung mehrerer Frontend-Komponenten respektive der ihnen zugrundeliegenden Dienste. Die Orchestrierung ergibt dann beispielsweise eine Suchergebnisseite, die Informationen aus gleichen oder verschiedenen Quellen auf unterschiedliche Weisen visualisiert.

Im Falle der Suchergebnisseite werden die anzuzeigenden Ergebnisse meist durch die manuelle Eingabe von einem oder mehreren Suchbegriffen getriggert.

In einem Vorverarbeitungsschritt wird versucht, den Suchbegriffen eine Semantik zuzuordnen, z.B. thematische Begriffe wie „Bodenerhaltung“ auf bekannte Deskriptoren wie „Bodenschutz“ abzubilden oder geographischen Begriffen wie „Karlsruhe“ den entsprechenden Ort zuzuordnen. Bei der Vorverarbeitung können die Begriffe semantisch angereichert werden, z.B. dem erkannten Ortsnamen „Karlsruhe“ auch dessen geografischer Mittelpunkt (Center), eine Bounding-Box sowie ein amtlicher Gemeindeschlüssel (Gemeindekennziffer); thematische Begriffe können ebenfalls um Attribute ergänzt werden, z.B. Schlüssel wie ein Objektartencode oder ein Fachführungscode. Mehrdeutigkeiten, beispielsweise mehrere erkannte Ortsnamen zu einem Suchbegriff wie „Neuhausen“, können ggf. erst nach einer Nutzerinteraktion aufgelöst werden.

Der Kontext einer Abfrage kann auch durch die Anwendung ergänzt werden, z.B. einen durch den Nutzer vorgegebenen bevorzugten Standort, der im System oder einem Cookie gespeichert ist und allen Suchanfragen automatisiert hinzugefügt wird.

Das wird in Abbildung 10 handel des Umweltinformationsnetzes Sachsen-Anhalt veranschaulicht. Hier wurde nach den Suchbegriffen „hochwasser tangermünde“ gesucht. Neben einer klassischen Volltextsuche-Trefferliste (rechts unten) sind hier unter ande-
rem eine Kartenansicht (rechts oben) mit der zugehörigen Kartenlayerauswahl (links oben) sowie passende Treffer konkreter Objekte (darunter, z.B. „Risiko (407)“) sowie aus dem Metadatenkatalog des Landes Sachsen-Anhalt (links unten) zu sehen.


Abbildung 10: Suchergebnisseite mit Karte, Layer-Auswahl, Volltext- und Metadaten-Trefferlisten im Umweltinformationsnetz Sachsen-Anhalt (Screenshot Umweltinformationsnetz Sachsen-Anhalt)


Insbesondere die Kommunikation per Ortsangabe bzw. thematischen Begriffen stellt zwar eine relativ lose Kopplung dar, sie kommt in vielen Fällen jedoch der Datenlage insofern entgegen, dass ein Zusammenhang zwischen zwei örtlich benachbarten Objekten (Windkraftanlage in der Nähe eines Naturschutzgebietes) häufig in den Daten nicht explizit dargestellt ist.


Auf der anderen Seite erwarten die Nutzer der Umweltportale in den meisten Fällen zwar eine Unterstützung beim Auffinden der passenden Informationen zu ihrem Anliegen, stellen dabei aber selbst eine aktive Filterinstanz dar, welche die angezeigten Informationen sichten, bewerten und sich passende Teile herauspicken kann. Eine – nicht zu große – Obermenge der tatsächlich relevanten Ergebnisse ist für sie in den meisten Fällen akzeptabel. Des Weiteren hat sich gezeigt, dass bei einem großen Anteil der Suchanfragen die Beziehungen zwischen den passenden Ergebnissen auf einem sehr hohen Abstraktionsniveau darstellbar sind, z.B. ihrer örtliche Nähe zueinan-
Architekturvarianten


Abbildung 11: Verknüpfung von Windkraftanlagen und (Natur-)Schutzgebieten durch die Suche nach „windrad schutzgebiet langenburg“ im Umweltportal Baden-Württemberg (Screenshot)

Die lose Kopplung von Umweltobjekten funktioniert also nur für menschliche Nutzer und nur in solchen Anwendungsfällen, in denen sich Beziehungen auf relativ hohen Abstraktionsniveaus darstellen lassen, z.B. ihre örtliche Nähe oder die Zuordenbarkeit zu einem bekannten Thema (Windkraft).

3.5.5 Verknüpfung semantischer Objekte und Klassen

Die Umweltportale haben in den meisten Fällen keinen oder nur wenig Einfluss auf die Systeme, aus denen sie ihre Daten beziehen. Das bedeutet, dass Erweiterungen des
Datenmodells, z.B. um Beziehungen zu anderen Objekten, in solchen Systemen in der Regel nicht möglich sind. Um dennoch eine engere Kopplung der Daten aus den Systemen zu erreichen, bietet es sich an, die Beziehungen in zusätzlichen Systemen („Beziehungsdiensten“ oder „Link-Services“) abzulegen. Dort kann eine Beziehung als Tripel (Objekt A, Typ der Beziehung, Objekt B) gespeichert werden, z.B. („Windkraftanlage Nr. 4711“, „liegt-in“, „Gemeinde Nr. 08127047“), was der Mechanik des Semantic Web (RDF-Tripel) entspricht (W3C 2004c). Speichert man zusätzlich in einem Metadatensystem, in welchem konkrete System die Windkraftanlagen gespeichert sind und wie technisch darauf zugriffen werden kann (z.B. Service-Adresse und Schlüsselattribut), dann kann ein Service bereitgestellt werden, der seinerseits explizite Verknüpfungen zu anderen konkreten Objekten (z.B. einzelnen Naturschutzgebieten) bereitstellt, selbst wenn die Beziehungen in der Original-Datenquelle nicht vorhanden sind.


Damit entstehen insgesamt drei Arten von Beziehungen:

- Objekt : Objekt (n:m)
- Klasse : Objekt (1:n)
- Klasse : Klasse (1:1).


Die Orchestrierung der Suche im Client geschieht über die Kommunikation der generischen Komponenten, die sich jeweils auf die Abfrage und Anzeige bestimmter Datentypen beschränken.

Einzelne (Micro-)services sind sehr gut geeignet, um den Zugriff auf Daten und Objekte von ihren jeweiligen Zielsystemen zu entkoppeln. In ihrer Gesamtheit bilden sie eine einheitliche Zugriffsschicht, die über generische Dienste realisiert werden kann und die potenziell große und heterogene Anzahl von Schnittstellen der Zielsysteme erheblich reduziert, was den Zugriff durch ebenfalls generische Frontend-Komponenten sehr vereinfacht.

Aus den bisherigen Untersuchungen ergeben sich die folgenden Aussagen:

- Die Entkopplung der Zugriffsschicht von den Zielsystemen reduziert die Last auf die Quellsysteme auf ein für die Synchronhaltung der Daten notwendiges Minimum. Gleichzeitig kann die Laufzeitinfrastruktur für einzelne Services individuell gewählt und konfiguriert werden, z.B. um eine lastabhängige horizontale Skalierbarkeit einzelner Services gewährleisten zu können.
- Die Entkopplung der Zugriffsschicht (API) von den Quellsystemen sorgt für stabile Schnittstellen, auch wenn sich Schnittstellen oder Datenmodelle von Quellsystemen ändern. Anpassungen müssen teilweise nur für die Anbindung des Quellsystems an den Service vorgenommen werden, massive Änderungen oder Erweiterungen, z.B. am Datenmodell, können über eine Versionierung der Zugriffsschicht (API) abgefangen werden, die eine Rückwärtskompatibilität gewährleisten kann.
- Der Aufbau einer Zugriffsschicht mit eigener, redundanten Datenhaltung erzeugt zunächst Aufwand,
  - Notwendigkeit zur Bereitstellung und zum Betrieb weiterer Systeme (Services),
  - Redundante Datenhaltung und daher die Notwendigkeit eines Konzepts (Regeln, Definition eines Konsistenzbegriffes für potenziell jede einzelne Datenquelle) und eines Systems zum Sicherstellen von Datenkonsistenz,
 ARCHITEKTURVARIANTEN

Notwendigkeit zur Konzeption und Implementierung entsprechender Synchronisationsmechanismen, bietet jedoch auch sehr gute Chancen:

- Einheitliche und stabile Zugrifffsschicht (APIs, ggf. versioniert) erleichtert die Implementierung von generischen Frontend-Komponenten und weiteren Anwendungen. Damit wird auch das Potenzial für Synergien, d.h. Mehrfachnutzung von Daten für verschiedene Zwecke, erhöht.
- Bereitstellung der Daten über standardisierte URLs (z.B. RESTful) als notwendige Voraussetzung für Linked Data.
- Vorverarbeitung der Daten bei der Synchronisation, z.B. Erweiterung um semantische Angaben wie Konzepte, Schlagworte etc., Normalisierung von Attributen (Datumsangaben, Geokoordinaten, …), Abbildung auf standardisierte oder generische Attribute (Titel, Kurzbeschreibung, Link etc.), potenziell auch die Erweiterung um Referenzen zu verknüpften Objekten.

- Die Wahl geeigneter Mechanismen (Container) gewährleistet einen flexiblen Betrieb, z.B. auf dezidierten Servern, auf Rechnerclustern oder in der Cloud und ermöglicht einen automatisierten Build- und Deploymentprozess.
- Nicht vollständig instrumentierte Tool-Unterstützung (Ziel: Einheitlichkeit, Konfiguration vor Programmierung) für die Synchronisation und die Sicherstellung der Datenkonsistenz.
- Objekte sind zwar per eindeutiger URL adressierbar, die Beschaffenheit der URLs entspricht jedoch noch nicht in allen Fällen den Best Practises für den Aufbau von RESTful URLs (Fredrich 2013).
- Die durch die Services unterstützten Formate sind zwar standardisiert und maschinenlesbar (z.B. GeoJSON), es wird jedoch z.B. noch kein RDF unterstützt.
- Die Daten enthalten noch keine expliziten Verweise (Links) auf andere Objekte im Sinne von Linked Data.
- Die Versionierung der Zugrifffsschicht ist noch nicht umgesetzt.
• Die Orchestrierung von Services erfolgt derzeit ausschließlich in den konsumierenden Anwendungen. Es liegen noch keine Erfahrungen in der Inter-Service-Kommunikation vor.


In der folgenden vierten Architekturvariante soll die dritte Variante dahingehend erweitert werden, dass die Daten aus dem Webcache auch im Sinne des Semantic Web (Berners-Lee et al. 2001) über standardisierte Schnittstellen als Linked Data (Berners-Lee 2006) bereitgestellt werden können.

3.6 Vierte Architekturvariante: Ausbau zu semantischen Diensten / Linked Data

Die Repräsentation von Daten mit Hilfe von RESTful Webservices (Bayer 2002) bietet gute Chancen, die Daten in Form von Linked Data zur Verfügung stellen zu können. Dabei sind nach Tim Berners-Lee (Berners-Lee 2006) vier Regeln zu beachten:

1. Verwendung von URIs zum Benennen von Objekten
2. Verwendung von HTTP-URIs zum tatsächlichen Auffinden von Objekten
3. Verwendung von Standards wie RDF und SPARQL
4. Bereitstellung von Links in Form von URLs, um Verbindungen zu weiteren Objekten finden zu können.

Wenn alle verfügbaren Daten (Objekte) bereits über (Micro-)Services mit RESTful URIs bereitgestellt werden, können an die ersten beiden Regeln bereits Haken gemacht werden.

Die Bereitstellung der Daten in Standardformaten wie RDF ist dann nur noch eine Frage der formalen Repräsentation, die relativ leicht zu implementieren ist. Wesentliche Randbedingung zur globalen Nutzbarkeit der Daten ist allerdings deren Anbindung an
globale Vokabulare, d.h. deren direkte Nutzung bzw. ein Mapping darauf, um eine Interoperabilität mit weiteren Datenquellen gewährleisten zu können.

Die Verknüpfung von Daten mit weiteren Objekten, d.h. die Bereitstellung von expliziten Links, stellt die größte Herausforderung dar, da in vielen Datenquellen solche Verknüpfungen nicht bestehen, schon gar nicht in Form von URIs. Verknüpfungen zwischen Objekten müssen also zunächst erzeugt werden. Die notwendigen Regeln bzw. zugehörigen Operationen zur Generierung von Verknüpfungen müssen entweder aus den Daten selbst bzw. den zugehörigen Metadaten des Quellsystems gewonnen werden, oder - im schlimmsten Fall - neu erzeugt werden, was in der Regel Aufwände erzeugt.

Ziel der vierten Architekturvariante ist die Bereitstellung aller Daten als Linked Data, was die explizite Bereitstellung von Verknüpfungen zwischen Daten beinhaltet. Abbildung 12 zeigt die vierte Architekturvariante mit der entsprechenden Erweiterung.

Abbildung 12: Umsetzung als serviceorientierte Architektur mit zusätzlichem Link-Service (neue Entwicklungen und eigene Anteile in rot)


3.6.1 Identität von Objekten

Grundlage für die Darstellung von Beziehungen zwischen Objekten ist, dass jedes Objekt über eine Identität verfügt, die meist durch einen eindeutigen Schlüssel ausgedrückt wird, der sich bei verschiedenen Objekten auch dann unterscheidet, wenn alle anderen Attribute gleiche Werte aufweisen.

In den meisten Quellsystemen werden Objekte bereits eine Identität besitzen. Dabei kann es sich um explizit vorgegebene Identitäten handeln (z.B. die eindeutige MAC-Adresse eines Netzwerkgerätes oder die eindeutig vergebene Nummer einer Messsta-
Sind in einem System keine expliziten Identitäten vorhanden, können sie in der Regel nach dem Muster von Primärschlüsseln relationaler Datenbanken gewonnen oder komplett künstlich erzeugt werden, z.B. einen Universally Unique Identifier, UUID.

Doch selbst wenn Objekte in einzelnen Quellsystemen bereits über Identitäten verfügen, müssen sie jedoch über Systemgrenzen hinweg nicht eindeutig sein, z.B. handelt es sich bei künstlichen Primärschlüsseln in relationalen Datenbanken meist um fortlaufende natürliche Zahlen, die Wahrscheinlichkeit der Mehrfachnutzung desselben numerischen Wertes ist bei mehreren relationalen Datenbanken daher sehr hoch. Um systemübergreifend eindeutige Identitäten zu schaffen, bietet es sich daher an, jedem Quellsystem eine Identität zu geben, z.B. eine einmalig künstlich generierte, eindeutige ID, und dann eine Verkettung der Quellsystem-ID mit der Objekt-Identität innerhalb des Quellsystems als systemübergreifend eindeutige Identität zu verwenden. Das ist nachfolgend veranschaulicht.

\[ \text{<uniqueObjectID>} ::= \text{<systemID>}.${\text{<systemObjectID>}} \]

Die Identitäten aller Quellsysteme müssen daher an einer zentralen Stelle verwaltet werden, z.B. zusammen mit weiteren Metadaten der Quellsysteme.

### 3.6.2 Nutzung bzw. Generierung von Verknüpfungen

Beziehungen zwischen Objekten sind in den Quellsystemen in verschiedenen Varianten abgelegt. Sie können explizit oder implizit vorhanden sein und sich auf dasselbe oder auf ein anderes Quellsystem beziehen. Die wichtigsten davon werden nachfolgend zusammengestellt.

