Mobile Systeme
Grundlagen und Anwendungen standortbezogener Dienste

Location Based Services in the Context of Web 2.0

Department of Informatics - MIN Faculty - University of Hamburg
Lecture Summer Term 2007

Dr. Thilo Horstmann

CLDC

NMEA

MIDP

Google Earth

OpenGIS

SQL

KML

Bluetooth

Mash-Ups

Web 2.0

J2ME

Loxodrome

Euclidean Spaces

RDMS

GPS

PostGIS

GPX

Maps

JSR 179

API

Threads

Polar Coordinates

Today: Building fancy Location Based Web 2.0 Mash-Ups

- Web 2.0: What is this all about?

- (Geo-) Standards and Formats in Web 2.0

- The Web as a Platform:
  RESTful architectures, JSON

Web 2.0:
Just another meaningless marketing buzzword...?

- The Term Web 2.0 was introduced by Tim O'Reilly in 2005:

- „Design Patterns and Business Models for the Next Generation of Software"

  - the term „Web 2.0" around since 2004

- Google finds more than 250,000,000 citations for „Web 2.0"

  - 9.5 million in Sep 2005

- Who knows what Web 2.0 is?

Definitions by Wikipedia
(one of the firsts Web 2.0 application!):

- The transition of web-sites from isolated information silos to sources of content and functionality, thus becoming computing platforms serving web applications to end-users

- a social phenomenon embracing an approach to generating and distributing Web content itself, characterized by open communication, decentralization of authority, freedom to share and re-use, and "the market as a conversation"

- a pronounced distinction between functionality and web technology, enabling significantly easier creation of new business models and processes by using readily available intuitive modular elements

- enhanced organization and categorization of content, emphasizing deep linking

- a rise in the economic value of the Web, possibly surpassing the impact of the dot-com boom of the late 1990s

Web 2.0 by Example

Web 1.0	Web 2.0
DoubleClick	Google AdSense
Ofoto (now Kodak Gallery)	Flickr
Akamai	BitTorrent
mp3.com	Napster
Britannica Online	Wikipedia
personal websites	blogging
page views	cost per click
screen scraping	web services
publishing	participation
content management systems	wikis
directories (taxonomy)	tagging ("folksonomy")
stickiness	syndication

Key Concepts of Web 2.0

- **The Web As Platform**

- Harnessing Collective Intelligence

- Data is the Next Intel Inside

- End of the Software Release Cycle

- **Lightweight Programming Models**

- Software Above the Level of a Single Device

- Rich User Experiences

(Geo-) Standards and Formats in Web 2.0

Keyhole Markup Language: KML

- KML is a file format used to display geographic data in an Earth browser, such as Google Earth, Google Maps, and Google Maps for mobile.

- KML uses a tag-based structure with nested elements and attributes and is based on the XML standard.

- Currently in Version 2.1 (2.2 in beta)

- Submitted by Google to the Open Geospatial Consortium (OGC) as an open standard (KML 3.0)

- OGC expects KML 3.0 to be released as a standard early next year.

- Standardized and interoperable with current GIS standards, e.g. Geographic Markup Language (GML)

- let's walk through some examples ...

The KML object hierarchy

- as KML is based on XML all objects are expressed as tags, e.g.:

