APIS, GEOTAGGING & XML / RSS

29.05.2014 @ SAE INSTITUTE

BJOERN ZAPADLO

ABOUT ME

Bjoern Zapadlo
Konstanz
34 Jahre

Development Manager / Consultant @ HolidayCheck AG

Informatik Studium 1999 - 2002
3 Agenturen in Stuttgart

HolidayCheck International Websites / new Framework
Neckermann / Thomas Cook

Dozent an der SAE, Dualen Hochschule Stuttgart,
Hochschule Furtwangen

PHP, Java, Scala, Javascript, CSS, Html,MySQL, MongoDB,
Elasticsearch, ...

CONTACT ME

bjoern.zapadlo@gmail.com
http://www.zapadlo.de

@BjoeZap

https://www.xing.com/profile/Bjoern_Zapadlo
http://de.linkedin.com/pub/bjoern-zapadlo/36/889/1a5

Facebook
Google+

HOLIDAYCHECK AG

Größtes deutsches Meinungsportal für Reise und Urlaub
Vermittlung von Reisen
Sitz in der Schweiz, direkt am Bodensee
Börsennotiert über Tomorrow Focus AG
Existiert seit 1999
Ausgründungen in mehreren europäischen Ländern
Über 330 Mitarbeiter

YES, WE HIRE ;)

HTTP://WWW.HOLIDAYCHECK.DE/JOBS

JETZT ABER SCHLUSS MIT DER WERBUNG... ;)

UND IHR?

NAME, ERWARTUNGEN, KENNTNISSE

AGENDA

1. XML
2. Geotagging
3. APIs
4. Q&A

XML

AGENDA

1. Warum?
2. Aufbau
3. Übung XML
4. Validität
5. CSS
6. Suchen
7. Transformation
8. Namensräume
9. XML-Sprachen
10. Einsatzgebiete
11. XHTML
12. RSS
13. Podcasts
14. Übung RSS

WARUM XML

WER VON EUCH HAT HEUTE SCHON XML BENUTZT?

AUF FACEBOOK GEWESEN

ADRESSEN ODER TERMINE AUF DEM HANDY SYNCHRONISIERT

GOOGLE EARTH ODER MAPS BENUTZT

EIN OFFICE DOKUMENT GESCHRIEBEN

IM INTERNET GESURFT

DAS PROBLEM

STRUKTURIERUNG UND AUSZEICHNUNG VON TEXT

Das Wort fett soll fett sein

Hier möchte ich gerne

einen Absatz haben

DIE LÖSUNG?

```
Das Wort FETTSTARTfettFETTSTOP soll fett sein

Hier möchte ich gerneABSATZeinen Absatz haben
```

NAJA, FAST

Eine spezielle Maskierung zur Abgrenzung von Nutz und Strukturierungsdaten muss her

DAS TAG IST <GEBOREN>

AUSZEICHNUNGSSPRACHEN

Eine Auszeichnungssprache (englisch: Markup Language, ML) dient zur Beschreibung des Inhalts eines Dokumentenformates und teilweise zur Beschreibung des Verfahrens, welches zur Bearbeitung dieser Daten benötigt wird. Ursprünglich dienten die Auszeichnungen im Text als Anweisungen für die Setzer im Drucksatz, mit der Weiterentwicklung in der Typografie für digitale Texte wurden daraus jedoch komplexe Sprachen.

Sind keine Programmiersprachen (PHP, ...) meistens ;)

Können trotzdem sehr komplex sein

Sind Dank des Internets sehr populär

WAS KANN XML?

SEMANTIK

Durch die Auszeichnung wird z.B. eine Überschrift zur Überschrift

MASCHINENLESBARKEIT

Durch die Auszeichnung kann eine (Such)Maschine Texte und deren Bedeutung interpretieren

MENSCHENLESBARKEIT

Okay, es gibt Sachen, die machen mehr Spass

WOFÜR

Datenaustausch (in heterogenen Systemen)

Datenbankersatz

VOR- & NACHTEILE

VORTEILE

Einfach & Flexibel

Menschen und Maschinenlesbar

Es ist Text

NACHTEILE

Es ist Text: Encoding, Escaping von Steuerzeichen

Kann Komplex werden

Performance

Overhead

GESCHICHTE VON XML

Vorgänger SGML (ISO 8879:1986):
Standard Generalized Markup Language
IBM Forschungsprojekt
sehr flexibel aber komplex und performancehungrig

1989 für das CERN HTML
Tim Berners-Lee
Einfach, aber unsauber, extreme Verbreitung
Versionschaos, dann Standartisierung durch das W3C

Parallel XML
Vereinfachtes SGML

XML + HTML = XHTML

STANDARDS

Standardisiert vom W3C

1.0 (5th Edition)

1.1 (2nd Edition)

AUFBAU

DAS ERSTE XML DOKUMENT

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<verzeichnis>
    <titel>Wikipedia Städteverzeichnis</titel>
    <eintrag>
        <stichwort>Genf</stichwort>
        <eintragstext>Genf ist der Sitz von ...</eintragstext>
    </eintrag>
    <eintrag>
        <stichwort>Köln</stichwort>
        <eintragstext>Köln ist eine Stadt, die ...</eintragstext>
    </eintrag>
</verzeichnis>
```

PHYSISCHE STRUKUTUR

XML-DEKLARATION

Eine XML-Deklaration ist eine Erkennungszeichenfolge im Prolog einer XML-Datei

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

VERSION="1.0"

definiert die Versionsnummer der zugrundeliegenden XML-Spezifikation. Sinnvolle Werte sind momentan 1.0 und 1.1. Der Parameter version muss stets angegeben werden.

ENCODING="ZEICHENKODIERUNG"

bestimmt die Kodierung der XML-Datei. Wird dieser Parameter ausgelassen, wird UTF-8 angenommen. Zu beachten ist, dass der Wert von encoding vom Parser unterstützt werden muss.

STANDALONE="YES|NO"

Eher selten verwendet. Gültige Werte sind yes oder no. Wird der Parameter standalone ausgelassen, wird der Standardwert no angenommen. Der Attributwert yes wird verwendet, wenn ein Dokument über keine DTD verfügt. Des Weiteren kann sich die DTD auch in derselben Datei

ENTITÄTEN

Die XML-Datei selbst

ENTITÄTENREFERENZEN

Verknüpfungen auf wiederkehrende Zeichenketten und ganze Dateien

STANDARD-ENTITÄTEN:

' = '
& = &
" = "
< = <
> = >

ENTITÄTENREFERENZEN

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!ENTITY c "Chris">
<!ENTITY chap1 SYSTEM "chapter-1.xml">
<verzeichnis>
    <titel>Hallo &c;</titel>
    <!-- Pull in the chapter content: -->
    &chap1;
</verzeichnis>
```

DTD

Eine Dokumenttypdefinition (englisch Document Type Definition, DTD, auch Schema-Definition oder DOCTYPE) ist ein Satz an Regeln, der benutzt wird, um Dokumente eines bestimmten Typs zu deklarieren.

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE hallo SYSTEM "hallo.dtd">
<hallo>Hallo Welt!</hallo>
```

LOGISCHE STRUKTUR

Der logische Aufbau entspricht einer Baumstruktur und ist damit hierarchisch organisiert.

VERARBEITUNGSANWEISUNG

Verarbeitungsanweisung (engl. processing instruction) oder auch Steueranweisung ist ein Konstrukt von SGML und XML, um Daten abzulegen, die an eine bestimmte (XML-)Anwendung gerichtet ist.

Eine XML-Deklaration ist keine Verarbeitungsanweisung, auch wenn sie genauso aussieht.

```
<?xml-stylesheet type="text/xsl" href="show_book.xsl"?>
```

ELEMENTE

Auszeichnung via Start- und End-Tag oder einem sog. Empty-Element-Tag

```
<Tag-Name></Tag-Name>
<Empty-Element />
```

ATTRIBUTE

Attribute als bei einem Start-Tag oder Empty-Element-Tag geschriebene Schlüsselwort-Werte-Paare (Attribut-Name="Attribut-Wert") für Zusatz-Informationen über Elemente

```
<Tag-Name attribute-key="attribute-value"></Tag-Name>
<Empty-Element attribute-key="attribute-value" />
```

TEXT

Normaler Text ;) bzw. die Nutzinformation

```
<Tag-Name>ganz normaler Text</Tag-Name>
```

CDATA

Mit einem CDATA(Akronym für engl. character data „Zeichendaten“)-Abschnitt wird einem Parser mitgeteilt, dass kein Markup folgt, sondern normaler Text. Dieser wird nicht interpretiert und kann daher auch "illegales" XML (Tags, Sonderzeichen, ...) enthalten.

