



deegree Web Map Service v2.2

lat/lon GmbH

Aennchenstr. 19
53177 Bonn
Germany
Tel ++49 - 228 - 184 96-0
Fax ++49 - 228 - 184 96-29
info@lat-lon.de
www.lat-lon.de

Dept. of Geography
Bonn University
Meckenheimer Allee 166
53115 Bonn

Tel. ++49 228 732098

Change log

| Date | Description | Author |
|-------------|---|-----------------|
| 2006-11-10 | Default behaviour for layer data sources adapted to changes in deegree implementation | Andreas Poth |
| 2006-10-18 | Description for accessing cascaded GetFeatureInfo request added | Andreas Poth |
| 2006-11-03 | Added details of supported SLD functionalities (Remote WFS servers) in section 4.1.4. | Markus Müller |
| 2006-12-22 | Added passed/added parameters for data sources. | Andreas Schmitz |
| 2007-01-10 | Update using new formatting style | Markus Müller |
| 2007-05-22 | Revision for demo version 2.1 | Hanko Rubach |
| 2007-12-11 | Remove typos; reformatting | Judit Mays |
| 2008-04-15 | Update to v2.2 | Judit Mays |
| | | |

Table of Contents

| | |
|--|-----------|
| 1 Introduction..... | 4 |
| 2 Download / Installation..... | 6 |
| 2.1Prerequisites..... | 6 |
| 2.2deegree Web Map Service release..... | 6 |
| 2.3Testing the installation..... | 6 |
| 3 Architecture..... | 9 |
| 4 Basic configuration..... | 10 |
| 4.1Structure of the configuration files..... | 10 |
| 4.2The deegree WMS configuration document..... | 10 |
| 4.2.1The deegree WMS configuration document..... | 10 |
| 4.2.2Service Parameter..... | 13 |
| 4.2.3Capability/Request-Parameter..... | 13 |
| 4.2.4Exceptions / UserDefinedSymbolization..... | 15 |
| 4.2.5Layer..... | 16 |
| 4.3Defining Styles..... | 26 |
| 4.3.1Label placement..... | 28 |
| 5 Advanced configuration..... | 31 |
| 5.1Manual Tomcat integration..... | 31 |

| | |
|---|-----------|
| 5.2 Offer own vector data (short description)..... | 34 |
| Appendix A Example wms_configuration.xml..... | 36 |
| Appendix B Supported SRS..... | 54 |
| Appendix C Featuretype Definition..... | 55 |
| Appendix D Deployment Descriptor (web.xml)..... | 57 |

Index of Tables

| | |
|--|---|
| Table 1: Directory structure of the WMS release..... | 6 |
|--|---|

Illustration Index

| | |
|---|----|
| Figure 1: Response of blue lake example GetMap request..... | 7 |
| Figure 2: Response of blue lake example GetMap request..... | 8 |
| Figure 3: WFS, WCS and WMS as data sources for deegree WMS..... | 9 |
| Figure 4: Files affecting the deegree WMS configuration..... | 10 |

1 Introduction

deegree is a Java Framework offering the main building blocks for Spatial Data Infrastructures (SDIs). Its entire architecture is developed using standards of the Open Geospatial Consortium (OGC) and ISO Technical Committee 211 – Geographic information / Geoinformatics (ISO/TC 211). deegree encompasses OGC Web Services as well as clients. deegree is Free Software protected by the GNU Lesser General Public License (GNU LGPL) and is accessible at <http://www.deegree.org>.

deegree2 is the new release of deegree supporting a number of features deegree1 was not able to handle. This documentation describes setup and configuration of deegree Web Map Service (WMS), an implementation of OGC's Web Map Service Implementation Specification 1.1.1. deegree WMS is the official Reference Implementation of the OGC for the mentioned standard.

deegree's WMS is able to render vector data as well as raster data from different storage formats and deliver it to any client that is able to perform a HTTP GET or POST request. The currently supported storage formats are:

- PostgreSQL / PostGIS
- Oracle (Spatial / Locator)
- databases allowing JDBC-connections
- ESRI Shapefiles
- (all provided by the deegree WFS)
- GML2 and GML3 provided by a OGC WFS
- JPEG, GIF, PNG, BMP, TIFF and GeoTIFF images (provided by the deegree WCS)
- Images provided by any OGC compliant WCS
- Cascaded OGC WMS 1.0.0/1.1.0/1.1.1

Besides a WMS, deegree comprises a number of additional services and clients. A complete list of deegree components can be found at:

<http://www.lat-lon.de> → Products

Downloads of packaged deegree components can be found at:

<http://www.deegree.org> → Download

deegree's Web Map Service offers great flexibility regarding its configuration and adaption to different data sources, formats and server environments. The configuration of WMS is similar to the configuration of other deegree web

services and requires editing of different XML files which control the functionality of the server.

The web services of deegree are realized as Java modules controlled by one central servlet (a “dispatcher”). This servlet has to be integrated into the respective web server/servlet engine. Most of the common web servers support servlet technology, thus making deegree a universal product. The Apache-Tomcat 5.5 Servlet-Engine is recommended due to its widespread use and its status as an open-source product.

2 Download / Installation

2.1 Prerequisites

For deegree2 Web Map Service to run you need:

- Java (JRE or JSDK) version 1.5.x
- Tomcat 5.5.x

For installation of these components refer to the corresponding documentation at java.sun.com and tomcat.apache.org.

2.2 deegree Web Map Service release

deegree Web Map Service can be downloaded from <http://www.deegree.org>. The release is packed as a WAR-archive. Simply put this file into your `$TOMCAT_HOME$/webapps` directory and (re-)start Tomcat. The installation of deegree WMS is already done with this.

Note: It is also possible to extract the WAR archive into another place of your computer and direct Tomcat to this place. Because of this possibility, in the remainder of this document, the directory you extracted the files to is referred to as `wms_home` (`=$TOMCAT_HOME$/webapps/deegree-wms` in the standard case).

Your `wms_home` will contain the following structure:

| directory | Content |
|----------------------|---|
| ./WEB-INF | Required by Tomcat, containing all libraries, configuration- and data-files |
| ./WEB-INF/conf/wms | WMS configuration files |
| ./WEB-INF/data | Example datasets |
| ./WEB-INF/conf/tools | SQL scripts for creating and filling example database |

Table 1: Directory structure of the WMS release

2.3 Testing the installation

deegree WMS comes with two sample data sets:

- The 'blue lake' dataset that is used by OGC for testing WMS compliance
- the 'UTAH' dataset comprising spatial data of the state Utah (sources: Shape, Hsqldb, WCS)

All of this data is automatically extracted and installed when unpacking the deegree WMS demo.

If installation succeeded, deegree WMS should be running by now with the sample data sets. Try the following request in your web browser:

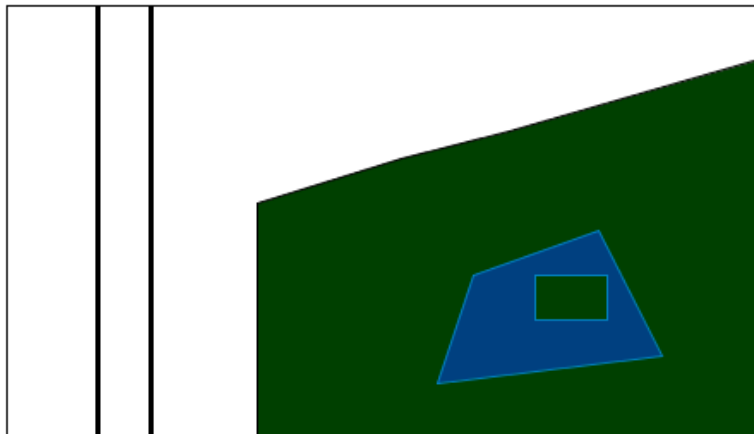
```
http://127.0.0.1:8080/deegree-  
wms/services?SERVICE=WMS&VERSION=1.1.1&REQUEST=GetCapabilities
```

This should provide a valid XML Capabilities file (if your browser asks for an application to open the file, try your browser again or any text editor). If this worked fine try a map request:

blue lake

```
http://127.0.0.1:8080/deegree-  
wms/services?REQUEST=GetMap&SERVICE=WMS&VERSION=1.1.1&WIDTH=467&HEIGHT=321&LAYERS=Buildings,DividedRoutes,Forests,Lakes,MapNeatline&TRANSPARENT=TRUE&FORMAT=image/png&BBOX=-0.004398,-  
0.003327182608695652,0.004531852173913042,0.0028108956521739124&SRS=EPSG:4326&STYLES
```

This request should deliver the following picture:

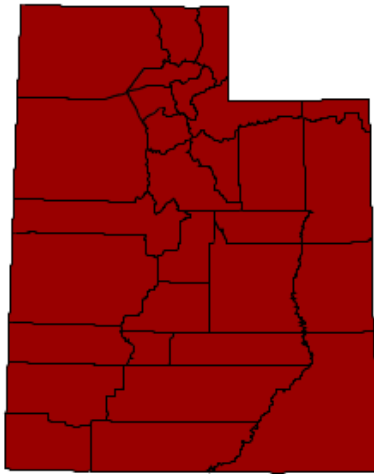


deegree 2.1 WMS 2007

Figure 1: Response of blue lake example GetMap request

UTAH

```
http://127.0.0.1:8080/deegree-  
wms/services?REQUEST=GetMap&SERVICE=WMS&VERSION=1.1.1&WIDTH=467&HEIGHT=321&LAYERS=Counties&TRANSPARENT=TRUE&FORMAT=image/png&BBOX=-  
27471.97559929872,4039616.6884999997,928973.8818492987,4697045.211499  
999&SRS=EPSG:26912&STYLES
```



deegree 2.1 WMS 2007

Figure 2: Response of blue lake example GetMap request

If all of this worked fine, your deegree WMS is running and you can now add your own data. But let's first have a brief look at the architecture of deegree WMS.

3 Architecture

degree2 WMS does not implement direct data access. All data that is used to create a map will be accessed via a WFS, WCS or another WMS. WFS and WCS can be realized locally – i.e. in the same Virtual Machine (VM) as the degree WMS – or remotely as a web service (Figure 3). Different WMS layers can be delivered by different WFS, WCS, or WMS servers.

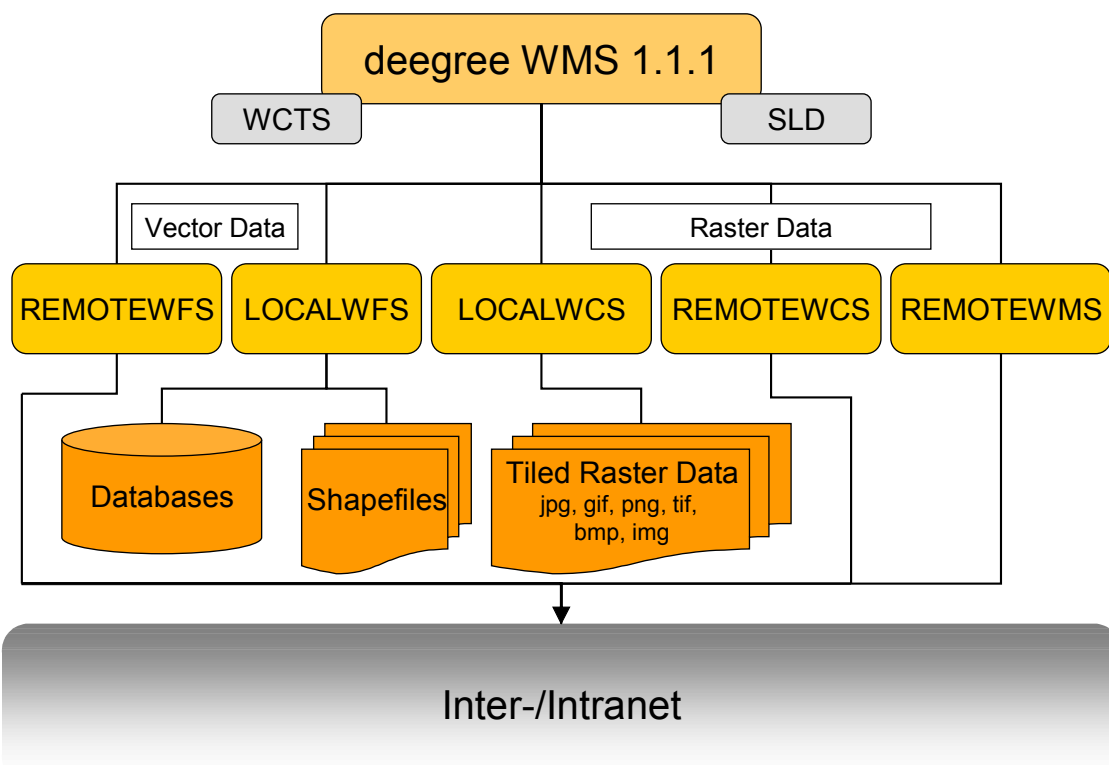


Figure 3: WFS, WCS and WMS as data sources for degree WMS

For configuration purposes this means that if setting up a degree WMS you also have to configure at least a LOCALWFS or LOCALWCS. Therefore it is recommended to consult these documentations as well if interested in advanced configuration options.

4 Basic configuration

4.1 Structure of the configuration files

The following figure shows the relationships between the different configuration files that have to be adapted:

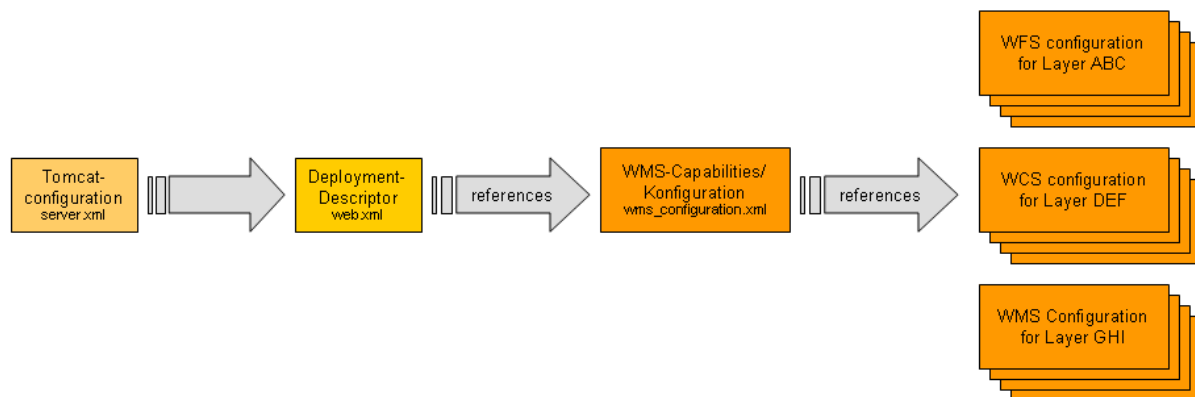


Figure 4: Files affecting the deegree WMS configuration

Besides the files shown in the diagram above, two additional files are part of the WMS configuration, the LOCALWFS_capabilities.xml and the LOCALWCS_capabilities.xml, if you use the default names. These two files are responsible for configuring the local WFS and WCS; please have a look at the deegree WFS- and WCS-documentation for details on how to do that.