#### Explizite Beziehungen zwischen Objekten innerhalb eines Systems

Explizite Beziehungen zwischen zwei Objekten innerhalb eines Quellsystems bestehen dann, wenn beide Objekte in einem einzigen Quellsystem vorhanden und über eine Referenz oder mehreren Referenzen miteinander verbunden sind. Die technische Umsetzung der Referenzen hängt dabei vom Quellsystem ab und kann z.B. über Fremdschlüssel (relationale Datenbank), Zeiger (objektorientiertes System) oder Kanten (Graphdatenbank) realisiert sein.

Eine Beziehung zwischen zwei Objekten lässt sich in Form ihrer systemübergreifenden Identitäten

\[ \text{<systemID>}.${\text{<systemObjectIDa>}} \sim \text{(Beziehungstyp)} \sim \text{<systemID>}.${\text{<systemObjectIDb>}} \]

darstellen, wenn es sich dabei um eine ungerichtete Verknüpfung handelt, bzw.

\[ \text{<systemID>}.${\text{<systemObjectIDa>}} \rightarrow \text{(Beziehungstyp)} \rightarrow \text{<systemID>}.${\text{<systemObjectIDb>}} \]

derart...
im Falle einer gerichteten Beziehung (beide Richtungen sind möglich) handelt. Die SystemID ist dabei in beiden Fällen gleich und für die systemObjectID werden jeweils die Identitäten der Objekte verwendet. Der Beziehungstyp erlaubt die Unterscheidung verschiedenartiger Beziehungen zwischen Objekten und entspricht dem Prädikat eines RDF-Tripels (W3C 2004c). Alle expliziten Beziehungen zwischen Objekten innerhalb eines Systems lassen sich also bei Kenntnis der inneren Struktur des Systems automatisiert auf systemübergreifende Art darstellen.

**Explizite Beziehungen zwischen Objekten über Systemgrenzen hinweg**

Explizite Beziehungen zwischen zwei Objekten, die sich in verschiedenen Quellsystemen befinden, werden meist über die Nutzung von gemeinsamen (übergreifend festgelegten) Schlüsseln realisiert, z.B. die eindeutige Nummer einer Messstation. Werden in beiden Systemen dieselben Schlüssel verwendet, lässt sich das als unge richtete Beziehungen zwischen Objekten interpretieren, d.h. man kann von einem Objekt das andere erreichen und umgekehrt. Zum Beispiel können in einem System die Metadaten zu einer Messstation gespeichert sein, in einem zweiten System die zugehörigen Messdaten. Eine solche Verknüpfung ist ungerichtet und lässt sich über Systemgrenzen hinweg in der Form:

\[
<\text{systemIDa}.<\text{systemObjectID}> \rightarrow \langle\text{systemIDb}.<\text{systemObjectID}\rangle
\]

ausdrücken.

Eine zweite Variante der expliziten Beziehungen über Systemgrenzen hinweg ist die einseitige, gerichtete Beziehung eines Objekts a im Quellsystem A zu einem Objekt b im Quellsystem B. Dann ist, z.B. in einem Attribut von a, die Referenz (z.B. ID) auf das Objekt b im Quellsystem B gespeichert, das Objekt b besitzt jedoch seinerseits keine Referenz auf das Objekt a in A. Die gerichtete Beziehung von a zu b lässt sich folgendermaßen ausdrücken:

\[
<\text{systemIDA}.<\text{systemObjectIDa}> \rightarrow (\text{Beziehungstyp}) \rightarrow \langle\text{systemIDB}.<\text{systemObjectIDb}\rangle
\]

Ein typisches Beispiel für eine gerichtete Beziehung ist die Zuordnung eines Objektes zu einer benannten Örtlichkeit (Lage des Objekts), z.B. Objekt a befindet sich im Landkreis b.

Explizite Beziehungen zwischen Objekten in verschiedenen Systemen lassen sich bei Kenntnis der inneren Struktur beider Systeme ebenfalls automatisiert darstellen, je nach Art der Verknüpfung in der gerichteten oder ungerichteten Form.
Implizite Beziehungen zwischen Objekten

Implizite Beziehungen zwischen zwei Objekten a und b (im Allgemeinen aus verschiedenen Quellsystemen A und B) bestehen dann, wenn aus den Eigenschaften von a über eine Abbildung (ggf. unter Hinzuziehung von Zusatzwissen) eine Beziehung zum Objekt b hergestellt werden kann. Die Abbildung kann operational außerhalb der Quellsysteme erfolgen, z.B. durch einen externen Dienst.

Stellen zum Beispiel alle Objekte aus A ihre geographische Position in Form von Geokoordinaten (z.B. als Latitude-Longitude-Paar) zur Verfügung und haben alle Objekte in B (z.B. Landkreise) eine flächenhafte Geometrie, so lässt sich über eine simple geometrische/geographische Operation bestimmen, in welchem Landkreis b ein Objekt a liegt. Sind die Landkreise flächendeckend verfügbar und haben paarweise disjunkte Flächen, ist die Abbildung eindeutig. Aus den Identitäten von a und b sowie deren Quellsysteme A und B lassen sich dann wieder explizite Beziehungen konstruieren, z.B. der Art

\[ \text{systemIDA}.\text{systemObjectIDa} \rightarrow \text{systemIDB}.\text{systemObjectIDb} \]

wobei "liegt in" den (neuen) Beziehungstyp darstellt.

Implizite Beziehungen und lose Kopplung von Objekten


Sind Objekte bezüglich ihrer (Geo-)Attribute abfragbar, lassen sich Beziehungen direkt in der konsumierenden Anwendung verwenden, z.B. um alle erreichbaren Objekte in einem gegebenen Kartenausschnitt anzuzeigen.

Einen weiteren Kandidaten für lose Kopplungen stellen zeitliche Bezüge (Zeitpunkte, Zeiträume) dar.

Indirekte Beziehungen zwischen Objekten und Ableitung neuer Beziehungen

Gegebenenfalls lassen sich aus bestehenden Beziehungen weitere Beziehungen konstruieren bzw. ableiten, insbesondere wenn mittelbare Beziehungen bestehen. Hat ein Objekt a eine Beziehung zu Objekt b und Objekt b eine Beziehung zu Objekt c, so lässt sich daraus eine Beziehung zwischen den Objekten a und c konstruieren (Transitivität).
Zum Beispiel kann ein Objekt a im Landkreis b liegen und der Landkreis b Teil des Bundeslandes c sein. Dann lässt sich eine Beziehung „Objekt a liegt in Bundesland c“ bilden. Zur Konstruktion solcher Beziehungen bedarf es jedoch weiterer Informationen bzw. des entsprechenden Fachwissens, das z.B. durch eine formale Semantik oder durch Inferenzdienste bereitgestellt werden kann, so dass Ableitungen ggf. auch automatisiert gewonnen werden können (Reasoning).

### 3.6.3 Beziehungsdienst

Wenn Daten aus Quellsystemen über einheitliche (Micro-)Services mit RESTful URLs bereitgestellt werden, sollen Objektidentitäten und Beziehungen zwischen Objekten selbstverständlich erhalten bleiben. Dazu wird der Beziehungsdienst benötigt.

Auf der Seite der Quellsysteme werden Identitäten von Objekten nach dem Vorschlag aus Abschnitt 4.5.1 anhand von Schlüsseln nach dem Muster

\[
<\text{uniqueObjectID}> := <\text{systemID}>.<\text{systemObjectID}>
\]

eindeutig identifiziert.

Die RESTful URLs der einzelnen Dienste haben eine Form nach dem Muster

https://<baseurl>/<function>/<domain>/<type>/<id>/<aspect>

Die <baseurl> gibt dabei eine reale Server Adresse an, über die alle Anfragen be-dient werden (z.B. ein Gateway bzw. ein Dispatcher-Dienst).

Die <function> dient der Unterscheidung von (generischen) Grundfunktionalitäten bzw. -diensten, z.B. zur Bereitstellung von Objektdaten, Zeitreihen oder Mediendaten.

Die <domain> kann der weiteren Strukturierung von Daten dienen, z.B. wenn gleichartige Daten für verschiedene Bundesländer vorliegen und danach unterschieden werden sollen. Die Angabe einer Domain ist optional, eine <domain> kann aber ggf. auch mehrere Hierarchiestufen einnehmen, z.B. "/bw/ka" oder (intern) "bw.ka" wenn innerhalb eines Bundeslandes auch nach Regierungspräsidien unterschieden werden soll.

Die <type> dient der inhaltlichen Unterscheidung von Objekten, zum Beispiel nach deren Klasse/Konzept und ist zwingend notwendig.

Die <id> bezeichnet ein konkretes Objekt des Typs. Ist keine <id> angegeben, so werden alle Objekte des Typs geliefert bzw. alle die zu den weiteren Parametern pas-sen.

Der optionale Bestandteil <aspect> kann der weiteren Diskriminierung der Anfrage dienen.

Er kann durch weitere Parameter ergänzt werden.

Konkret liefert z.B. die URL

https://linked-energy.org/objects/bw/windturbines/303
die Objektdaten zu einer bestimmten Windkraftanlage in Baden-Württemberg, während die URL

https://linked-energy.org/measures/bw/windturbines/303/power
die zugehörigen Messwerte (z.B. die aktuelle Einspeiseleistung) liefert.

Um eine eindeutige Zuordnung von Daten aus den Quellsystemen und den über die RESTful APIs bereitgestellten Repräsentationen gewährleisten zu können, z.B. zur Synchronisation der Daten bei Änderungen auf der einen oder der anderen Seite oder zur Auflösung von Beziehungen, besteht grundsätzlich die Notwendigkeit einer Abbildung der eindeutigen, aus den Quellsystemen konstruierten <uniqueObjectID>, die zur Adressierung der Daten im Quellsystem benötigt wird und der entsprechenden RESTful URL, welche die Daten im (Micro-)Service adressiert. Die Abbildung muss in beiden Richtungen funktionieren.

Mithilfe einer solchen Abbildung ist es zum Beispiel möglich, bei der Synchronisierung von Daten aus dem Quellsystem mit einem Service die Beziehungen zu anderen Objekten zunächst durch Verwendung der eindeutigen <uniqueObjectID>s und anschließend durch deren Substitution durch RESTful URLs die Anforderung 4 zu Linked Data (s. Abschnitt 3.6) zu erfüllen.

Der Beziehungsdienst muss dazu Funktionen in beiden Abbildungsrichtungen bereitstellen, z.B. nach den URL-Mustern

https://linked-energy.org/links/uniqueid/<urlFragment>

oder

https://linked-energy.org/links/uniqueid?url=<encodedUrl>
bzw.

https://linked-energy.org/links/url/<uniqueId>
or

https://linked-energy.org/links/url?id=<uniqueId>

Die notwendigen Daten für die Abbildung können beim Anschluss eines Quellsystems gewonnen werden. Es muss eine eindeutige ID für das Quellsystem vergeben und das Schlüsselattribut (ggf. künstlich) für die Objekte innerhalb des Quellsystems festgelegt werden. Bei der Definition des URL-Raums des zugehörigen Dienstes bzw. der zugehörigen Dienste müssen ggf. die mögliche(n) Funktion(en), die Domain sowie der Typ der Daten festgelegt werden.

Damit stehen alle notwendigen Informationen für die Abbildung zur Verfügung und bei der Synchronisation der Daten können dem Beziehungsdienst sowohl uniqueObjectIDs als auch URLs zur Verfügung gestellt werden.

In vielen Fällen lässt sich der Aufwand durch die Nutzung von Konventionen deutlich reduzieren, dann lassen sich uniqueObjectIDs und URLs auf rein syntaktischer Ebene ineinander überführen, was im Folgenden dargestellt wird.
Konvention für die Abbildung von Objektschlüsseln auf RESTful URLs

In einem konkreten Beispiel sind die Daten aller Windkraftanlagen aus Baden-Württemberg in einem einzigen Quellsystem verfügbar und mit eindeutigen numerischen IDs versehen.

Wählt man den Schlüssel für das Quellsystem mit Bedacht, z.B. den Namen eines Konzeptes, das dem Inhalt der Datenquelle entspricht, wie “windturbine”, und verwendet zur Identifikation der einzelnen Objekte deren Schlüssel im Quellsystem, dann entstehen eindeutige Schlüssel für das Quellsystem nach dem Muster

\[ <\text{uniqueObjectID}> ::= <\text{systemID}>().<\text{systemObjectID}> \]

für die Windturbine mit der ID 303 in der konkreten Form

windturbine.303

Für den in der Praxis durchaus häufigen Fall, dass gleichartige Objekte aus genau einem Quellsystem stammen, lässt sich mit Hilfe der vorgestellten Konvention erreichen, dass eine Abbildung von RESTful-URLs und den uniqueObjectIDs zur eindeutigen Identifikation von Objekten trivial, d.h. eine rein syntaktische Umwandlung, ist. Ggf. müssen dazu Metadaten des Quellsystems genutzt werden, z.B. die zugehörige Domäne (<domain>).

Abbildungen zwischen uniqueObjectIDs und URLs können so leicht berechnet werden, d.h. sie müssen nicht explizit gespeichert werden.

3.6.4 Metadatendienst

Metadaten sollen die Inhalte eines Informationssystems und ggf. die Möglichkeiten zum Zugriff darauf genauer beschreiben. Sie sollen die Nutzung der Daten durch andere Systeme ermöglichen, d.h. sie liefern einen wichtigen Beitrag zur Interoperabilität von Systemen. Daher beinhalten Metadaten Informationen zur Semantik, zum Datenmodell und zur Syntax der Daten.

Im Sinne der vorgestellten Architektur kann es Metadaten auf zwei Ebenen geben, der Ebene der Quellsysteme sowie der bereitgestellten (Micro-)Services.

Erstere sollten eigentlich zu jedem Quellsystem existieren, in der Realität liegen jedoch häufig keine Metadaten vor, zumindest nicht in einer standardisierten bzw. maschinell verarbeitbaren Form.

Für alle (Micro-)Services bzw. für alle durch sie verfügbar gemachten Datenquellen sollen ebenfalls Metadaten bereitgestellt werden. Zwei Varianten bieten dabei unterschiedliche Sichten auf die Systeme - die öffentlich verfügbaren Metadaten, welche die oben beschriebenen Informationen zum Zugriff (Semantik, Datenmodell und Syntax) auf den Service enthalten, sowie eine interne Sicht, die darüber hinaus weitere Informationen, z.B. über die verwendeten Schlüsselattribute im Quellsystem, den eindeutigen Schlüssel für des zugehörigen Quellsystems und weitere für die Datensynchronisation notwendige Angaben (z.B. Abbildungsvorschriften für Attribute, Aktualisierungintervalle etc.) enthält.
Der Metadatendienst dient in der zweiten Sicht somit auch der Verwaltung der ange- 
schlossenen Quellsysteme.

3.7 Gegenüberstellung der vier Architekturvarianten

Die vier in den vorigen Abschnitten vorgestellten Architekturvarianten bauen aufeinan-
der auf bzw. in den späteren Varianten wurden die Schwächen ihrer Vorgänger in der 
Architektur beseitigt.