```
<Tag-Name>
    <![CDATA[ Inhalt ]]>
    </Tag-Name>
    <Tag-Name>
    <![CDATA[ Inhalt]] ]]><![CDATA[ >Inhalt ]]>
</Tag-Name>
```

KOMMENTARE

```
<!-- Kommentar-Text -->
```

REGELN FÜR XML

XML-Deklaration muss vorhanden sein

Baumstruktur muss eingehalten werden:
Wurzelelement darf genau einmal vorhanden sein
Tags müssen geschlossen und die korrekte Verschachtelung muss eingehalten werden
XML ist „Case Sensitive“ und nicht auf ASCII beschränkt

Kommentare müsssen korrekt sein

Der Zeichensatz sollte angegeben werden

"WOHLGEFORMT"

Wenn keine Regel Verletzt wird.
Prüfung erfolgt durch den XML-Parser

KLASSIFIZIERUNG

DOKUMENTZENTRIERT

Das Dokument ist an ein Textdokument angelehnt, das für den menschlichen Leser größtenteils auch ohne die zusätzliche Metainformation verständlich ist. XML-Elemente werden hauptsächlich zur semantischen Markierung von Passagen des Dokuments genutzt, das Dokument ist nur schwach strukturiert. Aufgrund der schwachen Strukturierung ist eine maschinelle Verarbeitung schwierig.

DATENZENTRIERT

Das Dokument ist hauptsächlich für die maschinelle Verarbeitung bestimmt. Es folgt einem Schema, das Entitäten eines Datenmodells beschreibt und definiert, in welcher Beziehung die Entitäten zueinander stehen, sowie, welche Attribute die Entitäten haben. Das Dokument ist

SEMISTRUKTURIERT

Semistrukturierte Dokumente stellen eine Art Mischform dar, die stärker strukturiert ist als dokumentzentrierte Dokumente, aber schwächer als datenzentrierte Dokumente.

ÜBUNG XML

PHP: ZUGRIFF VIA DOM

PHP: ZUGRIFF VIA SIMPLE-XML

JAVASCRIPT: ZUGRIFF VIA AJAX & NATIVE JS

JAVASCRIPT: ZUGRIFF VIA JQUERY

VALIDITÄT

DEFINITION

Soll XML für den Datenaustausch verwendet werden, ist es von Vorteil, wenn das Format mittels einer Grammatik (z. B. einer Dokumenttypdefinition oder eines XML-Schemas) definiert ist. Der Standard definiert ein XML-Dokument als gültig (oder englisch valid), wenn es wohlgeformt ist, den Verweis auf eine Grammatik enthält und das durch die Grammatik beschriebene Format einhält.

DTD

Eine Dokumenttypdefinition (englisch Document Type Definition, DTD, auch Schema-Definition oder DOCTYPE) ist ein Satz an Regeln, der benutzt wird, um Dokumente eines bestimmten Typs zu deklarieren. Ein Dokumenttyp ist dabei eine Klasse ähnlicher Dokumente, wie beispielsweise Telefonbücher oder Inventurdatensätze. Die Dokumenttypdefinition besteht dabei aus Elementtypen, Attributen von Elementen, Entitäten und Notationen.

Konkret heißt das, dass in einer DTD die Reihenfolge, die Verschachtelung der Elemente und die Art des Inhalts von Attributen festgelegt wird – kurz gesagt: die Struktur des Dokuments.

Diese können direkt im XML-Dokument definiert werden oder von dort verlinkt werden.

XML-DOKUMENT

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE  Person SYSTEM "toll.dtd">

<Person>
    <Name>Manfred Müller</Name>
    <Kind>
        <Name>Olivia Müller</Name>
    </Kind>
    <Kind>
        <Name>Marc Müller</Name>
    </Kind>
</Person
```

DTD

```
<!ELEMENT Person (Name, Kind*)>
<!ELEMENT Kind (Name)>
<!ELEMENT Name (#PCDATA)>
```

AHA...

Diese DTD sagt aus, dass es ein Element Person gibt, mit den Unterelementen Name und Kind, wobei Kind öfter als einmal vorkommen darf.
Desweiteren bestimmt diese DTD, dass Name mit Text oder weiteren Unterknoten gefüllt sein darf während CDATA für reinen Text steht.

XSD

XML Schema, abgekürzt XSD (XML Schema Definition), ist eine Empfehlung des W3C zum Definieren von Strukturen für XML-Dokumente. Anders als bei den klassischen XML-DTDs wird die Struktur in Form eines XML-Dokuments beschrieben. Darüber hinaus wird eine große Anzahl von Datentypen unterstützt.

XML Schema beschreibt in einer komplexen Schemasprache Datentypen, einzelne XML-Schema-Instanzen (Dokumente) und Gruppen solcher Instanzen. Ein konkretes XML Schema wird auch als eine XSD (XML Schema Definition) bezeichnet und hat als Datei üblicherweise die Endung „.xsd“.

XML-DOKUMENT

```
...
<pc-Typ>
    <name>Ein Name, aber Element ist optional</name>
    <hersteller>Ein Hersteller, aber Element ist optional</hersteller>
    <id>123</id>
</pc-Typ>
...
```

XSD

```
<xsd:complexType name="pc-Typ">
    <xsd:sequence>
        <xsd:element name="name" type="xsd:string"/>
        <xsd:element name="hersteller" type="xsd:string"/>
    </xsd:sequence>
    <xsd:attribute name="id" type="xsd:integer"/>
</xsd:complexType>
```

CSS

XML UND CSS

Analog zu Html lassen sich auch XML Dokument mit CSS stylen. Die Interpretation erfolgt dann z.B. per Browser, welcher XML darstellen kann. Allerdings ist das ein eher ungebräuchlicher Ansatz.

So sieht es aus...

XML-DOKUMENT

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<?xml-stylesheet type="text/css" href="cd_catalog.css"?>
<CATALOG>
    <CD>
        <TITLE>Empire Burlesque</TITLE>
        <ARTIST>Bob Dylan</ARTIST>
        <COUNTRY>USA</COUNTRY>
        <COMPANY>Columbia</COMPANY>
        <PRICE>10.90</PRICE>
        <YEAR>1985</YEAR>
    </CD>
    .
    .
    .
</CATALOG>
```

CSS-DATEI

```
CATALOG {
    background-color: #ffffff;
    width: 100%;
}
CD {
    display: block;
    margin-bottom: 30pt;
    margin-left: 0;
}
TITLE {
    color: #FF0000;
    font-size: 20pt;
}
```

SUCHEN

XPATH

Die XML Path Language (XPath) ist eine vom W3-Konsortium entwickelte Abfragesprache, um Teile eines XML-Dokumentes zu adressieren. XPath dient als Grundlage einer Reihe weiterer Standards wie XSLT, XPointer und XQuery.

Ein XPath-Ausdruck adressiert Teile eines XML-Dokuments, das dabei als Baum betrachtet wird.

XML-DOKUMENT

