

Moderne Geodatenbereitstellung im Kontext der europäischen Geodateninfrastruktur INSPIRE

Jürgen Weichand

1. Einleitung

Gemäß des Zeitplans der europäischen Geodateninfrastruktur INSPIRE müssen seit Dezember 2013 für alle in den Anhängen der Richtlinie aufgeführten Themen Darstellungs- und Downloaddienste bereitstehen. Dieser Beitrag behandelt die Realisierung von INSPIRE-Netzdiensten auf Grundlage der OGC-Standards Web Map Tile Service 1.0 und Web Feature Service 2.0.

2. Darstellungsdienste

Die Technical Guidance 3.11 [1] sehen für die Umsetzung von INSPIRE-Darstellungsdiensten die OGC-Standards Web Map Service (WMS) und Web Map Tile Service (WMTS) vor. Für die Realisierung von INSPIRE-Darstellungsdiensten werden zunehmend OGC-konforme WMTS [2] eingesetzt. Im Gegensatz zu den etablierten WMS gelten „gecachte“ Dienste als einfach und performant.

2.1 Grundlagen WMTS 1.0

Ein WMTS ist ein Webdienst für den Zugriff auf in einer Kachelstruktur abgelegte Rasterbilder. Des Weiteren ist korrespondierend zum WMS-Standard eine Sachdatenabfrage (GetFeatureInfo) möglich.

2.1.1 Vor- und Nachteile im Vergleich zu WMS

Vorteile

- Durch die Ausgabe von vorprozessierten Kacheln sind sehr kurze Antwortzeiten möglich.
- Durch die Ausgabe von vorprozessierten Kacheln wird wenig Rechenleistung benötigt.

Nachteile

- Die vorprozessierten Kacheln können nur in festen Bodenauflösungen („Zoomstufen“) zur Verfügung gestellt werden.
- Aus Speicherplatzgründen können nur für wenige Projektionen Kacheln vorprozessiert werden.
- U. U. sind komplexe Mechanismen für partielle Kachelaktualisierungen notwendig.
- Der WMTS-Standard wird im Vergleich zum WMS-Standard von deutlich weniger Clients implementiert.

Der schnelle und ressourcenschonende Zugriff wird zu Lasten der Flexibilität erreicht. Daher hat sich die Bereitstellung von WMTS-Diensten über WMS-Dienste bewährt. Der WMS dient hierbei als Kachelproduzent für den WMTS, d. h. die GetTile-Anfrage wird in eine GetMap-Anfrage umgewandelt und das Ergebnisbild für eine bestimmte Zeit gecacht. Kunden, die auf einen möglichst flexiblen Zugriff angewiesen sind, kann so weiterhin der flexiblere WMS zur Verfügung gestellt werden.

2.1.2 Schnittstellen (Encoding)

Im WMTS-Standard sind die prozessorientierten Schnittstellen KVP und SOAP sowie die ressourcenorientierte Schnittstelle REST vorgesehen. Der Standard sieht die Unterstützung von mindestens einer der drei Schnittstellen vor, wobei die Unterstützung von KVP und/oder REST empfohlen wird.

„A WMTS server SHOULD support KVP and/or RESTful. SOAP support is optional.“ [2]

Jedoch unterstützen nicht alle Clients KVP und REST gleichermaßen, so dass die exklusive Bereitstellung einer der beiden Schnittstellen die Anzahl geeigneter Clients zusätzlich einschränkt.

2.1.3 Operationen / Ressourcen

Der WMTS 1.0.0 Standard definiert die drei Operationen GetCapabilities, GetTile und GetFeatureInfo (KVP, SOAP) bzw. die drei Ressourcen ServiceMetadata, Tile und FeatureInfo (REST).

