

Dynamische Karten mit AJAX

Björn Broscheit

15. März 2007

Ausblick Teil I

1 Was ist Mapbuilder

- Entstehung
- So kann's aussehen
- Was kann's?

2 Verwendung

- Model
- View
- Controller
- Beispiel

3 Pläne für die kommende Version (1.5) und die Zukunft

Ausblick Teil II

4 Was ist OpenLayers?

- Entstehung
- Und so kann's aussehen ...

5 Die Features von OpenLayers

- Unterstützte Layers
- Die Kontrollsteuerung
- Was gibts noch?

6 Beispiele für das Einbinden verschiedener Layer

- WMS
- Google Maps
- Nasa WorldWind
- Yahoo! Maps

7 benutzerdefinierte Kontrollsteuerung

8 Outlook

Teil I

Mapbuilder



Was ist Mapbuilder?

- AJAX framework für die Implementierung geographischer Klient-Applikationen in einem Webbrowser
- JavaScript/DHTML, XML/XSL
- Minimale Server seitige Anforderungen (Apache/PHP oder Jakarta Tomcat/J2EE)
- Modulares Design durch Objekt-orientierte Architektur (Model-View-Controller (MVC) design pattern)
- Freie Software (GNU LGPL)

Was ist Mapbuilder?

- AJAX framework für die Implementierung geographischer Klient-Applikationen in einem Webbrowser
- JavaScript/DHTML, XML/XSL
- Minimale Server seitige Anforderungen (Apache/PHP oder Jakarta Tomcat/J2EE)
- Modulares Design durch Objekt-orientierte Architektur (Model-View-Controller (MVC) design pattern)
- Freie Software (GNU LGPL)

Was ist Mapbuilder?

- AJAX framework für die Implementierung geographischer Klient-Applikationen in einem Webbrowser
- JavaScript/DHTML, XML/XSL
- **Minimale Server seitige Anforderungen** (Apache/PHP oder Jakarta Tomcat/J2EE)
- Modulares Design durch Objekt-orientierte Architektur (Model-View-Controller (MVC) design pattern)
- Freie Software (GNU LGPL)

Was ist Mapbuilder?

- AJAX framework für die Implementierung geographischer Klient-Applikationen in einem Webbrowser
- JavaScript/DHTML, XML/XSL
- **Minimale Server seitige Anforderungen** (Apache/PHP oder Jakarta Tomcat/J2EE)
- Modulares Design durch Objekt-orientierte Architektur (Model-View-Controller (MVC) design pattern)
- Freie Software (GNU LGPL)

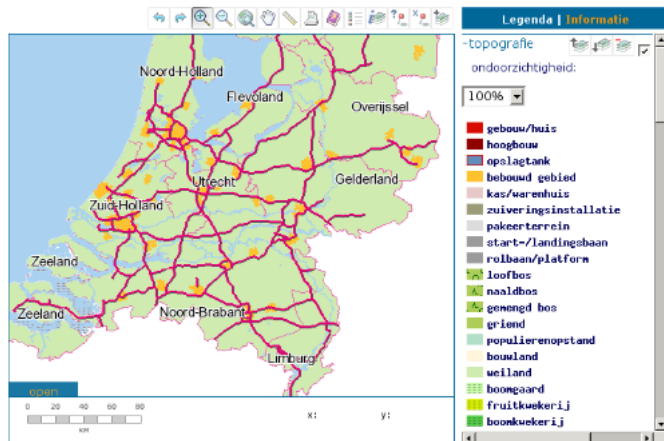
Was ist Mapbuilder?

- AJAX framework für die Implementierung geographischer Klient-Applikationen in einem Webbrowser
- JavaScript/DHTML, XML/XSL
- **Minimale Server seitige Anforderungen** (Apache/PHP oder Jakarta Tomcat/J2EE)
- Modulares Design durch Objekt-orientierte Architektur (Model-View-Controller (MVC) design pattern)
- Freie Software (GNU LGPL)

Entstehung

- 2001: Community Mapbuilder Webseite 2001 Die Idee: Geo-Wiki mit WMS/WFS-T Komponenten in einer einfach zu konfigurieren Anwendung.
- 2003: erster Prototyp welcher direkt mit einer Datenbank kommuniziert.
- 2005: Geoserver, ein freier WMS/WFS-T Server nimmt mapbuilder-lib in die Distribution mit auf.

