



Bundesamt für
Kartographie und Geodäsie

Nutzung der BKG WebMap in eigenen Anwendungen und Infrastrukturen

Dirk Thalheim

Agenda

1. Grundlagen

1. Webdienste des BKG
2. Mapping im Browser
3. Vorstellung des BKG WebMap Frameworks
4. Einbindung geschützter Dienste

2. Komponenten einer Kartenanwendung am Beispiel

1. Karte
2. Layer
3. Controls

Unterlagen

Folien und Übungen stehen unter:

http://sg.geodatenzentrum.de/web_bkg_webmap/tutorials/workshop.zip



Webdienste des BKG

Bereitstellung von Geobasisdaten über leistungsfähige standardisierte Webdienste

Unterstützte Standards: WMS, WMTS, WFS

Online-Dienste für Geodaten:

WebAtlasDE	Digitale Landschaftsmodelle
Digitale Orthophotos	Verwaltungsgebiete
Digitale Topographische Karten	Geographische Namen
Digitale Höhenmodelle	...






Webdienste des BKG Überblick

<http://www.geodatenzentrum.de>

→ Auskunft über Daten & Dienste
→ Dokumentation Dienste

- Aktuelles
- Auskunft über Daten & Dienste
- Metadaten BKG & Bundesländer
- Aktualitätsübersichten
- Dokumentation Daten
- Dokumentation Dienste
- Testdaten
- Web-Anwendungen
- Online-Shops
- Open Data
- Infos und Hinweise
- Über uns
- Links
- Informationsmaterial
- Glossar
- Seite drucken
















Dokumentation Dienste

Nachfolgend erhalten Sie einen Überblick über unsere grundlegenden Online-Dienste für Geodaten. Ihr konkretes Interesse an der Nutzung von Diensten können Sie in unserem **Dienste-Shop** bekunden. Bitte beachten Sie unsere **Allgemeinen Geschäfts- und Nutzungsbedingungen** und die **Adv-Entgeltrichtlinie**.

Der Name des Dienstes gibt Auskunft über den Dienstyp. Die Bezeichnungen bedeuten:

WMS	Standardisierter <i>Web Map Service</i> nach Spezifikation des <i>Open Geospatial Consortium (OGC)</i> zur Anfrage und Bereitstellung von Karten in Form von Rasterdaten. Ein WMS-Server kann dabei Karten aus Rasterdaten und Vektordaten generieren. Auf der Basis von Vektordaten können auch vorhandene Sachattribute über einen WMS abgefragt werden.
WMTS	Standardisierter <i>Web Map Tile Service</i> nach Spezifikation des <i>Open Geospatial Consortium (OGC)</i> zur Anfrage und Bereitstellung von Karten in Form von kachelbasierten Rasterdaten. Dabei wird die Karte serverseitig in den angebotenen Zoomstufen vorgefertigt und in kleine Kacheln (engl. tiles) zerlegt. Die Anwendung greift performant auf die fertigen Bildkacheln zu.
WFS	Standardisierter <i>Web Feature Service</i> nach Spezifikation des <i>Open Geospatial Consortium (OGC)</i> zur Anfrage und Bereitstellung von Geodaten in Form von objektsstrukturierten Vektordaten (Features). Für die Rücksendung des Anfrageergebnisses wird die XML-basierte Geography Markup Language (GML) eingesetzt.
OLS	Standardisierter <i>Open Location Service</i> nach Spezifikation des <i>Open Geospatial Consortium (OGC)</i> zur Anfrage und Bereitstellung von Geodaten in Form von objektsstrukturierten Vektordaten (Features). Für die Rücksendung des Anfrageergebnisses wird die XML-basierte Geography Markup Language (GML) eingesetzt.
CSW	Standardisierter <i>Web Catalogue Service</i> nach Spezifikation des <i>Open Geospatial Consortium (OGC)</i> und der International Organization for Standardization (ISO) zur Anfrage und Bereitstellung von Informationen (Metadaten) über Geodaten und Geodienste.