```
<?xml version="1.0" encoding="utf-8" standalone="yes" ?>
<dok>
    <kap title="Nettes Kapitel">
        <pa>Ein Absatz</pa>
        <pa>Noch ein Absatz</pa>
        <pa>Und noch ein Absatz</pa>
        <pa>Nett, oder?</pa>
    </kap>
    <kap title="Zweites Kapitel">
        <pa>Ein Absatz</pa>
        <pa format="bold">Erste Zeile</pa>
        <pa format="bold">Zweite Zeile</pa>
        <pa format="italic">Dritte Zeile</pa>
    </kap>
</dok>
```

XPATH BEISPIELE

Ausdruck	selektiert ...
`/dok`	das erste Element `dok`
`/*`	das Wurzel-Element unabhängig vom Namen (jedes wohlgeformte XML-Dokument hat genau ein Wurzel-Element)
`//dok/kap`	alle `kap`-Elemente innerhalb eines `dok` Elements
`//dok/kap[1]`	erstes `kap`-Element innerhalb eines `dok` Elements
`//pa`	alle `pa`-Elemente auf allen Ebenen
`//kap[@title='Nettes`	alle Absätze der Kapitel mit

`child::*`	alle Kindelemente des gegenwärtigen Knotens
`child::pa`	alle `pa`-Kinder des gegenwärtigen Knotens
`child::text()`	alle Textknoten des gegenwärtigen Knotens
`.`	der gegenwärtige Knoten
`./*`	alle Unterelemente des gegenwärtigen Knotens
`./pa`	alle `pa`-Kinder des gegenwärtigen Knotens
`pa`	alle `pa`-Kinder des gegenwärtigen Knotens
`attribute::*`	alle Attribute des gegenwärtigen Knotens
`namespace::*`	alle Namespaces des gegenwärtigen Knotens

TRANSFORMATION

UMWANDLUNG VON XML-DOKUMENTEN

XSLT

XSL Transformation, kurz XSLT, ist eine vollständige Programmiersprache zur Transformation von XML-Dokumenten. Sie ist Teil der Extensible Stylesheet Language (kurz XSL, eine in XML notierte Familie von Transformationssprachen zur Definition von Layouts für XML-Dokumente).

XSLT baut auf der logischen Baumstruktur eines XML-Dokumentes auf und dient zur Definition von Umwandlungsregeln. XSLT-Programme, sogenannte XSLT-Stylesheets, sind dabei selbst nach den Regeln des XML-Standards aufgebaut.

Die Stylesheets werden von spezieller Software, den XSLT-Prozessoren, eingelesen, die mit diesen Anweisungen ein oder mehrere XML-Dokumente in das gewünschte Ausgabeformat umwandeln. XSLT-Prozessoren sind auch in

XML-DOKUMENT + XSLT-STYLESHEET => XSLT-PROZESSOR = NEUES DOKUMENT

So sieht es aus...

XML-DOKUMENT

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<catalog>
    <cd>
        <title>Empire Burlesque</title>
        <artist>Bob Dylan</artist>
        <country>USA</country>
        <company>Columbia</company>
        <price>10.90</price>
        <year>1985</year>
    </cd>
    ...
</catalog>
```

XSLT-STYLESHEET

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform
    <xsl:template match="/">
        <html><body>
        <h2>My CD Collection</h2>
        <table border="1">
        <tr bgcolor="#9acd32"><th>Title</th><th>Artist</th></tr>
        <xsl:for-each select="catalog/cd">
        <tr><td><xsl:value-of select="title" /></td><td><xsl:value-of select=
        </xsl:for-each>
        </table>
        </body></html>
    </xsl:template>
</xsl:stylesheet>
```

NAMENSRÄUME

PROBLEMSTELLUNG

Mischung mehrere XML-Sprachen in einem Dokument.

Überschneidung bei den Tag-Names

Wer soll welches Tag bearbeiten

Siehe Telefon-Vorwahl ;)

Vorsicht: DTDs unterstützen dies nicht

```
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:svg="http://www.w3.org/2000/
...
<p>Und noch ein kleines Bild dazu:</p>

<svg:svg>
    <svg:rect x="0" y="0" width="1"  height="1" />
</svg:svg>
```

XML-SPRACHEN

Im Zusammenhang mit XML wurden vom W3-Konsortium auf Basis von XML viele Sprachen definiert, welche XML-Ausdrücke für häufig benötigte allgemeine Funktionen anbieten wie etwa die Verknüpfung von XML-Dokumenten. Zahlreiche XML-Sprachen nutzen diese Grundbausteine.

Verknüpfung von XML-Ressourcen: XPointer, XLink und XInclude

Selektion von Daten aus einem XML-Datensatz: XQuery

Datenmanipulation in einem XML-Datensatz: XUpdate

Abfassen von elektronischen Formularen: XForms

Definition von XML-Datenstrukturen: XML Schema (= XSD, XML Schema Definition Language), DTD und RELAX NG

Signatur und Verschlüsselung von XML-Knoten: XML Signature und XML-Encryption

Aussagen zum formellen Informationsgehalt: XML Infoset

Formatierte Darstellung von XML-Daten: XSL-FO

Definition zum Methoden- bzw. Funktionsaufruf durch verteilte Systeme: XML-RPC

Standardisierte Attribute: XML Base und ID (DTD)

XML-basierte deklarative Programmiersprache: MXML

EINSATZGEBIETE

TEXT

DocBook
DITA
XHTML (XML-konformes HTML)
TEI (Text Encoding Initiative)
NITF (News Industry Text Format)
OPML (Outline Processor Markup Language)
OSIS (Open Scripture Information Standard)
OpenDocument-Austauschformat (OASIS Open Document Format for Office Applications)

GRAFIK

SVG (Vektorgrafiken)
X3D (3D-Modellierungssprache)
Collada (Austauschformat für Daten zwischen verschiedenen 3D-Programmen)

GEODATEN

Geography Markup Language (GML)
GPS Exchange Format (GPX): XML für GPS-Daten
Keyhole Markup Language (KML): Koordinaten-Spezifikation für Google Earth
City Geography Markup Language (CityGML)
OpenStreetMap (OSM)

MULTIMEDIA

MusicXML (Notendaten, aufgeschriebene Musik)
SMIL (zeitsynchronisierte, multimediale Inhalte)
MPEG-7 (MPEG-7 Metadaten)
Laszlo (LZX)

SICHERHEIT

Security Assertion Markup Language (sicherheitsbezogene Informationen beschreiben und übertragen)
XML Signature (XML-Schreibweise für digitale Signaturen)
XML Encryption

INGENIEURWISSENSCHAFTEN

AutomationML, ein Format zur Speicherung von Anlagenplanungsdaten
CAEX, ein Format zur Speicherung hierarchischer Objektinformationen
GSDML, ein Format zur Beschreibung von Automatisierungsgeräten, die mit Profinet kommunizieren können
IODD, ein Format zur Beschreibung von Sensoren und Aktoren

WEITERE

Webservices (z. B. SOAP, WSDL und WS-*)
Einbindung von Java-Code in XML-Dokumente (XSP)
Synchronisation von Kalenderdaten SyncML
mathematische Formeln (MathML)
Repräsentation von Graphen (GraphML)
Verfahren im Bereich des Semantischen Webs (RDF, OWL, Topic Maps, UOML)
Service Provisioning (SPML)
Austausch von Nachrichten (XMPP)
Finanzberichten wie bspw. Jahresabschlüssen (XBRL)
Automobilindustrie (ODX, MSRSW, AUTOSAR-Templates, QDX, JADM)
automatisierter Test z. B. von Schaltkreisen (ATML)
Landwirtschaft (AgroXML)
Verlagswesen (ONIX)
Chemie (CIDX)

XHTML

DEFINITION

Der W3C-Standard Extensible HyperText Markup Language (erweiterbare HTML; Abkürzung: XHTML) ist eine textbasierte Auszeichnungssprache zur Strukturierung und semantischen Auszeichnung von Inhalten wie Texten, Bildern und Hyperlinks in Dokumenten. Es ist eine Neuformulierung von HTML 4.01 in XML: Im Gegensatz zu HTML, welche mittels SGML definiert wurde, verwendet XHTML die strengere und einfacher zu parsende SGML-Teilmenge XML als Sprachgrundlage. XHTML-Dokumente genügen also den Syntaxregeln von XML.

UNTERSCHIEDE ZU HTML

Tags immer klein geschrieben:

Start- und Ende-Tag oder Empty-Element-Tag: ,

Attributwert in Anführungszeichen angeben:

Attributname als -wert angeben: <input type="radio" checked="checked" />

name Attribut wird durch id ersetzt

Vollständige Kopfdaten:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
```

RSS

RSS ist eine seit dem Anfang des Jahres 2000 kontinuierlich weiterentwickelte Familie von Formaten für die einfache und strukturierte Veröffentlichung von Änderungen auf Websites in einem standardisierten Format.