So kann's ausssehen ...



Mapbuilder in Action!

Was kann's?

- Web Map Services (WMS)
- transaktionale Web Feature Services (WFS-T)
- Web Map Context (WMC) laden, speichern und verändern
- Time series WMS

Was kann's?

- Web Map Services (WMS)
- transaktionale Web Feature Services (WFS-T)
- Web Map Context (WMC) laden, speichern und verändern
- Time series WMS

Was kann's?

- Web Map Services (WMS)
- transaktionale Web Feature Services (WFS-T)
- Web Map Context (WMC) laden, speichern und verändern
- Time series WMS

Was kann's?

- Web Map Services (WMS)
- transaktionale Web Feature Services (WFS-T)
- Web Map Context (WMC) laden, speichern und verändern
- Time series WMS

5 Schritte

- 1 Bibliothek herunterladen und auf einem Webserver installieren.
- 2 HTML für die Webseite erstellen und Elemente identifizieren welche Mapbuilder Inhalte aufnehmen sollen.
- 3 Die JavaScript Bibliothek hinzufügen und initialisieren
- 4 Ein Konfigurationsdokument erzeugen.
- 5 Ein Datendokument erzeugen.

MVC Design Pattern

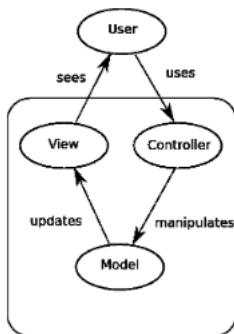


Abbildung: MVC

Model

XML-Dokument in dem werden die Daten verwaltet (z.B. Verweise auf WMS-Dienste).

View

Was der Benutzer sieht.

Controller

Womit der Benutzer die Daten "ändern" kann (XML-Dokument).

MVC Design Pattern

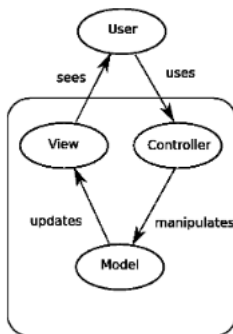


Abbildung: MVC

Model

XML-Dokument in dem werden die Daten verwaltet (z.B. Verweise auf WMS-Dienste).

View

Was der Benutzer sieht.

Controller

Womit der Benutzer die Daten "ändern" kann (XML-Dokument).

MVC Design Pattern

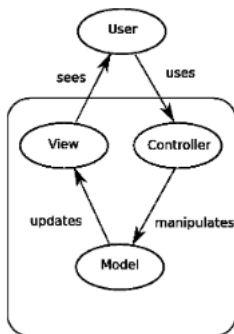


Abbildung: MVC

Model

XML-Dokument in dem werden die Daten verwaltet (z.B. Verweise auf WMS-Dienste).

View

Was der Benutzer sieht.

Controller

Womit der Benutzer die Daten "ändern" kann (XML-Dokument).

Die Daten (Frida-WMS)

```
<ViewContent>
<General>
...
<Window width="600" height="340"/>
<BoundingBox SRS="EPSG:4326" minx="7.92881" miny="52.2131" maxx="8.18349" maxy="52.341"/>
...
</General>
<LayerList>
<Layer queryable="1" hidden="0">
<Server service="OGC:WMS" version="1.1.1" title="Gruenflachen_Osnabrueck">
<OnlineResource xlink:type="simple" xlink:href="http://demo.intevation.org/cgi/frida-wms"/>
</Server>
<Name>gruenflaechen</Name>
<Title>Grünflächen</Title>
<SRS>EPSG:4326</SRS>
<FormatList>
<Format current="1">image/png</Format>
</FormatList>
</Layer>
...
</LayerList>
</ViewContext>
```