Kostenfreie Dienste für Jedermann

Inhalt	Name	Capabilities	Beispiel	Bilder
Digitales Geländemodell 1:1 000 000 in 2 Visualisierungen (Relief, Schummerung)	wms_dgm1000	x	x	
Digitales Geländemodell 1:200 000 in 2 Visualisierungen (Relief, Schummerung)	wms_dgm200	x	x	
DLM1000	wms_dlm1000	x	x	
Digitale topographische Karten 1:200 000, 1:500 000, 1:1 000 000 in abgestimmter Maßstabsfolge	wms_dtk_bkg	x	x	
Digitale topographische Karte 1 : 1 000 000	wms_dtk1000	x	x	
Digitale topographische Karte 1 : 1 000 000 in Grauwerten	wms_dtk1000_grau	x	x	
Digitale topographische Karte 1 : 1000 000 in Palettenfarben (ermöglicht Transparenz des weißen Hintergrunds)	wms_dtk1000_palette	x	x	
Digitale topographische Karte 1:200.000 - 1:1.000.000	wms_dtk_klein	x	x	
Digitale topographische Karte 1 : 200 000	wms_dtk200	x	x	
Digitale topographische Karte 1 : 200 000 in Grauwerten	wms_dtk200_grau	x	x	
Digitale topographische Karte 1 : 200 000 in Palettenfarben (ermöglicht Transparenz des weißen Hintergrunds)	wms_dtk200_palette	x	x	
Digitale topographische Karte 1 : 500 000	wms_dtk500	x	x	
Digitale topographische Karte 1 : 500 000 in Grauwerten	wms_dtk500_grau	x	x	
Digitale topographische Karte 1 : 500 000 in Palettenfarben (ermöglicht Transparenz des weißen Hintergrunds)	wms_dtk500_palette	x	x	
Digitale Übersichtskarte 1 : 2 500 000	wms_d2500	x	x	

Fragen zur Webintegration von Online-Diensten

Integration (Kunde):

Wie integriert man eine Karte oder Geodienste in seinen Webauftritt?

Zugriff (BKG):

Wie können wir geschützte Dienste den Webnutzern unseres Kunden zugänglich machen?



Integration von Webdiensten im Browser

HTML bietet keine native Unterstützung für interaktive Karten

- Erfordert Plug-Ins oder dynamisches HTML
- Programmierkenntnisse nötig
(z.B. in JavaScript, Java, Flash, ...)
- Kenntnis über Schnittstelle des Webdienst

Ziel: mit wenig Aufwand interaktive Karten für unsere Dienste erstellen

Mapping Frameworks für JavaScript

Google Maps API



Leaflet *Leaflet* 

Bing Maps API 

OpenLayers 

ArcGIS API
for JavaScript



...

Frameworks unterscheiden sich in:

Einsatzzweck

Flexibilität

Lizenzmodelle

...

OpenLayers

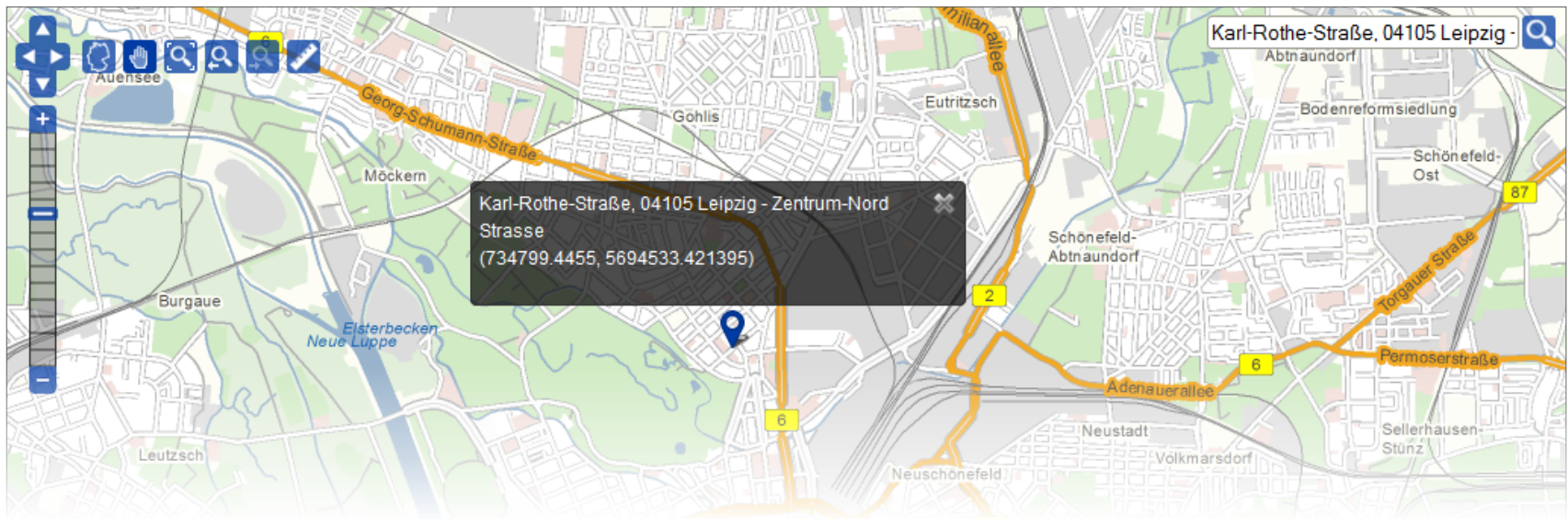
verbreitetes Framework zur Erzeugung interaktiver Karten