XML-basiert

Quasi-Standard (zusammen mit Atom)

Feeds sind meist in Channels organisiert

Rich Site Summary vs. RDF Site Summary vs. Really Simple Syndication

VERSIONEN

1999: RSS 0.90

Erfinder: Netscape, basiert auf RDF (Resource Description Framework)

2000: RSS 0.91 - 0.94

basiert auf einfachem XML

2000: RSS 1.0

wieder RDF-basiert

2002: RSS 2.0

Basis ist 0.91, nicht RDF-basiert, Quasi-Standard, nicht vollständig abwärtskompatibel

SOFTWARE

Blog-Software, z.B. Wordpress, um einen Feed anzubieten

Firefox für Live-Bookmarks

RSSReader (NewsRob, ...), zum on- und offline lesen

Aggregation mehrere Newsfeeds z.B. per Feedly oder früher (GoogleReader)

AUFBAU (VERSION 2.0)

```
<?xml version="1.0" encoding="utf-8"?>
<rss version="2.0">
    <channel>
        <title>Titel des Feeds</title>
        <link>URL der Webpräsenz</link>
        <description>Kurze Beschreibung des Feeds</description>
        <language>Sprache des Feeds (z. B. "de-de")</language>
        <copyright>Autor des Feeds</copyright>
        <pubDate>Erstellungsdatum("Tue, 8 Jul 2008 2:43:19")</pubDate>
        <image>
            <url>URL einer einzubindenden Grafik</url>
            <title>Bildtitel</title>
            <link>URL, mit der das Bild verknüpft ist</link>
        </image>

        <item>
            <title>Titel des Eintrags</title>
```

VERKNÜPFUNG AUS HTML

Man kann eine RSS-Datei in der HTML-Seite, deren Inhalte sie maschinenlesbar enthält, verknüpfen. Dieses Verfahren wurde für RSS nie spezifiziert, allerdings können fast alle Aggregatorprogramme dadurch selbständig die Adresse des RSS-Feeds eines Webangebots herausfinden (genannt auto-discovery). Moderne Browser ermöglichen es dem Seitenbesucher, den so verknüpften RSS-Feed zu abonnieren. Beispielsweise wird in der Adress- oder Statusleiste des Browser-Fensters eine RSS-Schaltfläche angezeigt.

```
<link rel="alternate" type="application/rss+xml" title="RSS" href="http://www
```

ATOM

Atom Syndication Format (ASF) wird entwickelt, um die Nachfolge von RSS antreten

Atom entstand aus dem Bedürfnis heraus, die Vorteile der unterschiedlichen RSS-Formate in einem neuen Format zusammenzufassen und um neue Elemente zu ergänzen. Dabei haben die Entwickler – in überwiegender Mehrzahl Blogger – ASF auch so gestaltet, um den speziellen Bedürfnissen von Weblogs und Nachrichtenseiten gerecht zu werden. Die maßgeblichen Unterstützer von Atom sind in der Industrievereinigung AtomEnabled Alliance organisiert.

Die aktuelle Version des Atom Syndication Formats ist der IETF-Entwurf vom 11. August 2005, welcher von der IESG im August 2005 als Proposed Standard verabschiedet und im Dezember 2005 als RFC 4287 veröffentlicht wurde. Die meisten größeren Feed-Anbieter arbeiten bereits daran, das

PODCASTS

WAS IST EIN PODCAST?

Podcasting bezeichnet das Anbieten abonnierbarer Mediendateien (Audio oder Video) über das Internet. Das Kofferwort setzt sich zusammen aus der englischen Rundfunkbezeichnung Broadcasting und der Bezeichnung für bestimmte tragbare MP3-Spieler, iPod, mit deren Erfolg Podcasts direkt verbunden sind und die heute stellvertretend für jegliche tragbare MP3-Spieler stehen. Ein einzelner Podcast besteht aus einer Serie von Medienbeiträgen (Episoden), die über einen News Feed (meistens RSS) automatisch bezogen werden können. Alternativ sind Podcasts auch unter dem markenneutralen Begriff Netcast bekannt.

WORAUS BESTEHT PODCAST?

Ein Podcast besteht aus einer Reihe von Medien-Dateien (Audio / Video / Text) und einer XML-Meta-Datei, die diese referenziert.

PODCAST-XML

```
<?xml version="1.0" encoding="utf-8"?>
<rss xmlns:atom="http://www.w3.org/2005/Atom" xmlns:itunes="http://www.itunes
    <channel>
        <link>http://www.YourSite.com</link>
        <language>en-us</language>
        <copyright>&#xA9;2013</copyright>
        <webMaster>your@email.com (Your Name)</webMaster>
        <managingEditor>your@email.com (Your Name)</managingEditor>
        <image>
            <url>http://www.YourSite.com/ImageSize300X300.jpg</url>
            <title>Title or description of your logo</title>
            <link>http://www.YourSite.com</link>
        </image>
        <itunes:owner>
            <itunes:name>Your Name</itunes:name>
            <itunes:email>your@email.com</itunes:email>
        </itunes:owner>
            <itunes:category text="Education">
```

DIE EINZELNEN EPISODEN WERDEN MIT DEN ITEM-TAGS BESCHRIEBEN

TOOLS

Natürlich muss man nicht zwingend das XML selber schreiben, sondern kann das erledigen lassen. Z.B. durch Podcast Generator

DER EIGENE PODCAST

1. Erstelle deine erste Episode in den Formaten: .m4a, .mp3, .mov, .mp4, .m4v, .pdf, and .epub
2. Erstelle einen RSS Feed, der deinen Podcast beschreibt. Dieser muss valides RSS 2.0 sein, die empfohlenen iTunes RSS Tags enthalten und auf deine Episode verweisen.
3. Erstelle ein Cover in der Größe 1400 x 1400 Pixel als JPG oder PNG.
4. Lege die Dateien auf einem öffentlichen Server mit Streaming-Support (byte-range support) ab.
5. Veröffentliche die URL deines Podcasts bei iTunes (durchläuft einen Freigabeprozess)

MEHR INFOS

https://www.apple.com/itunes/podcasts/specs.html
https://odee.osu.edu/resourcecenter/digital-media-production/how-write-podcast-rss-xml

ÜBUNG RSS

PHP: ERSTELLEN EINES RSS 2.0 FEEDS

PHP: AUSLESEN EINES RSS 2.0 FEEDS

JAVASCRIPT: AUSLESEN EINES RSS 2.0 FEEDS (JQUERY)

GEOTAGGING

AGENDA

1. Definition
2. Allgemeines
3. Koordinaten
4. GPS
5. Grundformen
6. Beziehungen
7. GeoJSON

DEFINITION

Unter dem Vorgang der Georeferenzierung, Geokodierung, Geotagging oder Verortung versteht man die Zuweisung raumbezogener Informationen, der Georeferenz, zu einem Datensatz. Der Vorgang spielt eine wichtige Rolle in der Computerkartografie, Fernerkundung und bei Geoinformationssystemen, kommt aber auch Heimanwendungen vor, z. B. der Archivierung von Fotos und Videos.

WARUM???

DATEN (BILDER, VIDEOS, ...) BEKOMMEN DURCH GEODATEN EINE NEUE DIMENSION, LASSEN SICH GENAUER ZUORDNEN UND ANDERS INTERPRETIEREN. Z.B. EINE REISEROUTE LÄSST SICH NACHVOLLZIEHEN UND ANZEIGEN.

ALLGEMEINES

Es gibt im Wesentlichen drei mögliche Gründe, warum man eine Georeferenzierung durchführen möchte:

- Man möchte Daten in ein geodätisches Referenzsystem einpassen, d.h. mit Realweltkoordinaten versehen (geokodieren).
- Man möchte geometrische Verzerrungen in Datensätzen, insbesondere in Bilddaten, eliminieren (rektifizieren).
- Man möchte zwei unterschiedlich orientierte bzw. skalierte Datensätze aneinander anpassen (transformieren).

ARTEN

Folgende Arten der Georeferenzierung sind zu unterscheiden:

- die Zuweisung einer Postanschrift (Adresskodierung)
- die Zuweisung einer Koordinate (Geokodierung, Geocoding, Geotagging, Geo-Imaging)

ADRESSKODIERUNG

Bei der Adresskodierung wird dem raumbezogenen Datensatz eine Postanschrift zugewiesen und damit ein indirekter Raumbezug geschaffen. Mithilfe geokodierter Adressen (das sind Punkte, die sowohl Postanschrift als auch Realweltkoordinaten tragen) lässt sich der direkte Raumbezug der Daten herstellen (Daten ↔ Adresse ↔ Koordinate). Adresspunktdatensätze werden unter anderem von der Katasterverwaltung oder von Navigationsdatensatzherstellern erstellt.