Die folgende Tabelle stellt die vier Architekturvarianten kompakt gegenüber und zeigt 
ihre Vor- und Nachteile auf:

<table>
<thead>
<tr>
<th>Architekturvariante mit wesentlichen Merkmalen</th>
<th>Vorteile</th>
<th>Nachteile</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Erste Architekturvariante:</strong></td>
<td>Schnelle Umsetzung auf Basis eines vorhandenen Portals und einer Suchmaschine, seit über 10 Jahren produktiv</td>
<td>Keine echte Nutzung von Semantik OneBoxes profitieren nicht vom Umweltthesaurus Einschränkungen bei der Ergebnispräsentation (OneBoxes nicht in Trefferliste integriert) Geringe Bandbreite von Datentypen (keine Karten, Zeitreihen etc.) durch Beschränkung auf Volltextsuche Fehlende Standardisierung: Zur Anbindung jedes Zielsystems als OneBox ist ein Adapter notwendig</td>
</tr>
<tr>
<td>Semantische Erweiterung einer kommerziellen Volltextsuchmaschine, Nutzung externer Datenquellen über OneBox-Mechanismus</td>
<td>Umweltthesaurus leicht in die Volltextsuche integrierbar Schnelle Anbindung neuer Zielsysteme über OneBox-Adapter Erweiterung der Suchanfrage lässt sich ergänzend, aber auch unabhängig von anderen Suchmechanismen nutzen</td>
<td></td>
</tr>
<tr>
<td>Serverseitige Verarbeitung der Suchanfrage, SearchBroker und Ontologiesystem</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dritte Architekturvariante: Serviceorientierung, „Webcache“, clientseitige Verarbeitung</td>
<td>Flexible und leistungsfähige Sammlung von Diensten (SOA) („Webcache“)</td>
<td>Nicht in das Semantic Web integriert, kein Linked Data</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td></td>
<td>Modularer, Komponenten-basierter Client mit rein clientsitierer Kommunikation (EventBus) und vielen Darstellungsvarianten</td>
<td>Semantische Hintergrundinformationen werden fast nur beim Transformationsprozess (Data Ingestion) genutzt</td>
</tr>
<tr>
<td></td>
<td>Asynchrone Verarbeitung von Anfragen an die Services</td>
<td>Große Anzahl von Schnittstellen beim Transformationsprozess (Data Ingestion) zur Anbindung vieler heterogener Zielsysteme</td>
</tr>
<tr>
<td></td>
<td>Umfangreiche Unterstützung aller in den Domänen „Umwelt“ und „Energie“ benötigten Datentypen</td>
<td>Manuelle Arbeit bei der Harmonisierung von Vokabularen</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Vierte Architekturvariante: Ausbau zu semantischen Diensten, Linked Data</th>
<th>Unterstützung von Linked Data</th>
<th>Noch nicht vollständig umgesetzt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Generierung zusätzlichen Wissens (z.B. Beziehungen) gegenüber den originalen Zielsystemen</td>
<td>Große Anzahl von Schnittstellen beim Transformationsprozess (Data Ingestion) zur Anbindung vieler heterogener Zielsysteme</td>
</tr>
<tr>
<td></td>
<td>Einbeziehung globaler Schemata, z.B. schema.org, dadurch Erhöhung möglicher Interoperabilität</td>
<td>Manuelle Arbeit bei der Harmonisierung von Vokabularen</td>
</tr>
</tbody>
</table>

Tabelle 1: Die Vor und Nachteile der vier Architekturvarianten gegenübergestellt.

Bei der vierten Architekturvariante bleiben am Ende keine wesentlichen inhaltlichen Nachteile stehen, außer denen, die sich, wie auch bei den vorhergehenden Varianten, direkt aus der Heterogenität der betrachteten Zielsysteme ergeben. Die vierte Architekturvariante ist jedoch noch nicht vollständig, d.h. etwa zu 90%, umgesetzt. Im Gegensatz dazu gibt es für die ersten drei Architekturvarianten zahlreiche praktische Umsetzungsbeispiele, die im folgenden Kapitel 4 beschrieben werden.
4 Umsetzungsbeispiele


4.1 Energieportal Baden-Württemberg


17 https://www.umwelt-bw.de
18 https://www.umwelt.sachsen-anhalt.de
19 http://www.umweltportal.thueringen.de
Umsetzungsbeispiele


Mit der Inbetriebnahme des Energieatlas Baden-Württemberg (Abschnitt 4.3) wurde das Energieportal außer Betrieb genommen.

4.2 Semantische Suche nach Umweltinformationen (SUI)

Im Projekt „Semantische Suche nach Umweltinformationen“ (SUI) des Fraunhofer Instituts IOSB und des Instituts für Angewandte Informatik (IAI) am KIT wurden in Zusammenarbeit mit der LUBW und dem UM Baden-Württemberg Informationstechnologien entwickelt und evaluiert, mit denen die Volltextsuche in Umweltportalen durch den Einsatz von semantischen Technologien und Serviceorientierung verbessert werden kann.

Das Konzept von SUI basiert auf der Architekturvariante 2 (Serverseitige Verarbeitung der Suchanfrage, SearchBroker und Ontologiesystem), d.h. darauf, dass die Semantik der Informationen von ausgewählten Fachinformationssystemen (sogenannter Zielsysteme) über Zielsystembeschreibungen, die eine Kategorisierung des Inhalts des Ziel-
Umsetzungsbeispiele

90


Ontologien und Ontologiemapping

Im SUI-System werden zurzeit die folgenden Ontologien eingesetzt:

- **GEMET (GEneral Multilingual Environmental Thesaurus):** Der Thesaurus ist im SKOS-Format verfügbar und bietet eine umfassende Basis von Suchterminen. Er ist auch Bestandteil des Semantic-Network-Service (SNS) des Umweltbundesamtes (Umweltbundesamt 2016).

- **Objektarten-Katalog (OK):** In integrierten Umweltinformationssystemen und Geodatenverbünden werden für die Fachbereiche alle Objektarten, die Realweltphänomene repräsentieren, in sog. Objektarten-Katalogen (bei ISO/OGC auch Feature Type Catalogues genannt) geführt und damit der Datenaustausch erleichtert. Wenngleich nicht ursprünglich für den Einsatz in Suchportalen konzipiert, liefert der OK wichtige Informationen für den Zugriff auf Umweltinformationen in Datenbanken, z.B. Objekttarncodes und Fachführungscodes. Im Rahmen von SUI wurde der OK in eine Ontologie transformiert.


Die Liste eingesetzter Ontologien ist offen, d.h., es gibt Pläne zur Integration weiterer verfügbarer Systeme. Die einzelnen Ontologien müssen jedoch miteinander in Verbindung gebracht werden, was im Folgenden beschrieben wird (Abbildung 15):
Die eingesetzten Ontologien (bzw. die zugrundeliegenden originalen Begriffssysteme) werden unabhängig voneinander durch unterschiedliche Experten-Gremien entwickelt, z.B. Lebenslagen, Objektartenkatalog (OAK) oder der Umweltthesaurus (SNS) auf der rechten Seite. Sie modellieren spezifisches Wissen, das auf die jeweilige Domäne fokussiert ist. Querbezüge zwischen Domänen werden hierdurch nicht erfasst. Prinzipiell können identische Phänomene in unterschiedlichen Domänen durch unterschiedliche Begriffsstrukturen und Relationen modelliert sein. Das SUI-System muss daher in der Lage sein, Querbezüge zur Nutzung durch die semantische Suche explizit zu machen.


Im SUI-System kann nun durch Herstellung einer expliziten Beziehung zwischen beiden Begriffen das Ergebnis der thematischen Aufbereitung entscheidend verbessert werden.

Für die Erzeugung der Querbezüge steht eine breite Palette automatisch arbeitender Werkzeuge für das Ontologie-Mapping zur Verfügung. In SUI wird ein Werkzeug eingesetzt, das im Rahmen des deutschen Forschungsprogramms THESEUS (Bundesministerium für Wirtschaft und Energie 2017) u.a. durch das Fraunhofer-Institut für Opt-
Umsetzungsbeispiele


Abbildung 16: Ontology-Mapping im SUI-System; aus: (Bügel et al. 2011b)

Da Mappings zwischen Ontologien bilateral sind, andererseits aber in der Anwendung beliebig viele Ontologien zum Einsatz kommen können, wurde für SUI eine einfach handhabbare, Web-basierte Verwaltung automatisch erzeugter Mappings entwickelt und in das SUI-System integriert. Mit Hilfe eines Workflows können Mappings definiert, automatisch erzeugt, nachbearbeitet und für die semantische Suche aktiviert werden. Insbesondere der Möglichkeit zur Nachbearbeitung kommt besondere Bedeutung zu. Obwohl die verfügbaren Werkzeuge teilweise sehr gute Trefferquoten vorweisen, sollten automatisch erzeugte Mappings lediglich als initiale Vorschläge aufgefasst, kritisch
hinterfragt und anwendungsspezifisch angepasst werden. Abbildung 16 zeigt ein Beispiel eines Mappings zwischen GEMET (linke Bildhälfte) und der Lebenslagen-Ontologie (rechte Bildhälfte).

Die in der Bildmitte dargestellten automatisch gefundenen Mappings können akzeptiert, verändert, selektiv gelöscht und durch weitere manuell erzeugte Mappings ergänzt werden.
4.3 Energieatlas 2015


4.3.1 Ziele und Zielgruppen des Energieatlas

Der Energieatlas richtet sich sowohl an interessierte Bürgerinnen und Bürger, als auch an Fachleute und Entscheidungsträger in Verwaltung, Forschung und Wirtschaft und
Umsetzungsbeispiele

stellt wichtige Informationen zum Stand der dezentralen Energieerzeugung und zum regionalen Energiebedarf zur Verfügung. Darüber hinaus bietet er mit seinem landesweiten Überblick Energieberatern, Planern und anderen interessierten Akteuren Hintergrundinformationen und Handreichungen an.

Lokale, kommunale und regionale Planungen können durch den Energieatlas nicht ersetzt werden, insbesondere stellt er keine Planungsgrundlage für die Regional- und Bauleitplanung dar. Vielmehr ist es Ziel des Energieatlas, alle an der Energiewende beteiligten Akteuren Daten und Informationen bereit zu stellen, auf deren Basis Strategien und Maßnahmen zur Erfüllung der gemeinsamen Klimaschutzziele entwickelt werden können.


Mit Hilfe vorkonfigurierter Links kann von jeder Seite des Energieatlas aus das erweiterte Daten- und Kartenangebot aufgerufen werden. Dabei wird der aktuelle Informationskontext beibehalten, indem dort automatisch die zum momentan ausgewählten Thema passenden Inhalte geladen werden und ggf. der gerade betrachtete Kartenausschnitt eingestellt wird.

Abbildung 17: Daten zu bestehenden Windkraftanlagen im erweiterten Daten- und Karrenangebot des Energieatlas (Screenshot)


4.3.2 Erscheinungsbild des Energieatlas


Durch die Verwendung derselben Farben und eines ähnlichen Seitenablaufs mit horizontalem Hauptmenü ist die Nähe des Energieatlas zu LUBW Homepage klar zu erkennen. Vom LUBW Webauftritt aus ist der Energieatlas von den Themen Seiten Erneuerbare Energien direkt verlinkt. Er wirkt somit quasi in die LUBW Seiten integriert, behält aber trotzdem den Charakter eines eigenen Webauftrittes bei.

Das Layout der Seiten ist zweigeteilt und verwendet ein Navigationsmenü auf der linken Seite sowie einen Inhaltsbereich auf der rechten Seite. Die Bereiche sind durch verschiedene Hintergrundfarben klar unterscheidbar. Die Breite des Inhaltsbereiches wurde auf 1000 Pixel erweitert, damit sich Karten möglichst groß darstellen lassen. Der
Navigationsbereich wurde dafür im Vergleich zu den LUBW Webseiten auf 170 Pixel Breite verkleinert.

Ein Merkmal des neuen Layouts ist die Verwendung eines einleitenden Bildes pro Thema über die gesamte verfügbare Bildschirmbreite. Dadurch wird die Seite optisch aufgewertet und zeigt sich moderner. Das Bild wird bei Verwendung von verschiedenen breiten Endgeräten mit unterschiedlicher Auflösung immer so skaliert, dass die Bildmitte zentriert bleibt und der linke und rechte Rand abgeschnitten werden. Die Bilder wurden im Vorfeld so ausgewählt, dass deren Informationsgehalt dabei nicht verloren geht.


Umsetzungsbeispiele

99

geschrieben (SCSS) und werden von einem SASS-Preprocessor (Syntactically Awesome Style Sheets) in CSS-Dateien konvertiert.

Für die Redakteure wurden außerdem zahlreiche Vorlagen entwickelt (Liferay Webcontent Strukturen und Application Display Templates), mit denen Inhaltskomponenten identisch darstellt werden können. Alle Projektbeschreibungen für Einzelprojekte und Biogasanlagen basieren beispielsweise auf nur einer Vorlage und werden so klar strukturiert und auf sehr übersichtliche Art und Weise angezeigt.

4.3.3 Systemarchitektur


Für die Bereitstellung von Daten kommen verschiedene Hintergrunddienste zum Einsatz, die überwiegend durch eine Cloud-basierte Infrastruktur bereitgestellt werden. Sämtliche Geodaten werden in Form von Kartenlagen durch eine auf CartoDB (CAR-TO 2017) beruhenden Server-Anwendung bereitgestellt. Originaldaten aus verschiedenen Fachsystemen werden so vom Energieatlas entkoppelt und im „Webcache“ (s. 3.5.1) redundant vorgehalten. CartoDB bietet dabei neben vektor- und kachelbasierten Ansichten der Geodaten auch Programmierschnittstellen (APIs) zum Abruf der Objekt- daten, die z.B. für die Breitstellung von Detailansichten genutzt werden.


Aufgrund der serviceorientierten Architektur werden viele Inhalte des Energieatlas mehrfach verwendet, zum Beispiel stehen viele Kartenlagen ebenfalls im Umweltportal Baden-Württemberg zur Verfügung oder werden in der mobilen App „Meine Umwelt“ zur lokalisierten Information über erneuerbare Energien angeboten. Neue Inhalte und
Umsetzungsbeispiele

Dienste lassen sich durch das modulare, auf Web-Widgets bzw. Portlets basierende Konzept leicht in den Energieatlas integrieren.

4.3.4 Weiterentwicklung und Flexibilisierung der Liferay-Portlets


Zum Informationsaustausch kommunizieren die Frontend-Komponenten des LUPO-Baukastens über einen ereignisbasierten Kommunikationsbus auf Basis eines Publish-Subscribe-Modells (Eugster et al. 2003), d.h. sie melden sich an einer clientseitigen Kommunikationsplattform („Eventbus“) an, und können darüber Nachrichten versenden bzw. empfangen, z.B. bei Nutzerinteraktionen. Klickt ein Nutzer in der Karte auf ein Objekt oder ändert die Zoomstufe und damit den angezeigten Kartenausschnitt, wird ein Ereignis ausgelöst, das an andere UI-Komponenten weitergeleitet wird, die dann autonom darauf reagieren können, z.B. Sachdaten des in der Karte angeklickten Objektes anzeigen.