- Darstellung von Raster- und Vektordaten
- Unterstützung offener Standards: WMS, WMTS, WFS, GML, ...
- Zahlreiche weitere Formate/APIs: ArcGIS, Bing, GeoJSON, Google, OSM, ...
- Cross-Browser-Kompatibilität
- Erweiterbar
- **Nachteil:**
 - Einarbeitungsaufwand für Schnelleinstieg
 - Entwickler als Zielgruppe

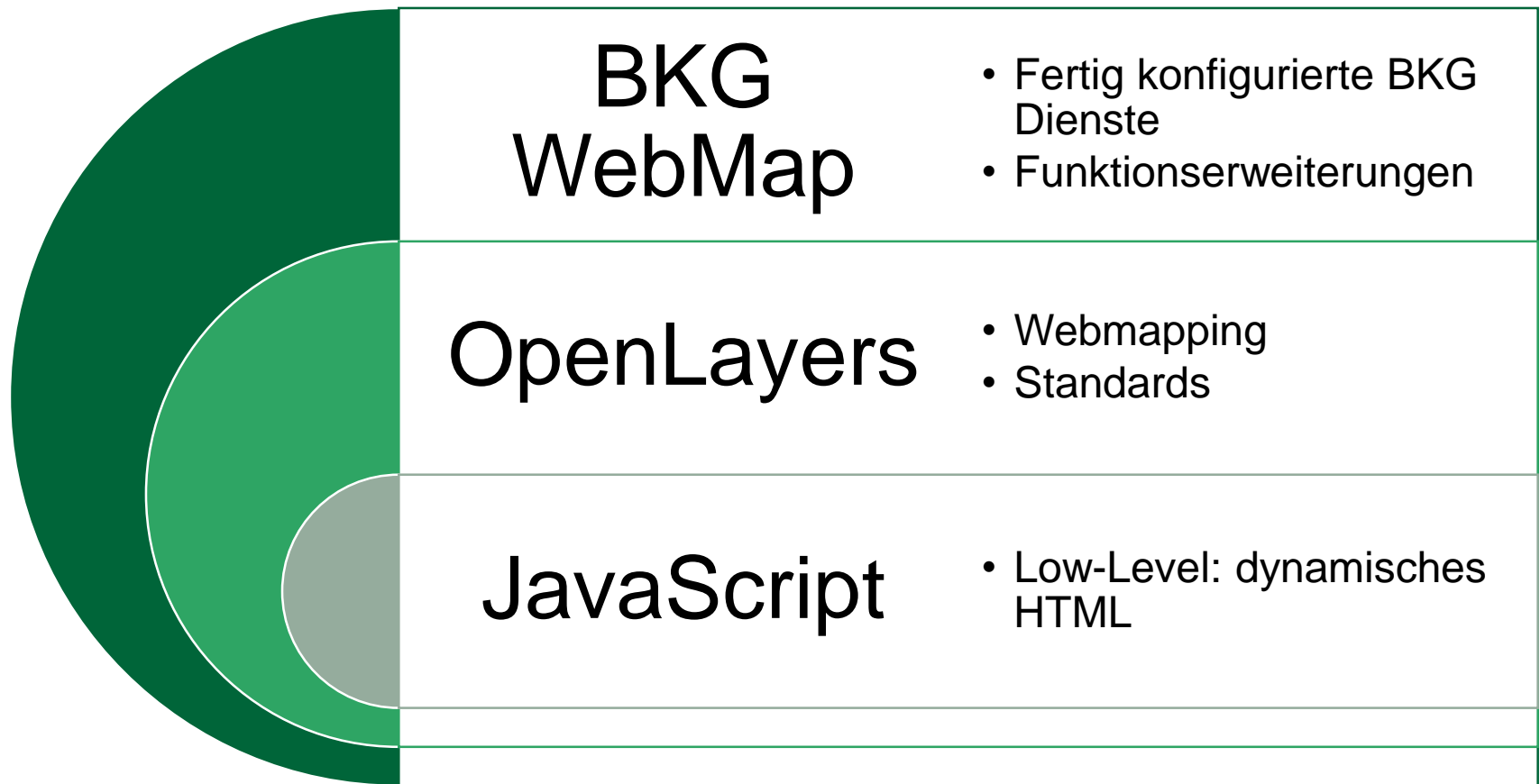


BKG WebMap

- Erweiterung von OpenLayers
- Stellt Standardkonfigurationen für Dienste und Karten bereit
- Ermöglicht einfache Integration der Webdienste des BKG