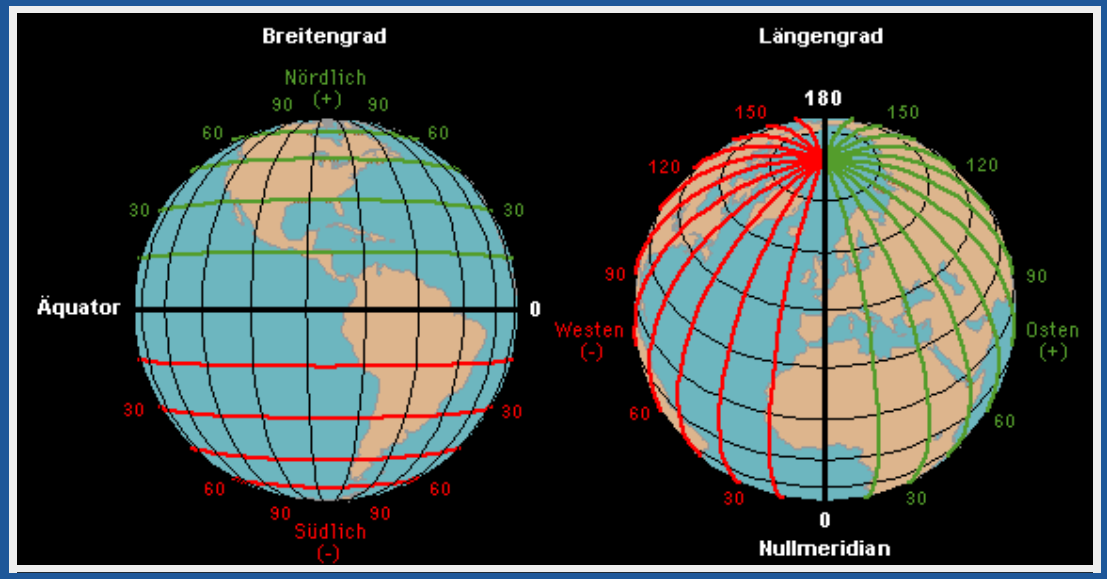
GEOTAGGING (GEOCODING)

Beim Geotagging wird ein raumbezogener Datensatz (z. B. ein Bild, eine Webseite, ein Artikel) mit einer Koordinate versehen. Die Koordinate wird als Tag, Attribut bzw. Metainformation beigefügt. Sie ermöglicht die räumliche Einordnung der Information. Die Daten lassen sich so zum Beispiel in einer digitalen Karte (wie z. B. Google Earth) an der richtigen Stelle platzieren. Im Internet haben sich hierfür die Begriffe Geotagging und Geocoding verbreitet. Das beigefügte Attribut wird entsprechend Geotag oder Geocode genannt. Für die Geokodierung von Webseiten gibt es in HTML z. B. das Meta-Element geo.

KOORDINATEN

Mit den geographischen Koordinaten (geographische Breite und geographische Länge) lässt sich die Lage eines Punktes auf der Erde beschreiben. Die Erde wird dabei in 360 Längengrade und 180 Breitengrade aufgeteilt. Längengrade verlaufen durch Nord- und Südpol, Breitengrade parallel zum Äquator.

Geographische Koordinaten werden häufig im Sexagesimalsystem angegeben, d. h. 1 Grad ist unterteilt in sechzig Minuten, 1 Minute wiederum in 60 Sekunden.


Breitengrad
Nördlich
(+)
90
90
60
60
30
30
Äquator
0
30
30
60
60
90
90
Südlich
(-)
Längengrad
180
150
150
120
120
90
90
Westen
(-)
Osten
(+)
60
60
30
30
0
Nullmeridian

GPS

WIE FUNKTIONIERT GPS?

TOTAL EINFACH, ODER?

$$\begin{matrix} (x_1 - x_0)^2 + (y_1 - y_0)^2 + (z_1 - z_0)^2 = [c (t_1 - t_0)]^2 \quad (1)\\ (x_2 - x_0)^2 + (y_2 - y_0)^2 + (z_2 - z_0)^2 = [c (t_2 - t_0)]^2 \quad (2)\\ (x_3 - x_0)^2 + (y_3 - y_0)^2 + (z_3 - z_0)^2 = [c (t_3 - t_0)]^2 \quad (3)\\ (x_4 - x_0)^2 + (y_4 - y_0)^2 + (z_4 - z_0)^2 = [c (t_4 - t_0)]^2 \quad (4)\\ \end{matrix}$$

FRAGEN WIR DOCH JEMAND, DER SICH DAMIT AUSKENNT

Wie funktioniert GPS

ZUSAMMENGEFASST

- 4 Satelliten werden benötigt, wenn die Zeit unbekannt ist
- 3 Satelliten wenn die Zeit bekannt ist
- Dann kann die Position trianguliert werden

ABER

- Die Welt ist 3D
- Genauigkeit steht und fällt mit der Anzahl der Satelliten
- Das Militär hat Einfluß auf die Genauigkeit
- Alternativen: GLONASS und Galileo

GRUNDFORMEN

Alle existierenden zu verortenden Elemente lassen sich mit folgenden Grundformen abbilden:

- Punkt
- Multi-Punkt
- Linie
- Multi-Linie
- Fläche / Polygon
- Multi-Polygon

BEZIEHUNGEN

Zwischen den Grundformen existieren folgende Beziehungen:

- A ist unabhängig von B
- A berührt B
- A enthält B
- A schneidet B
- A befindet sich innerhalb von B

GEOJSON

GeoJSON ist ein offener Standard, um geometrische Formen als JSON darzustellen und wird in vielen Anwendungen und dem Internet benutzt.

```
{
    "type": "Point",
    "coordinates": [30, 10]
}
{
    "type": "LineString",
    "coordinates": [[30, 10], [10, 30], [40, 40]]
}
{
    "type": "Polygon",
    "coordinates": [[[35, 10], [45, 45], [15, 40], [10, 20], [35, 10]], [[20,
}
{
    "type": "MultiPoint",
    "coordinates": [[10, 40], [40, 30], [20, 20], [30, 10]]
}
```

NAVIGATION

Ein Navigationssystem ist ein technisches System, das mit Hilfe von Positionsbestimmung (Satellit, Funk, GSM bzw. inertes oder autonomes System) und Geoinformationen (Topologie-, Straßen-, Luft- oder Seekarten) eine Zielführung zu einem gewählten Ort oder eine Route unter Beachtung gewünschter Kriterien ermöglicht.

Neben der Bestimmung der Position wird hier ein Computersystem benötigt, welches in der Lage ist, den kürzesten Pfad zu berechnen.

Zusatzanwendungen für Navigationssysteme:

- TMC oder TMCpro
- POIs
- Multimedia-Anwendungen
- Geocaching

GEODATEN IN BILDERN / VIDEOS

Unter Geotagging, auch Geocoding oder Geo-Imaging, versteht man bei fotografischen Aufnahmen die Zuordnung von geographischen Koordinaten. Als Punkte in einer elektronischen Karte lassen sich die so georeferenzierten Bilder anschließend leichter suchen und auswählen.

