



deegree Tools

(Bonn, June⁶ 2006)

latlon GmbH

Aennchenstr. 19

53177 Bonn

Tel. ++49 228 18496-0

Dept. of Geography

Bonn University

Meckenheimer Allee 166

53115 Bonn

Tel. ++49 228 732098

Inhaltsverzeichnis

1 WMS mit Shapefiles aufsetzen.....	3
1.1 Vorraussetzungen:.....	3
1.2 Vorgehen.....	3
1.3 Erzeugen der räumlichen Indizierung der Shapefiles.....	3
1.4 Erzeugen der deegree Datastore-Konfigurationen.....	3
1.5 Aufsetzen eines deegree WFS.....	4
1.6 Erzeugen der deegree WMS capabilities bzw. der <Layer>- Tags und Style-Definen aus den WFS Capabilities.....	5
1.7 Erzeugen eines Web Map Context Dokuments aus den WMS Capabilities.....	5

1 WMS mit Shapefiles aufsetzen

1.1 Voraussetzungen:

- Shapefiles mit Punkten, Linien und Polygonen (3D-Shapes funktionieren nur eingeschränkt und haben eine schlechte performance)
- Der Name der Shapefiles muss ein gültiger XML-Elementname sein (u.a. darf der Name nicht mit einer Zahl beginnen und keine Sonderzeichen wie ,&', ,|' oder ,<' enthalten)
- Shapefiles sollten keine ,NULL'-Geometrien beinhalten
- Die Feldnamen in den assoziierten DBase-Files sollten keine Umlaute enthalten; sie müssen ebenfalls gültige XML-Elementnamen sein.

1.2 Vorgehen

1. Erzeugen der räumlichen Indizierung der Shapefiles
2. Erzeugen der deegree Datastore-Konfigurationen
3. Aufsetzen eines deegree WFS
4. Erzeugen der deegree WMS capabilities bzw. der <Layer>-Tags und Style-Definieren aus den WFS Capabilities
5. Erzeugen eines Web Map Context Dokuments aus den WMS Capabilities

1.3 Erzeugen der räumlichen Indizierung der Shapefiles

1. Zunächst alle Shapefiles in ein Verzeichnis kopieren (z.B. deegree/WEB-INF/data)
2. Das Programm org.deegree.tools.shape.IndexShapeFile starten.

```
C:\Programme\Java\jdk1.5.0_06\bin\java -classpath  
.;..\..\deegree.jar;..\..\lib\xml\jaxen-1.1-beta-7.jar  
org.deegree.tools.shape.IndexShapeFile
```
3. In dem sich öffnenden Dateidialog ein shapefile auswählen und bestätigen
4. In dem sich öffnenden Fenster die zu ,Geometry' gehörende Checkbox aktivieren und den Button ,Start Indexing' anklicken.
5. Den Vorgang für jedes vorhandene Shapefile wiederholen.

1.4 Erzeugen der deegree Datastore-Konfigurationen

1. In das Verzeichnis deegree/WEB-INF/conf/wms wechseln
2. Das Programm org.deegree.tools.datastore.DBSchemaToDatastoreConf für jedes der zu konfigurierenden Shapefiles aufrufen:

```
java -classpath ../lib/deegree.jar;../lib/jaxen-1.1-beta-7.jar org.deegree.tools.datastore.DBSchemaToDatastoreConf -driver SHAPE -srs EPSG:31467 -output ./datastores/t12_16alle.xsd -tables D:/java/projekte/map-topomatik_x-pos/webapps/deegree/WEB-INF/data/myshapefile
```

(Der Name der Shapefiles wird ohne Endung angegeben)

Das Programm erzeugt für jedes Shapefile eine xsd-Datei (XML-Schemadefinition), die gleichzeitig auch Konfigurationsdatei des deegree WFS ist. In den xsd-Dateien kann der absolute Pfad zum Shapefile durch einen relativen Pfad ersetzt werden, wenn dies gewünscht ist.

