

Google Maps API

VO Softwareentwicklung WS09/10

Michael Haider, Klaus Paccagnel, Andreas Richter

Inhalt

- Einführung
- Funktionsweise
- Ereignisse
- Bedienelemente
- Overlays
- Services

Einführung (1)

API = Application Programming Interface

„Eine Programmierschnittstelle ist eine Schnittstelle, die von einem Softwaresystem anderen Programmen zur Anbindung an das System zur Verfügung gestellt wird.“

Einführung (2)

- Google Maps API
- Einbettung von Google Maps in Websites

- **Java Script-API:** dynamisch
- **Flash-API:** Einbindung in Flash-Anwendungen
- **Static-API:** als Bild über URL-Parameter

Funktionsweise (1)

```
<!DOCTYPE html "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="content-type" content="text/html; charset=utf-8"/>
    <title>Google Maps JavaScript API Example</title>
    <script src="http://maps.google.com/maps?file=api&v=2&key=abcdefghijklmnop&sensor=true_or_false"
      type="text/javascript"></script>
    <script type="text/javascript">

function initialize() {
  if (GBrowserIsCompatible()) {
    var map = new GMap2(document.getElementById("map_canvas"));
    map.setCenter(new GLatLng(37.4419, -122.1419), 13);
  }
}

</script>
</head>
<body onload="initialize()" onunload="GUnload()">
  <div id="map_canvas" style="width: 500px; height: 300px"></div>
</body>
</html>
```

Funktionsweise (2)

API laden

```
<script src="http://maps.google.com/maps?file=api&v=2&key=abcdefg&sensor=true_or_false"  
    type="text/javascript">  
</script>
```

- URL zeigt Speicherort der Java Script-Datei
- Enthält Schlüssel
- Sensor-Parameter für mobile Geräte

```
<div id="map_canvas" style="width: 500px; height: 300px"></div>
```

- Reserviert Platz für Karte (body)

Funktionsweise (3)

Klassenerstellung

```
var map = new GMap2(document.getElementById("map_canvas"));
```

- Grundlegende Klasse GMap2
- Erstellt neue Karte im HTML-Container
- Karte muss initialisiert werden

```
map.setCenter(new GLatLng(37.4419, -122.1419), 13);
```

Funktionsweise (4)

Objektpositionierung

```
var myGeographicCoordinates = new GLatLng(myLatitude, myLongitude)
```

- Platziert Objekt GLatLng auf der Karte
- GLatLng kann mehrere Aufgaben erfüllen
 - Marker zuweisen
 - Bounding Box zuweisen
 - ...

Ereignisse

- In Java Script: Interaktion => Ereignis
- Ereignis-Listener registriert & dokumentiert
=> dazugehöriger Code wird ausgeführt

```
var map = new GMap2(document.getElementById("map"));
map.setCenter(new GLatLng(37.4419, -122.1419), 13);
GEvent.addListener(map, "click", function() {
    alert("You clicked the map.");
});
```

<http://code.google.com/intl/de/apis/maps/documentation/examples/event-simple.html>

Bedienelemente (1)

- Benutzeroberflächenelemente
- Erlauben Interaktion mit der Karte
- Google Maps enthält integrierte Bedienelemente:
 - GLargeMapControl
 - GSmallMapControl
 - GSmallZoomControl
 - GHierarchicalMapTypeControl
 - GOverviewMapControl

Bedienelemente (2)

- Drei Standard-Kartentypen
 - G_NORMAL_MAP
 - G_SATELLITE_MAP
 - G_HYBRID_MAP
 - G_PHYSICAL_MAP
- Eigene Bilder oder Overlays können eingefügt werden

Overlays

- Positionsgebundene Objekte auf der Karte
- Overlay-Typen:
 - Markierungen
 - Linien
 - Polygone
 - Kachel-Overlays (Kartenhintergrund)
 - Infofenster

Services

- Einfügen von Funktionen in die Karte
 - XML-Ausgabe
 - Geocoding
 - Street View
 - Google Earth-Plug Ins
 - Suchfunktion
 - Routing
 - ...