4.2 The deegree WMS configuration document

The basic configuration allows use of the full set of parameters that can be used for controlling the deegree WMS using the deegree WMS configuration document. This includes the parameters mentioned in the OGC WMS 1.1.1 specification for the Capabilities document, but also additional specific technical parameters as for example descriptions of data sources for each layer or the maximum cache size. You can find the wms_configuration.xml file under \$wms-home\$/WEB-INF/conf/wms.

In the following, the elements of the configuration file will be described in detail. Appendix A includes the complete configuration document.

4.2.1 The deegree WMS configuration document

```

<?xml version="1.0" encoding="UTF-8"?>
<WMT_MS_Capabilities
  xmlns:deegree="http://www.deegree.org/wms"
  xmlns:sld="http://www.opengis.net/sld"
  xmlns:gml="http://www.opengis.net/gml"

```

```
xmlns:xlink="http://www.w3.org/1999/xlink" version="1.1.1"
updateSequence="1.1.0">
```

In this code fragment, the namespace bindings are defined. It is usually preferable to just copy this fragment and use the prefixes defined here elsewhere in the document.

After the root element, some deegree-WMS-specific parameters are defined. The `<DefaultOnlineResource>` is the URL by which the WMS operations can be invoked. This parameter can be overwritten by the URLs defined in the request-definitions. You have to adjust the `<DefaultOnlineResource>` to your system.

```
<deegree:DeegreeParam>
  <!-- The default online resource will be used for the following: If no
  LegendURL is provided, a GetLegendGraphic request with this URL will be
  deprecated automatically. If no OnlineResource element is specified within the
  Service element, this one will be inserted automatically. The operation
  specific online resources will NOT be filled automatically, since they also
  specify the HTTP method to be used. -->
  <deegree:DefaultOnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
  xlink:type="simple" xlink:href="http://localhost:8080/deegree-wms/services"/>
  <!-- default = 100 (MB) -->
  <deegree:CacheSize>100</deegree:CacheSize>
  <!-- maximum lifetime of the processes in the WMS; default = 3600 (sec)-->
  <deegree:MaxLifeTime>3600</deegree:MaxLifeTime>
  <!-- maximum time for the execution of a request until an exception of time-
  exceed is thrown default 15 seconds -->
  <deegree:RequestTimeLimit>45</deegree:RequestTimeLimit>
  <!-- determines the quality of the map/image generated from a GetMap request.
  the parameter will only be evaluated if the desired format is able to handle
  it. The range of values is 0 ... 1, where 1 is best and 0 is worst. default is
  0.95 -->
  <deegree:MapQuality>0.95</deegree:MapQuality>
  <!-- maximum map width that can be requested. default = 1000 -->
  <deegree:MaxMapWidth>1000</deegree:MaxMapWidth>
  <!-- maximum map height that can be requested. default = 1000 -->
  <deegree:MaxMapHeight>1000</deegree:MaxMapHeight>
  <deegree:AntiAliased>true</deegree:AntiAliased>
  <!-- copyright note that will be drawn to the left bottom side of the maps ;
  you can also reference a graphic file using absolute path to file e.g.
  c:/images/mylogo.jpg -->
  <deegree:Copyright>deegree-WMS V.2.2 2008</deegree:Copyright>
  <!-- radius of the circle around the point a user has clicked to that will
  be considered for creating a search area for a GetFeatureInfo request;
  default = 5 -->
  <deegree:FeatureInfoRadius>10</deegree:FeatureInfoRadius>
  <!-- returns the URL where the DTD for OGC WMS capabilities DOCTYPE definition
  is located. default =
  http://schemas.opengis.net/wms/1.1.1/WMS_MS_Capabilities.dtd You can also
  reference this document locally if you run this service without internet
  connection -->
  <deegree:DTDLocation>
    <deegree:OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
    xlink:type="simple"
    xlink:href="http://schemas.opengis.net/wms/1.1.1/WMS_MS_Capabilities.dtd"/>
  </deegree:DTDLocation>
  <!-- define a proxy server if the WMS is separated by a firewall from the DTD-
  location or the locations of connected REMOTEWFS, REMOTEWCS and REMOTEWMS
  as an alternative you can set command line parameters at the call of the java
  interpreter like this: java -DproxyHost=131.220.106.104 -DproxyPort=1234
  -classpath ... -->
  <!--Proxy proxyHost="131.220.106.104" proxyPort="1234"/-->
```

`</degree:DegreeParam>`

The `<CacheSize>` parameter defines the size of the cache available to degree-WMS in megabyte (this does not affect the cache for data sources). This parameter is optional, its default value is 100 MB. With the `<MaxLifeTime>` the user defines the time interval after which the internal processes of the WMS will be stopped if there are no active requests. If this parameter is not supplied, its default is 3600 seconds. By `<RequestTimeLimit>` the maximum time span is defined after which a request has to be processed. If this value is exceeded the processing is cancelled and an exception will be thrown. Its default value is 15 seconds.

Afterwards a `<MapQuality>` parameter is defined, by which the quality of the created maps can be controlled. Values for this parameter are between 0 (lowest quality) and 1 (best quality) with the default value being 0.95. This parameter is only used for image formats supporting different quality values (e.g. jpeg).

The following two parameters limit the maximum size of a map that can be requested via WMS. It has to be considered that requesting large maps is very taxing on a server as the processing cost increases with the square of the border length of a map. For example, the creation a map having 1000x1000 pixel takes four times the memory than creating a map with size 500x500 pixel. Default is 1000 pixel for width and height.

The `AntiAliased` parameter defines whether degree should use an anti aliasing filter while rendering the map. This usually improves the output for vector data. It is advisable to try both "true" and "false" with your data and see which looks better.

With the `<Copyright>` parameter it is possible to write a copyright mark in the lower left corner of each map. A copyright note can be either a text fragment or a reference to an image (e.g. `c:/images/copyright.gif`). In case this parameter is not supplied no output is produced.

The `FeatureInfoRadius` parameter defines a tolerance for the `GetFeatureInfo` request. If the user tries to click on a point and just misses it, he normally does not get back any data. Increasing this parameter to a value greater than 1 improves this behaviour by expanding the search area around a point by the specified radius.

The `DTDLocation` parameter can be used to override the `DTDLocation` that is sent back in response to a `GetCapabilities` request. Since degree nowadays does not try to load the DTD any more when encountering such a document, you'll usually drop this optional parameter, but if you wish to control the URL that is used in the `DOCTYPE` header of the `Capabilities`-document, here is where you set it.

If the WMS runs behind a firewall/proxy it is necessary to publish the proxy server address and port to the WMS (Java runtime environment) if data from remote

data sources outside the firewall are included in the WMS. Because in this case the WMS has to access servers from outside the firewall the network connection have to be passed through the proxy. Alternatively the proxy parameters can be passed to the Java runtime environment by command line parameters when starting the servlet engine (see java documentation).

4.2.2 Service Parameter

The following definition of service metadata is adopted from the WMS 1.1.1 specification.

```
<Service>
  <Name>WMS</Name>
  <Title>deegree2.2 demowms</Title>
  <!-- abstract and keywords are optional -->
  <Abstract>WMS reference implementation</Abstract>
  <KeywordList>
    <Keyword>deegree</Keyword>
    <Keyword>wms</Keyword>
  </KeywordList>
  <!-- The default online resource above will fill this one in for us, as well as
  in capabilities section. -->
  <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink" xlink:type="simple"
  xlink:href="http://localhost:8080/deegree-wms/services?" />
  <!-- the following service information is optional -->
  <ContactInformation>
    <ContactPersonPrimary>
      <ContactPerson>Andreas Poth</ContactPerson>
      <ContactOrganization>lat/lon</ContactOrganization>
    </ContactPersonPrimary>
    <ContactPosition>Technical Director</ContactPosition>
    <ContactAddress>
      <AddressType>Postal</AddressType>
      <Address>Aennchenstr. 19</Address>
      <City>Bonn</City>
      <StateOrProvince>NRW</StateOrProvince>
      <PostCode>53177</PostCode>
      <Country>Germany</Country>
    </ContactAddress>
    <ContactVoiceTelephone>0049228184960</ContactVoiceTelephone>
    <ContactFacsimileTelephone>00492281849629</ContactFacsimileTelephone>
    <ContactElectronicMailAddress>info@lat-lon.de
    </ContactElectronicMailAddress>
  </ContactInformation>
  <Fees>none</Fees>
  <AccessConstraints>none</AccessConstraints>
</Service>
```

4.2.3 Capability/Request-Parameter

The request-definition (GetCapabilities, GetMap and GetFeatureInfo) is in its form very similar to the definitions in OGC WMS 1.1.1 specification. However, some of the mandatory elements are declared optional. In case they are not supplied, deegree uses the corresponding default values. It is for example possible to not reference any of the requests explicitly. By default, GetCapabilities, GetMap and GetFeatureInfo will be initialized.

The following example is used to explain this by means of a GetMap request.

```
<GetMap>
  <!-- default = image/gif; image/png; image/jpg -->
  <!-- mandatory = image/png -->
  <Format>image/gif</Format>
  <Format>image/png</Format>
  <Format>image/jpeg</Format>
  <Format>image/jpg</Format>
  <!--Format>image/tif</Format-->
  <!--Format>image/tiff</Format-->
  <!--Format>image/bmp</Format-->
  <DCPType>
    <HTTP>
      <Get>
        <!--
        <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
          xlink:type="simple">
            <xlink:href="http://localhost:8080/deegree-wms/services?" />
          -->
        </Get>
        <Post>
          <!--
          <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink" xlink:type="simple"
            xlink:href="http://localhost:8080/deegree-wms/services?" />
          -->
        </Post>
      </HTTP>
    </DCPType>
  </GetMap>
```

You have to adjust the "xlink:href" attribute of the <OnlineResource> element to your system or leave the element commented out.

In case no <GetMap> element is defined, a GetMap with the formats image/gif, image/png and image/jpg will be initialized using the <DefaultOnlineResource> as web address for http-POST and GET. If a <GetMap> element is defined supplying information regarding Online-Resources but not the format, the above mentioned formats will be supported and initialized.

If only http-GET is defined, only this protocol is supported. Otherwise, if only http-POST is defined, http-GET is complemented as it is mandatory. The behaviour of the server regarding the image formats is likewise, all mandatory image formats will be initialized even if not defined by the user. An exception to this behaviour is the format image/png that is initialized anyway, as the OGC WMS 1.1.1 specification suggests the support of one format able to define an alpha-channel.

The behaviour of the initialization for GetCapabilities is similar to GetMap. Configuration of the operations DescribeLayer, GetLegendGraphic, GetStyles and PutStyles is also optional, but they will not be filled in by default (if missing, the WMS will not support these operations).

The format definition of GetFeatureInfo-Requests is distinctly different to the other format definitions. Its default format is application/vnd.ogc.gml. However, further user-defined formats are possible. You can use XSLT scripts to support

other formats. The XSLT scripts should take the standard GML3-Format as input and produce the desired format as output. You can specify the XSLT script by annotating the <Format> element with a “filter” attribute as in the following (commented) example:

```
<GetFeatureInfo>
  <!-- default & mandatory = application/vnd.ogc.gml -->
  <Format>application/vnd.ogc.gml</Format>
  <Format>text/plain</Format>
  <Format>text/html</Format>
  <!-- <Format
  filter="file:///c:/deegree/wms/toXML.xsl">application/vnd.ogc.wms_xml</Format>
  -->
  <DCPType>
    <HTTP>
      <Get>
        <!-- <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
        xlink:type="simple" xlink:href="http://localhost:8080/deegree-wms/services?"
        /> -->
      </Get>
    </HTTP>
  </DCPType>
</GetFeatureInfo>
```

In this example the XSLT-script file:///c:/deegree/wms/toXML.xsl is used as filter for the output format 'application/vnd.ogc.wms_xml', but there is no filter definition for 'text/html'. In this special case deegree is using a default filter defined by \$wms-home\$/WEB-INF/conf/wms/featureinfo2html.xsl.

The <OnlineResource> element can be dropped, and will be filled in with the value of the “DefaultOnlineResource” as described above.

4.2.4 Exceptions / UserDefinedSymbolization

You can also define which exception formats are to be used in case a service exception occurs. The three possible values are shown below:

```
<Exception>
  <!-- default & mandatory= application/vnd.ogc.se_xml -->
  <Format>application/vnd.ogc.se_xml</Format>
  <Format>application/vnd.ogc.se_inimage</Format>
  <Format>application/vnd.ogc.se_blank</Format>
</Exception>
```

If this information is not supplied, the XML format will be used as default.

The next element defines whether the WMS is capable of supporting SLDs.

```
<UserDefinedSymbolization SupportSLD="1" UserLayer="1" UserStyle="1"
RemoteWFS="1"/>
```

A value of '1' means “true”, a value of '0' false.

This element as well as its attributes are optional. Its lacking indicates that the WMS will not support SLD.

SLD allows three alternatives to access remote servers (using GetMap Parameters, <remoteOWS> element or some default WFS); deegree WMS only supports the second option: a <remoteOWS-element that is part of a <UserLayer>.

4.2.5 Layer

In the following, the layers known to the WMS are defined. Some elements here are not defined by the WMS 1.1.1 specification, especially those creating the association between layer and its data source.

It is possible, that a WMS supplies no layer definitions, indicating it serves no data of its own (a 'component' WMS as defined in the SLD specification). Such a server is nonetheless capable of serving maps by obtaining data directly or indirectly from the SLD documents sent to it.

