deegree CRS library

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
07/05/08	Initial version	Rutger Bezema

Table of Contents

Index of Tables

Illustration Index

1 Introduction

The deegree Java framework offers 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). The deegree framework encompasses OGC web services, clients as well as a large variety of GIS-application software. It is free and open 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 the configuration and usage of the CRS (Coordinate Reference System) library, which allows the projection of coordinates as well as the transformation of coordinates between different CRS's.

Besides the CRS library, 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

The deegree CRS package offers great flexibility regarding its configuration and adaptation to user defined projections and transformation as well as allowing user defined Coordinate System components like ellipsoid, axis or datums. The configuration of the CRS package requires editing an XML file.

2 Download / Installation

2.1 Prerequisites

To use the deegree2 CRS library you need following libraries:

- Java (JRE or JSDK) version 1.5.x
- vecmath.jar (version > 1.5) - needed for various calculations on matrices and vectors
- commons-logging.jar - commons logging implementation
- log4j-1.2.9.jar – specific logging implementation
- jaxen-1.1-beta-8.jar – needed for the evaluation of xpaths
- xerces_2_5_0.jar – for parsing of the xml configuration document

For installation instructions on java 1.5.x please refer to the corresponding documentation at java.sun.com. The other libraries can be downloaded from the deegree repository at http://deegree.wald.intevation.org

2.2 deegree CRS releases

Because deegree CRS still is work in progress, two versions with different behaviour currently exist, 2.2_testing and unstable.

The most differences are to found in the configuration: the 2.2 testing branch (short to be stable release deegree 2.2, and here after called deegree2.2) supports a different layout of the configuration file as the current trunk (unstable).

Another difference between the 2.2 release and the current trunk is the lack of the stereo graphic projection as defined by the EPSG database. The stereographic projection supported in deegree2.2 is implemented after Snyder, which can lead up to difference of a few meters with the one suggested by EPSG.

The last remaining difference between the two implementations is the ability to integrate user-defined projections and transformations, which is not possible in deegree2.2.

2.3 Installing the package

There are different ways of installing the deegree CRS library. The easiest way is creating a directory say 'crs'. Download the deegree2.jar of your likings (either deegree2.2.jar or a nightly_build) and copy it with the above mentioned libraries into your newly created directory.

In the future we might create a complete package which will contain the jars, this documentation and different scripts to let you interact with the deegree CRS library, by calling the appropriate tools, like transforming and projecting coordinates, features or coverage or find the definition of some coordinate system.

2.4 Testing the installation

The deegree library comes with a few tools to interact with the actual CRS library. They can be used to verify if your installation has been successful. We will use the tool SRSInfo to check if the coordinate system EPSG:31466 (DHDN/Gauss-Kruger zone 2) is available in your deegree installation (it should be available). This test assumes you created your own installation and have no scripts available.

Start by opening a command shell:

On a windows based system:

- Press Start->Execute and type cmd
- Change dir (cd) to your newly created directory for example:
  - cd c:\crs
- type:
  - java -cp deegree2.jar org.deegree.tools.srs.SRSInfo -isAvailable EPSG:31466

On a linux based system:

- Open a shell (see your window manager documentation to open an xshell, probably under the main menu->applications xterm, gnometerm or kterm)
- Change dir (cd) to your newly created directory for example:
  - cd /path/to/crs
- type:
  - java -cp deegree2.jar org.deegree.tools.srs.SRSInfo -isAvailable EPSG:31466

The expected response is:

- Coordinates System: EPSG:31466 is available in deegree

2.4.1 Errors and Warning

Following is an (incomplete) list of possible errors and warnings, indicating something went wrong with your set-up:

- Exception in thread "main" java.lang.NoClassDefFoundError: org/deegree/tools/srs/SRSInfo
  - The deegree2.jar library is not on your classpath, please go to your installation directory and see if the deegree2.jar is present
- Exception in thread "main" java.lang.NoClassDefFoundError: org/apache/commons/logging/LogFactory or any other class
  - Are all above libs in your current directory? If, for some reason, you put them in another directory you must put them on the classpath (the -cp parameter). Imagine for example you put vecmath.jar in a directory below 'crs' called 'lib'. Your Java command should be adopted to the following (notice the colon ':' on linux systems and the semicolon ';' on windows systems):
    - java -cp deegree2.jar:lib/vecmath.jar org.deegree.tools.srs.SRSInfo -isAvailable EPSG:31466
- Coordinates System: EPSG:31466 is **not** available in deegree
  - could it be you misspelled 'EPSG:31466' ? If not please redownload the current deegree2.jar and overwrite your current one. And try again.