Tab 1.: WMTS-Operationen

Operation	Ressource	m/o	Bedeutung
GetCapabilities	ServiceMetadata	m	Anforderung der Dienstbeschreibung
GetTile	Tile	m	Anforderung einer Kachel (Tile)
GetFeatureInfo	FeatureInfo	o	Anforderung von Sachinformationen

2.1.4 Tile Matrix Sets

Für jeden WMTS ist mindestens ein Tile Matrix Set („Kachelarchiv in einer Projektion“) bereitzustellen. Ein Tile Matrix Set besteht aus n-Tile Matrices („Zoomstufen“), für die jeweils folgende Festlegungen in den Capabilities beschrieben werden [2]:

- Identifikator: Identifier
- Maßstab: ScaleDenominator
- Obere, linke Ecke der BoundingBox (X, Y): TopLeftCorner
- Breite der Kachel [px]: TileWidth
- Höhe der Kachel [px]: TileHeight
- Anzahl der Kachelspalten: MatrixWidth
- Anzahl der Kachelreihen: MatrixHeight

2.1.5 Beispielaufufe

Die folgenden Beispielaufufe ermöglichen die Abfrage der Dienstbeschreibung bzw. einer Kachel. Für die Abfrage der Kachel wird das Tile Matrix Set („Kachelarchiv“) pmerc („Pseudo-Mercator“) und die Tile Matrix („Zoomstufe“) 2 verwendet.

Beispielaufufe KVP

GetCapabilities

<http://.../wmts?service=WMTS&version=1.0.0&request=GetCapabilities>

GetTile

[http://.../wmts?service=WMTS&version=1.0.0&request=GetTile
&layer=webatlas&style=default&format=image/png
&TileMatrixSet=pmerc&TileMatrix=2&TileRow=1&TileCol=1](http://.../wmts?service=WMTS&version=1.0.0&request=GetTile&layer=webatlas&style=default&format=image/png&TileMatrixSet=pmerc&TileMatrix=2&TileRow=1&TileCol=1)

Beispielaufufe REST

ServiceMetadata

<http://.../wmts/1.0.0/WMTSCapabilities.xml>

Tile

<http://.../wmts/1.0.0/webatlas/default/pmerc/2/1/1.png>

2.2 Anforderungen an einen INSPIRE-WMTS

Die Technical Guidance zu den INSPIRE-Darstellungsdiensten [1] erweitern den OGC-WMTS-Standard um:

- Mehrsprachigkeit
- Integration der INSPIRE-Metadatenelemente
- Daten-Dienste-Kopplung

Zudem wird das Tile Matrix Set InspireCRS84Quad als einheitliches Kachelarchiv empfohlen.

Der Arbeitskreis Geodienste der GDI-DE arbeitet derzeit an einer Erweiterung der bestehenden *Handlungsempfehlungen für die Bereitstellung von INSPIRE-konformen Darstellungsdiensten* [4] in Bezug auf WMTS.

2.3 WMTS 1.0 - Server

Im Open-Source-Bereich stehen u. a. folgende serverseitige WMTS-Implementierungen zur Verfügung:

- deegree 3 (deegree.org)
- GeoServer - GeoWebCache (geoserver.org)
- MapProxy (mapproxy.de)
- UMN MapServer - MapCache (mapserver.org)

3. Downloaddienste

Die Technical Guidance 3.1 zu den INSPIRE-Downloaddiensten [3] empfehlen die Umsetzungsvarianten Pre-defined Atom, Pre-defined WFS und Direct WFS auf Basis der Standards Atom (GeoRSS) und WFS 2.0. Für eine ausführliche Erläuterung der Standards sowie der drei Umsetzungsvarianten wird auf [5] verwiesen. Der folgende Abschnitt behandelt den Status Quo der INSPIRE-geeigneten FOSS WFS 2.0 Implementierungen.

3.1 WFS 2.0 - Server

Die WFS 2.0 Implementierungen von GeoServer (geoserver.org) und deegree (deegree.org) wurden weiter verbessert. Zudem wurde im Oktober 2013 mit der WFS 2.0 Implementierung für den UMN MapServer (mapserver.org) begonnen. Diese soll ab UMN MapServer 7.0 zur Verfügung stehen [6].

Die WFS 2.0 Implementierung des UMN MapServer ermöglicht die Bereitstellung von einfachen Feature-Modellen in Anlehnung an die Simple Feature Specification (SFS) [7]. Für die minimale Konfiguration eines FeatureTypes (Layer in Sinne der MapServer-Konfiguration) ist die Angabe des Namens, der Datenquelle und der Projektion notwendig. Hierbei wird standardmäßig der Namensraum ms (<http://mapserver.gis.umn.edu/mapserver>) verwendet. Über den Parameter `gml_include_items all` werden automatisiert alle Attribute der Datenquelle zur Verfügung gestellt und über `gml_featureid` das GML-Schlüsselattribut festgelegt.