The additional elements of the deegree-WMS configuration define the association of a layer with a data source (i.e. from a WFS or WCS). If a <Layer> has a <Name> but does not define a data source, using this layer with a GetMap or a GetFeatureInfo request causes that all <Layer>s nested within this layer will be affected by a request. For GetLegendGraphic requests only the top level layer will be used since GetLegendGraphic only allows the specification of one layer.

Simple reduced overview of example configuration:

```
<Layer>
  <Layer>nested Cite layers
    <Layer>Cite layer 1
    </Layer>
    <Layer>Cite layer 2
    </Layer>
    <Layer>Cite layer X
    </Layer>
  </Layer>
  <Layer>nested Utah layer
    <Layer>nested Utah administration
      <Layer>Utah county
      </Layer>
      <Layer>Utah Municipalities
      </Layer>
    </Layer>
    <Layer>Cite layer2
    </Layer>
    <Layer>Cite layerX
    </Layer>
  </Layer>
</Layer>
```

Full example for one nested layer:

```
...
  <Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
    <Name>citelayers</Name>
    <Title>Cite Layers</Title>
    <Abstract>deegree demo WMS</Abstract>
    <KeywordList>
      <Keyword>deegree</Keyword>
```



```

    <Keyword>layer</Keyword>
</KeywordList>
<MetadataURL type="ISO19115:2003">
  <Format>text/html</Format>
  <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
    xlink:type="simple" xlink:href="http://www.deegree.org" />
</MetadataURL>
<!-- queryable="1" sets GetFeatureInfo to enabled. If set to "1" at least
one datasource must be set to "1" too -->
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
<!-- please choose a unique <name> without spaces and a title which can
contain more details -->
  <Name>BasicPolygons</Name>
  <Title>BasicPolygons</Title>
  <Abstract>deegree demo WMS</Abstract>
  <KeywordList>
    <Keyword>deegree</Keyword>
    <Keyword>layer</Keyword>
  </KeywordList>
  <MetadataURL type="ISO19115:2003">
    <Format>text/html</Format>
    <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
      xlink:type="simple" xlink:href="http://www.deegree.org" />
  </MetadataURL>
  <MinScaleDenominator>0</MinScaleDenominator>
  <MaxScaleDenominator>10000000</MaxScaleDenominator>
  <!-- queryable="1" sets GetFeatureInfo to enabled for this datasource -->
  <deegree:DataSource failOnException="0" queryable="1">
    <deegree:Name>app:BasicPolygons</deegree:Name>
    <deegree:Type>LOCALWFS</deegree:Type>
    <!--name of the property that contains the geometries for this layer
this element will only be resolved if the data source is a LOCALWFS or
a REMOTEWFS -->
    <deegree:GeometryProperty>app:geom</deegree:GeometryProperty>
  </deegree:DataSource>
  <Style>
    <Name>default:BasicPolygons</Name>
    <Title>default:BasicPolygons</Title>
    <deegree:StyleResource>cite.xml</deegree:StyleResource>
  </Style>
  <!-- In case you do not wish to use the automatic legend generation you
need to set the <sld:GetLegendGraphic> here. Refer to the documentation
for details -->
</Layer>
<!-- The following Layer definition could be used as default for your own
layers -->
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
  <Name>Bridges</Name>
  <Title>Bridges</Title>
  <Abstract>deegree demo WMS</Abstract>
  <KeywordList>
    <Keyword>deegree</Keyword>
    <Keyword>layer</Keyword>
  </KeywordList>
  <MetadataURL type="ISO19115:2003">
    <Format>text/html</Format>
    <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
      xlink:type="simple" xlink:href="http://www.deegree.org" />
  </MetadataURL>
  <deegree:DataSource failOnException="0" queryable="1">
    <deegree:Name>app:Bridges</deegree:Name>
    <deegree:Type>LOCALWFS</deegree:Type>
    <deegree:GeometryProperty>app:geom</deegree:GeometryProperty>
  </deegree:DataSource>
  <Style>

```

```

        <Name>default:Bridges</Name>
        <Title>default:Bridges</Title>
        <degree:StyleResource>cite.xml</degree:StyleResource>
    </Style>
</Layer>
...

```

The configuration given in the example above will enable to perform a GetMap or GetFeatureInfo request on layer 'citelayers'. Because 'citelayers' itself does not define its own data source but nests two layers its resulting map/info result will be the same as performing a GetMap or GetFeatureInfo request against 'BasicPolygons' **and** 'Bridges'.

The following configuration snippet shows how to define a cascading WMS layer, i.e., a layer which gets its data from a remote WMS layer. This is done by specifying a DataSource element as shown. Note that this layer also has **two** data sources. Requesting this layer will paint the two results one after another.

```

<Layer xmlns:app="http://www.deegree.org/app" queryable="1" cascaded="0" opaque="1"
noSubsets="1">

    <Name>Europe</Name>
    <Title>Datenlayer 1</Title>
    <Abstract>first testing layer</Abstract>
    <degree:DataSource xmlns:deegree="http://www.deegree.org/wms"
        failOnException="1" queryable="1">
        <!-- default = equals the layer name and must be equal to the feature type
        or the layer name of the connected OWS -->
        <deegree:Name>app:Europe</deegree:Name>
        <!--name of the property that contains the geometries for this layer .
        this element will only be resolved if the data source is a LOCALWFS or a
        REMOTEWFS, the value is case sensitive -->
        <deegree:GeometryProperty>app:geom</deegree:GeometryProperty>
        <!--possible values: LOCALWFS, LOCALWCS, REMOTEWFS, REMOTEWCS,
        REMOTEWMS -->
        <!-- default = LOCALWFS -->
        <deegree:Type>LOCALWFS</deegree:Type>
        <!-- default is %Type%_capabilities.xml at the directory
        $wms-home$/WEB-INF/xml;
        default can only be used if data source type is LOCALXXX otherwise
        reference to the capabilities of the remote service must be set -->
        <deegree:OWSCapabilities>
            <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
                type="simple"
                xlink:href="http://www.lat-lon.de/documents/capabilities.xml"/>
        </deegree:OWSCapabilities>
        <deegree:ScaleHint min="0" max="1000"/>
        <!-- optional; the filter expression will be completed with the requested
        bounding box -->
        <deegree:FilterCondition>
            <wfs:Query typeName="app:Europe" xmlns:gml="http://www.opengis.net/gml"
                xmlns:ogc="http://www.opengis.net/ogc"
                xmlns:wfs="http://www.opengis.net/wfs">
                <wfs:PropertyName>app:Name</wfs:PropertyName>
                <wfs:PropertyName>app:Area</wfs:PropertyName>
                <wfs:PropertyName>app:Border</wfs:PropertyName>
                <ogc:Filter>
                    <ogc:And>

```

```

        <ogc:PropertyIsLessThan>
            <ogc:PropertyName>app:Area</ogc:PropertyName>
            <ogc:Literal>50000</ogc:Literal>
        </ogc:PropertyIsLessThan>
        <ogc:PropertyIsLike wildcard="*" singleChar="?"
escape="\ ">
            <ogc:PropertyName>
                app:Administrator/app:City</ogc:PropertyName>
            <ogc:Literal>53115 Bonn</ogc:Literal>
        </ogc:PropertyIsLike>
    </ogc:And>
</ogc:Filter>
</wfs:Query>
</deegree:FilterCondition>
</deegree:DataSource>
<deegree:DataSource failOnException="1" queryable="0">
    <!-- default = equals the layer name and must be equal to the feature type
        or the layer name of the connected OWS -->
    <deegree:Name>app:EuropeanRivers</deegree:Name>
    <!-- name of the property that contains the geometries for this layer .
        this element will only be solved if the data source is a LOCALWFS or a
        REMOTEWFS
        default = 'app:GEOM' -->
    <deegree:GeometryProperty>app:the_geom</deegree:GeometryProperty>
    <!-- default is %Type%_capabilities.xml at the directory
        $wms-home$/WEB-INF/xml;
        default can only be used if data source type is LOCALXXX otherwise
        reference to the capabilities of the remote service must be set -->
    <deegree:OWSCapabilities>
        <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
            type="simple"
            xlink:href="http://www.lat-lon.de/documents/another_capabilities_file.xml"/>
        </deegree:OWSCapabilities>
        <deegree:ScaleHint min="0" max="1000"/>
    </deegree:DataSource>
    <deegree:DataSource failOnException="0" queryable="0">
<deegree:Name>TK500</deegree:Name>
<deegree:Type>REMOTEWMS</deegree:Type>
<deegree:OWSCapabilities>
    <deegree:OnlineResource xlink:href="http://www.geoserver.nrw.de/GeoOgcWms1.3/
        servlet/NRW500?SERVICE=WMS&VERSION=1.1.0&REQUEST=GetCapabilities"
        xlink:type="simple" xmlns:xlink="http://www.w3.org/1999/xlink"/>
</deegree:OWSCapabilities>
<deegree:FilterCondition>
    <deegree:WMSRequest><![CDATA[Version=1.1.0&FORMAT=image/jpeg&
        TRANSPARENT=false&EXCEPTIONS=application/vnd.ogc.se_inimage&BGCOLOR=
        0xffffffff&LAYERS=Raster:UEK500:UEK500&STYLES=]]></deegree:WMSRequest>
<deegree:VendorspecificParameterDefinition>

    <deegree:PassedVendorspecificParameter>
        <deegree:Name>sessionid</deegree:Name>
        <deegree:Name>another</deegree:Name>
    </deegree:PassedVendorspecificParameter>

    <deegree:AddedVendorspecificParameter>
        <deegree:VendorspecificParameter>
            <deegree:Name>user</deegree:Name>
            <deegree:Value>aUser</deegree:Value>
        </deegree:VendorspecificParameter>
        <deegree:VendorspecificParameter>
            <deegree:Name>password</deegree:Name>
            <deegree:Value>aPassword</deegree:Value>
        </deegree:VendorspecificParameter>
    </deegree:AddedVendorspecificParameter>

```

```

</deegree:VendorSpecificParameterDefinition>
</deegree:FilterCondition>
<deegree:FeatureInfoTransformation>
  <deegree:OnlineResource
    xlink:href="file:///C:/deegree/wms/transformRemoteWMS.xsl"
    xlink:type="simple" xmlns:xlink="http://www.w3.org/1999/xlink"/>
  </deegree:FeatureInfoTransformation>
<deegree:ValidArea>
  <gml:Polygon srsName="EPSG:4326">
    <gml:outerBoundaryIs>
      <gml:LinearRing>
        <gml:coordinates>0,0 100,0 100,100 0,100 0,0</gml:coordinates>
      </gml:LinearRing>
    </gml:outerBoundaryIs>
  </gml:Polygon>
</deegree:ValidArea>
<deegree:TransparentColors>
  <deegree:Color>#000000</deegree:Color>
  <deegree:Color>#FF0000</deegree:Color>
</deegree:TransparentColors>
</deegree:DataSource>
</Layer>

```

4.2.5.1 DataSource

After the optional element `<Abstract>`, zero, one or more `<deegree:DataSource>` elements are defined. `<deegree:DataSource>` can be provided multiple times in case a scale-dependent definition or an overlay is needed that uses different feature types and/or layers. A `<deegree:DataSource>` may have two attributes:

- `failOnException` is of type boolean and defines if the processing of a GetMap request shall fail if accessing this data source fails. This is useful if a layer accesses more than one data source where at least one of these sources is not very stable (e.g. a REMOTEWMS) and one likes to avoid permanent failure of request processing. You can use '0' for "false" and '1' as "true", as above.
- `queryable` is also of type boolean and defines if a data source will be used for GetFeatureInfo requests. If the layer itself is queryable at least one of its data sources must be queryable.

The `<deegree:Name>` element of the DataSource supplies the name of the FeatureType, Coverage or Layer of the data that is supplied by the afterwards described OWS. The `<Name>` must be defined.

After the FeatureType/Coverage/Layer name the properties of the data source are defined. If a WFS will be used as data source `<deegree:GeometryProperty>` may be defined. As a WFS can use arbitrarily defined properties to store geometries, this element is only needed if the data store is a LOCALWFS or a REMOTEWFS. Its default value is 'app:GEOM'.

The third sub-element of `<deegree:DataSource>` declares the type of the data source. Five different values are possible for this:

- LOCALWFS → deegree WFS running in the same Virtual Machine
- REMOTEWFS → any other WFS
- LOCALWCS → deegree-WCS running in the same VM
- REMOTEWCS → any other WCS (not implemented yet)
- REMOTEWMS → other WMS which is used for cascading

To be able to use the full set of opportunities the deegree WMS offers you should read the deegree WFS and WCS documentation. There will be explained how to configure access to a database and file-based raster and vector data sources. If you use a remote OWS as data source you may have to study its configuration documentation.

<deegree:Type> is also optional, its default value is LOCALWFS.

<deegree:OWSCapabilities> specifies access to the capabilities of a remote or local data source. If for example a REMOTEWFS is defined as data source, the <deegree:OWSCapabilities> - respectively the <OnlineResource> - has to point to the Capabilities-document of the WFS.

If no <deegree:OWSCapabilities> is supplied, only the values LOCALWFS and LOCALWCS are valid, since the other types depend on the information supplied in this element.

In case a WCS or a REMOTEWMS is defined as data source, it is necessary to assure that they can deliver their data as image/gif, image/png, image/tif, image/jpeg or as image/bmp, in order to ensure that deegree can actually read the image data.

```
<deegree:OWSCapabilities>
  <OnlineResource xlink="http://www.w3.org/1999/xlink" type="simple"
    xlink:href="http://www.lat-lon.de/documents/capabilities.xml"/>
</deegree:OWSCapabilities>
```

or

```
<deegree:OWSCapabilities>
  <OnlineResource xlink="http://www.w3.org/1999/xlink" type="simple"
    xlink:href="http://remote-wfs-server.de/servicename?SERVICE=WFS&
    VERSION=1.1.0&REQUEST=GetCapabilities"/>
</deegree:OWSCapabilities>
```

In case you use a local WFS but do not use the default capabilities (LOCALWFS_capabilities.xml) a file URL will be used to reference the capabilities document of the WFS.

```
<deegree:OWSCapabilities>
  <OnlineResource xlink="http://www.w3.org/1999/xlink" type="simple"
    xlink:href="file:///c:/documents/capabilities.xml"/>
</deegree:OWSCapabilities>
```

In case neither <deegree:Type> nor <deegree:OWSCapabilities> are supplied it is assumed that the data source is a LOCALWFS whose Capabilities are in a

document named 'LOCALWFS_capabilities.xml' in the directory \$wms-home\$/WEB-INF/conf/wms/. If LOCALWCS is defined as <Type> and no <OWSCapabilities> element is supplied it is assumed that the Capabilities of the WCS are in a document named 'LOCALWCS_capabilities.xml' in the directory \$wms-home\$/WEB-INF/conf/wms/.