1.5 Aufsetzen eines deegree WFS

Hierzu muss eigentlich nicht mehr viel getan werden, da die grundsätzliche Konfiguration `deegree/WEB-INF/conf/wms/LOCALWFS_capabilities.xml` bereits vorhanden ist. Insofern es nicht ohnehin der Fall ist, muss das Verzeichnis in dem sich die zuvor erzeugten xsd-Dateien befinden, noch registriert werden. Dazu wird das Element `<DataDirectoryList>` um ein zusätzliches Element `<DataDirectory>` ergänzt:

```
<wfs:WFS_Capabilities xmlns:deegree="http://www.deegree.org/wfs"
xmlns:ows="http://www.opengis.net/ows" xmlns:ogc="http://www.opengis.net/ogc"
xmlns:wfs="http://www.opengis.net/wfs" xmlns:gml="http://www.opengis.net/gml"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wfs ../wfs.xsd" version="1.1.0"
updateSequence="0">
  <deegree:deegreeParams>
    <deegree:RootDirectory>d:/wfs</deegree:RootDirectory>
    <deegree:DefaultOnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple" xlink:href="http://127.0.0.1:8080/deegreewfs"/>
    <deegree:RequestTimeLimit>120</deegree:RequestTimeLimit>
    <!--
list of directories to be scanned for featuretypes/datastores to be served by
a WFS. deegree will look for datastore configuration files in these
directories and add the contained feature types to the featuretype list if not
already present.
optional; default: same directory as configuration
-->
    <deegree:DataDirectoryList>
      <deegree:DataDirectory>./datastores</deegree:DataDirectory>
    </deegree:DataDirectoryList>
  </deegree:deegreeParams>
  ...
```

Nach Neustart des Tomcats stehen alle zuvor konfigurierten Shapefiles als Featuretype zur Verfügung.

1.6 Erzeugen der deegree WMS capabilities bzw. der <Layer>-Tags und Style-Definieren aus den WFS Capabilities

1. Nach Ausetzen des deegree WFS können des Capabilities mittels eine GetCapabilities-Requests abgerufen werden (http://localhost:8080/deegree/services?service=WFS&request=GetCapabilities); das Ergebnis als XML-Datei irgendwo auf der Platte abspeichern.

2. Das Programm org.deegree.tools.wms.WFS2WMS wie folgt aufrufen:

```
java -classpath ../lib/deegree.jar;../lib/jaxen-1.1-beta-7.jar org.deegree.tools.wms.WFS2WMS -parentLayer LAYER -wfsCaps e:/temp/ogcwebservice.xml -outFile e:/temp/out.Tags angepasste werden.xml -full false
```

Das Programm wertet die über den Parameter -wfsCaps definierten WFS capabilities aus und erzeugt für jeden vorhandenen Featuretype einen Layereintrag für eine deegree WMS Konfiguration. Über den Parameter -full kann gesetzt werden, ob eine vollständige deegree WMS Konfiguration erzeugt wird (-full true) oder nur ein Wurzellayer, der dann in eine bereits vorhandene Konfiguration eingebettet werden kann (-full false).

Gleichzeitig wird ein SLD-Dokument erzeugt (und referenziert), das default-Styledefinition für jeden erzeugten Layer enthält (funktioniert zur Zeit nur für Layer mit Polygon-Geometrien);

Ggf. müssen noch Pfadangaben in erzeugten Layer

3. Wurde ‚-full true‘ gesetzt, kann der deegree WMS direkt mit der erzeugten Konfiguration gestartet werden; alle Featuretypes des WFS stehen nun auch als gleichnamige Layer über den WMS zur Verfügung (Die Namen der Layer können manuell angepasst werde).
4. Wurde ‚-full false‘ gesetzt, so kann das erzeugte XML-Fragment in eine bereits bestehende deegree WMS Konfiguration kopiert werden. Hierbei ist ggf. eine Anpassung der namespace-prefixes erforderlich.

1.7 Erzeugen eines Web Map Context Dokuments aus den WMS Capabilities

Das Programm org.deegree.tools.wms.WMS2WMC erzeugt aus einem WMS capabilities Dokument ein Web Map Context Dokument, das alle vom WMS angebotenen Layer umfasst. Der Aufruf sieht wie folgt aus:

```
java -classpath ../lib/deegree.jar;../lib/jaxen-1.1-beta-7.jar org.deegree.tools.wms.WMS2WMC -inFile myWMSCapabilities.xml -outFile myWMC.xml
```

Das erzeugte Web Map Context Dokument kann direkt vom deegree iGeoPortal portlet edition gelesen werden.