Die für den Energieatlas gemachten Erweiterungen, z.B. die Anzeige von Detailinformationen beim Klick auf Objekte im Kartenclient oder in einer Liste, flossen an den LUPO-Baukasten zurück und erweiterten so dessen Flexibilität und Funktionsumfang um z.B. die Möglichkeit zur Integration eines Orts-Suchschlitzes in die Kartenkomponente, die Anzeige von Default-Inhalten in der Objektinformation-Komponente, die Bereitstellung des aktuellen Kartenausschnitts als Event für weitere Komponenten oder die Möglichkeit zum orts- und themenscharfen Einsprung in das Fachsystem Umwelt- und -karten online (UDO). Darüber hinaus wurden die Möglichkeiten zur Attribuierung von Kartenkonfigurationen erweitert, z.B. um Legenden in den Kartenansichten per Vorkonfiguration ein- bzw. auszublenden.

Zusätzlich zur Weiterentwicklung bestehender Komponenten, z.B. des Kartenwidgets, der Kartenlayer-Auswahl und der Objektinfo-Anzeige, wurden einige neue Komponenten spezifisch für den Energieatlas entwickelt, beispielsweise die Anzeige von Kennzahlen für die Einspeisung von Wind- und Solarenergie auf der Startseite (Abbildung 18). Die Komponente ist ein typisches Beispiel für die Umsetzung der serviceorientierten Architektur und damit auch eine Art Blaupause für die Entkopplung der Datenbereitstellung per Service und der Anzeige der Daten im Portal. Die Originaldaten werden dabei in Form von XML-Dokumenten durch den Netzbetreiber bereitgestellt. Ein Update-Service transferiert die Daten zu in der Google Cloud gehosteten relationalen Da-
Ein weiterer Dienst konsumiert die Daten aus der Datenbank und stellt sie in Form einer definierten REST-Programmierschnittstelle als JSON- (JavaScript-) Objekte zur Verfügung, die durch die Anzeige-Komponenten direkt verarbeitet, d.h. Template-basiert dargestellt werden können. Durch die so erreichte Entkopplung wird der Dienst, der die Originaldaten bereitstellt, nur minimal belastet. Die potenziell zahlreichen Anfragen aus dem Energieatlas werden durch eine leistungsfähige, skalierbare Infrastruktur (App Engine und Cloud SQL) in der Cloud bearbeitet. Mögliche Latenzen bei Updates der Originaldaten werden durch eine hinreichend häufige Abfrage der Daten durch den Update-Service minimiert.

### 4.4 LUPO-Portale


Ein Schwerpunkt des Umbaus war die Einführung von Services, die den Umweltportalen Zugang zu Daten, insbesondere Messwerten, Sachdaten sowie Kartendaten bieten. Dabei kommen auch bewährte Dienste wie die klassische Volltextsuchmaschine zum Einsatz, die nun über eine unabhängige Schnittstelle angebunden ist und durch eine Suchmaschine für strukturierte Daten ergänzt wird.

Im Bereich der Frontend-Komponenten wurde auf eine von konkreten CMS- und Portalisystemen unabhängige Lösung gesetzt. Alle entwickelten UI-Komponenten sind als Web Widgets verfügbar und damit grundsätzlich in beliebigen Webseiten verwendbar (s. 3.5.3). Für die Nutzung in modernen Portalsystemen wie Liferay Portal stehen jedoch sogenannte Wrapper-Portlets für alle Widgets zur Verfügung, welche den Komfort bei der Einbindung und Konfiguration der Widgets innerhalb von Portalen deutlich erhöhen.

Die Klammer für die unabhängig voneinander nutzbaren, generischen Frontend-Komponenten bildet eine Ereignis-basierte Kommunikationsschicht in Form eines Eventbusses. Der Eventbus bietet Kanäle zum Nachrichten- und Datenaustausch unter
den Komponenten und damit auch die Möglichkeit, ein Zusammenwirken von selbstständigen Komponenten mit dem Ziel zu erreichen, dem Nutzer eine schlüssige Gesamtanwendung präsentieren zu können. Um die Anforderungen an die Daten und Dienste dabei möglichst gering zu halten, z.B. um bestehende Dienste und Datenbestände einbeziehen zu können, wurde zunächst auf eine relativ lose Kopplung auf Basis von allgemeinen Nachrichten (Ortsbezug, Themenbezug, Suchbegriffe) gesetzt.


**Diskussion der Event-basierten Kommunikation von Komponenten**

Für die Landesumweltportale hat die Event-basierte Kommunikation mit einer losen Kopplung von (Umwelt-)Objekten einen entscheidenden Vorteil: Sie reduziert den Aufwand bei der Einbindung von Umweltdaten in die Landesumweltportale auf ein leistbares Niveau. Die Landesumweltportale bieten auf der einen Seite Zugang zu einer äußerst heterogenen Landschaft von Umweltdaten:

- Struktur (strukturierte, semistrukturierte, unstrukturierte Daten)
- Ortsbezug (Daten mit und ohne expliziten Ortsbezug in einer Vielzahl von Präsentationen)
• Daten aus unterschiedlichen technischen Systemen mit einer Vielzahl von Schnittstellen und technischen Formaten, verschiedenen IDs oder Schlüsselis-
ten etc.

Die Daten- bzw. Systemlandschaft bietet in den meisten Fällen keine expliziten Bezie-
hungen zwischen Daten und Objekten bzw. zumindest keine technisch nutzbare Um-
setzung dafür.

Auf der anderen Seite erwarten die menschlichen Nutzer der Umweltportale in den
meisten Fällen zwar eine Unterstützung beim Auffinden der passenden Informationen
zu ihrem Anliegen, stellen dabei aber selbst eine aktive Filterinstanz dar, welche die
angezeigten Informationen sichten, bewerten und sich passende Teile herauspicken
kann. Eine – nicht zu große – Obermenge der tatsächlich relevanten Ergebnisse ist für
sie in den meisten Fällen akzeptabel. Des Weiteren hat sich gezeigt, dass bei einem
großen Anteil der Suchanfragen die Beziehungen zwischen den passenden Ergebnis-
sen auf einem sehr hohen Abstraktionsniveau darstellbar sind, z.B. ihrer örtliche Nähe
zueinander. So kann beispielsweise die Frage, ob die Windkraftanlagen einer Gemein-
de innerhalb oder in der Nähe von Naturschutzgebieten liegen, mit Hilfe der Umwelt-
portale sehr leicht beantwortet werden: Allein durch Eingabe der Suchbegriffe „windrad
schutzgebiet langenburg“ erhält der Nutzer bereits die gewünschten Informationen,
allerdings tatsächlich mehr als verlangt, da neben Naturschutzgebieten auch Objekte
anderer Schutzgebietstypen (Biotope, Nationalparke etc.) dargestellt werden.

Durch das einfache Abwählen der nicht benötigten Schutzgebietstypen lässt sich die
obige Frage klären, denn alle Naturschutzgebiete und Windkraftanlagen im Bereich der
Gemeinde Langenburg werden angezeigt. Zwar muss der Nutzer die Beziehung zwis-
chen Windkraftanlagen und Schutzgebieten noch selbst herstellen, allerdings gelingt
das dank Kartenansicht „auf einen Blick“ – obwohl in der verwendeten Datengrundlage
eine Beziehung wie „liegt in/bei“ nicht vorhanden ist; im Gegenteil: Informationen über
Windkraftanlagen und Naturschutzgebiete kommen aus völlig unterschiedlichen Sys-
temen und sind nur über die gemeinsame Darstellung innerhalb des Kartenclients mit-
einander verbunden.

Die lose Kopplung von Umweltobjekten funktioniert also nur für menschliche Nutzer
und nur in solchen Anwendungsfällen, in denen sich Beziehungen auf relativ hohen
Abstraktionsniveaus darstellen lassen, z.B. ihre örtliche Nähe oder die Zuordenbarkeit
to einem bekannten Thema (Windrad ➔ Windkraft).
4.5 Mobile Apps


Im Folgenden wird zunächst die App „Meine Umwelt“ vorgestellt und dann auf die Funktionen der neuen App „Meine Pegel“ eingegangen. Schließlich werden die gemeinsamen Grundlagen beider Apps im Rahmen des App-Baukastens „LUPO mobil“ (für „Landesumweltportale mobil“) dargestellt.

4.5.1 App „Meine Umwelt“


Die Ausweitung auf weitere Bundesländer über die Integration von bundesweiten Themen sowie über die Aufnahme weiterer Partner in der LUPO-Kooperation ist in Planung.


Abbildung 21: Bereich Informieren beinhaltet Karten mit Unterthemen (links), Detailinformationen zu ausgewählten Objekten (mittig) sowie aktuelle Messwerte (rechts) (Screenshots der App „Meine Umwelt“)

Abbildung 22: Verschiedene Meldethemen (links), Formular zum Erfassen von Standort, Sachdaten (mittig) im Bereich „Melden“, sowie die Anzeige von eingegangenen Meldungen im Bereich „Informieren“ (Screenshots der App „Meine Umwelt“)


4.5.2 LHP-App „Meine Pegel“


In Abbildung 23 werden die Startseite des Portals zum Stand des Hochwassers im Juni 2013 sowie das für Nutzung auf mobilen Endgeräten optimierte Layout des Portals gezeigt.

\(^{20}\) https://www.hochwasserzentralen.de

Umsetzungsbeispiele

Abbildung 24: Übersicht der Pegel als Karte (links), Pegeldetails mit Ganglinie (mittig) und Favoritenliste (rechts) (Screenshots der App „Meine Pegel“)

In der Detailansicht ist es möglich, den gewählten Pegel in die eigene Favoritenliste zu übernehmen, die in der Abbildung rechts dargestellt ist. Weitere Funktionen, die in der Detailansicht verfügbar sind, werden in Abbildung 25 dargestellt, z.B. die Einrichtung einer Warnung bei einem bestimmten Pegelstand (links), die Einrichtung von Nachrichtenabonnements (Mitteilungen) zu bestimmten Pegeln (mittig) sowie der Empfang solcher Nachrichten (rechts).

Abbildung 25: Einrichtung einer Pegelwarnung (links), Einrichtung von Abonnement (mittig) und Eingang von Mitteilungen (rechts) (Screenshots der App „Meine Pegel“)

Hier ist es möglich, einen individuellen Grenzwert für den Pegel einzutragen, bei dessen Über- oder Unterschreitung eine Benachrichtigung auf das vorliegende mobile Endgerät erfolgt. Über die Hauptnavigation im unteren Bereich gelangt man zur Ansicht

Die App lässt sich auch mit sogenannten „wearable devices“ (Heise online 2017) wie einer Smartwatch koppeln. Unterstützt werden Android Wear und die Apple Watch. Hierbei erhält der Anwender die Benachrichtigung direkt auf die Uhr an seinem Handgelenk und kann bei Bedarf detaillierte Informationen auf dem Smartphone anschauen.


4.5.3 Technischer Rahmen zur App-Entwicklung


Um eine möglichst agile Entwicklung (Gründerszene Lexikon 2017) zu ermöglichen, ist der Buildprozess zu einem hohen Grad automatisiert (Abbildung 28). Zusätzlich sollen bei der Entwicklung neueste Sprachfeatures von JavaScript (ECMAScript 6, ES 6) eingesetzt werden können. Es wurde eine Buildpipeline entwickelt, in der verschiedene Build-Werkzeuge wie Gulp.js, Webpack und Bable.js zum Einsatz kommen (Abbildung 28 links). Konzeptionell ist die Buildpipeline in drei Schritte aufgeteilt. Im „Compile“-Schritt (links) wird der Quellcode aus JavaScript, SASS etc. kompiliert. Im „Bundle“-Schritt (mittig) werden die einzelnen Bestandteile zu einer vollständigen, im Browser lauffähigen, WebApp zusammengeführt. Im Schritt „Package“ (rechts) wird die WebApp dann schlussendlich über Cordova als native App für die verschiedenen Plattformen verpackt. Das Ergebnis sind binäre Pakete, die in die jeweiligen App Stores hochgeladen werden können.

Da die WebApp auf Webtechnologien basiert, ist es möglich, sie für die Nutzung per Webbrowser über das Internet bereitzustellen, was hauptsächlich zu Test- und Demozwecken dient. Die WebApp unterliegt allerdings Beschränkungen, da die nativen Funktionen der mobilen Endgeräte, wie z.B. die Kamera, im Browser nicht zur Verfügung stehen. Solche Features können erst in der nativen App über das Framework Cordova (Apache Cordova 2017) genutzt werden.

5 Evaluation und Diskussion


5.1 Anwendungsszenarien (Use-Cases)

Die folgenden Anwendungsszenarien stellen typische Suchanfragen an ein Umwelt- bzw. Energieportal dar. Die Nutzer stammen aus der interessierten Öffentlichkeit (Szenarien 2, 3, 5) bzw. haben eine Bezug zur beruflichen Tätigkeit des Nutzers (Szenarien 1, 4), jedoch ohne tiefen fachlichen Bezug (Vorwissen, Expertenwissen) zur durchsuchten Domäne.

5.1.1 Szenario 1 „Politiker“

Der Lokalpolitiker Ernst Maier interessiert sich für den Ausbau mit erneuerbarer Energie in den Bereichen Windkraft und Photovoltaik, insbesondere den aktuellen Bestand und die damit erzeugte Leistung im Vergleich seines eigenen Landkreises Zollernalb mit den umliegenden Landkreisen.

Beispiele für mögliche Suchanfragen:
- „windkraft zollernalbkreis vergleich“
- „photovoltaik zollernalbkreis vergleich“
- „erneuerbare energie zollernalb“.

Erwartete Suchergebnisse:
- Windkraftanlagen und deren installierte Leistung im Zollernalbkreis und den umliegenden Landkreisen (Tübingen, Reutlingen, Sigmaringen, Tuttlingen, Rottweil und Freudenstadt), z.B. in MWh je Landkreis.
- Photovoltaikanlagen und deren installierte Leistung im Zollernalbkreis und den umliegenden Landkreisen (wie oben).
- Anteil der erneuerbaren Energien innerhalb der eigenen Kommune oder des eigenen Landkreises.

5.1.2 Szenario 2 „Bauen“

Das junge Ehepaar Silke und Jens Schmidt möchte im Konversionsgelände Karlsruhe-Knieblingen einen Bauplatz kaufen, um darauf ein Haus zu bauen. Neben den Möglich-
keiten zum Einsatz erneuerbarer Energien und energieeffizientem Bauen interessieren sie sich für mögliche Umweltbelastungen am Standort, z.B. Altlasten, Lärm, Luftqualität.

Beispiele für mögliche Suchanfragen:
- „bauen in karlsruhe-knielingen“
- „energieeffizient bauen knielingen“
- „erneuerbare energie knielingen“
- „umwelt knielingen“
- „belastung konversionsgelände knielingen“.

Erwartete Suchergebnisse:
- Informationen über Baugebiete im Karlsruher Stadtteil Knielingen
- Informationen zum energieeffizienten Bauen (inkl. speziellen Informationen für das Baugebiet Karlsruhe-Knielingen)
- Informationen zu Möglichkeiten für den Einsatz erneuerbaren Energien (inkl. speziellen Informationen für das Baugebiet Karlsruhe-Knielingen)
- Beschreibung der Situation der Umwelt und möglicher Umweltbelastungen (Altlasten, Lärm, Luftqualität und weitere) rund um das Baugebiet Karlsruhe-Knielingen
- Weitere mögliche Standortfaktoren aus den Bereichen Umwelt und Energie.