If none of the above works for you, please feel free to post a message on the deegree users list on deegree-users@lists.sourceforge.net

3 Architecture

The deegree2 CRS package is designed to be independent of the rest of the deegree2 framework it can therefore be easily used in other GIS-projects which need to transform and/or project coordinates in Java.

Figure 1 shows the simplified internal architecture of the deegree2 crs components and a common workflow of getting a coordinate system from the package.

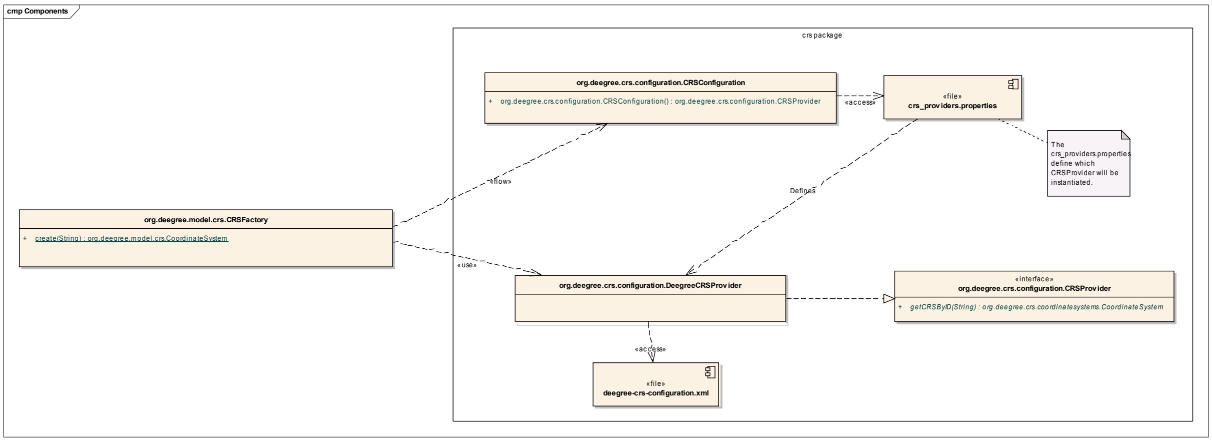

Figure 1: Shows the simplified architecture of the crs configuration.

To transform coordinates from one crs into another, you need to create both Coordinatesystems as well as a transformation path between the two. The deegree **model**[1] framework supplies the mechanisms for doing so.

3.1 Getting a configured coordinate system

As figure 1 shows, the CRSFactory can be given an identifier (for example EPSG:4326) of a coordinate system. The CRSFactory calls the org.deegree.crs.configuration.CRSConfiguration to determine the backend of the crs framework, by reading the classname of the CRSProvider from a file called crs_provider.properties. The instantiated CRSprovider (default = DeegreeCRSProvider) will in its turn be given the CRS identifier (e.g EPSG:4236). The Provider will try to read the definitions out of the deegree-crs-configuration.xml file and (on success) returns an instance containing the configured coordinate system.

[1]The org.deegree.model framework is the interface of deegree to the crs package.

3.2 Creating a transformation chain

After the successful retrieval of both source and target coordinate systems, a transformation chain (the concatenation of the operations necessary to transform coordinates from ones system into the other) needs to be created. The easiest way to transform coordinates is by using one of the transform method supplied by the org.deegreee.model.GeoTransformer class. Which must be instantiated with the target CRS.

This class will use the org.deegree.tranformations.TransformationFactory which will investigate the source and target Coordinate System and will create the chain accordingly.

3.3 Transformation tools

It is clear, that most of the use cases will be the transformation of existing files like Raster and/or Shape files without the will to write a scrap of java code. For this reason deegree2 supplies some tools to do the above steps.

3.3.1 Transforming shape files

The transforming of a shape file can be done with the org.deegree.tools.srs.TransformShapeFile tool.

The tool supports following parameters, (braces [] denote optional parameters):

1. -inCRS, the coordinate system of the source file
2. -outCRS, the target coordinate system, to which the source file will be transformed
3. -inFile, the shape file to transform.
4. [-outFile], the output file, if omitted the input file name will be extended with .outCRS

For example to transform the file with name original_shape_file.shp which is defined in EPSG:31466 to a shapefile defined in EPSG:4326 one should do the following:

- Change dir (cd) to your install directory for example:
  - cd /path/to/crs
- type:
  - java -cp deegree2.jar org.deegree.tools.srs.TransformShapeFile -inCRS EPSG:31466 -outCRS EPSG:4326 -inFile original_shape_file.shp

3.3.2 Transforming raster files

The transforming of a raster file (for example a .tiff or a .png file) can be done with the org.deegree.tools.srs.TransformRasterFile tool.