The optional element <degree:ScaleHint> indicates the span of scales for which the respective data source is valid, it corresponds to the element defined by OGC WMS 1.1.1. If multiple data sources are defined for one layer, the spans supplied by <degree:ScaleHint> must not overlap. If none is given, the ScaleHint of the layer itself is used (which may in turn inherit it from a parent layer).

As next sub-element of <degree:DataSource> a <degree:FilterCondition> can be defined, that constrains the data delivered by a data source. For example a layer might be associated only with those features of the feature type 'app:Europe' (Type = REMOTEWFS or LOCALWFS), whose areas are smaller than 50000 m² and whose data administrator is working in Bonn. Please refer wms_configuration.xml to find a different example.

```
<!-- optional; the filter expression will be enhanced by the bounding box of the
current request -->
<degree:FilterCondition>
  <wfs:Query typeName="app:Europe" xmlns:gml="http://www.opengis.net/gml"
    xmlns:ogc="http://www.opengis.net/ogc"
    xmlns:wfs="http://www.opengis.net/wfs">
    <wfs:PropertyName>app:Name</wfs:PropertyName>
    <wfs:PropertyName>app:Area</wfs:PropertyName>
    <wfs:PropertyName>app:Border</wfs:PropertyName>
    <ogc:Filter>
      <ogc:And>
        <ogc:PropertyIsLessThan>
          <ogc:PropertyName>app:Area</ogc:PropertyName>
          <ogc:Literal>50000</ogc:Literal>
        </ogc:PropertyIsLessThan>
        <ogc:PropertyIsLike wildCard="*" singleChar="?"
          escape="\ ">
          <ogc:PropertyName>
            app:Administrator/app:City</ogc:PropertyName>
            <ogc:Literal>53115 Bonn</ogc:Literal>
          </ogc:PropertyIsLike>
        </ogc:And>
      </ogc:Filter>
    </wfs:Query>
  </degree:FilterCondition>
```

The filter expression must be applicable to the defined type of the data source, meaning that for example the filter defined above can only be used in conjunction with a WFS, that stores features belonging to the feature type app:Europe. The typeName of the query in this case overwrites the value of the element <degree:Name> defined above in the DataSource tag.

The following example shows the <FilterCondition> for a REMOTEWMS:

```
<FilterCondition>
  <WMSRequest><![CDATA[VERSION=1.1.1&LAYERS=Ortsteile, Bezirke&
    STYLES=style1, default&FORMAT=jpg&BGCOLOR=0xFFFFFF&
    TRANSPARENT=true]]></WMSRequest>
  <degree:VendorspecificParameterDefinition>

    <degree:PassedVendorspecificParameter>
      <degree:Name>sessionid</degree:Name>
      <degree:Name>another</degree:Name>
    </degree:PassedVendorspecificParameter>

    <degree:AddedVendorspecificParameter>
      <degree:VendorspecificParameter>
        <degree:Name>user</degree:Name>
        <degree:Value>aUser</degree:Value>
      </degree:VendorspecificParameter>
      <degree:VendorspecificParameter>
        <degree:Name>password</degree:Name>
        <degree:Value>aPassword</degree:Value>
      </degree:VendorspecificParameter>
    </degree:AddedVendorspecificParameter>

  </degree:VendorspecificParameterDefinition>
</FilterCondition>
```

In this case the filter expression is overwriting several parameters of the layers (Name of the data store, styles) and of an incoming request (VERSION, FORMAT, BGCOLOR, TRANSPARENT). Other parameters (SRS, WIDTH, HEIGHT, BBOX, EXCEPTION, REQUEST) must not be overwritten, as this might lead to inconsistencies when processing GetMap and GetFeatureInfo-Requests.

It is also possible to tell degree which vendor specific parameters of an incoming request will be passed on to the remote server (the rest will be ignored) and which ones shall be automatically added to each request. This may be useful if the remote WMS does not know about a session ID, but requires extra login information, for example.

In the above example, the parameters “sessionid” and “another” would be passed along with their values, as well as the additional “user” and “password” parameters with their values “aUser” and “aPassword”.

To clarify, an “added” parameter is a parameter which will be added to each cascading request with its value. In the example, the remote WMS will receive GetMap requests like “...&request=GetMap&user=aUser&password=aPassword”.

A “passed” parameter will cause the degree WMS to make from a GetMap request like “...&request=GetMap&sessionid=abc123abc&another=andmore” a request to the remote WMS like “...TRANSPARENT=true&sessionid=abc123abc&another=andmore”.

The VendorspecificParameterDefinition element is optional.

A filter for a WCS is similar to one for WMS:

```
<FilterCondition>
  <WCSRequest>
```



```
<![CDATA[version=1.0.0&Coverage=world&Format=jpg]]>
</WCSRequest>
</FilterCondition>
```

To all filters the bounding box of the GetMap request will be added dynamically for each request.

When assigning a RemoteWMS as a data source you may not just perform GetMap requests against it but also GetFeatureInfo requests. Because OGC WMS specification does not define a clear format for GetFeatureInfo responses deegree WMS may not be able to process results returned by a cascaded WMS. If a cascaded server returns an XML format you have the option to register an XSLT-script to the corresponding data source that transforms incoming XML documents into flat GML2 compliant feature collections. Registration of such an XSL script is done by adding the following element:

```
<deegree:FeatureInfoTransformation>
  <deegree:OnlineResource
    xlink:href="file:///C:/deegree/wms/transformRemoteWMS.xsl"
    xlink:type="simple" xmlns:xlink="http://www.w3.org/1999/xlink"/>
  </deegree:FeatureInfoTransformation>
```

The element <deegree:ValidArea> defines the area for which the data source is valid. Any polygon can be used for defining a valid area. This can be useful if a layer integrates more than one data source where each data source is responsible for a different area because it may be served by different vendors.

```
<deegree:ValidArea>
  <gml:Polygon srsName="EPSG:4326">
    <gml:outerBoundaryIs>
      <gml:LinearRing>
        <gml:coordinates>0,0 100,0 100,100 0,100 0,0</gml:coordinates>
      </gml:LinearRing>
    </gml:outerBoundaryIs>
  </gml:Polygon>
</deegree:ValidArea>
```

At last the element <deegree:TransparentColors> can be used with raster data sources (WMS, WCS). It defines a list of colors that shall be turned to be transparent. For each color of a raster that shall be transformed to be transparent one <deegree:Color> element is added to <deegree:TransparentColors>. Colors are defined as hexadecimal RGB code with a leading '#'. The transparency will be a complete transparency with no opacity left.

```
<deegree:TransparentColors>
  <deegree:Color>#000000</deegree:Color>
  <deegree:Color>#FF0000</deegree:Color>
</deegree:TransparentColors>
```

4.2.5.2 SRS / BoundingBox / Style

The following elements are optional elements defined by WMS 1.1.1. They are not modified by the deegree-configuration and not initialized with default values,

save the SRS where a value of EPSG:4326 will be used if no other SRS is specified.

For bounding boxes care should be taken to use the correct SRS attribute, else clients won't be able to make use of it and the WMS may decide for (correct) requests to be outside and render the empty image instead of your data.

The LatLonBoundingBox should be specified at least once in the root layer, since the WMS specification says it's mandatory.

```
<!-- Just add the <SRS> you wish to support with your WMS -->
<!-- default = EPSG:4326 -->
<SRS>EPSG:4326</SRS>
<SRS>EPSG:26912</SRS>
<!-- It is strongly recommended that you set a LatLonBoundingBox; further
<BoundingBox>es are optional. Each layer inherits the BBoxes of the parent layer as
long as no explicit one is set. -->
<LatLonBoundingBox miny="-90" maxy="90" minx="-180" maxx="180" />
<!-- Setting <BoundingBox>es is optional but if set be sure to set the correct
SRS="EPSG:" -->
<BoundingBox SRS="EPSG:26912" miny="3581352" maxy="5432672" minx="0"
maxx="1504379" />
```

Note that the deegree WMS supports automatic coordinate reference system (CRS) for vector data (In WMS Specification the notion SRS instead of CRS is used). This means each layer based on a vector data set coming from a shape file or an Oracle Spatial database is available in nearly 300 CRS; you just have to define the ones you like for each layer or group of layers through <SRS> elements as in the code fragment above. A list of supported CRS is given in appendix B. There, you also find a java request to extract the latest list of supported CRS.

Next are the optional styles definitions for a layer. A <deegree:StyleResource> element was added here, giving information about the resource where the style is defined. The resource where a style is defined must be a valid Styled Layer Descriptor document (see below)

```
<Style>
  <Name>teststyle</Name>
  <Title>ein teststyle</Title>
  <!-- here you set the name of the style and the location <deegree:
StyleResource> where the WMS will find a style definition with the same
<UserStyle> name-->
  <deegree:StyleResource>styles.xml</deegree:StyleResource>
  <Abstract>dies ist nur ein test</Abstract>
  <LegendURL width="50" height="50">
    <Format>image/jpeg</Format>
    <OnlineResource xlink="http://www.w3.org/1999/xlink" type="simple"
      xlink:href="http://www.deegree.org/legend/teststyle.jpg"/>
  </LegendURL>
  <StyleSheetURL>
    <Format>/xsl</Format>
    <OnlineResource xlink="http://www.w3.org/1999/xlink" type="simple"
      xlink:href="http://www.deegree.org/legend/teststyle.xsl"/>
  </StyleSheetURL>
  <StyleURL>
    <Format>/css</Format>
```

```
<OnlineResource xlink="http://www.w3.org/1999/xlink" type="simple"
  xlink:href="http://www.deegree.org/legend/teststyle.css"/>
</StyleURL>
</Style>
```

Default value for `<deegree:StyleResource>` is `'wms_home/WEB-INF/conf/wms/styles.xml'`.

Besides individually defined styles, each layer knows a default-style, that can be address either by the name 'default' or with a missing style parameter. If there is no style defined for a layer and cannot be inherited, such a layer knows exclusively the default style. deegree-WMS allows definition of distinct default styles for each layer individually. For this purpose a style can be defined in `'wms_home/WEB-INF/conf/wms/styles.xml'` with its name beginning with 'default:' and displaying the name of the layer (default:%layername%). If for example a layer named 'Counties' is defined the name of its default style is default:Counties. If no individual default style is provided, the internal default style of deegree is used (area = 50% gray, lines = 1 pixel width an black, points = 7 pixel square 50% gray).

4.3 Defining Styles

Style definition for the deegree WMS will be done by creating one or more Styled Layer Descriptor (SLD) documents and assigning the UserStyles defined within it to one or more layers of the WMS. An SLD document that can be used for deegree WMS style definitions is valid against the OGC SLD 1.0.0 specification and defines one single 'NamedLayer'. The name of the `<NamedLayer>` element can be chosen freely and won't be evaluated currently. Within the `<NamedLayer>` element the user may define as many `<UserStyle>` elements as he likes. For better clarity we recommend to use more than one SLD document if the WMS offers a great amount of layers. The name of the 'UserStyles' corresponds to the name of the styles that can be assigned to a layer within the WMS configuration file as described above.

The following extract shows an example of an SLD

```
<?xml version="1.0" encoding="UTF-8"?>
<sld:StyledLayerDescriptor xmlns:sld="http://www.opengis.net/sld" xmlns:java="java"
  xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:wfs="http://www.opengis.net/wfs"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xslutil="de.latlon.deejump.plugin.style.XSLUtility"
  xmlns:deegreewfs="http://www.deegree.org/wfs"
  xmlns:fo="http://www.w3.org/1999/XSL/Format" xmlns:gml="http://www.opengis.net/gml"
  xmlns:app="http://www.deegree.org/app" xmlns:ogc="http://www.opengis.net/ogc"
  xmlns="http://www.opengis.net/sld" version="1.0.0">
<!-- Please be aware, that namespaces are used. So if you use featuretypes which
are bound to app: the property names in this document must use them too. e.g.
<ogc:PropertyName>app:geometry</ogc:PropertyName> -->
  <sld:NamedLayer>
    <sld:Name>Administration_etc</sld:Name>
    <sld:UserStyle>
```

```

<!-- The <sld:Name> of the <sld:UserStyle> must be identical to the
one referenced in the wms_configuration.xml -->
<sld:Name>default:StateBoundary</sld:Name>
<sld:Title>default:StateBoundary</sld:Title>
<sld:IsDefault>0</sld:IsDefault>
<sld:FeatureTypeStyle>
  <sld:Name>SGID024_StateBoundary</sld:Name>
  <sld:Rule>
    <sld:Name>State Boundary</sld:Name>
    <sld:MinScaleDenominator>0</sld:MinScaleDenominator>
    <sld:MaxScaleDenominator>99999999</sld:MaxScaleDenominator>
    <sld:PolygonSymbolizer>
      <ogc:Geometry>
        <ogc:PropertyName>app:geometry</ogc:PropertyName>
      </ogc:Geometry>
      <sld:Fill>
        <sld:CssParameter name="fill">#00b400</sld:CssParameter>
        <sld:CssParameter name="fill-opacity">1.0</sld:CssParameter>
      </sld:Fill>
      <sld:Stroke>
        <sld:CssParameter name="stroke">#007d00</sld:CssParameter>
        <sld:CssParameter name="stroke-opacity">1.0</sld:CssParameter>
        <sld:CssParameter name="stroke-width">3</sld:CssParameter>
        <sld:CssParameter name="stroke-dasharray">1</sld:CssParameter>
      </sld:Stroke>
    </sld:PolygonSymbolizer>
  </sld:Rule>
</sld:FeatureTypeStyle>
</sld:UserStyle>
<sld:UserStyle>
  <sld:Name>default:Municipalities</sld:Name>
  <sld:Title>default:Municipalities</sld:Title>
  <sld:IsDefault>0</sld:IsDefault>
  <sld:FeatureTypeStyle>
    <sld:Name>Municipalities 2004</sld:Name>
    <sld:Rule>
      <sld:Name>Municipalities</sld:Name>
      <sld:MinScaleDenominator>0</sld:MinScaleDenominator>
      <sld:MaxScaleDenominator>99999999</sld:MaxScaleDenominator>
      <sld:PolygonSymbolizer>
        <ogc:Geometry>
          <ogc:PropertyName>app:geometry</ogc:PropertyName>
        </ogc:Geometry>
        <sld:Fill>
          <sld:CssParameter name="fill">#e1b33f</sld:CssParameter>
          <sld:CssParameter name="fill-opacity">0.1</sld:CssParameter>
        </sld:Fill>
        <sld:Stroke>
          <sld:CssParameter name="stroke">#5f3d03</sld:CssParameter>
          <sld:CssParameter name="stroke-opacity">1.0</sld:CssParameter>
          <sld:CssParameter name="stroke-width">1</sld:CssParameter>
          <sld:CssParameter name="stroke-dasharray">1</sld:CssParameter>
        </sld:Stroke>
      </sld:PolygonSymbolizer>
    </sld:Rule>
    <sld:Rule>
      <sld:MinScaleDenominator>0</sld:MinScaleDenominator>
      <sld:MaxScaleDenominator>1600000</sld:MaxScaleDenominator>
      <sld:TextSymbolizer>
        <sld:Label>
          <ogc:PropertyName>app:name</ogc:PropertyName>
        </sld:Label>
        <sld:Font>
          <sld:CssParameter name="font-family">Arial</sld:CssParameter>