Die Daten (Frida-WMS)

```
<ViewContent>
<General>
...
<Window width="600" height="340"/>
<BoundingBox SRS="EPSG:4326" minx="7.92881" miny="52.2131" maxx="8.18349" maxy="52.341"/>
...
</General>
<LayerList>
<Layer queryable="1" hidden="0">
<Server service="OGC:WMS" version="1.1.1" title="Gruenflachen_Osnabrueck">
<OnlineResource xlink:type="simple" xlink:href="http://demo.intevation.org/cgi/frida-wms"/>
</Server>
<Name>gruenflaechen</Name>
<Title>Grünflächen</Title>
<SRS>EPSG:4326</SRS>
<FormatList>
<Format current="1">image/png</Format>
</FormatList>
</Layer>
...
</LayerList>
</ViewContext>
```

Die Daten (Frida-WMS)

```
<ViewContent>
<General>
...
<Window width="600" height="340"/>
<BoundingBox SRS="EPSG:4326" minx="7.92881" miny="52.2131" maxx="8.18349" maxy="52.341"/>
...
</General>
<LayerList>
<Layer queryable="1" hidden="0">
<Server service="OGC:WMS" version="1.1.1" title="Gruenflachen_Osnabrueck">
<OnlineResource xlink:type="simple" xlink:href="http://demo.intevation.org/cgi/frida-wms"/>
</Server>
<Name>gruenflaechen</Name>
<Title>Grünflächen</Title>
<SRS>EPSG:4326</SRS>
<FormatList>
<Format current="1">image/png</Format>
</FormatList>
</Layer>
...
</LayerList>
</ViewContext>
```

Erstellen einer einfachen View zur Präsentation des Frida-WMS

```
<html>
<head>
<script>
var mbConfigUrl='config.xml';
</script>
<script type="text/javascript" src="../../lib/Mapbuilder.js">
</script>
</head>
<body onload="mbDoLoad()">
  <div id="Kartenfenster"></div>
  <div id="Werkzeugleiste"></div>
</body>
</html>
```

Erstellen einer einfachen View zur Präsentation des Frida-WMS

```
<html>
<head>
<script>
var mbConfigUrl='config.xml';
</script>
<script type="text/javascript" src="../../lib/Mapbuilder.js">
</script>
</head>
<body onload="mbDoLoad()">
  <div id="Kartenfenster"></div>
  <div id="Werkzeugleiste"></div>
</body>
</html>
```


Erstellen einer einfachen View zur Präsentation des Frida-WMS

```
<html>
<head>
<script>
var mbConfigUrl='config.xml';
</script>
<script type="text/javascript" src="../../lib/Mapbuilder.js">
</script>
</head>
<body onload="mbDoLoad()">
  <div id="Kartenfenster"></div>
  <div id="Werkzeugleiste"></div>
</body>
</html>
```

Erstellen einer einfachen View zur Präsentation des Frida-WMS

```
<html>
<head>
<script>
var mbConfigUrl='config.xml';
</script>
<script type="text/javascript" src="../../lib/Mapbuilder.js">
</script>
</head>
<body onload="mbDoLoad()">
  <div id="Kartenfenster"></div>
  <div id="Werkzeugleiste"></div>
</body>
</html>
```

Erstellen einer einfachen View zur Präsentation des Frida-WMS

```
<html>
<head>
<script>
var mbConfigUrl='config.xml';
</script>
<script type="text/javascript" src="../../lib/Mapbuilder.js">
</script>
</head>
<body onload="mbDoLoad()">
  <div id="Kartenfenster"></div>
  <div id="Werkzeugleiste"></div>
</body>
</html>
```

Controller

```
<MapbuilderConfig>
<models>
<Context id="Hauptfenster">
<defaultModelUrl>../data/context/frida.xml</defaultModelUrl>
<widgets>
<MapPane id="mainMapWidget">
<htmlTagId>Kartenfenster</htmlTagId>
<mapContainerId>mainMapContainer</mapContainerId>
</MapPane>
...
</widgets>
...
</Context>
</models>
<widgets>
<ZoomIn id="zoomIn">
<buttonBar>Werkzeugleiste</buttonBar>
<targetModel>Hauptfenster</targetModel>
<mouseHandler>mainAoi</mouseHandler>
<class>RadioButton</class>
<selected>true</selected>
<enabledSrc>/images/ZoomInEnable.png</enabledSrc>
<disabledSrc>/images/ZoomInDisable.png</disabledSrc>
<zoomFactor>2</zoomFactor>
</ZoomIn>
...
</widgets>
<skinDir>../../lib/skin/default</skinDir>
<widgetTextUrl>widgetText.xml</widgetTextUrl>
</MapbuilderConfig>
```