This tool supports following parameters, (braces [] denote optional parameters):

1. -inCRS, the coordinate system of the source file
2. -outCRS, the target coordinate system, to which the source file will be transformed
3. -inFile, the raster file to transform.
4. [-outFile], the output file, if omitted the input file name will be extended with .outCRS
5. [-imageQuality], the quality of the output image, which defaults to 0.95
6. [-passpointGridSize], the root of the number of sample points to which a polynomial function will be calculated. Defaults to 5, which means, that 25 sample points of the virtual target image envelope will be mapped onto the original envelope.
7. [-polynomOrder], the order of the polynomial to use for warping from the virtual target envelope to the original envelope, defaults to a cubic value (3)

For example to transform the file with name original_raster_file.tiff which is defined in EPSG:31466 to a raster file defined in EPSG:4326 with 100 sample points and a square polynomial warp of one should do the following:

- Change dir (cd) to your install directory for example:
  - cd /path/to/crs
- type:
  - java -cp deegree2.jar org.deegree.tools.srs.TransformRasterFile -inCRS EPSG:31466 -outCRS EPSG:4326 -inFile original_raster_file.tiff -passpointGridSize 10 -polynomOrder 2

3.3.3 Transforming a single coordinate Pair

One other tool is available to transform coordinates from a file or from standard in. This tool is not available in deegree2.2 but will be in future releases, it is called org.deegree.tools.DemoCRSTransform, it supports following parameters, (braces [] denote optional parameters):

1. -sourceCRS, the coordinate system of the incoming coordinates
2. -targetCRS, the coordinate system to transform the coordinates to.

3. [-sourceFile], the file to read the coordinates from, if omitted the -coord parameter is required. This parameter has higher priority then the -coord parameter, which means that if it is supplied, the -coord parameter will be neglected. If supplied following parameters can be used:
    1. [-coordSep], the coordinate separator, indicating the character which separates the coordinates in a coordinate pair. Defaults to a comma (','). For example a valid file using the comma separator would contain following coordinate pairs:

       2.880, 42.822

       4.321, 33.32

       Invalid would be something like this:

       2.880; 42.822 (using a semicolon)

       4.321 33.32 (using a space)
4. [-coord], Only valid if the -sourceFile parameter is omitted. This parameter accepts a coordinate pair/triplet separated with a comma e.g -coord “1.8,9.7,78”

For example to transform the file with name coordinates_file.txt which contains coordinates defined in EPSG:31466 and are separated with a colon, to coordinates defined in EPSG:4326, one should do the following:

- Change dir (cd) to your install directory for example:
    - cd /path/to/crs
- type:
    - java -cp deegree2.jar org.deegree.tools.srs.TransformRasterFile -sourceCRS EPSG:31466 -targetCRS EPSG:4326 -sourceFile coordinates_file.txt -coordSep ':'

4 Basic configuration

As can be gathered from figure 1 the CRS package can be configured to have different CRSProviders. A crs provider can be understood as backend to a specific coordinate system configuration file (similar to a wfs-datastore), which is responsible for the decoding of the projection and transformation parameters into the appropriate deegree crs classes, resulting in a coordinate system.

Currently two different backends are supported the DeegreeCRSProvider and the PROJ4CRSProvider. The default implementation uses the DeegreeCRSProvider it has it's own configuration file format, which will be explained in chapter 4.2. How to enable other providers will be handled in chapter 4.1

4.1 Choosing a CRSProvider

A lot of different coordinate system definitions file formats currently exists, some examples are the EPSG-database, proj4, wkt (Well Known Text), gml:dictionaries and quite a lot more. It is of great advantage to be able to use them all, which is why we designed the deegree crs package to be able to handle them all, if a specific implementation is required. What is needed is a so called CRSProvider which understands the wanted file format. Once a Provider is implemented the crs library can be configured to use it as it's backend.

To configure the crs library to use the CRSProvider you can use a file named crs_providers.properties into the root directory of your deegree2.jar (your installation directory). This file must contain following key value pairs (kvp)[2]

- CRS_PROVIDER=org.deegree.crs.configuration.DeegreeCRSProvider
- CRS_FILE=

As stated above currently only two CRSProviders are supported, the DeegreeCRSProvider and the PROJ4CRSProvider.