Eingesetzt wird die Foto-Verortung in der Raumplanung und dem Tourismus, der Umweltplanung, dem Verkehr und dem Katastrophenschutz. Ein häufiges und gängiges Anwendungsbeispiel ist das zeitsparende Illustrieren eines in einem Stadtplan festgelegten Stadtrundganges mit Bildern von sehenswerten Baudenkmälern

Die Geodaten lassen sich mit einer GPS-fähigen Kamera direkt bei der Aufnahme oder nachträglich in den Metadaten

GEODATEN IN HTML

Als Metatag zur Verortung von Angeboten, z.B. einer Firmenwebseite

```
<meta name="geo.region" content="DE-BW" />
<meta name="geo.placename" content="Stuttgart" />
<meta name="geo.position" content="51.165691;10.451526" />
<meta name="ICBM" content="51.165691, 10.451526" />
```

Als Geo Mikroformat zum Auslesen aus Webseiten

```
<span class="geo">
    <span class="latitude">50.167958</span>;
    <span class="longitude">-97.133185</span>
</span>
```

GEODATEN IN HTML

Als RDF Metadaten für das semantische Web

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:geo="h
```

GEODATEN IN JAVASCRIPT

Zugriff über die Browser-API auf den Standort des Benutzers

So sieht es aus...

ANWENDUNGEN

- Google Maps, Google Earth, Google Street View
  Einbinden auf der Webseite
- SMS, Hangout, Whatsapp, ...
  Versand des aktuellen Standorts
- Flickr, Panoramio, ...
  Anzeigen wo Bilder aufgenommen wurden
- Wikipedia
  Auffinden von Artikeln über eine Karte
- Google, Bing, ...
  Relevante (nahe) Suchergebnisse werden bevorzugt angezeigt (Pubs in der Nähe)
- Geoblogging / Social Media
  Bezug des veröffentlichten Inhalts zu einem Standort

LOCATION BASED SERVICES

Standortbezogene Dienste (engl. Location-based Services (LBS), auch: Location Dependent Services (LDS)) sind mobile Dienste, die unter Zuhilfenahme von positionsabhängigen Daten dem Endbenutzer selektive Informationen bereitstellen oder Dienste anderer Art erbringen.

DIE ERDE IST RUND...

ABER DIE KOORDINATEN HABEN GRENZEN

WAS PASSIERT ALS Z.B. MIT EINEM POLYGON, DASS ÜBER DIESE GRENZE RAGT?

Meist wird ein solches in mehrere geteilt

ÜBUNG: GOOGLE MAP BUILDER

Erstellt einen schönen Kartenausschnitt für die SAE

http://googlemapbuilder.mynameisdonald.com/

APIS

AGENDA

1. Definition - API
2. Daten
3. Kommunikation
4. OAuth
5. Die eigene API
6. Famous APIs
7. Geotagging

DEFINITIONEN

WAS IST EINE API?

API... WTF?

Active Pharmaceutical Ingredient: englische Bezeichnung für einen Wirkstoff in einem Medikament

Adaptive Planungsintelligenz: mathematische Methoden für industrielle Planungsaufgaben

African Plants Initiative: ein Projekt zur Förderung botanisch-taxonomischer Arbeit in Afrika

anonima petroli italiana SpA: eine italienische Ölgesellschaft

Arbeiterkommunistische Partei Irans: eine iranische Exilpartei

Api: ein Berg im Westen Nepals

...

APPLICATION PROGRAMMING INTERFACE

ENGLISCH FÜR PROGRAMMIERSCHNITTSTELLE IN DER INFORMATIK

DEFINITION

Eine Programmierschnittstelle ist ein Programmteil, der von einem Softwaresystem anderen Programmen zur Anbindung an das System zur Verfügung gestellt wird. Oft wird dafür die Abkürzung API (engl. application programming interface, dt. „Schnittstelle zur Anwendungsprogrammierung“) verwendet. Im Gegensatz zu einer Binärschnittstelle (ABI) definiert eine API nur die Programmanbindung auf Quelltextebene.

FUNKTIONSORIENTIERT

Funktionsorientierte Programmierschnittstellen kennen nur Funktionen mit oder ohne Rückgabewert als Mittel der Kommunikation. Dabei wird fast immer das Konzept der Handles verwendet. Man ruft eine Funktion auf und bekommt ein Handle zurück. Mit diesem Handle lassen sich weitere Funktionen aufrufen, bis abschließend das Handle wieder geschlossen werden muss. Das BIOS eines Personal Computer ist die älteste Programmierschnittstelle für diesen Rechnertyp.

DATEIORIENTIERT

Dateiorientierte Programmierschnittstellen werden über die normalen Dateisystemaufrufe open, read, write und close angesprochen. Sollen Daten an ein Objekt gesendet werden, werden diese mit write geschrieben, sollen welche empfangen werden, werden sie mit read gelesen. Unter UNIX ist dieses Prinzip bei der Ansteuerung von Gerätetreibern weit verbreitet.

OBJEKTORIENTIERT

Objektorientierte Programmierschnittstellen verwenden Schnittstellenzeiger und sind damit deutlich flexibler als die funktionsorientierten Programmierschnittstellen. Häufig wird eine Typbibliothek mitgegeben.

PROTOKOLLORIENTIERT

Protokollorientierte Programmierschnittstellen sind unabhängig vom Betriebssystem und der Computerhardware. Allerdings muss das Protokoll stets neu implementiert werden. Um diesen Aufwand zu minimieren, wird die protokollorientierte Schnittstelle durch eine funktions- oder interfaceorientierte Schnittstelle gekapselt. Man kann hier weiterhin zwischen allgemeinen (z. B. SOAP) und anwendungsspezifischen (z. B. SMTP) Protokollen unterscheiden.

WOZU NE API?

ZUGÄNGLICH MACHEN VON FUNKTIONALITÄT ÜBER EINEN STANDARDISIERTEN WEG

WELCHE APIS KENNT IHR?

TWITTER JAVA ECLIPSE

YOUTUBE GOOGLE MAPS

AMAZON WINDOWS PHP

PYTHON ZEND FRAMEWORK

OPEN STREETMAP ADOBE AIR

GOOGLE GEARS FACEBOOK

ANDROID FLASH JQUERY

"INTERNE" APIS

Jede Programmiersprache und jedes Framework besitzen eine Dokumentation, in welcher der Zugriff auf Funktionen der Programmiersprache / des Frameworks erklärt werden.

In Objekt-orientierten Programmiersprachen (z.B. Java, PHP) nennt man das API.

"EXTERNE" APIS

Gerade im Bereich des Internets werden die „externe“ API's (Google Maps, Flickr, Facebook, ...) immer wichtiger. Hierbei handelt es sich entweder um Webservices (SOAP / Rest) oder um externe Bibliotheken, die eingebunden werden. Hierbei sind vor allem Bibliotheken für Javascript von Interesse.

Hierbei wird meist ein sog. API-Key benötigt.

API KEY

An application programming interface key (API key) is a code passed in by computer programs calling an API (application programming interface) to identify the calling program, its developer, or its user to the Web site. API keys are used to track and control how the API is being used, for example to prevent malicious use or abuse of the API (as defined perhaps by terms of service).

The API key often acts as both a unique identifier and a secret token for authentication, and will generally have a set of access rights on the API associated with it.

API keys can be based on the UUID system (e.g. 550e8400-e29b-41d4-a716-446655440000) to ensure they will be unique to each user.

DATEN

WELCHE DATENAUSTAUSCHFORMATE KENNT IHR?

Text

XML

Html

JSON

Binärdaten

TEXT

Einfachstes Format

Quasi schon antik ;)

Strukturierung schwierig => CSV

Kaum overhead

Encoding

XML

Strukturierter Text

Viel overhead

Eigene Tags möglich

Muss bestimmten Regeln entsprechen

Lange Zeite der quasi Standard

XML - BEISPIEL

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<verzeichnis>
     <titel>Wikipedia Städteverzeichnis</titel>
     <eintrag>
          <stichwort>Genf</stichwort>
          <eintragstext>Genf ist der Sitz von ...</eintragstext>
     </eintrag>
     <eintrag>
          <stichwort>Köln</stichwort>
          <eintragstext>Köln ist eine Stadt, die ...</eintragstext>
     </eintrag>
</verzeichnis>
```

JSON

Javascript Object Notation

Kurzschreibweise für Javascript Objekte

Sehr einfach

Kaum overhead

Native Unterstützung in vielen Programmiersprachen

JSON - BEISPIEL

```
{
    "titel": "Wikipedia Städteverzeichnis",
    "eintrag": [
        {
            "stichwort": "Genf",
            "eintragstext": "Genf ist der Sitz von ..."
        },
        {
            "stichwort": "Köln",
            "eintragstext": "Köln ist eine Stadt, die ..."
        }
    ]
}
```

UND WAS IST JETZT DAS BESTE?

Es kommt darauf an... ;)

Text bei einfachen Funktionalitäten

JSON mittlerweile quasi Standard

KOMMUNIKATION

WARUM BRAUCHT MAN EINEN STANDARD?

VERSTEHT EIN NORWEGER EINEN GRIECHEN? ;)

Kommunikation unabhängig von der Programmiersprache soll ermöglicht werden

WEBSERVICES

Das World Wide Web Consortium definiert die Bereitstellung eines Webservices als Unterstützung zur Zusammenarbeit zwischen verschiedenen Anwendungsprogrammen, die auf unterschiedlichen Plattformen und/oder Frameworks betrieben werden.