5.1.3 Szenario 3 „Öko-Urlaub“
Die ökologisch und technisch sehr interessierte Familie Müller plant einen Urlaub im Nordschwarzwald. Bei Tagesausflügen möchten sie einige Beispiele für nachhaltige Energieerzeugung und -nutzung aufsuchen.

Beispiele für mögliche Suchanfragen:
- „ökologische energieerzeugung schwarzwald“
- „nachhaltige energie nordschwarzwald“
- „energiesparen schwarzwald“.

Erwartete Suchergebnisse:
- Fakten zur Nutzung erneuerbarer Energien im Schwarzwald, z.B. Konzepte und Projekte, z.B. Ökodörfer, Energieagenturen.
- Standorte mit Anlagen zur nachhaltigen Energieerzeugung (z.B. Windkraftanlagen, Solarparks, Wasserkraftanlagen, Stauseen, Biomasse) beschränkt auf den Nordschwarzwald
- Informationen zum Energiesparen im Schwarzwald, z.B. Möglichkeiten zur Gebäudedämmung, energieeffiziente Heizanlagen, Nutzung erneuerbarer Energien etc. sowie weiterführende Informationen, z.B. Energieagenturen im Schwarzwald
5.1.4 Szenario 4 „Solardächer“

Beispiele für mögliche Suchanfragen:
- „eignung solaranlage dach“
- „photovoltaik heilbronn“
- „solarenergie heilbronn schweinsbergstraße 12“
- „solar untergruppenbach habichthöhe 9“.

Erwartete Suchergebnisse:
- Allgemeine Informationen zu Solaranlagen und unterschiedlichen Typen von Dächern.
- Spezifische Informationen zur Eignung der Dachflächen von einzelnen Gebäuden in Heilbronn für die Bestückung mit thermischen Solaranlagen bzw. Photovoltaikanlagen.
- Spezifische Informationen zur Eignung der Dachflächen an einer spezifischen Adresse, z.B. in Heilbronn bzw. Untergruppenbach.

5.1.5 Szenario 5 „Ökostrom“
Der Student Kevin Sauber ist zur Aufnahme seines Bauingenieur-Studiums von Hamburg in die Karlsruher Oststadt gezogen. Für seine erste Studentenbude interessiert er sich für CO₂-neutralen Strom und entsprechende Tarife.

Beispiele für mögliche Suchanfragen:
- „karlsruhe oststadt ökostrom“
- „CO₂-neutrale strom karlsruhe“
- „ökotrom tarife karlsruhe oststadt“.

Erwartete Suchergebnisse:
- Grundlegende Informationen zu CO₂-neutralem Strom („Ökostrom“) in der Karlsruher Oststadt, Informationen zum Stromnetzbetreiber in der Karlsruher Oststadt
- Informationen zu Anbietern von Ökostrom-Tarifen (in Karlsruhe bzw. in der Karlsruher Oststadt)
- Konkrete Ökostrom-Tarife für den Standort Karlsruhe Oststadt.

5.2 Evaluation und Bewertung der ersten Architekturvariante
Die Kernbestandteile der Architektur sind:
- Nutzung von domänenspezifischen Vokabularen
  - Ergänzung der Volltextsuche um ein domänenspezifisches Wörterbuch
  - Automatische Suchworterweiterung (Vorschläge für Suchbegriffe)
- Anbindung externer Datenquellen über die Suchmaschine (OneBoxen)
- Darstellung von Suchergebnissen externer Datenquellen.

Die Evaluation erfolgt anhand der folgenden Systeme:

- Energieportal Baden-Württemberg; Volltextsuche; automatische Suchworterweiterung. Bis zu dessen Abschaltung im Einsatz.
- Landesumweltportale (LUPO); Domäne per Wörterbuch in der Volltextsuche. Seit 2007 bis heute im Einsatz.
- Landesumweltportale (LUPO); OneBoxen. Bis zur Ablösung durch serviceorientierte Architektur (gemäß dritter Architekturvariante) im Einsatz.
- Volltextsuche in „Meine Umwelt“. Seit 2007 bis heute im Einsatz.

Die Nutzungsszenarien sind nachfolgend zusammengestellt:

<table>
<thead>
<tr>
<th>Szenario</th>
<th>Suchbegriffe</th>
<th>Volltext</th>
<th>Karte</th>
<th>Objekte</th>
<th>Bemerkung/Ziel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Politiker 1</td>
<td>windkraft zollernalb-kreis vergleich</td>
<td>✓</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Politiker 2</td>
<td>photovoltaik zollernalb-kreis vergleich</td>
<td>✓</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Politiker 3</td>
<td>erneuerbare energie zollernalb</td>
<td>✓</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Bauen 1</td>
<td>bauen in karlsruhe-knielingen</td>
<td>✓</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Bauen 2</td>
<td>energieeffizient bauen knielingen</td>
<td>✓</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Bauen 3</td>
<td>erneuerbare energie knielingen</td>
<td>✓</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Bauen 4</td>
<td>umwelt knielingen</td>
<td>✓</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Bauen 5</td>
<td>belastung konversionsgelände knielingen</td>
<td>✓</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Öko-Urlaub 1</td>
<td>ökologische energieerzeugung schwarzwald</td>
<td>✓</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Öko-Urlaub 2</td>
<td>nachhaltige energie nordschwarzwald</td>
<td>×</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Öko-Urlaub 3</td>
<td>energiesparen schwarzwald</td>
<td>✓</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
</tbody>
</table>

Quelle: Abfragen im Energieportal Baden-Württemberg

In einigen Fällen werden keine passenden Ergebnisse gefunden:

- Ortssuche zu spezifisch (Solardächer 3+4, Ökostrom 3), z.B. Straßennamen, die im Volltext nicht verfügbar sind
- Thema ist nicht in den Datenquellen (hier Volltextsuche) enthalten: Ökostrom 1, 2


### Bewertung und Erkenntnisse
Die Verbesserung beschränkt sich im Energieportal auf die Volltextsuche – hier allerdings ohne erkennbaren Mehrwert für die betrachteten Szenarien. Im Umweltportal gibt es dagegen eine ganze Reihe von Verbesserungen im Bereich der Abbildung von Umgangssprache auf Fachsprache, z.B. „Müll“ suchen, Informationen zum Thema „Abfall“ finden.


Die Anzeige von Treffern beschränkt sich auf die Volltextsuche und OneBoxen. Es werden keine Karten und keine (Energie-)Objekte gefunden, da keine Datenquellen zum Thema Energie per OneBox (Google Inc. 2015) an die Suchmaschine angebunden sind. Trigger für OneBoxen können nur die verwendeten Suchbegriffe sein, nicht jedoch z.B. explizite Geokoordinaten oder Datum/Zeitangaben.

Der Anschluss von Zielsystemen erfolgt (im Falle des Umweltportals) jeweils über Adapter, da OneBoxen keine standardisierte Schnittstelle darstellen. Der direkte Anschluss von langsamsten Zielsystemen als OneBox funktioniert wegen des vorgegebenen Timeouts nur über Caching, Indexbildung oder ähnlichen Techniken, d.h. durch redundante Datenbereitstellung.

Die Verbesserungen durch die erste Architekturvariante sind (erwartungsgemäß) beschränkt.

Dennoch lassen sich einige wertvolle Erkenntnisse daraus ziehen:

- Domänenspezifische Vokabulare können als Erweiterung des Wörterbuchs von suchmaschineneigenen Verbesserungen der Suche (Synonymketten), z.B. zur Auflösung von umgangssprachlichen Begriffen dienen.
- Suchbegriffe alleine reichen für die sinnvolle Anfrage vieler Zielsysteme nicht aus. Es ist notwendig, die Semantik der Suchbegriffe zu verstehen, d.h. sie einem Wortgut (Vokabular) zuzuordnen, oder sie anderweitig, z.B. als Ortsbegriffe, zu klasssifizieren. Zusätzlich ist ggf. eine Anreicherung bzw. Explizierung der Suchbegriffe notwendig, z.B. um aus Ortsnamen Geokoordinaten zu gewinnen.
- Zielsysteme müssen passende Schnittstellen bereitstellen. Die Daten in den Zielsystemen müssen semantisch beschrieben werden, minimal als Abbildung ihrer Konzepte auf eines der verwendeten Vokabulare.
5.3 Evaluation und Bewertung der zweiten Architekturv...
<table>
<thead>
<tr>
<th>Bauen 4</th>
<th>umwelt knielingen</th>
<th>✓</th>
<th>✓</th>
<th>✓</th>
<th>Viele Karten, z.B. Schutzgebiete</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bauen 5</td>
<td>belastung konversionsgelände knielingen</td>
<td>✓</td>
<td>×</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Öko-Urlaub 1</td>
<td>ökologische energieerzeugung schwarzwald</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Öko-Urlaub 2</td>
<td>nachhaltige energienordschwarzwald</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Öko-Urlaub 3</td>
<td>energiesparen schwarzwald</td>
<td>✓</td>
<td>×</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Solardächer 1</td>
<td>eignung solaranlage dach</td>
<td>✓</td>
<td>✓</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Solardächer 2</td>
<td>photovoltaik heilbronn</td>
<td>✓</td>
<td>✓</td>
<td>×</td>
<td>Mapping Adresse fehlt in Daten</td>
</tr>
<tr>
<td>Solardächer 3</td>
<td>solarenergie heilbronn schwarzwald</td>
<td>×</td>
<td>✓</td>
<td>×</td>
<td>Mapping Adresse fehlt in Daten</td>
</tr>
<tr>
<td>Solardächer 4</td>
<td>solar untergruppenbach habichhöhe 9</td>
<td>×</td>
<td>✓</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Ökostrom 1</td>
<td>karlsruhe oststadt ökostrom</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>Fehlende Daten</td>
</tr>
<tr>
<td>Ökostrom 2</td>
<td>co2-neutrale Strom karlsruhe</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>Fehlende Daten</td>
</tr>
<tr>
<td>Ökostrom 3</td>
<td>ökotrom tarife karlsruhe degenfeldstraße</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>Fehlende Daten</td>
</tr>
</tbody>
</table>

**Volltext**
✓ Gewünschte Information unter den ersten 20 Treffern.
× Gewünschte Information nicht unter den ersten 20 Treffern.

**Karte**
✓ Gewünschte Information auf Kartenansicht enthalten.
✓ Gewünschte Information auf Kartenansicht enthalten, Ortsauswahl korrekt
× Gewünschte Information nicht auf Kartenansicht enthalten.

* = nicht anwendbar

Die Volltextsuche (spalte „Volltext“) liefert praktisch die gleichen Ergebnisse wie in der ersten Architekturvariante, sie basiert auf derselben Suchmaschine unter Verwendung des erweiterten Wörterbuchs, in dem jedoch nicht alle Inhalte des Ontologiesystems (Lebenslagen, Artikulationsontologie) verfügbar sind.


**Bewertung und Erkenntnisse**


Das Einfügen konkreter Objekte aus den Zielsystemen in die Ontologie und die Verknüpfung der Objekte mit Konzepten aus den Domänenontologien haben sich nicht bewährt, da die verwendeten Ontologiesysteme mit einer großen Zahl enthaltener Instanzen erheblich an Performance (inakzeptable Antwortzeiten) einbüßen. Daneben
muss dauerhaft die Konsistenz zwischen den Daten in den Zielsystemen und denen in der Ontologie sichergestellt werden. Daher, und da in der zweiten Architekturvariante die Daten grundsätzlich aus den Zielsystemen abgerufen werden sollten, wurde die Integration der Daten in die Ontologie nicht weiter verfolgt.


Das grundsätzliche Funktionieren der Zielsystembeschreibungen für den parametrisierten Zugriff auf Zielsysteme demonstriert in der Praxis neben dem Prototypen der SUI-Suche die mobile App „Meine Umwelt“ (Schlachter et al. 2013), innerhalb der alle enthaltenen Datenquellen über erweiterte OpenSearch-Descriptions beschrieben werden.

Der SearchBroker als koordinierendes und prozessierendes Element zwischen Suchanfrage, deren semantische Erkennung und Anreicherung und der Anfrage von Zielsystemen funktioniert ebenfalls grundsätzlich.

Das prototypische System zur semantischen Suche innerhalb des Umweltportals Baden-Württemberg zeigt insgesamt zwei erhebliche Einschränkungen:

1. Die serverseitige Abfrage von Zielsystemen bildet einen Flaschenhals. Es muss auf „langsamen“ Zielsysteme, die teilweise erst mit erheblichen Latenzen von mehreren Sekunden antworten, warten. Das bedeutet, dass auch die Antwortzeit der Suche entweder durch Timeouts in den Anfragen an die Zielsysteme begrenzt – dafür jedoch auf mögliche Ergebnisse verzichtet – werden muss, oder dass die Suche immer so lange läuft, bis alle Zielsysteme Daten (oder zumindest eine ggf. negative Antwort) geliefert haben, was zu erheblichen, inakzeptablen Wartezeiten\(^{21}\) für den Nutzer führt.

2. Viele mögliche Zielsysteme bieten keine Schnittstellen, die sich zur maschinellen Abfrage eignen, sondern lediglich Weboberflächen für menschliche Nutzer. Sie sind als Zielsysteme für die zweite Architekturvariante ungeeignet. Da auch

\(^{21}\) Für die Landesumweltportale wurde festgelegt, dass die Suchmaschine nach spätestens 1000ms ein Ergebnis ausliefern muss. Experimente von Google (Brutlag 2009) und Microsoft/Bing (Schurman und Brutlag 2009) zeigten, dass bei Internet-Suchmaschinen bereits Wartezeiten von einigen 100ms zu einer messbaren Abnahme der Anzahl der Suchen je Tag führen.

Wenn entscheidende Systeme einen derart massiven Einfluss auf die Gesamtarchitektur haben können, muss das Paradigma, Daten direkt, d.h. zur Laufzeit der Anfrage, aus den Zielsystemen abzurufen, grundsätzlich infrage gestellt werden. Wenn der direkte Zugriff auf die Zielsysteme, z.B. mangels technischer Schnittstellen oder aus Gründen der Performanz, also nicht möglich ist, sollten die Daten über redundante Systeme bereitgestellt werden, die entsprechende Schnittstellen, Verfügbarkeit und Performanz bieten.


- Ontologiesysteme sind zur Darstellung und Verknüpfung mehrere Vokabulare mithilfe einer Artikulationsontologie grundsätzlich geeignet.
- Die Pflege der (Teil-)Ontologien und Bereitstellung der Artikulationsontologie macht Aufwand; das Hinzufügen weiterer Teilontologien vermehrt den Aufwand
sogar überproportional stark. Daher sollte hier soweit wie möglich automatisiert werden.

- Die rein serverseitige Implementierung der Suche funktioniert insbesondere wegen der synchronen Anfragen der Zielsysteme nicht ausreichend performant, auch wenn die einzelnen Teilkomponenten der Suche im Grundsatz funktionieren, sowohl einzeln als auch im Zusammenspiel.