BKG WebMap Architektur



BKG WebMap einfache Integration

OpenLayers

```
map = new OpenLayers.Map('map', {
  projection: "EPSG:25832",
  resolutions: [
    4891.96981025128,
    2445.98490512564,
    ...
  ]
});

map.addLayer(new OpenLayers.Layer.WMTS({
  name: 'WebAtlasDE',
  url: 'http://sg.geodatenzentrum.de/wmts_webatlasde/tile',
  layer: 'webatlasde',
  attribution: "scopy: GeoBasis-DE / BKG 2015",
  isBaseLayer: true,
  transitionEffect: 'resize',
  style: 'default',
  matrixSet: 'default028mm',
  format: 'image/png',
  requestEncoding: 'REST',
  tileOrigin: new OpenLayers.LonLat(-46133.17, 6301219.54),
  tileSize: new OpenLayers.Size(256,256),
}));

map.addLayer(new OpenLayers.Layer.WMS(
  "Verwaltungsgrenzen",
  "http://sg.geodatenzentrum.de/wms_vg1000",
  {
    layers: '0,1,2,3,4,5',
    format: "image/png32",
    transparent: true
  },
  {
    singleTile: true,
    visibility: true,
    isBaseLayer: false,
    attribution: "scopy: GeoBasis-DE / BKG 2015",
  }
));

map.addControls([...]);

// Karte auf maximale Ausdehnung zoomen
map.zoomToMaxExtent();
```



BKG WebMap

```
// Karte mit Layer definieren
map = new BKGWebMap.MapBuilder()
    .setLayers([type: 'WEBATLASDE.FARBE', {type: 'WMS.VG1000'}])
    .setControls([...])
    .create();

// Karte auf maximale Ausdehnung zoomen
map.zoomToMaxExtent();
```



100

- Schnelleinstieg
In drei Schritten zur Webkarte
- Beispiele
Baukasten für eigene Anwendungen
- API-Dokumentation
Für die, die mehr wollen

Namespaces

Classes

Class: DGM200

Klasse WMS Layer für Digitales Geländemodell mit Gitterweite von 200m

Parameters:

Source: [BKGWebMap/Layer/DGM/DGM200.js](#), line 5

```
<static, constant> DEFAULT_TITLE :string
```

Name des Layers

Source: [BKGWebMap/Layer/DGM/DGM200.js](#), line 37

```
<static, constant> DEFAULT_URL :string
```

Fragen zur Webintegration von Online-Diensten

Integration (Kunde):

Wie integriert man eine Karte oder Geodienste in seinen Webauftritt?

Zugriff (BKG):

Wie können wir geschützte Dienste den Webnutzern unseres Kunden zugänglich machen?

Freischaltung von geschützten Diensten erfolgt über das Dienstleistungszentrum (DLZ): dlz@bkg.bund.de

Und dann?



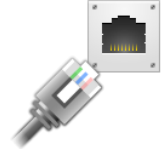
Klassische Authentifizierung

- Nutzeridentifikation

- IP-Adresse des Rechners

(wenn fest und eindeutig)

172.16.254.1



- Nutzeridentifikatoren

(als Parameter und in URL)



https://sg.geodatenzentrum.de/wms_webatlasde__abcdefgh-ijkl-mnop-qrst-uvwxyz123456

- Name + Passwort



- Nicht geeignet für Webintegration
- Benötigen zusätzliche Infrastruktur
- Kunde ist verantwortlich für Schutz gegen unautorisierte Nutzung