Controller

```
<MapbuilderConfig>
<models>
<Context id="Hauptfenster">
<defaultModelUrl>../data/context/frida.xml</defaultModelUrl>
<widgets>
<MapPane id="mainMapWidget">
<htmlTagId>Kartenfenster</htmlTagId>
<mapContainerId>mainMapContainer</mapContainerId>
</MapPane>
...
</widgets>
...
</Context>
</models>
<widgets>
<ZoomIn id="zoomIn">
<buttonBar>Werkzeugleiste</buttonBar>
<targetModel>Hauptfenster</targetModel>
<mouseHandler>mainAoi</mouseHandler>
<class>RadioButton</class>
<selected>true</selected>
<enabledSrc>/images/ZoomInEnable.png</enabledSrc>
<disabledSrc>/images/ZoomInDisable.png</disabledSrc>
<zoomFactor>2</zoomFactor>
</ZoomIn>
...
</widgets>
<skinDir>../../lib/skin/default</skinDir>
<widgetTextUrl>widgetText.xml</widgetTextUrl>
</MapbuilderConfig>
```

Controller

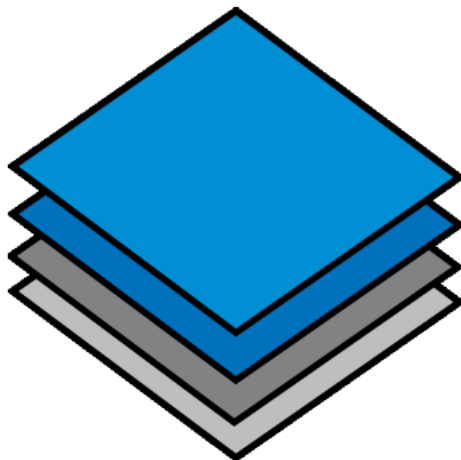
```
<MapbuilderConfig>
<models>
<Context id="Hauptfenster">
<defaultModelUrl>../data/context/frida.xml</defaultModelUrl>
<widgets>
<MapPane id="mainMapWidget">
<htmlTagId>Kartenfenster</htmlTagId>
<mapContainerId>mainMapContainer</mapContainerId>
</MapPane>
...
</widgets>
...
</Context>
</models>
<widgets>
<ZoomIn id="zoomIn">
<buttonBar>Werkzeugleiste</buttonBar>
<targetModel>Hauptfenster</targetModel>
<mouseHandler>mainAoi</mouseHandler>
<class>RadioButton</class>
<selected>true</selected>
<enabledSrc>/images/ZoomInEnable.png</enabledSrc>
<disabledSrc>/images/ZoomInDisable.png</disabledSrc>
<zoomFactor>2</zoomFactor>
</ZoomIn>
...
</widgets>
<skinDir>../../lib/skin/default</skinDir>
<widgetTextUrl>widgetText.xml</widgetTextUrl>
</MapbuilderConfig>
```

Und so sieht's aus ...



Pläne für die kommende Version und die Zukunft

- Unterstützung von weiteren models, widgets, tools
- SOAP/WSDL Unterstützung für andere Webdienste
- OGC Catalogue Klient
- SVG / VML rendering (1.5)
- Open Web Services Context (OWS Context) (1.5)
- Styled Layer Descriptor (SLD) editing (1.5)
- Geographic Really Simple Syndication (GeoRSS) (1.5)
- Verwendung von Kacheln (Tiling)
- Integration von OpenLayers(1.5)



Teil II

OpenLayers

Was ist OpenLayers?

- Freie API um dynamische Karten in beliebigen Webseiten zu integrieren.
- Ausschließlich klientseitiges JavaScript
- Unterstützung von freien und offenen Layern
- Außerdem Unterstützung von restriktiven Layern
- Benutzt Kachelung (Tiling) zur Darstellung
- Freie Software (BSD Lizenz)

Was ist OpenLayers?

- Freie API um dynamische Karten in beliebigen Webseiten zu integrieren.
- **Ausschließlich klientseitiges JavaScript**
- Unterstützung von freien und offenen Layern
- Außerdem Unterstützung von restriktiven Layern
- Benutzt Kachelung (Tiling) zur Darstellung
- Freie Software (BSD Lizenz)

Was ist OpenLayers?