- <Point>
  <coordinates>
  -90.8694, 48.2545
  </coordinates>
  </Point>

- compare the KML with the OpenGIS class hierarchy (last lecture) and GML
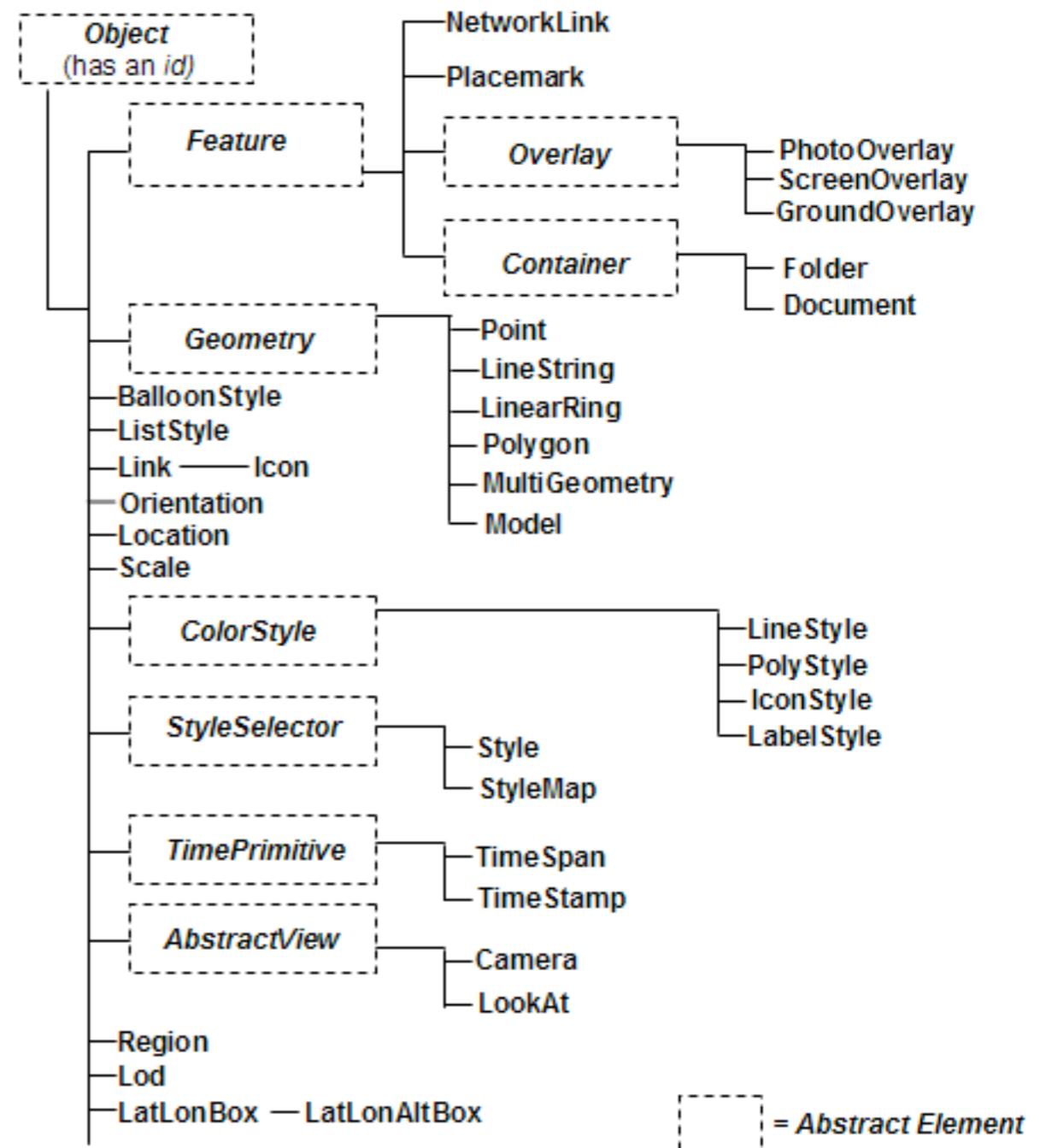
Geography Markup Language

- Based on the OpenGIS abstract specification

  - common model of geography to model and express geographic content

  - proposed by OGC

- Geographic Content is organized by features

  - geometry properties

  - topology properties

  - temporal properties

  - descriptive properties

GML class hierarchy

GeoRSS

- RSS is a family of web feed formats used to publish frequently updated content such as blog entries, news headlines or podcasts.

    - specified using XML

    - current version 2.0 (spec. frozen)

- GeoRSS extends RSS with Geo markup

    - 3 different specifications in use!

        - Otherwise it would be too simple ;-)

    - Of course, different GeoRSS aware applications use different standards :-(

Three ways to specify location information

```
Simple:
   <georss:point>45.256 -71.92</georss:point>

GML:
   <georss:where>
     <gml:Point>
       <gml:pos>45.256 -71.92</gml:pos>
     </gml:Point>
   </georss:where>

W3C (Basic Geo Vocabulary is a RDF vocabulary) :
   <geo:Point>
     <geo:lat>55.701</geo:lat>
     <geo:long>12.552</geo:long>
   </geo:Point>
```

GeoRSS examples

- GeoRSS in Google Maps or Google Earth

  - Example: Flickr feed http://www.flickr.com/services/feeds/geo/DE/Germany/Hamburg?format=rss_200 in Google Maps

  - Add network link in Google Earth

- Convert RSS in GeoRSS feed: reads the entries of an RSS feed and searches the Geonames Database to find a location for the entry text. If a relevant location is found, its latitude and longitude are added to the RSS feed using the GeoRSS encoding.

  - http://www.geonames.org/rss-to-georss-converter.html

The Web as a Platform: RESTful architectures

Some building blocks of Web 2.0 today

- Yahoo!

    - Flickr API Social photo sharing

    - del.icio.us, Social Bookmarking

    - Maps, Search

- Similar Google:

    - Google Data, Gmail Atom Feeds, Google Spreadsheets Data API, YouTube API and another 30 APIs...

- Microsoft, amazon, eBay, technorati, etc. etc

- Thousands more...

Some (Geo) Web 2.0 examples

- http://www.mibazaar.com/new7wonders/

  - Maps, Video, Wikipedia, ad sense

- http://exploreourpla.net/explorer/

  - Different Map overlays

- http://www.81nassau.com/apnews/

  - Geo coded AP news, Google Maps + Yahoo Geocoding

- http://www.flashearth.com/

  - Flash Integration of various map sources

Data integration technologies

- Mash-ups are basically composite applications that use the "programmable Web," including Web services.

- To build Mash-ups need to integrate various services provided by different companies

- The other way around, to provide services that can easily be integrated (mashed-up) by other applications on the web we should provide „Web 2.0" friendly services.