```

```

        <sld:CssParameter name="font-family">Sans-
        Serif</sld:CssParameter>
        <sld:CssParameter name="font-size">11</sld:CssParameter>
        <sld:CssParameter name="font-color">#000000</sld:CssParameter>
    </sld:Font>
    <sld:LabelPlacement>
        <sld:PointPlacement>
            <sld:Displacement>
                <sld:DisplacementX>0</sld:DisplacementX>
                <sld:DisplacementY>0</sld:DisplacementY>
            </sld:Displacement>
        </sld:PointPlacement>
    </sld:LabelPlacement>
</sld:TextSymbolizer>
</sld:Rule>
</sld:FeatureTypeStyle>
</sld:UserStyle>
</sld:NamedLayer>
</sld:StyledLayerDescriptor>

```

For a detailed description of defining a Styled Layer Descriptor document please have a look at the OGC SLD 1.0.0 specification. deegree supports most forms of style definition except for raster data (RasterSymbolizer).

One additional feature of the deegree WMS in context of defining styles that has been mentioned before is its capability of using an individual default style for each layer. To realize this for each layer a <UserStyle> must be defined that has a name starting with 'default:' followed by the layer's name (e.g. default:StateBoundary is the default style for the layer StateBoundary; see SLD fragment above). So if the default style is defined in a GetMap request the individual default style will be selected. If you don't define an individual default style 50% gray for fills and 1 pixel thick black lines will be used.

TIP:

To convert existing ESRI .avl visualization rules to SLD use the following command

```
java -classpath .:deegree2.jar org.deegree.tools.shape.AVL2SLD
```

There is a AVL2SLDtool.sh/.bat in the tools directory which you can adapt to your demands.

4.3.1 Label placement

The deegree WMS supports the automatic and optimized placement of labels for points and polygons. To force an automatic label placement an additional attribute has to be used at the <PointPlacement> element of the <TextSymbolizer>.

```

<sld:UserStyle>
    <sld:Name>default:Springs</sld:Name>
    <sld:Title>default:Springs</sld:Title>
    <sld:IsDefault>1</sld:IsDefault>
    <sld:FeatureTypeStyle>
        <sld:Name>Springs</sld:Name>

```

```

<sld:Rule>
...
</sld:Rule>
<sld:Rule>
  <sld:Name>Springs</sld:Name>
  <sld:MinScaleDenominator>0</sld:MinScaleDenominator>
  <sld:MaxScaleDenominator>500000</sld:MaxScaleDenominator>
  <sld:TextSymbolizer>
    <sld:Geometry>
      <ogc:PropertyName>app:geometry</ogc:PropertyName>
    </sld:Geometry>
    <sld:Label>
      <ogc:PropertyName>app:source</ogc:PropertyName>
    </sld:Label>
    <sld:Font>
      <sld:CssParameter name="font-family">Arial</sld:CssParameter>
      <sld:CssParameter name="font-family">Sans-Serif</sld:CssParameter>
      <sld:CssParameter name="font-style">italic</sld:CssParameter>
      <sld:CssParameter name="font-size">10</sld:CssParameter>
      <sld:CssParameter name="font-color">#222222</sld:CssParameter>
    </sld:Font>
    <sld:LabelPlacement>
      <!-- There are two options to place the text either automatically or
      by setting AnchorPoint/Displacement -->
      <sld:PointPlacement auto="true" />
      <!-- <sld:PointPlacement>
        <sld:Displacement></sld:Displacement>-->
        <!-- The coordinates are given as two floating-point numbers in the
        AnchorPointX and AnchorPointY elements each with values between 0.0
        and 1.0 inclusive. The bounding box of the label to be rendered is
        considered to be in a coordinate space from 0.0 (lower-left corner)
        to 1.0 (upper-right corner), and the anchor position is specified
        as a point in this space. The default point is X=0, Y=0.5, which is
        at the middle height of the left-hand side of the label. -->
        <!-- <sld:AnchorPoint>
          <sld:AnchorPointX>0</sld:AnchorPointX>
          <sld:AnchorPointY>0.5</sld:AnchorPointY>
        </sld:AnchorPoint>-->
        <!-- This will often be used to avoid over-plotting a graphic
        symbol marking a city or some such feature. The displacements are
        in units of pixels above and to the right of the point. A system
        may reflect this displacement about the X and/or Y axes to de-
        conflict labels. The default displacement is X=0, Y=0 -->
        <!--<sld:Displacement>
          <sld:DisplacementX>5</sld:DisplacementX>
          <sld:DisplacementY>-10</sld:DisplacementY>
        </sld:Displacement>
      </sld:PointPlacement>-->
    </sld:LabelPlacement>
    <sld:Halo>
      <sld:Fill>
        <sld:CssParameter name="fill">#0080C0</sld:CssParameter>
        <sld:CssParameter name="fill-opacity">0.6</sld:CssParameter>
      </sld:Fill>
    </sld:Halo>
  </sld:TextSymbolizer>
</sld:Rule>
</sld:FeatureTypeStyle>
</sld:UserStyle>

```

The <UserStyle> given above defines a label for the geometry of a feature named 'app:geometry'. The label will print the content of the features 'app:source' property. The geometry can be a point, a polygon or a multi-polygon.

As `<LabelPlacement>` `<PointPlacement>` is chosen so the label is related to a point which is the point geometry by itself or the centroid of the visible part of a polygon (`<LinePlacement>` isn't supported yet). It has an additional attribute named 'auto' not defined in the SLD specification that forces a label placement optimization if it is set to 'true'. If it is missing or set to 'false' no optimization will be done and the user is free to fix the position of the labels by its own definitions (e.g. `AnchorPoint`, `Displacement`, `Rotation`, ... ; see SLD specifications how to do this).

`LabelPlacement` for `Linestrings` is a little bit different, but not much:

```
<sld:Rule>
  <sld:Name>Textsymbolizer</sld:Name>
  <sld:MinScaleDenominator>0</sld:MinScaleDenominator>
  <sld:MaxScaleDenominator>500000</sld:MaxScaleDenominator>
  <sld:TextSymbolizer>
    <sld:Geometry>
      <ogc:PropertyName>app:geometry</ogc:PropertyName>
    </sld:Geometry>
    <sld:Label>
      <ogc:PropertyName>app:code</ogc:PropertyName>
    </sld:Label>
    <sld:Font>
      <CssParameter name="font-family">Serif</CssParameter>
      <CssParameter name="font-style">normal</CssParameter>
      <CssParameter name="font-weight">normal</CssParameter>
      <CssParameter name="font-size">12</CssParameter>
      <CssParameter name="font-color">#000000</CssParameter>
    </sld:Font>
    <sld:LabelPlacement>
      <sld:LinePlacement>
        <sld:PerpendicularOffset>above</sld:PerpendicularOffset>
        <!-- width of the line the label is associated with -->
        <sld:LineWidth>2.5</sld:LineWidth>
        <!-- gap between labels measured in label width -->
        <sld:Gap>5</sld:Gap>
      </sld:LinePlacement>
    </sld:LabelPlacement>
  </sld:TextSymbolizer>
</sld:Rule>
```

Instead of `<PointPlacement>` the Element `<LinePlacement>` will be used. At the moment `LinePlacement` doesn't support automatic label placement optimization but using the three elements contained in the `<LinePlacement>` element you can already realize good results. The `<PerpendicularOffset>` defines the relative position of the label according to the line. You can use a number as defined in the SLD specifications or set a predefined value (above, below, center). If you use one of predefined values it is useful to set a value for `<LineWidth>` to ensure that if you choose 'above' the label will be really above the line and not partially within it.

Using the `<Gap>` element you can determine the gap between the labels of one `Linestring`. The value of `<Gap>` is measured in the length of the label string. This means if a river shall be labeled with 'Rhine' one gap will have the length of the string 'Rhine'. The labeling will look like this Rhine Rhine Rhine. If you

choose `<Gap>0</Gap>` you will get a labelling like this Rhine Rhine Rhine Rhine Rhine.

5 Advanced configuration

5.1 Manual Tomcat integration

The location of deegree's libraries and the central deegree wms configuration file `wms_configuration.xml` should be registered with the Servlet Engine (in this case Apache Tomcat 5.5). Tomcat offers several possibilities to register and configure web contexts.

The easiest way to register deegree web services with Tomcat is to copy the `deegree-wms.war` file to the `$TOMCAT_HOME$/webapps` directory. You can do this either with running or stopped tomcat. If the tomcat is started afterwards, the application should be automatically deployed. Tomcat will unpack the `deegree-wms.war` file (which is nothing more than a .zip file) to the webapps directory. The name of the .war sets the name of the service address:

`http://localhost:8080/deegree-wms`

If you want to do the Tomcat installation process manually use the steps described in the following.

Unpack the `deegree-wms.war` to a directory of your choice (e.g. `c:/deegree/webapps/deegree-wms`).

Afterwards Tomcat needs information about the root directory of the WMS. The easiest way is to create an XML-file in the directory `$TOMCAT_HOME$/conf/Catalina/localhost`, named exactly as the service e.g. `deegree-wms.xml`, and fill it with the following information

```
<Context docBase="c:/deegree/webapps/deegree-wms" path="/deegree-wms">
</Context>
```

where the `docBase` attribute reflects the physical location of the deegree service in the file system and the `path` attribute describes the virtual location of the main directory of the deegree web service. In the example, the root directory of the service is accessible at `http://my.server.domain/deegree-wms/`. For further information have a look at the Tomcat documentation included in the installation.

The name of the deegree-service directory is arbitrary whereas Tomcat definitely looks for a subdirectory `WEB-INF` (in capital letters – even on a Windows system) in the root directory. You will find this directory after tomcat automatically unpacked the war archive. Here the `Deployment-Descriptor` (`web.xml`) is located, which is analysed by Tomcat to identify the servlet(s) belonging to the application, their names, the parameters that are delivered to the servlet(s) and information about the existing access restrictions.

Before starting deegree WMS the following dataset entry in `web.xml` is essential:

```
<?xml version="1.0"?>
```



```
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app>
  <display-name>deegree 2.2</display-name>
  <description>deegree 2.2 OWS</description>
  <servlet>
    <servlet-name>owservice</servlet-name>
    <servlet-class>org.deegree.enterprise.servlet.OGCServletController</servlet-
    class>

    <init-param>
      <param-name>services</param-name>
      <param-value>wms</param-value>
      <description>
        list of supported services, e.g.: wfs,wms,wcs (comma separated) always use
        lowercase
      </description>
    </init-param>

    <!-- WMS INITIALIZING PARAMETERS -->
    <init-param>
      <param-name>wms.handler</param-name>
      <param-value>org.deegree.enterprise.servlet.WMSHandler</param-value>
    </init-param>
    <init-param>
      <param-name>wms.config</param-name>
      <param-value>WEB-INF/conf/wms/wms_configuration.xml</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
  </servlet>
  <servlet-mapping>
    <servlet-name>owservice</servlet-name>
    <url-pattern>/services</url-pattern>
  </servlet-mapping>
  <welcome-file-list>
    <welcome-file>/index.jsp</welcome-file>
  </welcome-file-list>
  <error-page>
    <error-code>500</error-code>
    <location>/error.jsp</location>
  </error-page>
  <error-page>
    <exception-type>org.deegree.ogcwebservices.OGCWebServiceException</exception-
    type>
    <location>/error.jsp</location>
  </error-page>
</web-app>
```

The name of the servlet and of the java-class representing the servlet should be indicated in the <servlet> tags. The servlet-name can be user defined, but care should be taken that the same name that is defined here is also used in the servlet-mapping. The servlet is located in the deegree2.jar library.

The tag <init-param> defines parameters that are analyzed by the servlet, during initialization. The transferred parameters are

- 'services': The value of this parameter contains a comma separated list of OWS that will be made available through the context. In the example only a 'wms' is defined to be available (other possible values at the moment are: wfs, wcs, sos, wpvs and csw).

- For each service listed in the 'service' init-param a handler class and a configuration file must be referenced.
- The name of the init-param for defining the handler starts with the service name ('wms' in the example) followed by '.handler'. The value of this parameter is the name of the handler class to be used. It is possible to write a different class for this and reference it accordingly. As default 'org.deegree.enterprise.servlet.WMSHandler' should be used.
- The name of the init-param for defining the main configuration file of a service also starts with the service name followed by '.config'. Note that you can use a relative path to the configuration file starting at the WEB-INF directory of the context.