Als Datenquelle können u. a. OGR-Datenformate und PostGIS-Datenbanken eingesetzt werden. Bei der Bereitstellung von WFS-Diensten sollten aus Performanzgründen stets räumliche Datenbanken verwendet werden. Der WFS-Server kann hierdurch auf die optimierten Abfrage-Algorithmen des Datenbankmanagementsystems (RDBMS) zurückgreifen.

Minimale WFS 2.0 FeatureType-Konfiguration (Auszug aus dem Mapfile):

```
LAYER
  NAME "Gemeinden"
  TYPE POLYGON
  DATA "/tmp/gmd_ex.shp"
  STATUS ON
  PROJECTION
    "init=epsg:31468"
  END
  METADATA
    "wfs_title" "Gemeinden (Bayern)"
    "gml_include_items" "all"
    "gml_featureid" "SCH"
  END
END
```

Ergebnis (GML 3.2.1):

```
<wfs:member>
  <ms:Gemeinden gml:id="Gemeinden.09161000">
    <gml:boundedBy />
    <ms:msGeometry />
    <ms:LAND>BY</ms:LAND>
    <ms:MODELLART>Basis-DLM</ms:MODELLART>
    <ms:OBJART>75003</ms:OBJART>
    <ms:OBJART_TXT>AX_KommunalesGebiet</ms:OBJART_TXT>
    <ms:OBJID>DEBY8728005FI001</ms:OBJID>
    <ms:HDU_X>0</ms:HDU_X>
    <ms:BEGINN>2013-02-12T08:05:10Z</ms:BEGINN>
    <ms:ENDE></ms:ENDE>
    <ms:ADM>6003</ms:ADM>
    <ms:BEZ_GEM>Ingolstadt</ms:BEZ_GEM>
    <ms:BEZ_KRS>Ingolstadt</ms:BEZ_KRS>
    <ms:BEZ_LAN>Freistaat Bayern</ms:BEZ_LAN>
    <ms:BEZ_RBZ>Oberbayern</ms:BEZ_RBZ>
    <ms:SCH>09161000</ms:SCH>
  </ms:Gemeinden>
</wfs:member>
```

Zusätzlich können weitere Eigenschaften des einfachen Feature-Modells konfiguriert werden:

- Namensraum des FeatureTypes (Mapfile/Web-Sektion: `wfs_namespace_prefix`, `wfs_namespace_uri`)
- Attribute des FeatureTypes (Mapfile/Layer-Sektion: `gml_include_items`)
- Alias der Attribute (Mapfile/Layer-Sektion: z. B. Alias für Attribut `BEZ_GEM`: `gml_BEZ_GEM_alias`)
- Alias des Geometrieattributs (Mapfile/Layer-Sektion: `gml_geometries`)

Ergebnis einer erweiterten FeatureType-Konfiguration:

```
<wfs:member>
  <bvv:Gemeinden gml:id="Gemeinden.09161000">
    <gml:boundedBy />
    <bvv:geometry />
    <bvv:gemeindeName>Ingolstadt</bvv:gemeindeName>
    <bvv:landkreisName>Ingolstadt</bvv:landkreisName>
    <bvv:regierungsbezirkName>Oberbayern</bvv:regierungsbezirkName>
    <bvv:gemeindeSchluessel>09161000</bvv:gemeindeSchluessel>
  </bvv:Gemeinden>
</wfs:member>
```

Zur besseren Übersichtlichkeit wurde auf die Darstellung der Geometrien verzichtet. Datenquelle: Bayerische Vermessungsverwaltung – www.geodaten.bayern.de

Die in WFS 2.0 vorgesehenen Stored Queries können als XML-Dateien abgelegt und über das Mapfile (`wfs_storedqueries`) referenziert werden.

3.2 WFS 2.0 - Clients

Die Unterstützung von WFS 2.0 hat sich im Vergleich zum vergangenen Jahr ein wenig verbessert. Das QGIS WFS 2.0 Client Plugin wurde weiterentwickelt. Seit Version 0.9 steht das Plugin auch für QGIS 2.0 zur Verfügung [8]. Zudem unterstützen einige proprietäre Softwareprodukte inzwischen ebenfalls WFS 2.0 (z. B. ArcGIS 10.2.1, FME 2014).