Mögliche Ansatzpunkte hierfür sind die Verlagerung der Kernfunktionalitäten (semantische Verarbeitung der Suchanfrage, Koordination durch den Search-Broker, Anfrage der Zielsysteme, Aufbereitung der Trefferansicht (Mashup)) in den Client, insbesondere durch Nutzung asynchroner Aufrufe von Hintergrunddiensten und Zielsystemen.


### 5.4 Evaluation und Bewertung der dritten Architekturvariante

Die Kernbestandteile der Architektur sind:

- (Redundante) Bereitstellung von Daten über eine Reihe generischer Daten- dienste (Webcache)
- Transformation und Synchronisation der Daten zwischen Zielsystemen und Webcache mithilfe von definierten Prozessen (Data Ingestion)
- Nutzung von Vokabularen zur semantischen Beschreibung der Daten (bzw. von Klassen); Beschreibung der Daten durch Schemata
- Prozessierung und Koordination der Suchanfrage und der Anfrage von Zielsystemen im Client mithilfe spezialisierter Gazetteer-Dienste
- Ergebnispräsentation durch eine Reihe generischer Frontend-Komponenten
- Orchestrierung der Suche im Client und der Ergebnispräsentation durch Event-basierte Kommunikation zwischen den Frontend-Komponenten.

Die Evaluation erfolgt anhand der folgenden Systeme:

- Energieatlas Baden-Württemberg.
Als Nutzungsszenarien werden erneut die zuvor betrachteten verwendet:

Quelle: Umweltportal Baden-Württemberg 2015, Liferay-basierte Version

<table>
<thead>
<tr>
<th>Szenario</th>
<th>Suchbegriffe</th>
<th>Volltext</th>
<th>Karte</th>
<th>Objekte</th>
<th>Bemerkung/Ziel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Politiker 1</td>
<td>windkraft zollernalb-kreis vergleich</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>Kein Vergleich</td>
</tr>
<tr>
<td>Politiker 2</td>
<td>photovoltaik zollernalb-kreis vergleich</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>Kein Vergleich</td>
</tr>
<tr>
<td>Politiker 3</td>
<td>erneuerbare energie zollernalb</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Bauen 1</td>
<td>bauen in karlsruhe-knieelingen</td>
<td>✓</td>
<td>✓</td>
<td>× (✓)</td>
<td>Lebenslagen („Bauen“) nicht instrumentiert</td>
</tr>
<tr>
<td>Bauen 2</td>
<td>energieeffizient bauen knieelingen</td>
<td>✓</td>
<td>✓</td>
<td>× (✓)</td>
<td>Lebenslagen („Bauen“) nicht instrumentiert</td>
</tr>
<tr>
<td>Bauen 3</td>
<td>erneuerbare energie knieelingen</td>
<td>✓</td>
<td>✓</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Bauen 4</td>
<td>umwelt knieelingen</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>Viele Karten, z.B. Schutzgebiete</td>
</tr>
<tr>
<td>Bauen 5</td>
<td>belastung konversionsgelände knieelingen</td>
<td>✓</td>
<td>×</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Öko-Urlaub 1</td>
<td>ökologische energieerzeugung schwarzwald</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Öko-Urlaub 2</td>
<td>nachhaltige energie nordschwarzwald</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Öko-Urlaub 3</td>
<td>energiesparen schwarzwald</td>
<td>✓</td>
<td>×</td>
<td>✓ (✓)</td>
<td>Lebenslage</td>
</tr>
<tr>
<td>Solardächer 1</td>
<td>eignung solaranlage dach</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Solardächer 2</td>
<td>photovoltaik heilbronn</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Solardächer 3</td>
<td>solarenergie heilbronn schwinsbergstraße 12</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
<td>Mapping der Adresse fehlt in den Daten</td>
</tr>
<tr>
<td>Solardächer 4</td>
<td>solar untergruppen-bach habichtöhöhe 9</td>
<td>×</td>
<td>✓</td>
<td>×</td>
<td>Mapping der Adresse fehlt in den Daten</td>
</tr>
</tbody>
</table>

Im Szenario „Bauen“ sind die im Umweltportal gelieferten Suchergebnisse tatsächlich sogar schlechter als in Architekturvariante zwei, in der die Lebenslage „Bauen“ in die Ontologie integriert ist, und daher eine ganze Reihe von konkreten Themen in die Suche einbezieht. In der hier verwendeten Version des Umweltportals sind alle thematischen Bezüge ebenfalls über Konzepte aus der Domänenontologie beschrieben. Aus Performanzgründen wird die thematische Zugehörigkeit von Datensätzen zu Konzepten der Domäne bereits beim Transferieren der Daten in den Webcache festgelegt und dort gespeichert, teilweise durch Reduzierung der Konzepte auf ihre Namen (Labels), um sie besser indexieren zu können. Dabei kommt zwar dasselbe Ontologiesystem wie
in Architekturvariante zwei zum Einsatz, d.h. die berechnete „Umgebung“ eines Konzeptes (z.B. „Naturschutzgebiet“) erfolgt nach demselben Prinzip wie sie die dortige thematische Zuordnung bei der Verarbeitung der Suchanfrage liefert, allerdings wurde die Ontologie wegen des Pflegeaufwands gegenüber dem prototypischen SUI-System auf die Inhalte des GEMET reduziert, weshalb in der Suche des produktiven Systems keine Lebenslagen verfügbar sind.

**Bewertung und Erkenntnisse**

Der Webcache mit seinen (Micro-)Services ist sehr gut geeignet, um den Zugriff auf Daten und Objekte von ihren jeweiligen Zielsystemen zu entkoppeln und generell die Verfügbarkeit von Daten zu gewährleisten, weitgehend ohne die Zielsysteme zu belas-
ten. Das Aufsetzen und der Betrieb des Webcache sowie die Synchronisation der Da-
ten aus den Zielsystemen mit dem Webcache bedeuten zwar zusätzlichen Aufwand, der sich nach der ersten Einrichtung jedoch sehr gut automatisieren lässt.

Das modulare Konzept von auf bestimmte Datentypen spezialisierten, dennoch generi-
schen Datendiensten hat sich bewährt, die acht Haupt- und zwei Hilfsdienste des Microservice-Backends decken den Bedarf für alle im Energie- und Umweltbereich anfallenden Datentypen wie Stammdaten, Zeitreihen, Geodaten, digitalen Assets etc. ab. Die einzelnen Dienste sind durch ihre Konzeption und Implementierung als Micro-
services relativ unkompliziert und bieten einfache REST-Schnittstellen zur Nutzung, die über Versionierung dauerhaft stabil gehalten werden können. Durch Verwendung ska-
lbarer Backend-Systeme und Laufzeitinfrastrukturen können sie ausreichend Leis-
tung für unterschiedliche Nutzungsszenarien bieten. Die Nutzung von Containervirtua-
lisierung mittels Docker bietet Flexibilität beim Betrieb und ebenfalls Vorteile bei der Skalierung in entsprechenden Infrastrukturen wie Kubernetes (Kubernetes 2016).

Da der Webcache wenige, stabile Schnittstellen anbietet, reduziert sich der Aufwand für die Implementierung Schnittstellen im Client erheblich, was die Implementierung von Frontend-Komponenten wesentlich vereinfacht – die Schnittstellen zu den Zielsys-
temen müssen allerdings an anderer Stelle, bei der Datentransformation, implementiert werden.

Das Sucherlebnis für die Nutzer unterscheidet sich gegenüber der Architekturvariante zwei erheblich. Die Suchseite besteht aus einem Zusammenspiel verschiedener Kom-
ponenten im Client, die über eine Event-basierte Kommunikationsschicht miteinander verbunden sind und darüber orchestriert werden. Die einzelnen Komponenten werden so mit der rohen Suchanfrage, mit ihrer semantisch verarbeiteten Form sowie den Er-
gebnissen der verschiedenen Zusatzdienste (Gazetteers) versorgt. Sie können so au-
tark den Webcache anfragen und die gefunden Daten zur Anzeige bringen. Die Kom-
munikation mit den Hintergrunddiensten geschieht asynchron, so dass der Nutzer nicht mehr warten muss, bis das Gesamtergebnis vorliegt, sondern meist sehr schnell mit ersten Ergebnissen, z.B. der Trefferliste der Volltextsuchmaschine, versorgt wird, wäh-
rend aufwändigere Komponenten, wie z.B. die Kartenansicht, erst nach und nach die vollständigen Ergebnisse, z.B. die zum Thema passenden Kartenlayer, anzeigen.
Die per REST-URLs verfügbaren Daten bieten bereits eine gute Grundlage für die Nutzung als Linked Data im Sinne des Semantic Web, jedoch fehlen den Diensten derzeit noch die dafür notwendigen standardisierten Datenformate, z.B. RDF. Zusätzlich sollen in den Zielsystemen nicht enthaltene Beziehungen zwischen den Daten darstellbar sein, die jedoch erst erzeugt werden müssen, z.B. unter Ausnutzung von Zusatzwissen. Der in der Architektur beschriebene Link-Service wird in der Praxis der Umweltportale noch nicht genutzt, unter anderem, da für die Beschreibung und Konfiguration von Verknüpfungen und den notwendigen Implementierungen bei den Transformationen Aufwände anfallen, die beim erstmaligen Aufbau des Webcache nach der dritten Architekturvariante zunächst nicht leistbar waren.

5.5 Diskussion der vierten Architekturvariante

Die Kernbestandteile der Architektur sind:

- Zusätzlicher Link-Service
- Erweiterung der Datentransformation in den Webcache um die Erzeugung zusätzlicher Verknüpfungen
- Erweiterung der Anzeigekomponenten um die Darstellung zusätzlicher, verknüpfter Informationen.

Derzeit erfolgte noch keine Umsetzung in konkreten Systemen. Eine Evaluation bezüglich der konkreten Nutzungsszenarien hat daher noch nicht stattgefunden.

Diskussion


auf den verschiedenen Verwaltungsebenen aggregieren, ohne dabei erneut Geooperationen ausführen zu müssen. Für die prototypische Umsetzung wurde für die Darstellung der Beziehungen kein Triplestore, sondern eine neo4j-Graphdatenbank (neo4j 2016) verwendet, die allerdings noch nicht an den eigentlichen Linkservice angebunden wurde. Sie demonstriert jedoch eindrucksvoll das Potenzial der generierten Verknüpfungen, da Attribute für die verschiedenen Verwaltungsebenen aggregiert werden können. Die in der Graphdatenbank vorhandenen Verwaltungsstrukturen lassen sich so mit beliebigen Datensätzen verknüpfen und in analoger Weise nutzen.


Da zur Adressierung von Datensätzen die URLs der Datendienste verwendet werden, bietet das Gesamtszenario das Potenzial für die Verwendung als Linked Data. Dazu müssen die Datendienste lediglich die hierfür notwendigen Datenformate, d.h. RDF bzw. RDF/JSON, generieren, was für den Stammdatendienst bereits prototypisch erfolgt ist und damit die Erfüllung aller vier Kriterien für Linked Data:

1. Verwendung von URIs zum Benennen von Objekten
2. Verwendung von HTTP-URIs zum tatsächlichen Auffinden von Objekten
3. Verwendung von Standards wie RDF und SPARQL
4. Bereitstellung von Links in Form von URLs, um Verbindungen zu weiteren Objekten finden zu können.

Eine praktische Erweiterung ist die Möglichkeit zur erweiterten Abfrage von Daten aus allen Diensten, d.h. Anfragen, die alle oder bestimmte Verknüpfungen eines Objektes mit anderen Objekten enthält, auch wenn sie nicht im konkreten Datendienst, sondern nur im Link-Service verfügbar sind. Hierfür steht der Messaging-Kanal der Microservice-basierten Architektur zur Verfügung. Hierüber kann der Datendienst sich die notwendigen Informationen vom Link-Service holen und direkt in die Antwort einfügen. Die entsprechende Funktion muss jedoch noch umgesetzt werden.
Zusammenfassung

Das Ziel der vorliegenden Dissertationsschrift bestand darin, ein neues Konzept für die semantische Suche in heterogenen Informationssystemen zu Fragestellungen aus Umwelt und Energie zu entwickeln, d.h. die Konzeption und Entwicklung einer Suchfunktion für Webportale, die zwar für den Nutzer so einfach wie herkömmliche Internet-Suchmaschinen funktioniert, jedoch qualitativ bessere, ggf. mehr Ergebnisse liefert als eine konventionelle Volltextsuche. Dabei sollten folgende wissenschaftlichen Teilziele erreicht werden:

- Erkennen der Semantik einer Suchanfrage innerhalb einer gegebenen Domäne
- Beschreibung der Semantik von Daten gegebener Informationssysteme bezüglich einer vorgegeben Domäne, die sich ggf. aus mehreren Vokabularen zusammensetzt:
  - Mapping der Vokabulare untereinander (Artikulation) bzw. Harmonisierung
  - Abbildung der Daten auf die Vokabulare und ggf. Nutzung der zugehörigen Schemata
  - Nutzung von Zusatzwissen zu Orts- und Zeitbezug, auch als generische Zusammenhänge zwischen Daten
  - Harmonisierung der Darstellung von Daten aus verschiedenen Informationssystemen
  - Nutzung der in den Vokabularen enthaltenen Strukturen, z.B. zur Weiternavigation, Gruppierung, Facettierung etc.
- Beschreibung und Realisierung des technischen Zugriffs auf heterogene Informationssysteme (Datentypen, Schnittstellen und Formate)
- Nutzung generischer Komponenten zur Präsentation von Daten innerhalb eines Webportals
- Präsentation/Darstellung der Suchergebnisse in einer integrierten Trefferansicht
  - Koordination zwischen bzw. Orchestrierung von Komponenten
  - Möglichkeit zur Kommunikation zwischen Komponenten.

In Kapitel 2 wurde ausgehend von der gegebenen Zielstellung eine Grundarchitektur entwickelt, welche den Aufbau einer semantischen Suchfunktion auf einer allgemeinen, von technischen Randbedingungen unabhängigen Basis beschreibt.

Kapitel 3 beschreibt vier Varianten der Grundarchitektur. Ein erster Ansatz erweiterter eine bestehende konventionelle Volltextsuche um Fachvokabular in Form eines Umweltthesaurus sowie die Möglichkeit zum Anschluss mehrerer Zielsysteme.

Die zweite Architekturvariante setzt auf die Vorverarbeitung der Suchanfrage sowie die Beschreibung und den Anschluss von Zielsystemen mittels Zielsystembeschreibungen über einen SearchBroker. Vokabulare werden durch ein Ontologiesystem verwaltet und über eine Artikulationsontologie miteinander verknüpft. Die Generierung der Ergebnis-


Die Gegenüberstellung und Diskussion der verschiedenen Architekturvarianten findet in Kapitel 5 statt.