If you want to make more than one service available through a servlet context, web.xml looks like this (the example defines a 'wms' as well as a 'wfs' and if you uncomment the WCS section even this one is accessible):

```
<?xml version="1.0"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app>
  <display-name>deegree 2.2</display-name>
  <description>deegree 2.2 OWS</description>
  <servlet>
    <servlet-name>owservice</servlet-name>
    <servlet-class>org.deegree.enterprise.servlet.OGCServletController</servlet-
class>

    <init-param>
      <param-name>services</param-name>
      <param-value>wms,wfs</param-value>
      <description>
        list of supported services, e.g.: wfs,wms,wcs (comma separated) always use
        lowercase
      </description>
    </init-param>

    <!-- WMS INITIALIZING PARAMETERS -->
    <init-param>
      <param-name>wms.handler</param-name>
      <param-value>org.deegree.enterprise.servlet.WMSHandler</param-value>
    </init-param>
    <init-param>
      <param-name>wms.config</param-name>
      <param-value>WEB-INF/conf/wms/wms_configuration.xml</param-value>
    </init-param>

    <!-- WFS INITIALIZING PARAMETERS -->
    <init-param>
      <param-name>wfs.handler</param-name>
      <param-value>org.deegree.enterprise.servlet.WFSHandler</param-value>
    </init-param>
    <init-param>
      <param-name>wfs.config</param-name>
      <param-value>WEB-INF/conf/wms/LOCALWFS_capabilities.xml</param-value>
    </init-param>

    <!-- WCS INITIALIZING PARAMETERS -->
    <!--
```

```
<init-param>
  <param-name>wcs.handler</param-name>
  <param-value>org.deegree.enterprise.servlet.WCSHandler</param-value>
</init-param>
<init-param>
  <param-name>wcs.config</param-name>
  <param-value>WEB-INF/conf/wcs/LOCALWCS_capabilities.xml</param-value>
</init-param>
-->
<load-on-startup>1</load-on-startup>
</servlet>
<servlet-mapping>
  <servlet-name>owservice</servlet-name>
  <url-pattern>/services</url-pattern>
</servlet-mapping>
<welcome-file-list>
  <welcome-file>/index.jsp</welcome-file>
</welcome-file-list>
<error-page>
  <error-code>500</error-code>
  <location>/error.jsp</location>
</error-page>
<error-page>
  <exception-type>org.deegree.ogcwebservices.OGCWebServiceException</exception-
  type>
  <location>/error.jsp</location>
</error-page>
</web-app>
```

The tag `<servlet-mapping>` defines the alias name for the servlet. It is not necessary that the `<servlet-name>` and `<url-pattern>` are identical. `<url-pattern>` is the name for the parameter the servlet will be called through (part of the base-URL of the service that all requests have to use). Combined with the path settings in `$TOMCAT_HOME$/conf/Catalina/localhost/deegree-wms.xml` and respectively the name of the .war which you deployed in the `$TOMCAT_HOME$/webapps` (in our example `deegree-wms`) you should be able to point to your WMS (OWS) via the following URL: `http://my.server.domain/deegree-wms/services?`

5.2 Offer own vector data (short description)

This deegree 2.2 WMS release comes bundled with a configuration of WFS and WCS as these are the data sources of the WMS. It is not possible for the WMS to connect to databases or files directly. You will find detailed information on how to configure the deegree 2.2 WFS and 2.2 WCS in separate download packages/documentation. But still you find some scripts under `wms_home/WEB-INF/conf/tools` with some short hints to assist you to add own data. The three steps you have to take are

1. create a `featuretypedefinition.xsd` for the LOCALWFS and place it in `wms_home/WEB-INF/conf/wms/featuretypes`
2. Adapt the `wms_home/WEB-INF/conf/wms/wms_configuration.xml` to your demands (add needed SRS support, bounding boxes, etc) and add a layer definition

3. create a style.xml or use an existing file. Create a <UserStyle> and reference it in the wms_configuration.xml (refer Chapter 4.2 et seqq)
4. restart tomcat and you ready!

Featuretype definitions can be created with support by tools. Tools are supplied for Windows (xxx.bat) and Linux (xxx.sh) operating systems. Before you start one of the tools you have to adapt them to your demands. You find preconfigured commands for the demo datasets which you can uncomment and adapt. You can edit these scripts with any text editor. Afterwards just go with your shell/console to the mentioned tools directory and run the desired script. To create a featuretype definition on shape files use the script 'ShapetoFeatureTypeDef', to create Postgis database access use the script 03_DBtoFeatureTypeDef.

Open the created featuretypedefinition.xsd and do further manual configuration:

- set the CRS to EPSG:yourCode in the tag <deegreewfs:DefaultSRS> EPSG:26912</deegreewfs:DefaultSRS>
- check if the path to the shapefile is correct. Relative paths must start at the featuretypedefinition.xsd!
- if a database is used which stores the used CRS, then specify the table srs=' ' (without 'EPSG:' !). If no CRS is specified in DB or if used for shapefile access then leave srs='-1'
- ```
<xsd:element name='geometry' type='gml:GeometryPropertyType'>
 <xsd:annotation>
 <xsd:appinfo>
 <deegreewfs:Content>
 <deegreewfs:MappingField field='the_geom' type='GEOMETRY'
 srs='26912' />
 </deegreewfs:Content>
 </xsd:appinfo>
 </xsd:annotation>
</xsd:element>
```
- see Commented\_FeaturetypeDefiniton.xsd.txt under \$wms\_home\$/WEB-INF/conf/wms/featuretypes for further details
- restart tomcat and the WFS should deliver this featuretype.

Follow with the steps 2 & 3 and you are ready.

## Appendix A Example wms\_configuration.xml

Path of file: \$wms\_home\$/WEB-INF/conf/wms/

```
<?xml version="1.0" encoding="UTF-8"?>
<WMT_MS_Capabilities xmlns:deegree="http://www.deegree.org/wms"
 xmlns:sld="http://www.opengis.net/sld" xmlns:gml="http://www.opengis.net/gml"
 xmlns:xlink="http://www.w3.org/1999/xlink" version="1.1.1"
 updateSequence="1.1.0">

 <deegree:DeegreeParam>
 <!-- The online resource parameter must be set here: The 'deegree-wms'
 originates from the context file in apache under $Tomcat_home$/conf/Catalina/
 localhost/deegree-wms.xml or the name of the .war file under $Tomcat_home$/
 webapps/ depending on where the root directory of the service is located; the
 services url-pattern originates from the deployment descriptor under
 wms_home/WEB-INF/web.xml <url-pattern>/services</url-pattern> -->
 <deegree:DefaultOnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
 xlink:type="simple" xlink:href="http://localhost:8080/deegree-wms/services" />
 <!-- default = 100 (MB) -->
 <deegree:CacheSize>100</deegree:CacheSize>
 <!-- maximum lifetime of the processes in the WMS; default = 3600 (sec)-->
 <deegree:MaxLifeTime>3600</deegree:MaxLifeTime>
 <!-- maximum time for the execution of a request until an exception of time-
 exceed is thrown default 15 Sekunden -->
 <deegree:RequestTimeLimit>45</deegree:RequestTimeLimit>
 <!-- determines the quality of the map/image generated from a GetMap request.
 the parameter only will be evaluated if the desired format is able to handle it.
 the range of values is 0 ... 1, where 1 is best and 0 is worst. default is 0.95
 -->
 <deegree:MapQuality>0.95</deegree:MapQuality>
 <!-- maximum map width that can be requested. default = 1000 -->
 <deegree:MaxMapWidth>1000</deegree:MaxMapWidth>
 <!-- maximum map height that can be requested. default = 1000 -->
 <deegree:MaxMapHeight>1000</deegree:MaxMapHeight>
 <deegree:AntiAliased>true</deegree:AntiAliased>
 <!-- copyright note that will be drawn to the left bottom side of the maps ;
 you can also reference a graphic file using absolute path to file e.g.
 c:/images/mylogo.jpg -->
 <deegree:Copyright>deegree-WMS V.2.2 2008</deegree:Copyright>
 <!-- radius of the circle around the point a user has clicked to that will
 be considered for creating a search area for a GetFeatureInfo request;
 default = 5 -->
 <deegree:FeatureInfoRadius>10</deegree:FeatureInfoRadius>
 <!-- returns the URL where the DTD for OGC WMS capabilities DOCTYPE definition
 is located. default =
 http://schemas.opengis.net/wms/1.1.1/WMS_MS_Capabilities.dtd -->
 <deegree:DTDLocation>
 <deegree:OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
 xlink:type="simple" xlink:href="WMS_MS_Capabilities.dtd" />
 </deegree:DTDLocation>
 <!--deegree:DTDLocation>
 <deegree:OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
 xlink:type="simple"
 xlink:href="http://schemas.opengis.net/wms/1.1.1/WMS_MS_Capabilities.dtd"/>
 </deegree:DTDLocation-->
 <!-- define a proxy server if the WMS is separated by a firewall from the DTD-
 location or the locations of connected REMOTEWFS, REMOTEWCS and REMOTEWMS as an
 alternative you can set command line parameters at the call of the java
 interpreter like this: java -DproxyHost=131.220.106.104 -DproxyPort=1234
 -classpath ... -->
 <!--Proxy proxyHost="131.220.106.104" proxyPort="1234"/-->
 </deegree:DeegreeParam>
```

```
<Service>
 <Name>deegree wms</Name>
 <Title>deegree wms</Title>
 <!-- abstract und keywords are optional -->
 <Abstract>wms reference implementation</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>wms</Keyword>
 </KeywordList>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink" xlink:type="simple"
 xlink:href="http://localhost:8080/deegree-wms/services" />
 <!-- the following service information is optional -->
 <ContactInformation>
 <ContactPersonPrimary>
 <ContactPerson>Andreas Poth</ContactPerson>
 <ContactOrganization>lat/lon</ContactOrganization>
 </ContactPersonPrimary>
 <ContactPosition>Technical Director</ContactPosition>
 <ContactAddress>
 <AddressType>XXXX</AddressType>
 <Address>Aennchenstr. 19</Address>
 <City>Bonn</City>
 <StateOrProvince>NRW</StateOrProvince>
 <PostCode>53177</PostCode>
 <Country>Germany</Country>
 </ContactAddress>
 <ContactVoiceTelephone>0049228184960</ContactVoiceTelephone>
 <ContactFacsimileTelephone>00492281849629</ContactFacsimileTelephone>
 <ContactElectronicMailAddress>info@lat-lon.de</ContactElectronicMailAddress>
 </ContactInformation>
 <Fees>none</Fees>
 <AccessConstraints>none</AccessConstraints>
</Service>
<Capability>
 <Request>
 <!-- by default available -->
 <GetCapabilities>
 <!-- default and mandatory = application/vnd.ogc.wms_xml -->
 <Format>application/vnd.ogc.wms_xml</Format>
 <DCPType>
 <HTTP>
 <Get>
 <!-- If left empty, it will be automatically filled with the
 parameters of <deegree:DefaultOnlineResource> -->
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
 xlink:type="simple" xlink:href="http://localhost:8080/deegree-
 wms/services?" />
 </Get>
 <!-- POST isn't supported -->
 </HTTP>
 </DCPType>
 </GetCapabilities>
 <!-- by default available -->
 <GetMap>
 <!-- default = image/gif; image/png; image/jpg -->
 <!-- mandatory = image/png -->
 <Format>image/gif</Format>
 <Format>image/png</Format>
 <Format>image/jpg</Format>
 <Format>image/tif</Format>
 <Format>image/bmp</Format>
 <DCPType>
 <HTTP>
 <Get>
 <!-- If left empty, it will be automatically filled with the
```

```

 parameters of <deegree:DefaultOnlineResource> -->
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
 xlink:type="simple" xlink:href="http://localhost:8080/deegree-
 wms/services?" />
 </Get>
 <Post>
 <!-- If left empty, it will be automatically filled with the
 parameters of <deegree:DefaultOnlineResource> -->
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
 xlink:type="simple" xlink:href="http://localhost:8080/deegree-
 wms/services?" />
 </Post>
</HTTP>
</DCPType>
</GetMap>
<!-- by default available -->
<GetFeatureInfo>
 <!-- default & mandatory = application/vnd.ogc.gml -->
 <Format>application/vnd.ogc.gml</Format>
 <Format>text/plain</Format>
 <Format>text/html</Format>
 <DCPType>
 <HTTP>
 <Get>
 <!-- If left empty, it will be automatically filled with the
 parameters of <deegree:DefaultOnlineResource> -->
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
 xlink:type="simple"
 xlink:href="http://localhost:8080/deegree-wms/services?" />
 </Get>
 <!-- POST isn't supported -->
 </HTTP>
 </DCPType>
</GetFeatureInfo>
<GetLegendGraphic>
 <!-- default = image/gif; image/png; image/jpg -->
 <!-- mandatory = image/png -->
 <Format>image/gif</Format>
 <Format>image/png</Format>
 <Format>image/jpeg</Format>
 <Format>image/jpg</Format>
 <Format>image/tif</Format>
 <Format>image/bmp</Format>
 <DCPType>
 <HTTP>
 <Get>
 <!-- If left empty, it will be automatically filled with the
 parameters of <deegree:DefaultOnlineResource> -->
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
 xlink:type="simple" xlink:href="http://localhost:8080/deegree-
 wms/services?" />
 </Get>
 <!-- POST isn't supported -->
 </HTTP>
 </DCPType>
</GetLegendGraphic>
</Request>
<Exception>
 <!-- default & mandatory= application/vnd.ogc.se_xml -->
 <Format>application/vnd.ogc.se_xml</Format>
 <Format>application/vnd.ogc.se_inimage</Format>
 <Format>application/vnd.ogc.se_blank</Format>
</Exception>
<!-- optional; default = all false -->