Lets assume you would want to use the PROJ4CRSProvider instead of the default (DeegreeCRSProvider) the crs_provider.properties should contain following kvp's

- CRS_PROVIDER=org.deegree.crs.configuration.PROJ4CRSProvider
- CRS_FILE=c:/windows/proj4/nad/epsg

If you have chosen to use the default CRSProvider, you may leave the CRS_FILE property empty. The DeegreeCRSProvider will then search for the file "deegree-crs-config.xml". The configuration of this file will be explained in the next chapter.

[2]As is common in property files, all lines starting with a root character (#) will be ignored (handled as comments).

4.2 The deegree crs configuration document

The standard implementation of the deegree crs library uses the DeegreeCRSProvider as backend for the projection/transformation parameters. This provider (as explained in chapter 4.1) uses a deegree crs definitions file[3] which is an xml file, it's default name is deegree-crs-configuration.xml. It is located in the deegree2.jar under org/deegree/crs/configuration/deegree-crs-configuration.xml. It is possible to add your own definitions by editing this file[4] or by supplying your own deegree-crs-configuration.xml into the install directory.

The root Element of the deegree-crs-configuration file must be a <definitions> element bound to the **http://www.deegree.org/crs** namespace as following:

```
<crs:definitions xmlns:crs="http://www.deegree.org/crs">
```

The basic idea behind the deegree definition file is presented in figure 2. It illustrates that all direct child element of the definitions element must have following properties:

1. They are **identifiable**, which means, each of these elements must have at least one Identifier, and can have a name, description, area of use and a version.
2. A coordinate system element (indirectly) holds a reference to some datum element
3. A datum element holds references to
    1. an Ellipsoid element,
    2. a toWGS84 parameters element
    3. a Prime Meridian element

This layout makes it possible to reuse already defined elements without having to explicitly redefine them. For example the dutch “amersfoort” datum EPSG:6289 uses the same ellipsoid (bessel) EPSG:7004 as the german “hauptdreieck” datum EPSG:6314. Which is why they both reference the same ellipsoid with the following statement:

```
<crs:usedEllipsoid>EPSG:7004</crs:usedEllipsoid>
```

---

[3] We found the creation of a new file format (rather than using an old one) convenient, because we are not aware of an existing format which on the one side allows for the reusing of defined crs-components and on the other hand is not cluttered up with unnecessary information.

[4] It is also possible to create your own file with a different name. You must configure the Provider to use it though, see chapter 4.1 for more details.