- Freie API um dynamische Karten in beliebigen Webseiten zu integrieren.
- Ausschließlich klientseitiges JavaScript
- Unterstützung von freien und offenen Layern
- Außerdem Unterstützung von restriktiven Layern
- Benutzt Kachelung (Tiling) zur Darstellung
- Freie Software (BSD Lizenz)

Was ist OpenLayers?

- Freie API um dynamische Karten in beliebigen Webseiten zu integrieren.
- Ausschließlich klientseitiges JavaScript
- Unterstützung von freien und offenen Layern
- Außerdem Unterstützung von restriktiven Layern
- Benutzt Kachelung (Tiling) zur Darstellung
- Freie Software (BSD Lizenz)

Was ist OpenLayers?

- Freie API um dynamische Karten in beliebigen Webseiten zu integrieren.
- Ausschließlich klientseitiges JavaScript
- Unterstützung von freien und offenen Layern
- Außerdem Unterstützung von restriktiven Layern
- Benutzt Kachelung (Tiling) zur Darstellung
- Freie Software (BSD Lizenz)

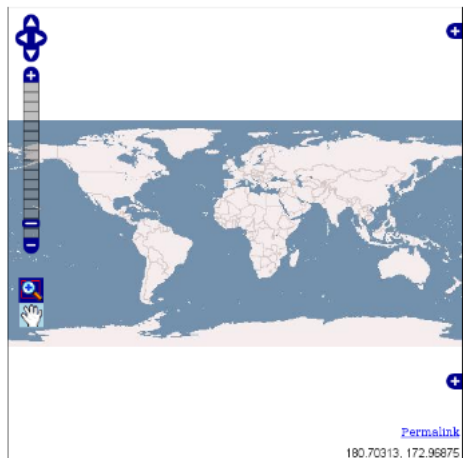
Was ist OpenLayers?

- Freie API um dynamische Karten in beliebigen Webseiten zu integrieren.
- Ausschließlich klientseitiges JavaScript
- Unterstützung von freien und offenen Layern
- Außerdem Unterstützung von restriktiven Layern
- Benutzt Kachelung (Tiling) zur Darstellung
- Freie Software (BSD Lizenz)

Entstehung

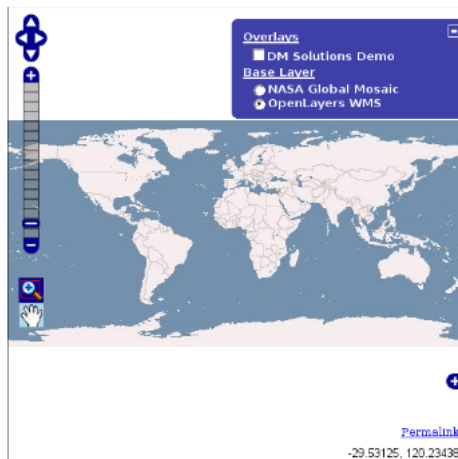
- Begonnen nach der Where 2.0 im Jahr 2005
- Motiviert durch MetaCarta's Geschäftsideen
- Erste Version veröffentlicht vor der Where 2.0 im Jahr 2006

Und so kann's aussehen ...



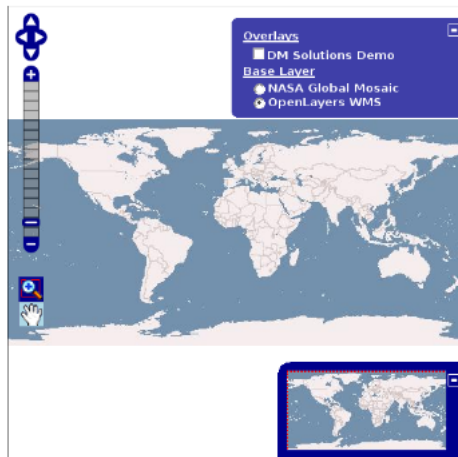
Openlayers in Action?!

Und so kann's aussehen . . .



Openlayers in Action?!

Und so kann's aussehen ...



Openlayers in Action?!

