1 Introduction

In the last decade, portals on the WWW have been extensively developed for different applications in e-business, e-commerce, e-government or science and are used as an information and communication base. Recently, due to the progression of web-technologies, new user-friendly features could be added into portals. Among them the incorporation of geographic informations became most important since many relevant data are related to space. In such a way the value and the appropriate visualization of the relevant information as well as the ability to investigate the interrelation between different data could be reached [Bernhardt, 02]. Geo-based web portals combine the features of common portals like searching for keywords or navigating with geo information systems (GIS). For a certain target group they provide a unique presentation of all relevant information within a region or a market, and provide user specific communication tools.

In this context, providing maps is one of the main characteristics of geo-based web portals. Creating a geo-referenced map is a challenging and complex task since all relevant geo data like streets, objects, information of the landscape etc. have to be known and incorporated into the GIS. Therefore, a geo-object in the geo-based web portal has to contain both the geo-data and the content related data. Geo-data describe the structure of an object in space whereas the content related data provide additional attributes associated with the properties of the object (Dickmann & Zehner 2001).

From the point of the architecture, a web-based map application consists of two essential components – the server and the client part. Typically, the server component is responsible for providing geographical information according to the client’s request and returns the data and images suitable for the visualization back to the client application. The client component provides the user with a graphical user interface (GUI). Its main tasks are collecting the user input data and communicating with the server in order to retrieve the information as well as to provide interactive mechanisms for representing analyzing the results. The communication between the client and the sever component via a HTTP port is based on a defined request-response protocol. Typical solutions are implemented using JavaScript, Java applets or Flash and are discussed in more detail in the next section.

In this contribution we describe an alternative approach using the Java Applet Technology, Asynchronous JavaScript and XML (Ajax). It’s application for offering informations of companies operating in the field of food technology and food industry is shown. In such a way it represents a realization of our recently developed concept of a geo-based web portal [Hartmann et al, 05].
2 Techniques for web-based map applications

Numerous commercial and non-commercial solutions for web based map applications can be found on the Internet. They vary in implementation techniques, speed of functioning and ease of viewing and processing the geographical information.

The design and the architecture of the application depend on the kind of geographical data which is to be published. In most of cases, the server component will use vector data (more seldom raster data) in order to generate raster images containing the response. An example of this solution is a GIS MapServer, an application for rendering spatial data (maps, images, and vector data) for the web [Mapserver, 05].

The response generated by the server is displayed by the client component in the web browser. The most common technologies used for providing web user-interface are:
- DHTML – combination of HTML, JavaScript and Cascading Style Sheets (CSS)
- Macromedia Flash
- Java Applet technology

The first technique – DHTML - may be supported by asynchronous data exchange (most common XML data) through the XMLHttpRequest object. This combination is called Ajax.

Each of the techniques has certain advantages and disadvantages. The Table 1 presents the comparison of the mentioned technologies.

Using dynamic HTML for providing interactive interfaces for map web applications is a difficult task and may result in browsers incompatibilities. This technology supported by asynchronous XML-HTTP requests is used in a Google product called Google Maps.

<table>
<thead>
<tr>
<th>Browser plug-in required</th>
<th>HTML, JavaScript</th>
<th>Macromedia Flash</th>
<th>Java Applet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Browser plug-in required</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Compatibility with different browsers</td>
<td>Problematic</td>
<td>100% if plug-in is present</td>
<td>100% if plug-in is present</td>
</tr>
<tr>
<td>Speed of loading the interface</td>
<td>High</td>
<td>Medium</td>
<td>Low (executing) Downloading for the first time only</td>
</tr>
<tr>
<td>Downloading the content from other domains than where the client is hosted</td>
<td>Possible</td>
<td>Impossible</td>
<td>Impossible</td>
</tr>
<tr>
<td>Ease of debugging and maintaining the code</td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td>Possibilities for providing an interactive interface</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Integration in standalone applications (without browser)</td>
<td>Impossible</td>
<td>Impossible</td>
<td>Possible</td>
</tr>
</tbody>
</table>

Macromedia Flash as well as the Java Applet technology allow providing interactive
and reliable graphical user interfaces (GUI’s). The requirement is installing appropriate browser plugging. The popularity of both solutions is significantly growing.

The choice of the technology implies different ways of sharing the processing of tasks by the client and server component and is reflected in the system’s architecture.

3 Architecture and Implementation using Ajax

The main aim of the project was to design and to implement a web based application for publishing the maps and the geo-data, supplying an interactive interface and enabling convenient customization and integration. The designed system was supposed to work with raster map images.

The application consists of client and server components which communicate through a specially defined protocol.

The server component is implemented in the Java Servlets technology, and supplies two essential services:

- Map service – providing map images to the client.
- GeoDB service – providing geographical data to the client.