Einbindung geschützter Dienste in Webanwendungen

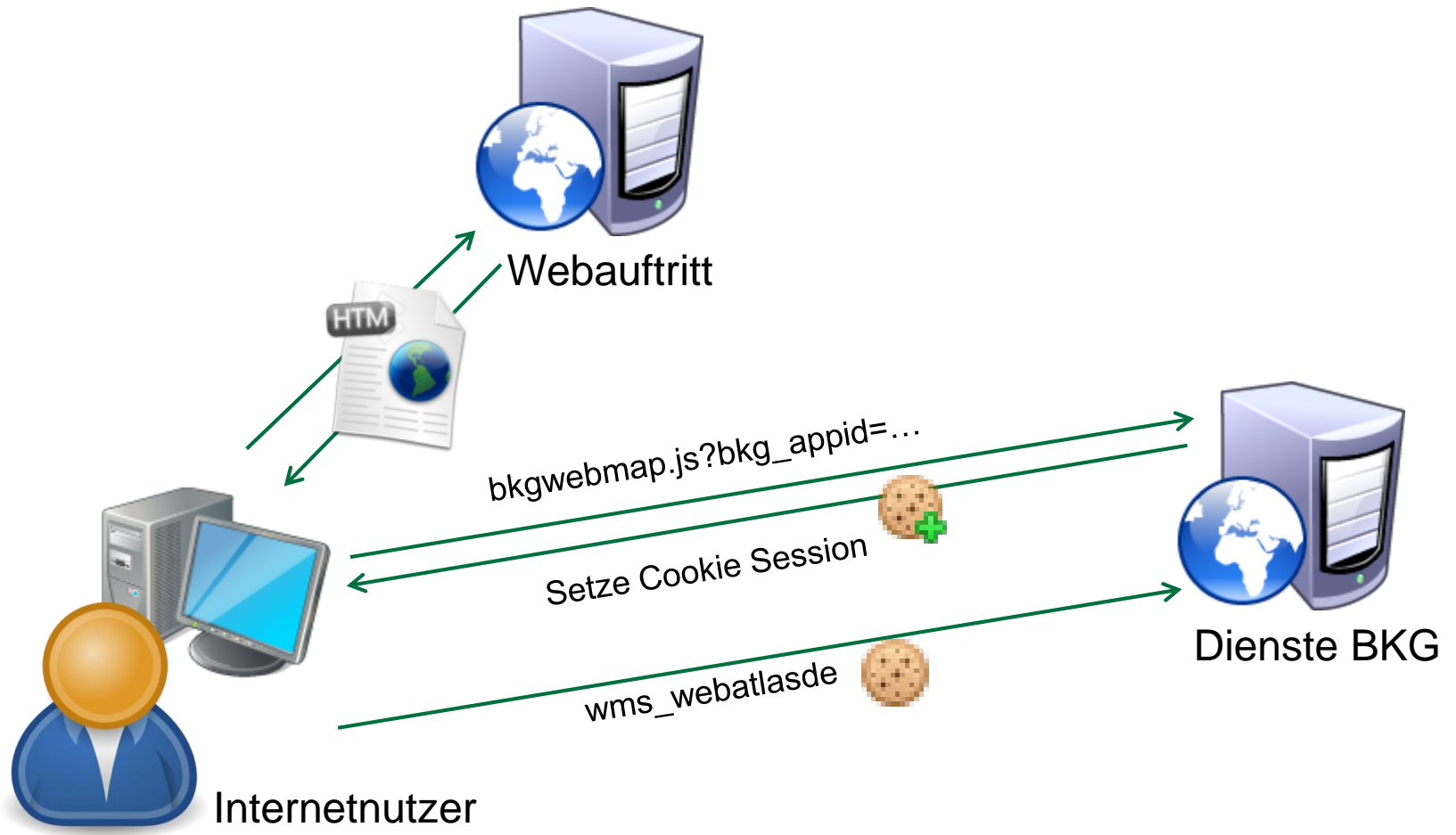
- Cookiebasierte-Session-Authentifizierung
- Initialisierung einer Session im Browser