- Among the various integration approaches (most notably SOA architectures based on Web Services, UDDI, WSDL found in the Enterprise context) the Web tend to a more simple architecture pattern

A RESTful Web application architecture

- What is REST?

    - Representational State Transfer

- And what is Representational State Transfer?

- Roy Fielding: "Representational State Transfer is intended to evoke an image of how a well-designed Web application behaves: a network of web pages (a virtual state-machine), where the user progresses through an application by selecting links (state transitions), resulting in the next page (representing the next state of the application) being transferred to the user and rendered for their use."

- REST is a set of guidelines for constructing Web applications. Those guidelines are mostly restrictions.

- goals is to develop Web applications that are more simple and more understandable both by the developers of the application and by its clients and users.

REST restrictions/guidelines

- Everything is a resource. The client/user requests a resource. The Web application delivers (the representation of) a resource. Even submitting information is done via requesting a resource. (For example, the user might submit information by requesting a form (the representation of a form resource), filling out the form, then requesting the updated resource by clicking a link in the form to submit the form.)

- Every resource has a URI. Think of the design process as determining which things to name. URI's are the names.

- The application produces representations, not resources. Furthermore, for any given resource, your application can expose multiple representations, for example, HTML and XML representations and even multiple HTML or multiple XML representations. Think of a representation as the content you generate from a resource; the representation/content is what you send to the client/requestor.

- State is carried on the client, not the server. One implication of this guideline is that each request from a client must contain all the information needed by the application (server) to satisfy the request.

REST - An Architectural Style, Not a Standard

- There is no REST standard and no REST toolkit. It's an architectural style!

  - However, modern Web development frameworks support and simply building RESTful web application (services), e.g. Ruby on Rails (1.2 upwards)

- HTTP's standard PUT/GET/POST/DELETE verbs translate directly to the basic CRUD Create/Read/Update/Delete model operations found in today's web frameworks

  - but do not think of REST in terms of SQL

- E.g. examine the twitter API (http://twitter.com/help/api):

- several representations of the same resource are available (XML, JSON, RSS and Atom). Each request from a client carries all necessary parameters.

Atom Publishing Protocol (APP)

- The 'blogosphere' — the universe of weblogs — is mostly REST-based

- Atom Publishing Protocol for publishing to blogs is considered the canonical RESTful protocol

- Atom also provides the Atom Publishing Protocol (APP), an HTTP-based protocol for publishing and editing web resources.

  - For example, using the APP, you can send an HTTP GET request to ask for a particular resource (such as a feed or entry); the server returns a representation of that resource in the Atom syndication format. You can also create, edit, and delete resources, using POST, PUT, or DELETE, respectively. This use of HTTP requests for a publishing and editing protocol is in the spirit of the REST approach to web service interfaces.

- Note, the APP specification isn't yet complete; it's an emerging standard, currently under development by the IETF.

JSON vs XML

- Another common simplification in Web 2.0 services: JSON

- JavaScript Object Notation

  - A Subset of ECMA-262 Third Edition.

- JSON (like XML) is a data interchange format

- it supports two structures:

  - objects (unordered collections of name/value pairs) and

  - arrays (ordered sequences of values),

- as well as four simple types: strings, numbers, booleans, and null.

JSON example

```
{
    "glossary": {
        "title": "example glossary",
      "GlossDiv": {
            "title": "S",
          "GlossList": {
                "GlossEntry": {
                    "ID": "SGML",
                  "SortAs": "SGML",
                  "GlossTerm": "Standard Generalized Markup Language",
                  "Acronym": "SGML",
                  "Abbrev": "ISO 8879:1986",
                  "GlossDef": {
                        "para": "A meta-markup language",
                      "GlossSeeAlso": ["GML", "XML"]
                   },
                  "GlossSee": "markup"
                }
            }
        }
    }
}
```

The Web as Platform

- Idea: Since data provided by „Web 2.0" application is based on open standards (RSS, Atom, XML) and common conventions it should be easy to mix it all up without much programming

- Example: Take a GeoRSS feed, display it in Google Maps and mix it with Wikipedia content („Mapplet")

- http://maps.google.com/?q=http://api.twibble.de/statuses/friends_timeline/twibble

- Taking this idea even further:

  - Yahoo! pipes and Google Mashup Editor (closed beta as of today)

- „Yahoo!'s new Pipes service is a milestone in the history of the internet."

  - Tim O'Reilly July 2007

This Lecture

- Tim O'Reilly, What Is Web 2.0:

    - http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html

- Roy Thomas Fielding, Architectural Styles and the Design of Network-based Software Architectures

    - http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm

- Küpper, A.: Location-based Services - Fundamentals and Operation, John Wiley & Sons, 2005

    - GML Chapter 3.6

- How to Create a REST Protocol

    - http://www.xml.com/pub/a/2004/12/01/restful-web.html

- JSON vs. XML

    - http://www.infoq.com/news/2006/12/json-vs-xml-debate

Thank you!

Dr. Thilo Horstmann

e-mail: thilo.horstmann@gmail.com
blog: http://www.das-zentralorgan.de