```

```
<UserDefinedSymbolization SupportSLD="1" UserLayer="1" UserStyle="1"
RemoteWFS="0" />
<!-- cascaded is optional and default=false -->
<!-- TOP LAYER As you are able to nest Layers as in this example please always
make sure that you have the appropriate number of opening layer tags <Layer>
and closing layer tags </Layer> -->
<Layer queryable="0" cascaded="0" noSubsets="0"
xmlns:app="http://www.deegree.org/app">
 <Title>deegree 2.2 Demo WMS</Title>
 <!-- abstract and keyword are optional -->
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <!-- Just add the <SRS> you wish to support with your WMS -->
 <!-- default = EPSG:4326 -->
 <SRS>EPSG:4326</SRS>
 <SRS>EPSG:26912</SRS>
 <!-- It is strongly recommended that you set a LatLonBoundingBox; further
 <BoundingBox>es are optional. Each layer inherits the BBoxes of the parent
 layer as long as no explicit one is set. -->
 <LatLonBoundingBox miny="-90" maxy="90" minx="-180" maxx="180" />
 <!-- Setting <BoundingBox>es ist optional but if set be sure to set the
 correct SRS="EPSG:" -->
 <BoundingBox SRS="EPSG:26912" miny="3581352" maxy="5432672" minx="0"
 maxx="1504379" />
 <!-- default 0 ; 9E99 -->
 <ScaleHint min="0" max="100000000" />
 <!-- This layer has a <Name>, but no data source. You can request this layer
 anyway as you get all the sub-layers. You even get the GetFeatureInfo but your
 client must be able to handle the result as each sublayer has it's own
 properties (attributes). If you don't want this layer to be accessible remove
 the name tag (<Name>citelayers</Name>). -->

 <Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>citelayers</Name>
 <Title>Cite Layers</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
 xlink:type="simple" xlink:href="http://www.deegree.org" />
 </MetadataURL>

 <LatLonBoundingBox miny="-0.004" maxy="0.004" minx="-0.004" maxx="0.004" />
 <!-- queryable="1" sets whether GetFeatureinfo is enabled or not. for this
 layer if set to "1" at least one data source must be set to "1" too -->
 <Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <!-- please choose a unique <name> without spaces and a title which can
 contain more details -->
 <Name>BasicPolygons</Name>
 <Title>BasicPolygons</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
```



```

 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
 xlink:type="simple" xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <!-- sets the range of scale, where the WMS will serve data for this
 layer -->
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>10000000</MaxScaleDenominator>
 <!-- queryable="1" sets whether GetFeatureinfo is enabled or not for this
 datasource -->
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:BasicPolygons</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <!--name of the property that contains the geometries for this layer
 this element will only be solved if the data source is a LOCALWFS or a
 REMOTEWFS default = 'app:GEOM' -->
 <deegree:GeometryProperty>app:geom</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <!-- here you set the name of the style and the location
 <deegree:StyleResource> where the WMS will find a style definition with
 the same <UserStyle> name-->
 <Name>default:BasicPolygons</Name>
 <Title>default:BasicPolygons</Title>
 <deegree:StyleResource>cite_style.xml</deegree:StyleResource>
 <!-- If LegendURL is set for layers the default automated legend graphic
 will be overwritten -->
 <!--<LegendURL width="50" height="50">
 <Format>image/jpeg</Format>
 <OnlineResource xlink="http://www.w3.org/1999/xlink" type="simple"
 xlink:href="http://www.yourdomain.com/legend/teststyle.jpg"/>
 </LegendURL-->
 </Style>
</Layer>
<!-- The following Layer definition could be uses as default for your own
layers -->
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Bridges</Name>
 <Title>Bridges</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
 xlink:type="simple" xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Bridges</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geom</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:Bridges</Name>
 <Title>default:Bridges</Title>
 <deegree:StyleResource>cite_style.xml</deegree:StyleResource>
 </Style>
</Layer>
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Buildings</Name>
 <Title>Buildings</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>

```

```

 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Buildings</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geom</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:Buildings</Name>
 <Title>default:Buildings</Title>
 <deegree:StyleResource>cite_style.xml</deegree:StyleResource>
 </Style>
</Layer>
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>BuildingCenters</Name>
 <Title>BuildingCenters</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Buildings</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geom</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:BuildingCenters</Name>
 <Title>default:BuildingCenters</Title>
 <deegree:StyleResource>cite_style.xml</deegree:StyleResource>
 </Style>
</Layer>
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>DividedRoutes</Name>
 <Title>DividedRoutes</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:DividedRoutes</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geom</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:DividedRoutes</Name>

```

```

 <Title>default:DividedRoutes</Title>
 <degree:StyleResource>cite_style.xml</degree:StyleResource>
 </Style>
</Layer>

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Forests</Name>
 <Title>Forests</Title>
 <Abstract>degree demo WMS</Abstract>
 <KeywordList>
 <Keyword>degree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.degree.org" />
 </MetadataURL>
 <degree:DataSource failOnException="0" queryable="1">
 <degree:Name>app:Forests</degree:Name>
 <degree:Type>LOCALWFS</degree:Type>
 <degree:GeometryProperty>app:geom</degree:GeometryProperty>
 </degree:DataSource>
 <Style>
 <Name>default:Forests</Name>
 <Title>default:Forests</Title>
 <degree:StyleResource>cite_style.xml</degree:StyleResource>
 </Style>
</Layer>

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Lakes</Name>
 <Title>Lakes</Title>
 <Abstract>degree demo WMS</Abstract>
 <KeywordList>
 <Keyword>degree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.degree.org" />
 </MetadataURL>
 <degree:DataSource failOnException="0" queryable="1">
 <degree:Name>app:Lakes</degree:Name>
 <degree:Type>LOCALWFS</degree:Type>
 <degree:GeometryProperty>app:geom</degree:GeometryProperty>
 </degree:DataSource>
 <Style>
 <Name>default:Lakes</Name>
 <Title>default:Lakes</Title>
 <degree:StyleResource>cite_style.xml</degree:StyleResource>
 </Style>
</Layer>

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>MapNeatline</Name>
 <Title>MapNeatline</Title>
 <Abstract>degree demo WMS</Abstract>
 <KeywordList>
 <Keyword>degree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>

```

```

<MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
</MetadataURL>
<deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:MapNeatline</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geom</deegree:GeometryProperty>
</deegree:DataSource>
<Style>
 <Name>default:MapNeatline</Name>
 <Title>default:MapNeatline</Title>
 <deegree:StyleResource>cite_style.xml</deegree:StyleResource>
</Style>
</Layer>

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>NamedPlaces</Name>
 <Title>NamedPlaces</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:NamedPlaces</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geom</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:NamedPlaces</Name>
 <Title>default:NamedPlaces</Title>
 <deegree:StyleResource>cite_style.xml</deegree:StyleResource>
 </Style>
</Layer>

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Ponds</Name>
 <Title>Ponds</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Ponds</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geom</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:Ponds</Name>

```

```

 <Title>default:Ponds</Title>
 <degree:StyleResource>cite_style.xml</degree:StyleResource>
 </Style>
</Layer>

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>RoadSegments</Name>
 <Title>RoadSegments</Title>
 <Abstract>degree demo WMS</Abstract>
 <KeywordList>
 <Keyword>degree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.degree.org" />
 </MetadataURL>
 <degree:DataSource failOnException="0" queryable="1">
 <degree:Name>app:RoadSegments</degree:Name>
 <degree:Type>LOCALWFS</degree:Type>
 <degree:GeometryProperty>app:geom</degree:GeometryProperty>
 </degree:DataSource>
 <Style>
 <Name>default:RoadSegments</Name>
 <Title>default:RoadSegments</Title>
 <degree:StyleResource>cite_style.xml</degree:StyleResource>
 </Style>
</Layer>

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Streams</Name>
 <Title>Streams</Title>
 <Abstract>degree demo WMS</Abstract>
 <KeywordList>
 <Keyword>degree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.degree.org" />
 </MetadataURL>
 <degree:DataSource failOnException="0" queryable="1">
 <degree:Name>app:Streams</degree:Name>
 <degree:Type>LOCALWFS</degree:Type>
 <degree:GeometryProperty>app:geom</degree:GeometryProperty>
 </degree:DataSource>
 <Style>
 <Name>default:Streams</Name>
 <Title>default:Streams</Title>
 <degree:StyleResource>cite_style.xml</degree:StyleResource>
 </Style>
</Layer>
</Layer>

<!-- ++++++ -->
<!-- ++++++ Utah ++++++ -->
<!-- ++++++ -->

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>utah</Name>
 <Title>Utah Demo Data</Title>

```

```

<Abstract>deegree demo WMS</Abstract>
<KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
</KeywordList>
<MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
</MetadataURL>

<!-- Example for a LOCALWCS -->
<Layer queryable="0" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>orthoimage</Name>
 <Title>Ortho Imagery of Salt Lake City</Title>
 <SRS>EPSG:26912</SRS>
 <deegree:DataSource failOnException="0" queryable="0">
 <deegree:Name>saltlakesatellite</deegree:Name>
 <deegree:Type>LOCALWCS</deegree:Type>
 <deegree:OWSCapabilities>
 <deegree:OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="LOCALWCS_capabilities.xml" />
 </deegree:OWSCapabilities>
 <deegree:FilterCondition>
 <deegree:WCSRequest>
 <![CDATA[VERSION=1.0.0&coverage=saltlakesatellite&TRANSPARENT=TRUE&FO
RMAT=jpg&EXCEPTIONS=application/vnd.ogc.se_xml]]>
 </deegree:WCSRequest>
 </deegree:FilterCondition>
 </deegree:DataSource>
</Layer>

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>StateBoundary</Name>
 <Title>StateBoundary</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>10000000</MaxScaleDenominator>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:StateBoundary</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:StateBoundary</Name>
 <Title>default:StateBoundary</Title>
 <deegree:StyleResource>state_styles.xml</deegree:StyleResource>
 </Style>
</Layer>
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>StateOverview</Name>
 <Title>StateOverview</Title>
 <Abstract>deegree demo WMS</Abstract>

```

```

<KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
</KeywordList>
<MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
</MetadataURL>
<MinScaleDenominator>0</MinScaleDenominator>
<MaxScaleDenominator>100000000</MaxScaleDenominator>
<deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:StateBoundary</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
</deegree:DataSource>
<deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:CountyBoundaries_edited</deegree:Name>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
</deegree:DataSource>

<Style>
 <Name>default:StateOverview</Name>
 <Title>default:StateOverview</Title>
 <deegree:StyleResource>state_styles.xml</deegree:StyleResource>
</Style>
</Layer>

<!-- Example for a REMOTEWMS; cascaded="1" indicates, that this layer
originates from a remote WMS -->

<Layer queryable="1" cascaded="1" noSubsets="0" fixedWidth="0"
fixedHeight="0">
 <Name>landcover</Name>
 <Title>National Land Cover Dataset (JUST EPSG:4326 => WGS 84)</Title>
 <SRS>EPSG:4326</SRS>
 <LatLonBoundingBox miny="37.88970184326172" maxy="42.91960144042969"
 minx="-115.55139923095703" maxx="-110.41089630126953" />
 <deegree:DataSource failOnException="1" queryable="1">
 <deegree:Name>3:5</deegree:Name>
 <deegree:Type>REMOTEWMS</deegree:Type>
 <deegree:OWSCapabilities>
 <deegree:OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
xlink:href="http://columbo.nrlssc.navy.mil/ogcwms/servlet/WMSServlet/U
tah_State_University_Water_Initiative.wms?SERVICE=WMS&VERSION=1.1.
1&REQUEST=GetCapabilities" />
 </deegree:OWSCapabilities>
 <deegree:FilterCondition>
 <deegree:WMSRequest>
 <![CDATA[Version=1.1.1&FORMAT=image/png&TRANSPARENT=true&EXCEPTIONS
=application/vnd.ogc.se_inimage&BGOLOR=0xffffffff&LAYERS=3:5&STYLES=]]>
 </deegree:WMSRequest>
 </deegree:FilterCondition>
 <deegree:ScaleHint min="0.0" max="9999999999" />
 <!-- Specifies the color to be transparent -->
 <!-- <deegree:TransparentColors>
 <deegree:Color>#000000</deegree:Color>
 <deegree:Color>#FF0000</deegree:Color>
 </deegree:TransparentColors>-->
 </deegree:DataSource>
 </Layer>

```

```

<!-- This layer contains several sublayers and is able load these as one as
it has a <name> -->

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Administration</Name>
 <Title>Administration</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <!-- This layer contains 3 style definitions -->
 <Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Counties</Name>
 <Title>County Boundaries</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>10000000</MaxScaleDenominator>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:CountyBoundaries_edited</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:Counties</Name>
 <Title>default:Counties</Title>
 <deegree:StyleResource>county_styles.xml</deegree:StyleResource>
 </Style>
 <Style>
 <Name>GreyCounties</Name>
 <Title>GreyCounties</Title>
 <deegree:StyleResource>county_styles.xml</deegree:StyleResource>
 </Style>
 <Style>
 <Name>ColourfulCounties</Name>
 <Title>ColourfulCounties</Title>
 <deegree:StyleResource>county_styles.xml</deegree:StyleResource>
 </Style>
 </Layer>
 <!-- This Layer contains 2 style definitions -->
 <Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>ZipCodes</Name>
 <Title>Zip Codes</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">

```



```

 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>10000000</MaxScaleDenominator>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:ZipCodes</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:ZipCodes</Name>
 <Title>default:ZipCodes</Title>
 <deegree:StyleResource>state_styles.xml</deegree:StyleResource>
 </Style>
 <Style>
 <Name>ZipCodesPop</Name>
 <Title>ZipCodes Population</Title>
 <deegree:StyleResource>state_styles.xml</deegree:StyleResource>
 </Style>
</Layer>
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Municipalities</Name>
 <Title>Municipalities</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>1000000</MaxScaleDenominator>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Municipalities_edited</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:Municipalities</Name>
 <Title>default:Municipalities</Title>
 <deegree:StyleResource>state_styles.xml</deegree:StyleResource>
 </Style>
</Layer>
</Layer>

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Geology</Name>
 <Title>Geology</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />