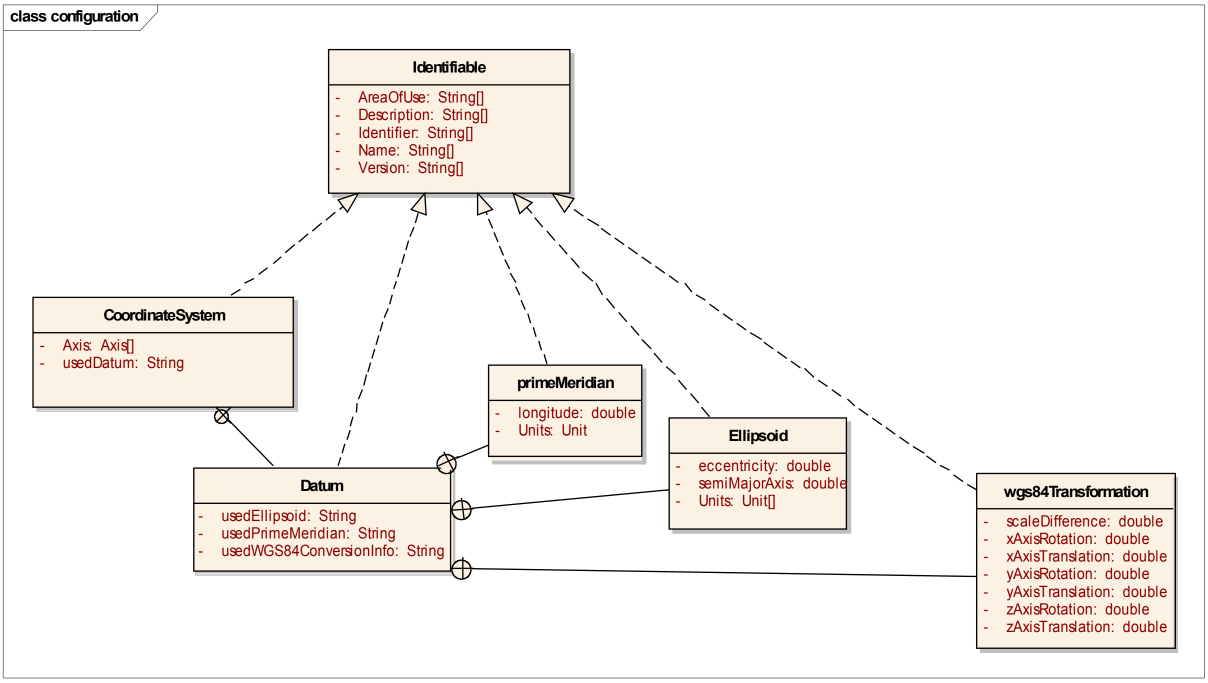

Figure 2: The relationships between the configuration elements

4.2.1 The identifiable elements

The above mentioned types are defined to be identifiable, which is why they all inherit from an abstract Identifiable type. Because this type is abstract an actual instantiation can not be created, but for exemplary purposes a virtual (**not valid**) <crs:Identifiable> element will be illustrative:

```
<crs:Identifiable>
  <crs:id>FirstID</crs:id>
  <crs:id>URN:OGC:DEF:CRS:NAMESPACE::97664884</crs:id>
  <crs:name>Optional Identifiable name</crs:name>
  <crs:version>2008-1-16T9:49</crs:version>
  <crs:version>Another version</crs:version>
  <crs:description>Explain the elements of an Identifiable type</crs:description>
  <crs:areaOfUse>NorthPole</crs:areaOfUse>
</crs:Identifiable>
```

This (**not valid**) element is associated with two different identifiers:

Be careful, the identifiers are cAsE sEnSiTiVe, and must be unique and unambiguous in the complete configuration file.

Each Identifiable Element **must have** at least one <crs:id>, but can have multiple identifiers as demonstrated above. Following are the optional name, two optional versions, the optional description and an optional area of use. Each of the optional elements may be declared indefinite.

4.2.2 Creating coordinate system elements

Taking a closer look at the definition for a coordinate system, it is evident, that all coordinate systems have certain similarities. This is why the deegree crs configuration uses a base class for all coordinate systems:

```
<xs:complexType name="AbstractCRS" abstract="true">
```

This complex type defines an axisOrder and the actual Axis which are common to all coordinate systems, furthermore it extends the identifiable type which is why all coordinate systems must have at least one identifier and can have different names, descriptions, areas-of-use and versions (see The identifiable elements chapter 4.2.1.)

As with the Identifiable example this type is also declared as abstract, no real instantiation can validly be created, but for exemplary purposes an AbstractCRS element with the common sub elements is displayed:

```
<crs:abstracCRS>
  <crs:id>EPSG:4691</crs:id>
  <crs:id>URN:OGC:DEF:CRS:EPSG::4691</crs:id>
  <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#4691</crs:id>
  <crs:id>URN:OPENGIS:DEF:CRS:EPSG::4691</crs:id>
  <crs:name>Abstract crs 87</crs:name>
  <crs:version>2008-1-16T9:49</crs:version>
  <crs:description>Handmade abstract crs, which is not valid.</crs:description>
  <crs:Axis>
    <crs:name>longitude</crs:name>
    <crs:units>degree</crs:units>
    <crs:axisOrientation>east</crs:axisOrientation>
  </crs:Axis>
  <crs:Axis>
    <crs:name>latitude</crs:name>
    <crs:units>degree</crs:units>
    <crs:axisOrientation>north</crs:axisOrientation>
  </crs:Axis>
  <crs:axisOrder>longitude, latitude</crs:axisOrder>
</crs:abstractCRS>
```

The abstractCRS element demonstrates following elements, first of all it must have at least one (at most three) <crs:Axis> defining a <crs:name>, <crs:units> and the orientation of the axis (it is therefore currently assumed that the axis are perpendicular to each order).

The <crs:name> is important, because it is used to define the <crs:axisOrder> element. Which –as can be interpreted from the name– defines the order of incoming coordinate tuples.