über Applikationsschlüssel



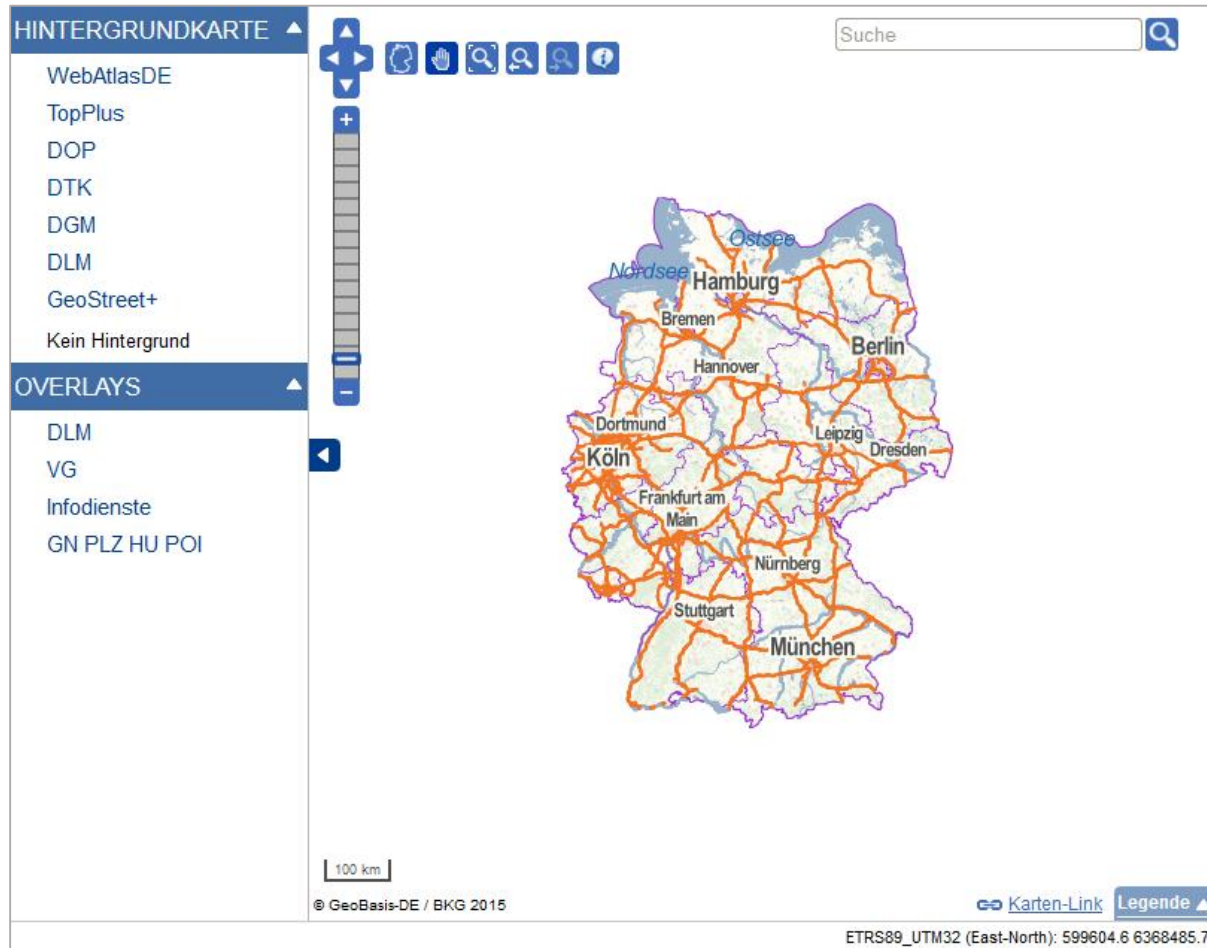
```
src="https://sg.geodatenzentrum.de/web_bkg_webmap/lib/bkgwebmap-0.10.js?bkg_appid=<API_KEY>"
```