```

```

</MetadataURL>
<!-- This layer contains OGC filter encoding. The WMS collects just
defined WFS properties (attributes) as well as selected content
(COAL*, PC, GF) -->
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>EnergyResources</Name>
 <Title>Energy Resources</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>10000000</MaxScaleDenominator>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:EnergyResources</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
 <deegree:FilterCondition>
 <wfs:Query typeName="app:EnergyResources"
xmlns:gml="http://www.opengis.net/gml"
 xmlns:ogc="http://www.opengis.net/ogc"
xmlns:wfs="http://www.opengis.net/wfs">
 <wfs:PropertyName>app:code</wfs:PropertyName>
 <!-- <wfs:PropertyName>app:shape_area</wfs:PropertyName> -->
 <wfs:PropertyName>app:shape_len</wfs:PropertyName>
 <ogc:Filter>
 <ogc:Or>
 <ogc:Or>
 <ogc:PropertyIsEqualTo>
 <ogc:PropertyName>app:code</ogc:PropertyName>
 <ogc:Literal>PC</ogc:Literal>
 </ogc:PropertyIsEqualTo>
 <ogc:PropertyIsEqualTo>
 <ogc:PropertyName>app:code</ogc:PropertyName>
 <ogc:Literal>GF</ogc:Literal>
 </ogc:PropertyIsEqualTo>
 </ogc:Or>
 <ogc:PropertyIsLike wildCard="*" singleChar="?" escape="\ ">
 <ogc:PropertyName>app:code</ogc:PropertyName>
 <ogc:Literal>COAL*</ogc:Literal>
 </ogc:PropertyIsLike>
 </ogc:Or>
 </ogc:Filter>
 </wfs:Query>
 </deegree:FilterCondition>
 </deegree:DataSource>
 <Style>
 <Name>default:EnergyResources</Name>
 <Title>default:EnergyResources</Title>
 <deegree:StyleResource>geology_styles.xml</deegree:StyleResource>
 </Style>
</Layer>
</Layer>

<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Hydrology</Name>
 <Title>Hydrology</Title>

```

```

<Abstract>deegree demo WMS</Abstract>
<KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
</KeywordList>
<MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
</MetadataURL>
<!-- Layer containing points -->
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Springs</Name>
 <Title>Springs of Utah</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>10000000</MaxScaleDenominator>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Springs</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:Springs</Name>
 <Title>default:Springs</Title>
 <deegree:StyleResource>hydro_styles.xml</deegree:StyleResource>
 </Style>
</Layer>
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Lake</Name>
 <Title>Lakes of Utah</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>10000000</MaxScaleDenominator>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Lake</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:Lake</Name>
 <Title>default:Lake</Title>
 <deegree:StyleResource>hydro_styles.xml</deegree:StyleResource>
 </Style>

```

```

 </Layer>
 </Layer>

 <Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>LandUse</Name>
 <Title>Land Use</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Vegetation</Name>
 <Title>Dominant Vegetation</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>10000000</MaxScaleDenominator>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Vegetation</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:Vegetation</Name>
 <Title>default:Vegetation</Title>
 <deegree:StyleResource>vegie_styles.xml</deegree:StyleResource>
 </Style>
 </Layer>
 </Layer>

 <Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Infrastructure</Name>
 <Title>Infrastructure</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Airports</Name>
 <Title>Airports of Utah</Title>

```

```

<Abstract>deegree demo WMS</Abstract>
<KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
</KeywordList>
<MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
</MetadataURL>
<MinScaleDenominator>0</MinScaleDenominator>
<MaxScaleDenominator>10000000</MaxScaleDenominator>
<deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Airports</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
</deegree:DataSource>
<Style>
 <Name>default:Airports</Name>
 <Title>default:Airports</Title>
 <deegree:StyleResource>infra_styles.xml</deegree:StyleResource>
</Style>
</Layer>
<!-- This layer has 2 data sources used for different scales -->
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Roads</Name>
 <Title>Roads of Utah</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>10000000</MaxScaleDenominator>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Roads</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
 <deegree:ScaleHint min="0.0" max="50" />
 </deegree:DataSource>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Roads500</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
 <deegree:ScaleHint min="50.0" max="2000000000" />
 </deegree:DataSource>
 <Style>
 <Name>default:Roads</Name>
 <Title>default:Roads</Title>
 <deegree:StyleResource>roads_styles.xml</deegree:StyleResource>
 </Style>
</Layer>
<Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Railroads</Name>
 <Title>Railroads of Utah</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>

```

```

 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>10000000</MaxScaleDenominator>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Railroads</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:geometry</deegree:GeometryProperty>
 </deegree:DataSource>
 <Style>
 <Name>default:Railroads</Name>
 <Title>default:Railroads</Title>
 <deegree:StyleResource>infra_styles.xml</deegree:StyleResource>
 </Style>
 </Layer>
 </Layer>

 <Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>Elevation</Name>
 <Title>Elevation</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <!-- This layer has 3 data sources used for different scale ranges;
 additional the geometry property is renamed in the featuretype definition
 file to contourLine which has also to be set in the featuretype
 definition -->
 <Layer queryable="1" noSubsets="0" fixedWidth="0" fixedHeight="0">
 <Name>ElevationContours</Name>
 <Title>Elevation Contours</Title>
 <Abstract>deegree demo WMS</Abstract>
 <KeywordList>
 <Keyword>deegree</Keyword>
 <Keyword>layer</Keyword>
 </KeywordList>
 <MetadataURL type="ISO19115:2003">
 <Format>text/html</Format>
 <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
 xlink:href="http://www.deegree.org" />
 </MetadataURL>
 <MinScaleDenominator>0</MinScaleDenominator>
 <MaxScaleDenominator>10000000</MaxScaleDenominator>
 <deegree:DataSource failOnException="0" queryable="1">
 <deegree:Name>app:Contours500Ft</deegree:Name>
 <deegree:Type>LOCALWFS</deegree:Type>
 <deegree:GeometryProperty>app:contourLine</deegree:GeometryProperty>
 <deegree:ScaleHint min="0.0" max="150" />
 </deegree:DataSource>
 <deegree:DataSource failOnException="0" queryable="1">

```

```

 <degree:Name>app:Contours1000Ft</degree:Name>
 <degree:Type>LOCALWFS</degree:Type>
 <degree:GeometryProperty>app:contourLine</degree:GeometryProperty>
 <degree:ScaleHint min="150" max="500" />
 </degree:DataSource>
 <degree:DataSource failOnException="0" queryable="1">
 <degree:Name>app:Contours2500Ft</degree:Name>
 <degree:Type>LOCALWFS</degree:Type>
 <degree:GeometryProperty>app:contourLine</degree:GeometryProperty>
 <degree:ScaleHint min="500" max="200000000" />
 </degree:DataSource>
 <Style>
 <Name>default:ElevationContours</Name>
 <Title>default:ElevationContours</Title>
 <degree:StyleResource>general_styles.xml</degree:StyleResource>
 </Style>
</Layer>
</Layer>
</Layer>
</Capability>
</WMT_MS_Capabilities>

```

## Appendix B Supported SRS

Change to `$wms_home$/WEB-INF/conf/tools` and execute:

Linux:

```
java -cp ../../lib/deegree2.jar:../../lib/jai/jai_core.jar org.deegree.tools.srs.SRSInfo
```

Windows

```
java -cp ../../lib\deegree2.jar;../../lib\jai\jai_core.jar org.deegree.tools.srs.SRSInfo
```

and you'll get a list of supported CRS in deegree. All CRS that are supported in proj4 are now also supported by deegree.



## Appendix C Featuretype Definition

Path of file: \$wms\_home\$/WEB-INF/conf/wms/featuretypes/

```
<xsd:schema targetNamespace="http://www.deegree.org/app"
xmlns:gml="http://www.opengis.net/gml"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:deegreewfs="http://www.deegree.org/wfs"
 xmlns:ogc="http://www.opengis.net/ogc" xmlns:app="http://www.deegree.org/app"
 elementFormDefault="qualified" attributeFormDefault="unqualified">
 <xsd:import namespace="http://www.opengis.net/gml"
 schemaLocation="http://schemas.opengis.net/gml/3.1.1/base/feature.xsd" />
 <xsd:import namespace="http://www.opengis.net/gml"
 schemaLocation="http://schemas.opengis.net/gml/3.1.1/base/geometryAggregates.xsd"
 />

 <!-- configuration for the persistence backend to be used -->
 <!-- This is an automatic generated and manually extended featuretype definition
 file; it associates the WFS with the data source(s) -->
 <xsd:annotation>
 <xsd:appinfo>
 <deegreewfs:Prefix>app</deegreewfs:Prefix>
 <!--Please refer the demo definitons for accessing HSQLDB, SHAPE or POSTGIS
 -->
 <deegreewfs:Backend>SHAPE</deegreewfs:Backend>
 <!-- This part varies depending on the backend type -->
 <deegreewfs:File>../../../../data/utah/vector/SGID500_EnergyResourcesPoly</deegr
 eewfs:File>
 <!-- This will NOT be set by the creation-tool so please insert the EPSG:
 appropriate to your data source SRS -->
 <deegreewfs:DefaultSRS>EPSG:26912</deegreewfs:DefaultSRS>
 </xsd:appinfo>
 </xsd:annotation>
 <!-- ===== -->
 <!-- here the featuretype name is set, under which the WFS will offer this
 featuretype. If you would
 like to change this featuretype name make sure, you also change the
 type='app:EnergyResourcesType' and
 further below the <xsd:complexType name='EnergyResourcesType'> -->
 <xsd:element name='EnergyResources' type='app:EnergyResourcesType'
 substitutionGroup="gml:_Feature">
 <xsd:annotation>
 <xsd:appinfo>
 <!-- the referenced table in the DB/file is set here. Never change this
 unless your sources have changed -->
 <deegreewfs:table>SGID500_EnergyResourcesPoly</deegreewfs:table>
 <!-- The unique identifier is set here -->
 <deegreewfs:gmlId prefix="ID_">
 <deegreewfs:MappingField field='ID' type="INTEGER" />
 </deegreewfs:gmlId>
 </xsd:appinfo>
 </xsd:annotation>
 </xsd:element>
 <!-- ===== -->
 <xsd:complexType name='EnergyResourcesType'>
 <xsd:complexContent>
 <xsd:extension base="gml:AbstractFeatureType">
 <xsd:sequence>
 <!-- The geometry property will be mapped here. As you can see property
 naming (<xsd:element name='geometry'> can differ from the source name
 (<deegreewfs:MappingField field='GEOM' ...> -->
 <xsd:element name='geometry' type='gml:GeometryPropertyType'>
 <xsd:annotation>
 <xsd:appinfo>
```

```

 <deegreevfs:Content>
 <!-- the srs='' specifies how the data is saved in the
 database/file table; for shapes this is -1 as this information is
 not stored in the file, for databases it depends on the DB-
 Settings, whether the SRS is saved or not. -->
 <deegreevfs:MappingField field='GEOM' type='GEOMETRY' srs='-1' />
 </deegreevfs:Content>
 </xsd:appinfo>
</xsd:annotation>
</xsd:element>
<xsd:element name='objectid' type='xsd:integer'>
 <xsd:annotation>
 <xsd:appinfo>
 <deegreevfs:Content>
 <deegreevfs:MappingField field='OBJECTID' type='NUMERIC' />
 </deegreevfs:Content>
 </xsd:appinfo>
 </xsd:annotation>
</xsd:element>
<!-- You are free to change the order of the elements or reduce the number
-->
<!-- By default the featurtypedefinition creation tool sets no
minOccurs=""; minOccurs sets whether or not a property CAN (minOccurs="0")
or MUST (minOccurs="1") be delivered by WFS. If not set minOccurs="1". In
the demos the EnergyResources WMS layer uses this mechanism as it has a
filter condition which requests just some properties of the WFS -->
<xsd:element name='code' type='xsd:string' minOccurs="0">
 <xsd:annotation>
 <xsd:appinfo>
 <deegreevfs:Content>
 <deegreevfs:MappingField field='CODE' type='VARCHAR' />
 </deegreevfs:Content>
 </xsd:appinfo>
 </xsd:annotation>
</xsd:element>
<xsd:element name='shape_area' type='xsd:double' minOccurs="0">
 <xsd:annotation>
 <xsd:appinfo>
 <deegreevfs:Content>
 <deegreevfs:MappingField field='SHAPE_AREA' type='NUMERIC' />
 </deegreevfs:Content>
 </xsd:appinfo>
 </xsd:annotation>
</xsd:element>
<xsd:element name='shape_len' type='xsd:double' minOccurs="0">
 <xsd:annotation>
 <xsd:appinfo>
 <deegreevfs:Content>
 <deegreevfs:MappingField field='SHAPE_LEN' type='NUMERIC' />
 </deegreevfs:Content>
 </xsd:appinfo>
 </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:schema>

```

## Appendix D Deployment Descriptor (web.xml)

Path of file: \$wms\_home\$/WEB-INF/

```
<?xml version="1.0"?>

<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app>
 <display-name>deegree 2.2</display-name>
 <description>deegree 2.2 OWS</description>
 <servlet>
 <servlet-name>owservice</servlet-name>
 <servlet-class>org.deegree.enterprise.servlet.OGCServletController</servlet-
class>

 <init-param>
 <param-name>services</param-name>
 <param-value>wms,wfs</param-value>
 <description>
 list of supported services, e.g.: wfs,wms,wcs (comma separated) allways use
 lowercase
 </description>
 </init-param>

 <!-- WMS INITIALIZING PARAMETERS -->
 <init-param>
 <param-name>wms.handler</param-name>
 <param-value>org.deegree.enterprise.servlet.WMSHandler</param-value>
 </init-param>
 <init-param>
 <param-name>wms.config</param-name>
 <param-value>WEB-INF/conf/wms/wms_configuration.xml</param-value>
 </init-param>

 <!-- WFS INITIALIZING PARAMETERS -->
 <init-param>
 <param-name>wfs.handler</param-name>
 <param-value>org.deegree.enterprise.servlet.WFSHandler</param-value>
 </init-param>
 <init-param>
 <param-name>wfs.config</param-name>
 <param-value>WEB-INF/conf/wms/LOCALWFS_capabilities.xml</param-value>
 </init-param>

 <!-- WCS INITIALIZING PARAMETERS -->
 <!--
 <init-param>
 <param-name>wcs.handler</param-name>
 <param-value>org.deegree.enterprise.servlet.WCSHandler</param-value>
 </init-param>
 <init-param>
 <param-name>wcs.config</param-name>
 <param-value>WEB-INF/conf/wcs/LOCALWCS_capabilities.xml</param-value>
 </init-param>
 -->
 <load-on-startup>1</load-on-startup>
 </servlet>
 <servlet-mapping>
 <servlet-name>owservice</servlet-name>
 <url-pattern>/services</url-pattern>
 </servlet-mapping>
 <welcome-file-list>
 <welcome-file>/index.jsp</welcome-file>
 </welcome-file-list>
</web-app>
```

```
</welcome-file-list>
<error-page>
 <error-code>500</error-code>
 <location>/error.jsp</location>
</error-page>
<error-page>
 <exception-type>org.deegree.ogcwebservices.OGCWebServiceException</exception-
 type>
 <location>/error.jsp</location>
</error-page>
</web-app>
```