Unterstützte Layers

freie Layer

- OGC WMS
- OGC WFS
- GeoRSS
- CSV
- ka-Map
- WorldWind (*)
- Canvas

restriktive Layer

- Google Maps
- MSN Virtual Earth
- Yahoo! Maps
- Multimap

Die Kontrollsteuerung

- Zoom / Pan
- Zoom Bar
- Mouse controls
- Layer Switcher (aka legend)
- Scale Ratio
- Scale Bar
- Permalink

Was gibts noch?

- Markers
- Popups
- WMS-T

Einbinden von WMS

```
<html>
<head>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var wms = new OpenLayers.Layer.WMS("OpenLayers WMS",
      "http://labs.metacarta.com/wms/vmap0", layers: 'basic');
    map.addLayer(wms);
    map.zoomToMaxExtent();
  </script>
</body>
</html>
```

Einbinden von WMS

```
<html>
<head>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var wms = new OpenLayers.Layer.WMS("OpenLayers WMS",
      "http://labs.metacarta.com/wms/vmap0", layers: 'basic');
    map.addLayer(wms);
    map.zoomToMaxExtent();
  </script>
</body>
</html>
```


Einbinden von WMS

```
<html>
<head>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var wms = new OpenLayers.Layer.WMS("OpenLayers WMS",
      "http://labs.metacarta.com/wms/vmap0", layers: 'basic');
    map.addLayer(wms);
    map.zoomToMaxExtent();
  </script>
</body>
</html>
```

Einbinden von WMS

```
<html>
<head>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var wms = new OpenLayers.Layer.WMS("OpenLayers WMS",
      "http://labs.metacarta.com/wms/vmap0", layers: 'basic');
    map.addLayer(wms);
    map.zoomToMaxExtent();
  </script>
</body>
</html>
```

Einbinden von WMS

```
<html>
<head>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var wms = new OpenLayers.Layer.WMS("OpenLayers WMS",
      "http://labs.metacarta.com/wms/vmap0", layers: 'basic');
    map.addLayer(wms);
    map.zoomToMaxExtent();
  </script>
</body>
</html>
```

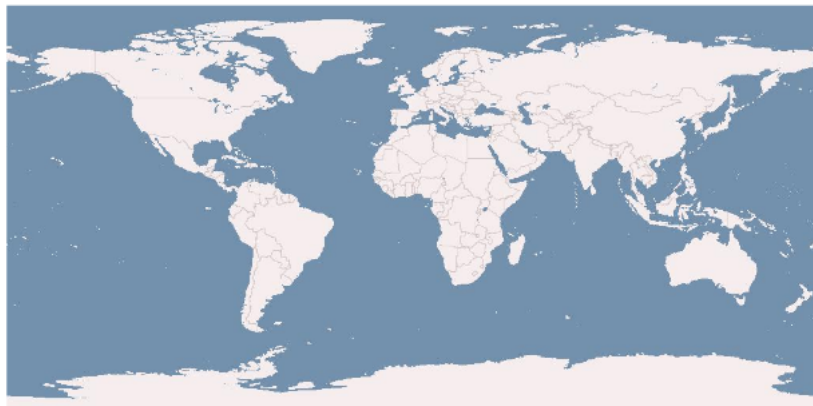
Einbinden von WMS

```
<html>
<head>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var wms = new OpenLayers.Layer.WMS("OpenLayers WMS",
      "http://labs.metacarta.com/wms/vmap0", layers: 'basic');
    map.addLayer(wms);
    map.zoomToMaxExtent();
  </script>
</body>
</html>
```

Einbinden von WMS

```
<html>
<head>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var wms = new OpenLayers.Layer.WMS("OpenLayers WMS",
      "http://labs.metacarta.com/wms/vmap0", layers: 'basic');
    map.addLayer(wms);
    map.zoomToMaxExtent();
  </script>
</body>
</html>
```

Beispiel WMS



Live!?