Ein Webservice oder Webdienst ist eine Software-Anwendung, die mit einem Uniform Resource Identifier (URI) eindeutig identifizierbar ist und deren Schnittstelle als XML-Artefakt definiert, beschrieben und gefunden werden kann. Ein Webservice unterstützt die direkte Interaktion mit anderen Software-Agenten unter Verwendung XML-basierter Nachrichten durch den Austausch über internetbasierte Protokolle.

WTF...?

Schnittstelle via Netzwerk ansprechbar

Externe Funktionalität

Kommunikation meist via XML, es gibt aber auch andere Formate

SOA in modernen Systemen

SOAP

SOAP (ursprünglich für Simple Object Access Protocol) ist ein Netzwerkprotokoll, mit dessen Hilfe Daten zwischen Systemen ausgetauscht und Remote Procedure Calls durchgeführt werden können. SOAP stützt sich auf XML zur Repräsentation der Daten und auf Internet-Protokolle der Transport- und Anwendungsschicht (vgl. TCP/IP-Referenzmodell) zur Übertragung der Nachrichten. Die gängigste Kombination ist SOAP über HTTP und TCP.

Die Abkürzung SOAP wird offiziell seit Version 1.2 nicht mehr als Akronym gebraucht, da es erstens (subjektiv) keineswegs einfach (Simple) ist und zweitens nicht nur dem Zugriff auf Objekte (Object Access) dient.

SOAP

Meist entspricht ein Serviceendpunkt einer URI, die mehrere Services anbietet. Diese werden per WSDL bekannt gegeben.

Die Web Services Description Language (WSDL) ist eine plattform-, programmiersprachen- und protokollunabhängige Beschreibungssprache für Netzwerkdienste (Webservices) zum Austausch von Nachrichten auf Basis von XML.

SOAP NACHRICHT

```
<?xml version="1.0"?>
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope">
   <s:Header>
       <m:RequestID xmlns:m="http://www.lecture-db.de/soap">a3f5c109b</m:Requ
   </s:Header>
   <s:Body>
       <m:DbResponse xmlns:m="http://www.lecture-db.de/soap">
           <m:title value="DOM, SAX und SOAP">
               <m:Choice value="1">Arbeitsbericht Informatik</m:Choice>
               <m:Choice value="2">Seminar XML und Datenbanken</m:Choice>
           </m:title>
       </m:DbResponse>
   </s:Body>
</s:Envelope>
```

REST

Representational State Transfer (mit dem Akronym REST) bezeichnet ein Programmierparadigma für Webanwendungen. Es gibt keine explizite Norm, daher gehen die Vorstellungen, was REST ist, auseinander. I m Grunde bezeichnet REST die Idee, dass eine URL genau einen Seiteninhalt als Ergebnis einer serverseitigen Aktion (etwa das Anzeigen einer Trefferliste nach einer Suche) darstellt, wie es der Internetstandard HTTP für statische Inhalte (Permalink) bereits vorsieht, ein Ziel, das für dynamisch erzeugte Seiten mitunter jedoch zusätzlichen Aufwand erfordert.

PRINZIPIEN VON REST

ADRESSIERBARKEIT

Jeder Dienst, den ein REST-konformer Server zur Verfügung stellt, hat eine eindeutige Adresse, den Uniform Resource Identifier (URI). Diese „Straße und Hausnummer im Netz“ durch einen URI stellt sicher, dass das Angebot eines Webservices auf einfache, standardisierte Art und Weise einer Vielzahl von Anwendungen (Clients) zur Verfügung steht. Eine konsistente Adressierbarkeit erleichtert es außerdem, einen Webservice als Teil eines Mashups weiterzuverwenden.

UNTERSCHIEDLICHE REPRÄSENTATIONEN

Die unter einer Adresse zugänglichen Dienste können unterschiedliche Darstellungsformen (Repräsentationen) haben. Ein REST-konformer Server kann je nachdem, was die Anwendung anfordert, verschiedene Repräsentationen ausliefern (z. B. HTML, JSON oder XML). Damit kann der Standard-Webbrowser über die gleiche URI-Adresse sowohl den Dienst, als auch die Beschreibung oder Dokumentation abrufen. Es wird nicht der interne Datenbankeintrag ausgeliefert, sondern das je nach Anfrage evtl. in eine bestimmte Kodierung oder Sprachversion übertragene Dokument.

ZUSTANDSLOSIGKEIT

REST setzt auf ein zustandsloses Client-Server-Protokoll. Dabei enthält jede HTTP-Botschaft alle Informationen, die notwendig sind, um die Nachricht zu verstehen. Deshalb muss weder der Server noch die Anwendung Zustandsinformationen zwischen zwei Nachrichten speichern. Eine derartig strikte Trennung der Zuständigkeiten zwischen Client und Server führt dazu, dass ein REST-konformer Webservice als zustandslos (stateless) bezeichnet werden kann: Jede Anfrage eines Clients an den Server ist in dem Sinne in sich geschlossen, als dass sie sämtliche Informationen über den Anwendungszustand beinhaltet, die vom Server für die Verarbeitung der Anfrage benötigt werden.

Zustandslosigkeit in der hier beschriebenen Form wirkt sich begünstigend auf die Skalierbarkeit eines Webservices aus. Beispielsweise können eingehende Anfragen im Zuge der Lastverteilung unkompliziert auf beliebige Maschinen verteilt werden: Da jeder Request in sich geschlossen ist und Anwendungsinformationen somit ausschließlich auf der Seite des Clients vorgehalten werden, ist auf der Seite des Servers keine Sitzungsverwaltung erforderlich. In der Praxis nutzen jedoch viele HTTP-basierte Anwendungen Cookies und andere Techniken, um Zustandsinformationen zu behalten.

VERWENDUNG VON HYPERMEDIA

Sowohl für Anwendungsinformationen als auch für Zustandsveränderungen werden Hypermedia benutzt: Repräsentationen in einem REST-System sind typischerweise im HTML- oder XML-Format, welche sowohl Informationen als auch Links zu anderen Ressourcen enthalten. Deshalb ist es oftmals möglich, von einer Ressource zu einer anderen zu navigieren, indem man einfach Verknüpfungen folgt, ohne dass dafür Registrierungsdatenbanken oder ähnliche Infrastrukturen erforderlich sind. Diese Verknüpfung von Ressourcen innerhalb einer REST-Architektur wird auch als Verbindungshaftigkeit bezeichnet.

OPERATIONEN

REST-konforme Dienste müssen einige Operationen vorsehen, die auf alle Dienste (Ressourcen genannt) angewendet werden können. HTTP, zum Beispiel, hat eine klar definierte Menge von Operationen, darunter GET, POST, PUT und DELETE.

HTTP schreibt vor, dass GET „sicher“ (englisch safe) sein muss, was bedeutet, dass diese Methode nur Informationen beschafft und keine sonstigen Effekte verursacht. Die Methoden GET, HEAD, PUT und DELETE müssen laut HTTP-Spezifikation idempotent sein, was in diesem Zusammenhang bedeutet, dass das mehrfache Absenden der gleichen Anforderung sich nicht anders auswirkt als ein einzelner Aufruf.

HTTP / REST OPERATIONEN

GET: fordert die angegebene Ressource vom Server an.

POST: fügt eine neue (Sub-)Ressource unterhalb der angegebenen Ressource ein. Da die neue Ressource noch keine URI besitzt, adressiert den URI die übergeordnete Ressource. Als Ergebnis wird der neue Ressourcenlink dem Client zurückgegeben. POST kann im weiteren Sinne auch dazu verwendet werden Operationen abzubilden, die von keiner anderen Methode abgedeckt werden.

PUT: die angegebene Ressource wird angelegt oder dazu geändert. Üblicherweise ohne Seiteneffekte.

DELETE: löscht die angegebene Ressource.

HEAD: fordert Metadaten zu einer Ressource vom Server an.

OPTIONS: prüft, welche Methoden auf einer Ressource zur Verfügung stehen.

PATCH: ein Teil der angegeben Ressource wird geändert, Seiteneffekte erlaubt.

UND WAS JETZT BENUTZEN?