- Vorteile:
 - Authentifizierung einer Anwendung (Domain-gebunden)
 - Dritte können freigeschaltete Dienste unserer Nutzer verwenden

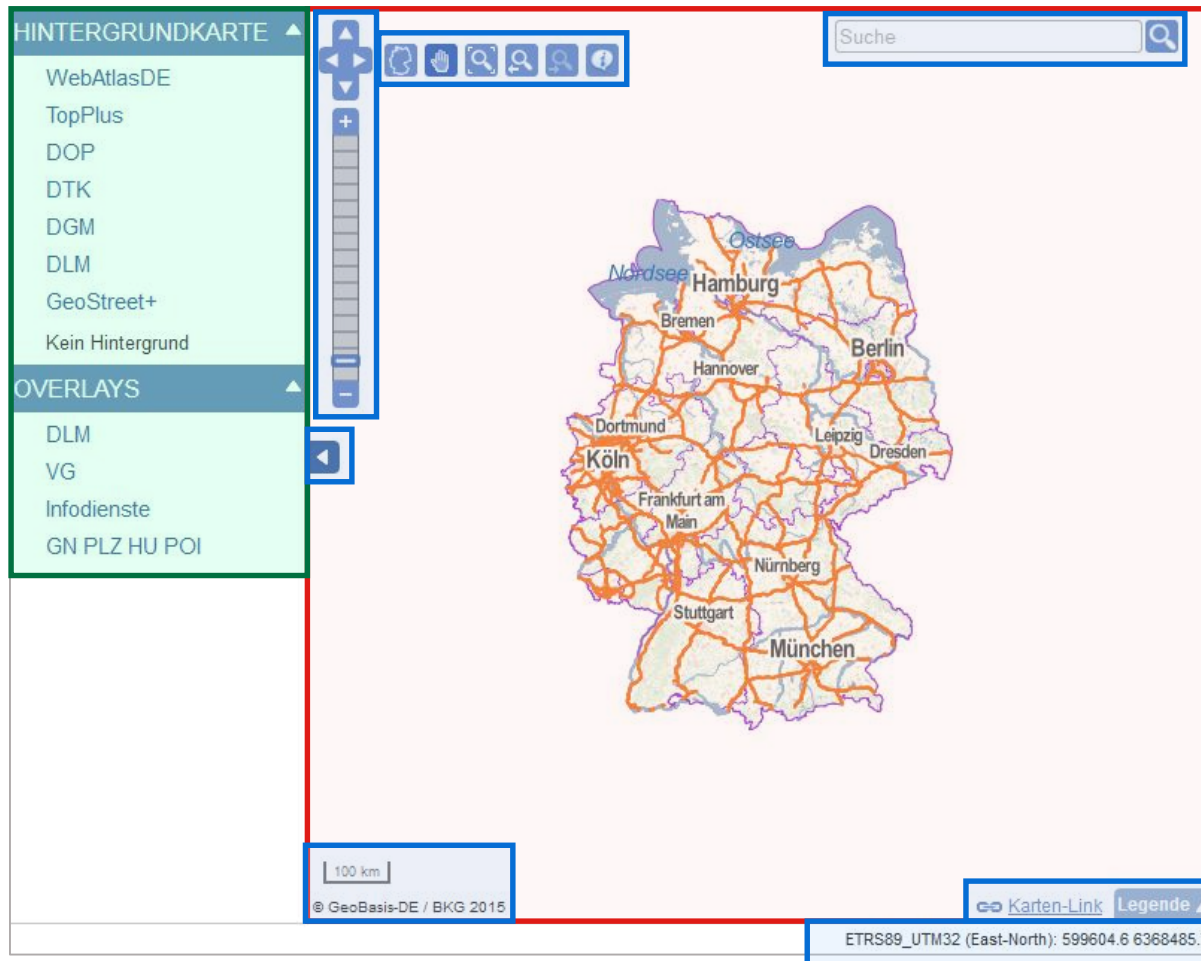
Anwendungsszenario



BKG WebMap im Detail



Komponenten einer Karte



- Kartencontainer
- Layer
- Controls

BKGWebMap.MapBuilder

- Abstrahiert Erzeugung von Karte, Layer und Controls in OpenLayers
- Generierung von Kartenanwendungen in wenigen Schritten:
 1. setMapConfig(...): Kartenkonfiguration setzen
 2. setDiv(...): HTML Element zuweisen
 3. setLayers(...): Layer konfigurieren
 4. setControls(...): Steuerelemente konfigurieren
 5. create(): Karte erzeugen
- Standardkonfiguration vorgegeben
- Standardkonfiguration kann überschrieben werden

http://sg.geodatenzentrum.de/web_bkg_webmap/doc/BKGWebMap/BKGWebMap.MapBuilder.html

Kartenkontainer

Stellt die Karte innerhalb der Webseite dar

Klasse: `OpenLayers.Map`

http://sg.geodatenzentrum.de/web_bkg_webmap/doc/OpenLayers2.13.1/files/OpenLayers/Map-js.html

- Benötigt HTML-Element zur Darstellung
- Verwaltet:
 - dargestellte Projektion
 - Maßeinheit
 - Kartenausschnitt
 - Layer
 - Controls

BKGWebMap.MapBuilder

Karte konfigurieren

MapBuilder.setDiv(id):

- Setzt das HTML-Element, in welchem die Karte dargestellt werden soll
- Beispiel:
`mapBuilder.setDiv('karte');`

MapBuilder.setMapConfig(config):

- Setzt die Konfigurationsparameter der Karte
- config s. Properties von [OpenLayers.Map](#)
- maxExtent und units werden von projection abgeleitet
- Beispiel:
`mapBuilder.setMapConfig({projection: 'EPSG:4326', units: 'degree'});`

Übung 1

Erstellung einer Karte

Vorlage: lessons/01_karte.html

Ziel:

- Voraussetzungen zur BKG WebMap kennenlernen
- Erstellen einer Karte in verschiedenen HTML Elementen
- Parameter der Karte ändern
(Georeferenzierung, Maßeinheit, ...)

Layer

Stellt die Daten oder Dienste innerhalb der Karte da

Basisklasse: `OpenLayers.Layer`

http://sg.geodatenzentrum.de/web_bkg_webmap/doc/OpenLayers2.13.1/files/OpenLayers/Layer-js.html

Konkrete Implementierungen in Packages:

- `OpenLayers.Layer`
 - `OpenLayers.Layer.WMS`
 - `OpenLayers.Layer.WMTS`
 - `OpenLayers.Layer.Vector`
 - ...
- `BKGWebMap.Layer`
 - `BKGWebMap.Layer.WMS`
 - `BKGWebMap.Layer.MarkerLayer`

Layer

Grundlagen

Darstellbare Daten:

- Rasterdaten (WMS, WMTS, OSM, ...)
- Vektordaten (Marker, GML, WFS, ...)