Einbinden von Google Maps

```
<html>
<head>
<script src="http://maps.google.com/maps?file=api">
</script>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var layer = new OpenLayers.Layer.Google("Google");
    map.addLayer(layer);
    map.zoomToMaxExtent();
  </script>
</body>
```

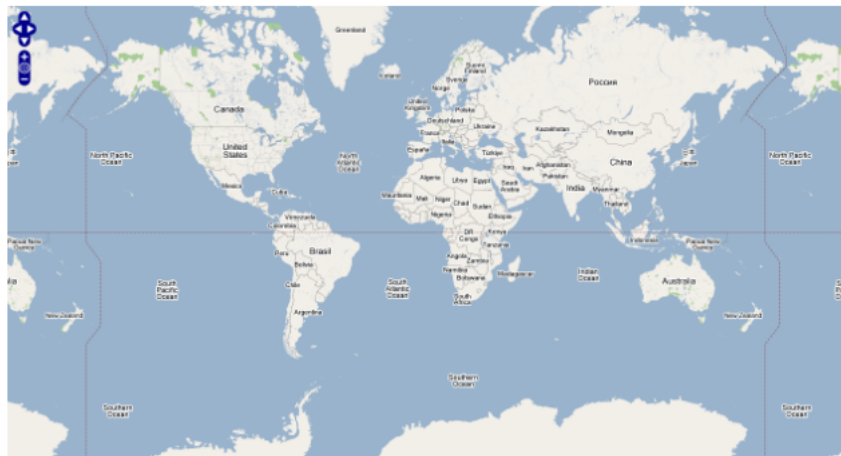
Einbinden von Google Maps

```
<html>
<head>
<script src="http://maps.google.com/maps?file=api">
</script>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var layer = new OpenLayers.Layer.Google("Google");
    map.addLayer(layer);
    map.zoomToMaxExtent();
  </script>
</body>
```


Einbinden von Google Maps

```
<html>
<head>
<script src="http://maps.google.com/maps?file=api">
</script>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var layer = new OpenLayers.Layer.Google("Google");
    map.addLayer(layer);
    map.zoomToMaxExtent();
  </script>
</body>
```

Beispiel Google Maps



Live?! Google Satellit

Einbinden von Nasa WorldWind

```
<html>
<head>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map', {'maxResolution': .28125,
                                           tileSize: new OpenLayers.Size(512, 512)});
    var layer = new OpenLayers.Layer.WorldWind("Bathy",
        "http://worldwind25.arc.nasa.gov/tile/tile.aspx?", 36, 4,
        T:"bmng.topo.bathy.200406");
    map.addLayer(layer);
    map.setCenter(new OpenLayers.LonLat(13.53460,52.43231),6);
  </script>
</body>
</html>
```

Einbinden von Nasa WorldWind

```
<html>
<head>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map', {'maxResolution': .28125,
                                          tileSize: new OpenLayers.Size(512, 512)});
    var layer = new OpenLayers.Layer.WorldWind("Bathy",
      "http://worldwind25.arc.nasa.gov/tile/tile.aspx?", 36, 4,
      T:"bmng.topo.bathy.200406");
    map.addLayer(layer);
    map.setCenter(new OpenLayers.LonLat(13.53460,52.43231),6);
  </script>
</body>
</html>
```

Einbinden von Nasa WorldWind

```
<html>
<head>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map', {'maxResolution': .28125,
                                          tileSize: new OpenLayers.Size(512, 512)});
    var layer = new OpenLayers.Layer.WorldWind("Bathy",
      "http://worldwind25.arc.nasa.gov/tile/tile.aspx?", 36, 4,
      T:"bmng.topo.bathy.200406");
    map.addLayer(layer);
    map.setCenter(new OpenLayers.LonLat(13.53460,52.43231),6);
  </script>
</body>
</html>
```

Beispiel Nasa WorldWind



Live?! Nasa WorldWind

Einbinden von Yahoo! Maps

```
<html>
<head>
<script src="http://api.maps.yahoo.com/ajaxymap?v=3.0&appid=euzuro-openlayers">
</script>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var layer = new OpenLayers.Layer.Yahoo("Yahoo");
    map.addLayer(layer);
    map.zoomToMaxExtent();
  </script>
</body>
</html>
```

Einbinden von Yahoo! Maps

```
<html>
<head>
<script src="http://api.maps.yahoo.com/ajaxymap?v=3.0&appid=euzuro-openlayers">
</script>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var layer = new OpenLayers.Layer.Yahoo("Yahoo");
    map.addLayer(layer);
    map.zoomToMaxExtent();
  </script>
</body>
</html>
```