Die wesentlichen Ergebnisse der Arbeit sind:

1. Entwicklung eines neuen Konzeptes zur semantischen Suche in heterogenen Informationssystemen zu Fragestellungen aus den Bereichen „Umwelt“ und „Energie“.
2. Herleitung einer allgemeinen (generischen) Architektur, aus der sich vier verschiedene Architekturvarianten ableiten lassen.
3. Ableitung der ersten Architekturvariante durch Erweiterung der bestehenden Volltextsuche um domänenspezifisches Fachvokabular sowie um Möglichkeiten zum Zugriff auf weitere Zielsysteme auf Basis der OneBox-Schnittstelle.
5. Ableitung einer dritten Architekturvariante, die Daten aus den Zielsystemen redundant über eine serviceorientierte Architektur („Webcache“) bereitstellt. Ein Transformationsprozess (Data Ingestion) sorgt für die Aufbereitung und seman-
Zusammenfassung

135


10. Bereitstellung einer leistungsfähigen semantischen Suche auf Basis der dritten Architekturvariante im Bereich des Umweltinformationssystems Baden-Württemberg als Beispiel für die Suche in heterogenen Informationssystemen innerhalb der Domänen „Umwelt“ und „Energie“.


Das vorliegende Konzept bietet Spielraum für künftige Erweiterungen, zunächst insbesondere für die vollständige Umsetzung der Erweiterungen aus der vierten Architekturvariante.

Darüber hinaus gibt es jedoch Potenzial für weitere Verbesserungen:


Mit der eindeutigen Adressierbarkeit der Daten über eindeutige RESTful URLs ist eine wesentliche Voraussetzung für die Verwendung einer großen Menge von Technologien des Semantic Web gegeben. Um eine globale Verknüpfbarkeit und Nutzbarkeit der Daten zu gewährleisten, ist es jedoch notwendig, die Daten auf global gültige Schemata abzubilden. Auch wenn es Kritik bezüglich der Nutzung von Ontologien und XML gibt (Shirky 2005; Swartz 2013), so gibt es leichtgewichtige bzw. pragmatische Ansätze (Mikroformate, Microdata), die jedoch ebenso die gegenseitige Nutzung von Daten und
damit die Interoperabilität von Anwendungen zum Ziel haben und bereits von kommerziell erfolgreichen Suchmaschinen unterstützt/genutzt werden.

Aus der entwickelten Architektur lassen sich nicht zuletzt Empfehlungen und Vorschläge für das Aufsetzen künftiger Informationssysteme ableiten, um eine möglichst einfache Integrierbarkeit von deren Inhalte in übergreifende Informationssysteme (Webportale) zu gewährleisten sowie eine Interoperabilität mit dritten Anwendungen bieten zu können.
Anhang: Grundlagen

A1 Das Semantic Web

Eine Vision des Semantic Web (Berners-Lee und Fischetti 1999; Berners-Lee et al. 2001) ist die Bereitstellung von Informationen in einer Art und Weise, die es ermöglicht, Informationen maschinell zu verarbeiten, zu kombinieren und Fragen durch entsprechende Schlussfolgerungsmechanismen zu beantworten bzw. mit deren Hilfe sogar weitere Informationen zu generieren.

Hierzu gibt es bereits eine ganze Reihe etablierter Technologien. Ihre gemeinsame Grundlage sind

- Linked Data,
- Vokabulare,
- Abfragen (Queries) und
- Inferenzen (Schlussfolgerungen).

A1.1 Linked Data

Berners-Lee schlug in seinen Überlegungen zu einem Semantischen Web (Semantic Web) (Berners-Lee und Fischetti 1999) das Konzept von Linked Data vor (Berners-Lee 2006). Danach sollen alle Objekte über eindeutige Bezeichner (URIs) repräsentiert werden. Solche eindeutig bezeichneten Objekte werden Ressourcen genannt. Für das Semantic Web wird die Verwendung von HTTP-URIs als Bezeichnerformat vorgeschlagen, so dass sich Informationen zu einem Objekt auch tatsächlich nachschlagen lassen, d.h. die HTTP-URI adressiert einen Server, der Informationen zu einem Objekt in einem standardisierten Format liefert. Die Informationen enthalten im Regelfall weitere URIs, die beispielsweise Beziehungen des Objekts zu anderen Ressourcen, d.h. weiteren Objekten, die mit den gelieferten URIs bezeichnet sind, ausdrücken.

Beziehungen zwischen Objekten werden im Semantischen Web als Tripel dargestellt. Darin werden ein Subjekt und ein Objekt bezüglich eines Prädikates miteinander verknüpft.

Versteht man die Objekte als Knoten und die Prädikate als Kanten eines Graphen, so spannen die Beziehungen zwischen allen möglichen Objekten einen gigantischen Graphen auf, der im Semantischen Web auch als „Giant Global Graph“ (Berners-Lee 2007) bezeichnet wird.

Zur Definition von Ressourcen und für die Darstellung von Beziehungen zwischen Objekten gibt es eine Reihe von standardisierten Formaten, insbesondere das Ressource Description Format (RDF) sowie Varianten davon (RDFa, RDF/XML), die z.B. eine Einbettung von semantischen Informationen als Linked Data innerhalb von für den
menschlichen Nutzer bestimmten HTML-Dokumenten ermöglichen. Das Semantische Web muss also nicht notwendigerweise parallel zum herkömmlichen Web entstehen, sondern kann sich damit gewissermaßen überschneiden.

### A1.2 Vokabulare


Im Semantic Web werden Vokabulare in Form von Ontologien beschrieben, die eine explizite Spezifikation einer Konzeptionalisierung darstellen.

### A1.3 Abfragen (Queries)

A1.4 Inferenzen (Schlussfolgerung)

Wenn Daten auf Basis von Vokabularen und Linked Data dargestellt werden, können aus ihnen und ihren Beziehungen automatisiert neue Beziehungen gewonnen werden (Semantic Reasoning). Dazu benötigt man Zusatzinformationen, z.B. erweiterte Vokabulare (Ontologie-Sprachen) oder spezielle Regeln (Beschreibungslogik). Die meisten Inferenzmaschinen (Semantic Reasoner) nutzen die Prädikatenlogik erster Stufe (Sowa 2014).


A2 Datentypen und der Strukturierungsgrad von Daten


\[
\text{<title>Schnelllaufzahl eines Dreiblattrotors</title>}
\]

für den Titel eines Dokuments. Schwach strukturiert bedeutet, dass die Auszeichnung von Inhalten einer typischen Dokumentstruktur (Titel, Überschriften, Absätze) entspricht, nicht jedoch spezifischen Attributen aus der semantischen Domäne des Dokuments\(^{22}\).

In vielen Dokumenten fehlt sogar die schwache Auszeichnung von Inhalten, man spricht dann von unstrukturierten Dokumenten bzw. unstrukturierten Daten. Das lässt sich in HTML-Dokumenten beispielsweise bei der verbreiteten exzessiven Verwendung von `<DIV>` oder `<SPAN>`-Tags beobachten, die zwar eine hierarchische Strukturierung von Dokumenten erlauben, denen jedoch keine Bedeutung zugeordnet ist.

\(^{22}\) Es gibt Mechanismen mit deren Hilfe auch Markup-Dokumenten (XML) eine Struktur und/oder Semantik zugeordnet werden kann, z.B. Dokumenttyp-Definitionen (DTD), XML-Schema oder über Microtagging-Mechanismen (schema.org 2016b, 2016c)
In vielen Fällen werden (stark) strukturierte Daten in weniger strukturierte Repräsentationen überführt, z.B. wenn eine (strukturierte) Datenbank als Basis für die Erzeugung von Webseiten (HTML-Dokumente) dient. Suchmaschinen können häufig nur auf die schwächer strukturierte Repräsentation zugreifen. Da eine Rücktransformation in die strukturierte Form im Allgemeinen nicht möglich ist (Hänsch 2014) steht den Suchmaschinen daher sehr häufig nicht die volle Semantik von Daten zur Verfügung.

Der Grad der Strukturierung von Daten hängt einerseits von der Ausprägung ihrer inhärenten Struktur, zur technischen Nutzung jedoch ebenfalls vom Grad der Standardisierung ihrer Strukturen, ab (Abbildung 29).

Abbildung 29: Strukturierung versus Standardisierung der Datenschemata; nach (Holzinger 2014)

Große Teile der im Internet verfügbaren Inhalte sind schwach strukturiert. Sie liegen zwar in standardisierten Datenformaten (meist HTML oder PDF) vor (Glöggler 2003), welche jedoch meist nicht die vollständige Semantik und Strukturierung der Informationen zum Ausdruck bringen, sondern sich häufig auf die allgemeine Struktur von Dokumenten (Titel, Überschriften, Absätze) beziehen, nicht jedoch auf den Inhalt. Teilweise werden schwach strukturierte Dokumente durch Metadaten ergänzt, die sie zu semi-strukturierten Daten machen können.

In den meisten verbreiteten Datenbanksystemen, insbesondere bei relationalen Datenbanken, muss im Datenbanksystem die Struktur der Daten durch Spezifikation von Schemas festgelegt werden. Die enthaltenen Daten sind hierdurch per se strukturiert. Die verwendeten Schemata können jedoch mehr oder weniger standardisiert sein, was wiederum den Datenaustausch mit anderen Systemen beeinflusst.

Sowohl stark strukturiert als auch standardisiert sind die Formate des Semantic Web wie RDF oder OWL.

A2.1 Grundlagen für die maschinelle Verarbeitung von Daten

Um die maschinelle (Weiter-)Verarbeitung der im Internet (WWW) vorhandenen Informationen zu ermöglichen, ist es notwendig, dass Maschinen den vorhandenen, heute häufig noch von Menschen zusammengetragenen Informationen deren Bedeutung (Semantik) eindeutig zuordnen können. Im Hinblick auf ein Internet der Dinge (Internet of Things) (Uckelmann et al. 2011b; Fleisch und Mattern 2005; Weber und Weber 2010) und den Einsatz von Computern in allen Bereichen des Lebens (Ubiquitous computing = „Rechnerallgegenwart“) (Greenfield 2006; Fleisch und Mattern 2005) ist es notwendig, menschliche Eingriffe (Interaktion) zu reduzieren oder sogar gänzlich überflüssig zu machen (Uckelmann et al. 2011a), d.h. eine Automatisierung durchzuführen.

Heute stehen Informationen in einer Vielzahl verschiedener elektronischer Systeme zur Verfügung. Informationssysteme stehen dabei in einem Beziehungsgefüge, das häufig als MAT-System (Mensch/Aufgabe/Technik-System), einem Dreieck zwischen Mensch, Aufgabe und (Informations-)Technik, bezeichnet wird. Die Ecken des Dreiecks betonen dabei verschiedene Schwerpunkte, die bei der Erstellung eines Informationssystems gesetzt werden können. Eine Verschiebung des Schwerpunktes in Richtung einer Ecke hat Einfluss auf die Betonung bzw. die Anforderungen an die jeweils anderen, z.B. bedeuten hohe Anforderungen im Bereich der Benutzerbarkeit eines Informationssystem durch den Endnutzer meist auch erhöhte Anforderungen bei der Definition und Ausgestaltung der Aufgabe (funktionale Anforderungen) sowie bei der Auswahl der verwendeten Technologie.

Umgekehrt wird bei Informationssystemen, die sich am menschlichen Nutzer orientieren, häufig kein oder wenig Wert auf die maschinelle Interpretierbarkeit und Weiterverarbeitbarkeit von Informationen gelegt, da die notwendige Interpretation als (i.A. leistungsfähiger) kognitiver Prozess beim Nutzer abläuft. Wieder umgekehrt, stellt eine einfache maschinelle Verarbeitbarkeit von Daten hohe Anforderungen an die Daten bzw. deren Strukturierung.

Der Grad der Strukturierung von Daten in Informationssystemen ist in der Realität häufig vom Anwendungsfall und der Zielgruppe (der Nutzer) abhängig. Gerade Fachinformationen sind oft für den Konsum (die Verarbeitung) durch menschliche Nutzer aufbereitet, und die Informationssysteme entsprechend dafür implementiert und optimiert.
Viele Daten sind schwach strukturiert und liegen z.B. in Form von Dokumenten (PDF, DOC) und zugehörigen Metainformationen vor.

**A2.2 Semantische Interpretation von Daten**

Die unmittelbare semantische Interpretation von schwach strukturierten Daten ist häufig nur auf Basis der verfügbaren Metadaten (Titel, Inhaltsangabe, Stand, Gültigkeitsbereich, Schlagworte etc.) möglich, die zur Verwendung durch menschliche Nutzer auch zum Sortieren, Filtern und Gruppieren herangezogen werden können, jedoch, insbesondere wenn fachliche Metadaten vorhanden sind, ein inhaltliches Grundverständnis des Nutzers voraussetzen.

In der Praxis werden Dokumentenbestände als Paradebeispiel für große Mengen unstrukturierter Daten (unter weitgehender Umgehung ihrer Semantik) häufig durch Volltextsuchmaschinen erschlossen. Die meisten Volltextsuchen basieren dabei auf zuvor gebildeten Indexen und dem lexikalischen Vergleich von im Index gespeicherten Begriffen und den verwendeten Suchbegriffen.


Andere Typen von Daten können stark strukturiert sein, z.B. die Daten dauerhafter Messprogramme (Zeitreihen zur Luftqualität, Pegelstände oder Wetterdaten), die meist in festen Strukturen, z.B. mit Hilfe relationaler Datenbankmodelle, abgelegt werden. Die Struktur der Daten bleibt in der Regel auch beim Zugriff erhalten, z.B. werden einzelne Messwerte, der zugehörige Zeitstempel und Stammdaten (Ort der Messung, Messmethodik) jeweils zusammenhängend ausgeliefert.

Die semantische Interpretation ist im Allgemeinen jedoch auch bei stark strukturierten Daten abhängig von Zusatzinformationen, d.h. die Semantik ergibt sich nicht alleine aus den Daten selbst, sondern z.B. nur unter Einbeziehung von Metainformationen, die Aufschluss über die Bedeutung von Spaltenbezeichnern, verwendete Einheiten und Ähnliches geben.

**A3 Webportale**

Bei Webportalen handelt es sich um eine spezielle Art von Websites, bei denen die Bündelung von Informationen zu einem Themengebiet im Vordergrund steht, zum Beispiel um den Nutzern einen zentralen Einstieg in das Thema zu bieten. Typischerweise stammen die im Portal präsentierten Informationen aus unterschiedlichen Datenquellen, zum Beispiel in einem Flugbuchungsportal aus den Systemen verschiedener Fluggesellschaften. Daher ist es eine Aufgabe des Portals bzw. seiner Entwickler, Informationen zu bündeln, ggf. zu harmonisieren und dem Benutzer in geeigneter Form, meist
einheitlich, zu präsentieren, z.B. durch Selektionsmechanismen wie die Facettierung (Stock und Stock 2008). Häufig sind Webportale mit einer ganzen Reihe von Funktionalitäten ausgestattet, z.B. der Möglichkeit zur Personalisierung (Einstellungen) durch den Nutzer, interaktiven Elementen wie Foren oder Bewertungssystemen, Suchmaschinen etc.

A4 Serviceorientierte Architekturen

Serviceorientierte Architekturen (SOA) stellen eine moderne Form verteilter Informationssysteme dar und wurden erstmals 1996 beschrieben (OASIS Open 2006; Schulte und Natis 1996). Die Grundlage für SOAs bilden Beschreibungen von Geschäftsprozessen auf verschiedenen Abstraktionsebenen, die jeweils entsprechende, aufeinander aufbauende Implementierungen haben. Ein „höherwertiger“ Prozess besteht also in der Regel aus einer Zusammensetzung einfacherer Dienste/Prozesse, die er zielführend verwendet.