Es gibt zwei Darstellungsarten für Layer:

- BaseLayer
 - Hintergrundkarte: es kann immer nur ein BaseLayer angezeigt werden
- Overlay
 - Weitere Kartenebenen: es können mehrere Layer angezeigt werden
 - Darstellungsreihenfolge: zuerst hinzugefügt → zuerst dargestellt

BKGWebMap.MapBuilder Layer konfigurieren

MapBuilder.setLayers(layers):

- Setzt die Layer in der Karte
- layers ist eine Liste mit
 - Layerkonfigurationen (JavaScript Objekt)
 - OpenLayers.Layer Instanzen

- Beispiel:

```
mapBuilder.setLayers([  
  {type: 'WMS.DGM200.RELIEF'},  
  {type: 'WMS.VG1000', isBaseLayer: false}  
])
```

- Konfigurationsreferenz:

http://sg.geodatenzentrum.de/web_bkg_webmap/doc/factories.html#layer

Analog dazu kann auch `BKGWebMap.Layer.create(layer, map)` verwendet werden

Übung 2

Rasterlayer

Vorlage: [lessons/02_layer_bkg.html](#)

Ziel:

- BKG-Layer als BaseLayer hinzufügen
- BKG-Layer als Overlays hinzufügen
- Layereigenschaften anpassen

MarkerLayer

- Setzt Positionsmarken innerhalb der Karte
- Markierungen können zusätzlichen Inhalt per PopUp darstellen
- Beispiel:

```
markers = new BKGWebMap.Layer.MarkerLayer('Layername');  
markers.mark( [x, y], popUpText );
```



Vektorlayer

Darstellung von Vektordaten

Klasse: `OpenLayers.Layer.Vector`

http://sg.geodatenzentrum.de/web_bkg_webmap/doc/OpenLayers2.13.1/files/OpenLayers/Layer/Vector-js.html

Wichtige Properties:

- `protocol`: Definiert wie auf Daten zugegriffen wird (z.B.: `OpenLayers.Protocol.WFS`)
- `strategies`: Aktualisierungsstrategien für Daten (z.B.: `OpenLayers.Strategy.BBOX`)
- `styleMap`: Visualisierungsregeln für Daten
- Im MapBuilder per type: 'WFS'

```
var wfs = BKGWebMap.Layer.create( {  
  type: 'WFS',  
  name: 'VG250',  
  wfs: {url: url, featureType: typeName, featureNS: nameSpace,  
        geometryName: geometryProperty }  
} );  
map.addLayer(wfs);
```



Controls

Controls beeinflussen Darstellung und Verhalten der Karte.
Umfasst alles von Kartenbewegung bis Copyright-Hinweise

Basisklasse: `OpenLayers.Control`

http://sg.geodatenzentrum.de/web_bkg_webmap/doc/OpenLayers2.13.1/files/OpenLayers/Control-js.html

Konkrete Implementierungen in Packages:

- `OpenLayers.Control`
 - `OpenLayers.Control.Attribution`
 - `OpenLayers.Control.Navigation`
 - ...
- `BKGWebMap.Control`
 - `BKGWebMap.Control.Legend`
 - `BKGWebMap.Control.LayerSwitcher`
 - ...

BKGWebMap.MapBuilder Controls konfigurieren

MapBuilder.setControls(controls):

- Setzt die Controls der Karte
- controls ist JavaScript Objekt mit
 - Attributname definiert was erzeugt werden soll
 - Attributwert enthält Konfiguration

- **Beispiel:**

```
mapBuilder.setControls({  
    zoomBar:true,  
    scaleLine: true,  
    layerSwitcher: {size:200, maximized:true}  
})
```

- **Konfigurationsreferenz:**

http://sg.geodatenzentrum.de/web_bkg_webmap/doc/factories.html#controls

Analog dazu kann auch BKGWebMap.Control.create(config, controls) verwendet werden

Übung 4

Interaktion erweitern

Vorlage: lessons/04_controls.html

Ziel:

- Anzeige der Mausposition integrieren
- Layerauswahl integrieren
- Messwerkzeug integrieren
- Weitere Navigationselemente integrieren



Distributionen

1. bkgwebmap-x.xx.min.js

- Minimaler Funktionsumfang zur Reduzierung der Dateigröße
- Es sind die wesentlichen Bestandteile von OpenLayers integriert, um mit den WMS-Dienste des BKG zu interagieren

2. bkgwebmap-x.xx.all.min.js

- Erweiterter Funktionsumfang
- Es sind alle Funktionen aus OpenLayers integriert
- Zusätzliche Integration der Ortssuche des BKG

3. bkgwebmap-x.xx.all.js

- Wie 1. nur unkomprimiert zum besseren Debugging

4. bkgwebmap-x.xx.js

- Wie 2. nur unkomprimiert zum besseren Debugging



Vielen Dank für Ihre Aufmerksamkeit!

BKG WebMap: http://sg.geodatenzentrum.de/web_bkg_webmap/

Dienstleistungszentrum: dlz@bkg.bund.de

<http://www.geodatenzentrum.de>

Kontakt

Bundesamt für Kartographie und Geodäsie
Referat GI 5 / Dienstleistungszentrum
Karl-Rothe-Str. 10 - 14,
04105 Leipzig

Ansprechpartner
Dirk Thalheim
dirk.thalheim@bkg.bund.de
www.bkg.bund.de
Tel. +49 (0)341 5634 311