Einbinden von Yahoo! Maps

```
<html>
<head>
<script src="http://api.maps.yahoo.com/ajaxymap?v=3.0&appid=euzuro-openlayers">
</script>
<script src="http://openlayers.org/api/OpenLayers.js">
</script>
</head>
<body>
  <div id="map"></div>
  <script defer="defer" type="text/javascript">
    var map = new OpenLayers.Map('map');
    var layer = new OpenLayers.Layer.Yahoo("Yahoo");
    map.addLayer(layer);
    map.zoomToMaxExtent();
  </script>
</body>
</html>
```

Beispiel Yahoo! Maps



Live?!

benutzerdefinierte Kontrollsteuerung

```
var controls = [  
    new OpenLayers.Control.PanZoomBar(),  
    new OpenLayers.Control.MouseDefaults(),  
    new OpenLayers.Control.Scale(),  
    new OpenLayers.Control.Permalink()  
];  
  
var map = new OpenLayers.Map('map', controls: controls );
```

benutzerdefinierte Kontrollsteuerung

```
var controls = [  
    new OpenLayers.Control.PanZoomBar(),  
    new OpenLayers.Control.MouseDefaults(),  
    new OpenLayers.Control.Scale(),  
    new OpenLayers.Control.Permalink()  
];  
  
var map = new OpenLayers.Map('map', controls: controls );
```

benutzerdefinierte Kontrollsteuerung

```
var controls = [  
    new OpenLayers.Control.PanZoomBar(),  
    new OpenLayers.Control.MouseDefaults(),  
    new OpenLayers.Control.Scale(),  
    new OpenLayers.Control.Permalink()  
];  
  
var map = new OpenLayers.Map('map', controls: controls );
```

Beachten Sie die Verwendung von `controls`.

Beispiel für eine benutzerdefinierte Kontrollsteuerung



Outlook

- Vektor integration (WFS-T)
- animated zooming
- animated panning
- KML Layer
- Map24 Layer (150 \$ Belohnung!)

Wo liegen sind die Gemeinsamkeiten?

Beide ...

- ... sind Web Mapping AJAX Klienten
- ... unterstützen WMS und WFS-T Services.
- ... haben ein Objektorientiertes Design.
- ... sind gut dokumentiert (JSDoc (Javadoc für javascript))
- ... unterstützen Javascript Kompression
- ... sind Freie Software!

Wo liegen die Stärken?

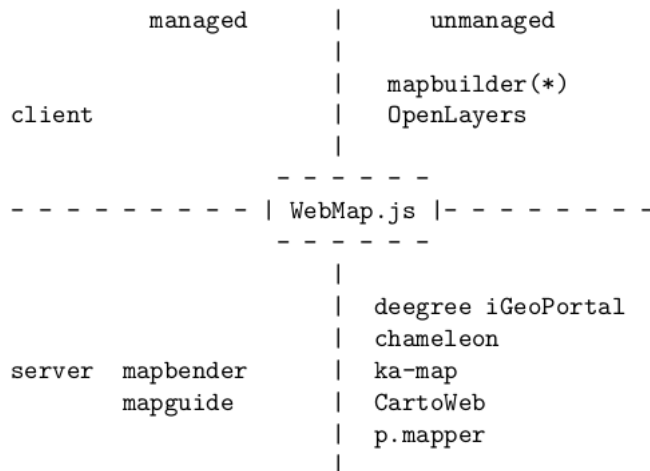
Mapbuilder

- Laden, speichern, ändern, von (Punkt, Linie, Fläche) in einem transaktionalen Web Feature Service (WFS-T)
- Gazetteer widget
- Laden von Web Map Context (WMC) Dokumenten.
- Web Map Context Builder
- Time series WMS

OpenLayers

- Kachelung
- Anzahl unterstützte Layer
- Einfache in beliebigen Webseiten zu integrieren.
- Modulares Bauen der Bibliothek (OpenLayers.js)
- Ausführliches testen mit Unit testing
- Sexy (animated panning + animated zooming)

Zukunft???



Quelle: Choosing a Web Mapping Platform

Das war meine letzte Folie ...

Viel Spaß mit ...!

Wo gibt's mehr Infos?

- <http://openlayers.org>
- <http://communitybuilder.org>