The architecture of the server component is presented on Figure 1.

The map service is responsible for providing raster map images according to the client’s request. Each image stored in the database is a slice of a bigger raster map and is described by geographical coordinates. In order to use the service the client must send a request to the MapServlet and pass the query in the URL. Three types of requests are defined in the communication protocol allowing to obtain the following data:

- Information about available maps.
- Information about map slices fulfilling the request.
- Images containing map slices.

![Figure 1 Application architecture](image)

The response for the two first requests is an XML data. The result of the third response is an image containing a slice of the map. In the case of an error in the request a XML
data is returned containing the description of the error.

An example of the request for the map service and the response XML message are presented in the following listing 1:

```
http://host/webmap/MapServlet?request=2&mapId=1&bBox=13.386,52.542,13.387,52.543

<?xml version="1.0" encoding="ISO-8859-1"?>
<wmresponse>
  <slices>
    <slice id="6">
      <width>229</width>
      <height>357</height>
      <minx>13.38546951</minx>
      <miny>52.52990689</miny>
      <maxx>13.4023</maxx>
      <maxy>52.5459</maxy>
      <zoomLevel>0</zoomLevel>
    </slice>
  </slices>
</wmresponse>
```

Listing 1

The GeoDB service is used to access the information about various geographical locations as: university institutes, private companies, etc. Two types of request are defined allowing to obtain information about location categories, and locations data. The server response in both cases is a XML data. An example of a request and the response is presented in listing 2.

The architecture of the server implies the design of the client. Significant part of data processing is transferred to the client end. The client engine must be able to make HTTP requests, parse the XML data and download images. The technologies used to implement the client are the Java Applet Technology and the Ajax technique. This approach allows flexible retrieving of the data. Only the necessary information is downloaded by the client and image transfer is optimised. Moreover, using the Java Applet Technology allows creating an interactive component for viewing the maps.

The client engine consists of two parts:
- MapViewer component – implemented as a Java Applet.
- Interface for inputting search query and displaying search results – Ajax technique used for asynchronous communication with the server.

The MapViewer component is a Java Swing application allowing convenient and interactive viewing of the geographical data returned by the server. It displays draggable maps and locations according to user preferences. Each location can be clicked in order to display the details or to link to the associated web site. The component includes an intelligent algorithm for downloading the slices of the map. While dragging the map by the user, the tendency in the movement direction is detected and only the necessary slices are requested from the server. The applet performs data caching to reduce the transfer over the network.
The main tasks of the GUI supported by Ajax technique are collecting the user search queries, requesting the information from the GeoDB service and displaying the results on the web site. The process of obtaining the data from the server runs asynchronously with the web server and ensures high efficiency. After the required data is retrieved, the document content is modified through the Document Object Model (DOM). As a result, the application is more responsive as typical web applications, since the amount of data exchanged between the web browser and web server is significantly reduced and reloading of the page is not necessary.

Additionally, the JavaScript code interacts with the applet and allows the configuration of the applet, finding and displaying the locations on the map. Due to the processing on the client side and taking advantage of the client’s CPU, application provides convenient real-time user interface.

The client enables interactive using of maps like zooming, dragging, map control, as well as customisation and extensibility (icons, coordinates, metadata) and the integration in web-applications. In brief, this web based map technology is supplying a highly interactive interface with a convenient way of integration and customisation.

4 Webportal for companies in Food Industry and Food Technology

For representing our research we are using a data base with data of companies in food production and food technology equipment. A big number of the food equipment companies are producer of image processing solutions. They offer solutions in thermal
imaging (e.g. infrared solutions), object identification, color check, pattern recognition, 
check of number of pieces and in different other quality control possibility.

The data base contains - apart from geo-data - information like postal address, web presentation and keywords.

Most of these companies, especially the food producing enterprises, are located in Germany but some of them are outside of Europe. As an example, we use two maps, one of middle Europe and a second one of Germany. Enterprises outside of the specified region can not be displayed but will be referenced by their data.

The solution we introduce is mainly oriented to show the use of Ajax technology in representing maps. Another project of our research activities in using Ajax technology is dedicated in faster content search solutions. This application can be checked on the internet too. For a final webportal we’ll combine both methods.

5 Conclusions

In this contribution we have developed a client-server architecture for web-based map applications. The approach involving the Java Applet Technology and Asynchronous JavaScript and XML (Ajax) ensures high efficiency of the system performance. Based on this technology, web-based portals will provide a highly interactive and user-friendly tool. Additionally, the designed communication protocol, which combines URL-based requests and HTTP XML responses, implies system flexibility and extensibility. A basic version of this solution is available on the web (http://130.149.197.67:8080/webmap2/).

References


[Xindice, 05] Xindice XML Database, 2005; Link: http://xml.apache.org/xindice/