REST

SO FUNKTIONIERT DAS INTERNET

OAUTH

WAS IST OAUTH?

OAuth ist ein offenes Protokoll, das eine standardisierte, sichere API-Autorisierung für Desktop-, Web- und Mobile-Applikationen erlaubt. Es wurde von Blaine Cook und Chris Messina initiiert.

Ein Endbenutzer (User) kann mit Hilfe dieses Protokolls einer Anwendung (Consumer) den Zugriff auf seine Daten erlauben (Autorisierung), die von einer anderen Anwendung (Service) verwaltet werden, ohne alle Details seiner Zugangsberechtigung zur anderen Anwendung (Authentifizierung) preiszugeben. Der Endbenutzer kann so Dritte damit beauftragen und dazu autorisieren, sich von ihnen den Gebrauchswert von Anwendungen erhöhen zu lassen. Typischerweise wird dabei die Übermittlung von Passwörtern an Dritte vermieden.

BEGRIFFE

SERVICE PROVIDER

(deutsch: „Dienstanbieter“) ist eine Website oder ein Web Service, bei dem die Informationen liegen, auf die ein kontrollierter Zugriff erlaubt werden soll. Er hat die volle Kontrolle und Verantwortung für die Implementierung von OAuth. Der Service Provider muss kein Identity Provider (Anbieter für ein zentrales Identitätsmanagement; zum Beispiel OpenID) sein.

USER

(deutsch: „Benutzer“) ist der Eigentümer der Informationen. Ihm soll mit OAuth die Kontrolle über seine Informationen ermöglicht werden. OAuth ist so konzipiert, dass nur die manuelle Freigabe des Users den Zugriff auf seine Daten ermöglicht.

CONSUMER

(deutsch: „Nachfrager“ oder „Konsument“) ist eine Applikation, die Zugriff auf die Informationen eines Users erlangen möchte. Es kann eine Website, ein Desktopprogramm, eine Mobile-Applikation, eine Set-Top-Box und so weiter sein, die in jedem Fall Zugang zum Internet haben muss. Ein Consumer Developer ist der Entwickler der Consumer-Applikation, die mit dem Service Provider interagiert.

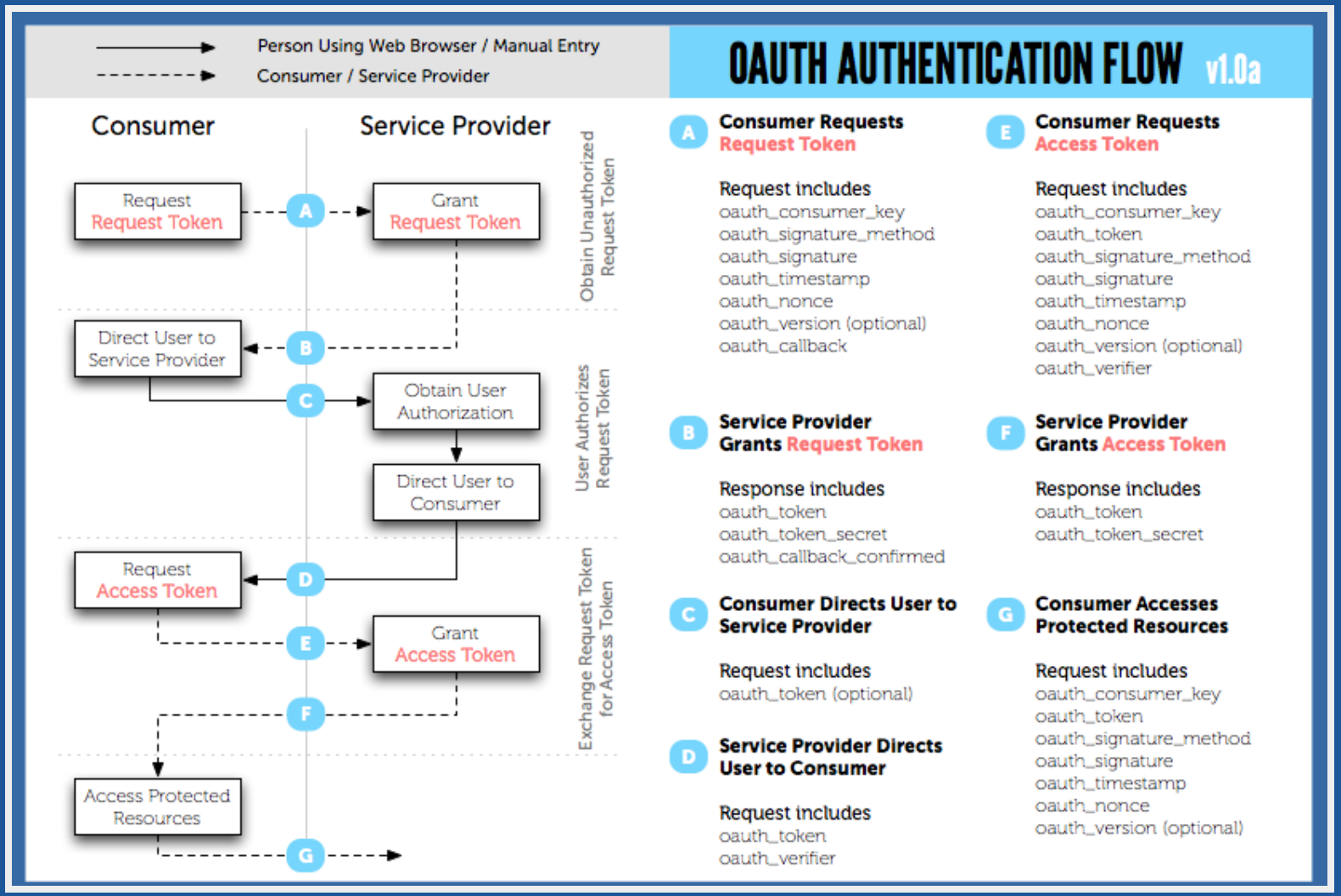
PROTECTED RESOURCES

(deutsch: „geschützte Ressourcen“) sind die Informationen des Users, auf die mit Hilfe von OAuth ein kontrollierter Zugriff erlaubt werden soll. Dabei kann es sich um Daten (Fotos, Dokumente, Adressen und so weiter), Aktivitäten (das Schreiben von Blogbeiträgen, der Transfer von Geld und so weiter) oder den geschützten Zugriff auf eine URL handeln.

TOKEN

werden an Stelle von Benutzername-Passwort-Kombinationen verwendet, um auf Ressourcen zuzugreifen. Ein Token ist meist eine Zeichenkette aus Buchstaben und Zahlen; Sonderzeichen können auch verwendet werden. Um es vor Missbrauch zu schützen soll es schwer zu erraten und passend zu einer Sicherheitsabfrage sein. OAuth unterscheidet zwischen Abfrage-Token und Zugangs-Token.

WIE FUNKTIONIERT OAUTH?



OAUTH 2.0

Einfacher

Mehrere Workflows (Javascript Application)

SSL

Regenerate Token

Aber noch in der Entwicklung

DIE EIGENE API

WANN / WARUM?

Eigenes Framework

Schnittstelle nach aussen

EINE GUTE API

Easy to learn

Easy to use, even without documentation

Hard to misuse

Easy to read and maintain code that uses it

Sufficiently powerful to satisfy requirements

Easy to extend

Appropriate to audience

UND WAS HEISST DAS?

Versionierung

Stabilität

Standards

Dokumentation

FAMOUS APIS

TWITTER

Abfragen von Tweets, Suchen nach Tweets, Schreiben von Tweets, ...

Twitter Dev Pages

Twitter Clients

FACEBOOK

Authentifizierung, Zugriff auf Profildaten & Freunde, Posting von Artikeln, ...

Facebook Dev

GOOGLEMAPS

Suche nach Stichworte, Anzeige von Kartenauschnitten,
Malen auf den Karten, ...

Google Maps API

OSM (OPENSTREETMAP)

Suche nach Stichworten, Zugriff auf versch. Karten / Kartenteile, ...

API Wiki

AMAZON

Zugriff auf den Artikelstamm von Amazon, Preise, Suchen, ...

Affiliate API

Q&A

FRAGEN?

THE END

VIELEN DANK FÜR DIE AUFMERKSAMKEIT UND STIMMUNG!

Made with reveal.js