Ein wesentliches Prinzip der Architektur ist dabei die Wiederverwendbarkeit von Diensten, bei denen es sich jedoch immer um Teile des jeweiligen Geschäftsprozesses, also inhaltlichen Aufgaben, handelt – im Gegensatz zu rein technischen Aufgaben (wie das Ausführen einer einzelnen Datenbankabfrage).

Sichtbar und nutzbar werden die einzelnen Geschäftsprozesse (Dienste) durch ihre Schnittstellen (Eingaben/Parameter und Ergebnisse). Serviceorientierte Architekturen spielen eine große Rolle bei der Implementierung von Internet-Anwendungen, die dabei auf eine ganze Reihe standardisierter (technischer) Protokolle aufsetzen können.


Eine verbreitete Definition von SOA (OASIS Open 2006) sieht vor, dass die einzelnen Dienste („verteilte Funktionalität“) einer SOA von mehreren Anbietern bereitgestellt werden können – also (zumindest) organisatorisch nur lose gekoppelt sind. Das erfordert eine bestimmte Generizität solcher Dienste, die sich durch Parametrisierbarkeit und die Standardisierung von Datenmodellen ausdrückt. Große Internet-Dienstleister wie Google bieten eine große Zahl solcher generischer Dienste an, z.B. Gazetteer-
Services zum Auflösen von Ortsnamen oder Standortdaten. Im Sinne einer SOA erfüllen solche Dienste durchaus inhaltliche Aufgaben. Deren Nutzung in eigenen Anwendungen ist also gerade in Bezug auf die Wiederverwendbarkeit von Softwarekomponenten und damit die Vermeidung redundanter Implementierungen im Sinne des Entwicklens von verteilten Anwendungen nach der SOA-Architektur sinnvoll. Die rechtlichen, sicherheitstechnischen bzw. lizenztechnischen Aspekte solcher Lösungen sind relevant, jedoch nicht Bestandteil der vorliegenden Diskussion.

Häufig werden allgemein nutzbare Dienste über große Cloud-Infrastrukturen (Baun et al. 2010) (Weinhardt et al. 2009) zur Verfügung gestellt, was meistens ein hohes Maß an Verfügbare und Skalierbarkeit sicherstellt.

Auch im Zusammenhang mit per se (oder de facto) verteilten Informationen bietet sich die Nutzung von serviceorientierten Infrastrukturen an. Daten und Prozesse, die noch nicht durch Dienste bereitgestellt werden, können häufig mit relativ geringem Aufwand als Adapter (Gamma 2004) (auch: „Wrapper“) an eine bestehende (nicht serviceorientierte) Anwendung gekoppelt werden. Wo eine solche Anpassung nicht möglich ist, reicht häufig auch die Entwicklung einer parallelen, ggf. abgespeckten, Variante der Original-Anwendung und einem Mechanismus zur Synchronisierung von Daten.


Proprietäre, d.h. nicht standardisierte, Schnittstellen, stellen ein Hindernis beim Aufsetzen einer serviceorientierten Architektur im Kontext bestehender Systeme dar. Auch wenn proprietäre Schnittstellen im Hinblick auf eine einzelne Anwendung durchaus sinnvoll sein können, z.B. was ihre Effizienz betrifft, so hinderlich ist die fehlende In-

---

23 [https://developers.google.com/maps/documentation/geocoding/](https://developers.google.com/maps/documentation/geocoding/)
24 Das ist normalerweise Bestandteil eines (ggf. auszuhandelnden) Service-Level-Agreements (SLA) zwischen dem Anbieter und den Anwendern solcher Dienste.
teroperabilität, wenn solche Systeme geöffnet und in einem größeren Kontext verfügbar gemacht werden sollen, z.B. wenn Daten aus einem Fachsystem im Zuge des E-Government verfügbar gemacht werden sollen.

Wenn interoperable Schnittstellen nicht direkt in den Originalsystemen implementiert werden können, kann die Interoperabilität durch das Vorschalten von Adapter-Programmen hergestellt werden, die auf der einen Seite die proprietäre Schnittstelle implementieren, und alle oder einzelne Dienste auf der anderen Seite z.B. als Web- oder RESTful Services zur Verfügung stellen. Um dabei auch eine semantische Interoperabilität zu erreichen, müssen die Adapter im Allgemeinen auch eine Abbildung des proprietären Datenmodells auf ein standardisiertes oder zumindest in der SOA bekanntes Datenmodell vornehmen.

A4.1 Microservices

Microservices stellen eine spezielle Art von Diensten dar (Fowler und Lewis 2015). Ihr wesentlichstes Merkmal ist, dass jeder Microservice für einen speziellen, kleinen und abgegrenzten Aufgabenbereich verantwortlich ist („Do one thing and do it well!“)\(^{26}\), und seine Funktion(en) über eine (möglichst sprachunabhängige) Schnittstelle zur Verfügung stellt. Microservices können sich über ihre Schnittstellen auch gegenseitig nutzen – sind aber grundsätzlich voneinander entkoppelt. Auf Microservices beruhende Anwendungen sind somit per se modular aufgebaut und einzelne Microservices sind leicht austauschbar. Bei der Entwicklung und Wartung bieten Microservices aufgrund ihrer überschaubaren Funktionalität und Größe erhebliche Vorteile: Kurze Entwicklungszeiten und die Möglichkeit des Einsatzes von Automatisierungswerkzeugen (Continuous Integration und/oder Continuous Delivery) (Fowler 2006; Humble et al. 2006).

\(^{26}\) Das Zitat stammt ursprünglich von Douglas McIlroy, dem Erfinder der Unix-Pipes, und bezieht sich auf Unix-Kommandos. Es wird jedoch auch häufig im Zusammenhang mit Microservices verwendet.


**A4.2 Schnittstellen und Protokolle**

(SOAP-)Webservices

Bei den SOAP-Webservices handelt es sich um einen Standard des W3C-Konsortiums (W3C 2004d). Dabei stützt sich SOAP auf verschiedene Internet-Basistechnologien zum Transport von Daten, bietet darüber hinaus jedoch einen hohen Grad an Standar-
disierung, der sich vor allem in der Festlegung der verwendeten technischen Datenformate (meist XML-basiert) ausdrückt. Die semantische Interpretation beschränkt sich dabei auf die technische Verarbeitung von Daten. Was die eigentlichen (Nutz-)Inhalte betrifft, macht SOAP keine Vorgaben, d.h. die inhaltliche Festlegung und Interpretation von Daten erfolgt anwendungsspezifisch.


Allerdings bietet SOAP auch Unterstützung komplexer, aus mehreren Teilen bestehender Anfragen, so dass sich der Aufwand in manchen Anwendungsszenarien wieder relativieren kann.


RESTful Services (REST)

Representational State Transfer (REST) stellt keine Normierung oder Standardisierung im engeren Sinne dar. Es handelt sich vielmehr um ein Programmierparadigma, das als Konvention für die Verwendung des HTTP-Protokolls durch Programme verstanden werden kann (Bayer 2002).

Im Gegensatz zur Verwendung des WWW durch menschliche Nutzer dienen REST-Services dem Datenaustausch zwischen Programmen, d.h. die übermittelten Daten dienen der Weiterverarbeitung durch die empfangende Anwendung 27. REST bedient sich dabei des HTTP-Protokolls und reduziert die möglichen Operationen zwischen Anwendungen auf eine kleine Menge von Operationen, die im Wesentlichen den CRUD-Operationen 28 für Datenbankanwendungen entsprechen, welche dabei im Wesentlichen den HTTP-Operationen PUT/POST, GET, PATCH/PUT bzw. DELETE zugeordnet werden.

Der Aufbau von REST-URLs ist nicht standardisiert, es gibt jedoch Best-Practise-Ansätze, die Empfehlungen für den über die URL-Syntax hinausgehenden Aufbau der REST-URLs geben 29 (Fredrich 2013).

---

27 Die Anwendung kann allerdings auch im Webbrowser eines Anwenders laufen und z.B. eine Ansicht für einen menschlichen Nutzer generieren.

28 C für das Anlegen (Create), R für das (wiederholte) Lesen (Read), U für das (ggf. wiederholte) Aktualisieren (Update) und D für das Löschen eines Datensatzes (Delete).

29 http://www.vinaysahni.com/best-practices-for-a-pragmatic-restful-api#restful
Zum Beispiel liefert die GET-Anfrage auf die URL

http://services-bw.de/water/gauging

eine Liste aller Pegelstationen. Die URL

http://services-bw.de/water/gauging/177

ilfert die Daten zu einer bestimmten Pegelstation mit der ID 177. Darüber hinaus kön-
nen die Anfragen über weitere URL-Parameter genauer spezifiziert werden, z.B.
schränkt die URL


die Anfrage auf einen bestimmten Zeitraum ein.

Auch die Datenformate für den Datenaustausch mittels RESTful Services sind nicht
standardisiert. Es kommen häufig XML-basierte oder JSON-Formate zum Einsatz, je-
doch auch HTML. Seit der Einführung von Ajax (Garrett 2005) und HTML5 (Hickson et
al. 2014) wird zunehmend JSON (ECMA International 2013) verwendet, da das leicht-
gewichtige Format in Webbrowsern direkt durch JavaScript-Programme verarbeitet
werden kann.

Im Allgemeinen können die durch einen RESTful Service gelieferten Daten Adressen
(URLs bzw. URIs) für weitere REST-Aufrufe enthalten, die z.B. Beziehungen zwischen
Objekten oder Kompositionen (ein Objekt besteht aus mehreren anderen Objekten)
ausdrücken können. Eine solche Darstellung von Beziehungen wird auch als HATEO-
AS (Hypermedia as the Engine of Application State) bezeichnet und stellt ein wichtiges
Prinzip von REST-basierten Anwendungen dar (Wikipedia 2016). HATEOAS ist auch
einen Brückenschlag zu „Linked Data“ aus dem Bereich des Semantic Web.

A5 Cloud-Dienste

Viele Anwendungen und Dienste werden inzwischen nicht mehr auf dedizierten Ser-
vorn zur Verfügung gestellt, sondern nutzen große Server-Infrastrukturen, die Res-
sourcen nach Bedarf dynamisch zuteilen. Das führt aus Sicht der Betreiber unter Ande-
rem zu einer erhöhten Effizienz, Skalierbarkeit und Ausfallsicherheit.

Cloud-Dienste lassen sich grob in drei Klassen unterscheiden (Abbildung 31):

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS).


Eidesstattliche Versicherung

Die vorliegende Arbeit wurde von mir selbstständig angefertigt und es wurden keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Wörtlich oder inhaltlich übernommenen Stellen wurden als solche kenntlich gemacht. Die Satzung des Karlsruher Instituts für Technologie (KIT) zur Sicherung guter wissenschaftlicher Praxis in der gültigen Fassung wurde beachtet.

_________________________  ____________________________
Ort, Datum                                      Unterschrift
Literaturverzeichnis


Angrick, Michael; Bös, Richard; Rüther, Maria; Bandholtz, Thomas (2002): Semantic Network Services (SNS). In: Klaus Tochtermann Werner Pillmann (Hg.): IGU/ISEP: IGU/ISEP, S. 78–84.


189. Online verfügbar unter https://dl.acm.org/citation.cfm?id=2889917, zuletzt geprüft am 01.06.2018.


Düpmeier, Clemens; Geiger, Werner; Greceanu, Claudia; Weidemann, Rainer; Ebel, Renate; Lehle, Manfred et al. (2009): Themenpark Umwelt. Optimierung der Volltext-


Gschwender, David; Kost, Florian; Schillinger, Wolfgang; Niemeier, Rüdiger; Koch, Lars; Düpmeier, Clemens; Schlachter, Thorsten (2016): Energieatlas Baden-Württemberg Daten und Fakten zur Energiewende. In: Kurt Weissenbach, Wolfgang Schillinger und Rainer Weidemann (Hg.): INOVUM Phase I 2014/16, S. 61–70. Online verfügbar unter http://www.fachdokumente.lubw.baden-wuerttemberg.de/content/119257/INOVUM_I_Endfassung.pdf.


Hickson, Ian; Berjon, Robin; Faulkner, Steve; Leithead, Travis; Navara, Erika Doyle; O’Connor, Edward; Pfeiffer, Silvia (2014): HTML5. A vocabulary and associated APIs for HTML and XHTML. W3C. Online verfügbar unter https://www.w3.org/TR/html5/, zuletzt aktualisiert am 24.10.2014, zuletzt geprüft am 02.06.2016.


Lemke, Matthias; Wiedemann, Gregor; Blätte, Andreas (2016): Text Mining in den Sozialwissenschaften. Grundlagen und Anwendungen zwischen qualitativer und quantitativer Diskursanalyse.


Nikolai, Ralf (2002): Thesaurusföderationen: Ein Rahmenwerk für die flexible Integrati-

delberg: Springer (International handbooks on information systems. Handbook on ono-
logies).


OASIS UDDI Specifications TC (2016): OASIS - Committees - OASIS UDDI Specifica-


Schlachter, Thorsten; Düpmeier, Clemens; Geiger, Werner; Weidemann, Rainer; Ebel, Renate; Tauber, Martina et al. (2011a): Concept of a universal mobile application accessing environmental information systems. In: Innovations in Sharing Environmental Observations and Information: EnvirolInfo 2011 : 25th Internat.Conf.on Environmental Informatics, S. 398–504.


Schlachter, Thorsten; Düpmeier, Clemens; Weidemann, Rainer; Schillinger, Wolfgang; Bayer, Nina; Hrebicek, J. (2013): ‘My environment’ - a dashboard for environmental information on mobile devices. In: Environmental Software Systems: Fostering Infor-
mation Sharing; 10th IFIP WG 5.11 International Symposium. (ISESS 2013), S. 197–203.


Sherman, Chris; Price, Gary (2001): The invisible web: uncovering sources search engines can't see. Hg. v. University of Illinois at Urbana-Champaign.


Uckelmann, Dieter; Harrison, Mark; Michahelles, Florian (2011a): An Architectural Approach Towards the Future Internet of Things. In: Dieter Uckelmann, Mark Harrison und Florian Michahelles (Hg.): Architecting the Internet of Things. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 1–24. Online verfügbar unter https://doi.org/10.1007/978-3-642-19157-2_1, zuletzt geprüft am 01.06.2018.

Uckelmann, Dieter; Harrison, Mark; Michahelles, Florian (2011b): Architecting the Internet of Things. Berlin: Springer-Verlag.


Weidemann, Rainer; Geiger, Werner; Greceanu, Claudia; Schlachter, Thorsten; Zilly, Gerd; Lautner, Petra et al. (2007): FADO BW - Realisierung erster Komponenten für ein verteiltes Fachdokumentenmanagement im Umweltinformationssystem Baden-Württemberg. In: Roland Mayer-Föll, André Keitel und Werner Geiger (Hg.): KEWA Phase II, S. 31–44.

Weidemann, Rainer; Geiger, Werner; Greceanu, Claudia; Schlachter, Thorsten; Zilly, Gerd; Lautner, Petra et al. (2008): FADO BW - Entwicklung der Basisversion für das