INSPIRE-Downloaddienste auf Basis von WFS 2.0 können relativ einfach zur Aktualisierung lokaler Geodatenrepositorien (z. B. Verzeichnis mit Shapedateien) eingesetzt werden. Im folgenden Beispiel werden über die Stored Query FlurstueckeAnhandGemarkungscodes alle Flurstücke der Gemeinde Oberschleißheim abgerufen und als `flurstuecke.gml` gespeichert. Mit Hilfe des Kommandozeilenwerkzeugs `ogr2ogr` [9] wird diese anschließend in eine ESRI-Shapefile umgewandelt.

Schritt 1: Download aller Flurstücke der Gemarkung Oberschleißheim (Gemarkungscodes: 8695) mit `wget`

```
wget --user='NUTZER' --password='PASSWORT' -O 'flurstuecke.gml'
'https://www.geodaten.bayern.de/wfs/ogc_flurkarte.cgi?
service=WFS&request=GetFeature&version=2.0.0&STOREDQUERY_ID=urn:bvv:def:que
ry:OGC-WFS::FlurstueckeAnhandGemarkungscodes&Gemarkungscodes=8695'
```

Schritt 2: Umwandlung GML 3.2.1 nach ESRI-Shapefile mit `ogr2ogr`

```
ogr2ogr -f 'ESRI Shapefile' flurstuecke flurstuecke.gml
```

Dieser Prozess kann über ein Shell-Skript automatisiert und über einen Cronjob täglich angestoßen werden.

Kontakt zum Autor:

Jürgen Weichand
Landesamt für Digitalisierung, Breitband und Vermessung
Alexandrastraße 4, 80538 München
juergen.weichand@ldbv.bayern.de

Literatur:

- [1] *Initial Operating Capability Task Force: Technical Guidance for the implementation of INSPIRE View Services*, Version 3.11, 2013, http://inspire.jrc.ec.europa.eu/documents/Network_Services/TechnicalGuidance_ViewServices_v3.11.pdf, letzter Zugriff: 04.02.2014
- [2] *Open Geospatial Consortium Inc. (OGC): OpenGIS Web Map Tile Service Implementation Standard*, OGC 07-057r7, Version 1.0.0, 2010, http://portal.opengeospatial.org/files/?artifact_id=35326, letzter Zugriff: 04.02.2014
- [3] *Initial Operating Capability Task Force: Technical Guidance for the implementation of INSPIRE Download Services*, Version 3.1, 2013, http://inspire.jrc.ec.europa.eu/documents/Network_Services/Technical_Guidance_Download_Services_v3.1.pdf, letzter Zugriff: 04.02.2014
- [4] *Geodateninfrastruktur Deutschland: Arbeitskreis Geodienste*, <http://www.geoportal.de/DE/GDI-DE/Arbeitskreise/Geodienste/geodienste.html?lang=de>, letzter Zugriff: 04.02.2014
- [5] *Weichand, Jürgen: Entwicklung und Anwendung von Downloaddiensten im Kontext der europäischen Geodateninfrastruktur INSPIRE*, Masterarbeit, Hochschule Anhalt, Dessau, 2013, <http://www.weichand.de/masterarbeit-inspire-downloaddienste/>, letzter Zugriff: 04.02.2014
- [6] *UMN MapServer: MS RFC 105: Support for WFS 2.0 (server side)*, <http://mapserver.org/development/rfc/ms-rfc-105.html>, letzter Zugriff: 04.02.2014
- [7] *Open Geospatial Consortium Inc. (OGC): OpenGIS Implementation Standard for Geographic information - Simple feature access - Part 1: Common architecture*, OGC 06-103r4, Version 1.2.1, 2011, http://portal.opengeospatial.org/files/?artifact_id=25355, letzter Zugriff: 04.02.2014
- [8] *Quantum GIS: WFS 2.0 Client*, <http://plugins.qgis.org/plugins/wfsclient/>, letzter Zugriff: 04.02.2014
- [9] *GDAL: OGR Simple Feature Library*, <http://www.gdal.org/ogr/>, letzter Zugriff: 04.02.2014