The <crs:units> element gives information about the units used for this crs. Because the deegree2 crs library internally works with metres and radians, this value is needed to convert from other measurable systems. Following table 1 shows which units are currently supported (the table is showing which units can (+) or cant (–) be converted into the internal used units of radians and metres):

Can convert to	**radians**	**metres**
metres	–	+
britishyard	–	+
usfoot	–	+
radian	+	–
degree	+	–
arc_sec	+	–

Table 1: Showing currently valid units and there if they can be converted to the deegree internal units.

The <crs:axisOrientation> element allows the definitions of an axis direction, currently only perpendicular (North-South, East-West, Front-Back, Up-Down, Perpendicular[5] and Other) axis are supported.

Currently following derived coordinate systems are defined, and can thus be defined in the crs-configuration:

```
1. <xs:element name="geographicCRS" type="crs:GeographicCRSType"/>
2. <xs:element name="projectedCRS" type="crs:ProjectedCRSType"/>
3. <xs:element name="geocentricCRS" type="crs:GeocentricCRSType"/>
4. <xs:element name="compoundCRS" type="crs:CompoundCRSType"/>
```

4.2.2.1 Creating a geographic coordinate system

A geographic reference system is based on a geodetic datum with coordinates given in geodetic longitude (λ – eastwards) and geodetic latitude (φ – northwards). Such a coordinate system sometimes provides a supplemental value perpendicular to the ellipsoid's surface, a so called 'height' value. To use these kind of coordinate tuples (λ, φ, h), you can define a compound crs explained in chapter 4.2.2.3.

The only needed supplemental information is the datum this geographic crs is defined upon. This can be accomplished by using the <crs:usedDatum> element, referencing the datum definition in the configuration file.

Attention it is up to the administrator of the configuration to make sure the identifier supplied in the <crs:usedDatum> element, points to an existing <crs:GeographicDatum> element. (See Creating a geodetic datum element chapter 4.2.3 on information about the configuration of datums). It is allowed to use any one of it's identifiers.

[5]The Perpendicular axis is used as the height axis in a compoundCRS.

Together with the information of the abstract crs type a valid example of a crs:GeographicCRS element can be put together:

```
<crs:geographicCRS>
  <crs:id>EPSG:4314</crs:id>
  <crs:id>URN:OGC:DEF:CRS:EPSG::4314</crs:id>
  <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#4314</crs:id>
  <crs:id>URN:OPENGIS:DEF:CRS:EPSG::4314</crs:id>
  <crs:name>DHDN</crs:name>
  <crs:version>2008-1-16T9:49</crs:version>
  <crs:description>Handmade proj4 geographic crs definition (parsed from
nad/epsg).</crs:description>
  <crs:Axis>
    <crs:name>longitude</crs:name>
    <crs:units>degree</crs:units>
    <crs:axisOrientation>east</crs:axisOrientation>
  </crs:Axis>
  <crs:Axis>
    <crs:name>latitude</crs:name>
    <crs:units>degree</crs:units>
    <crs:axisOrientation>north</crs:axisOrientation>
  </crs:Axis>
  <crs:axisOrder>longitude, latitude</crs:axisOrder>
  <crs:usedDatum>EPSG:6314</crs:usedDatum>
</crs:geographicCRS>
```

This example shows a <crs:geographicCRS> element with id EPSG:4314, which is defined with two axis, longitude and latitude, and uses the datum EPSG:6314.

4.2.2.2 Creating a projected coordinate system

A projected coordinate reference system adds a map projection to a geographic coordinate reference system. A map projection is (iso 19111): “A coordinate conversion from an ellipsoidal coordinate system to a plane”.

Because of the evident fact, that information is lost in the process of mapping to a plane, different map projections have been invented over the years. Each one with it's pros and contras for different areas of the world. Currently five different map projections are supported in the deegree2 crs library.

1. Lambert Azimuthal Equal Area
2. Lambert Conformal Conic
3. Stereographic Azimuthal
4. Stereographic Azimuthal Alternative[6]
5. Transverse Mercator

Again it is easily seen, that map projections have certain parameters in common, which is why a base projection (<crs:ProjectionBaseType>) type exist. An

---

[6]Two different versions of the stereographic projection are commonly used and supported by deegree. The **alternative stereographic azimuthal** projection is defined by EPSG where as Snyder (**stereographic azimuthal**) uses somewhat different approach, resulting in slightly different projected coordinates away from the natural origin.

example of an element using only 'standard' projection values would be a LambertAzimuthalEqualArea projection.

```
<crs:lambertAzimuthalEqualArea>
  <crs:latitudeOfNaturalOrigin inDegrees="true">45.0</crs:latitudeOfNaturalOrigin>
  <crs:longitudeOfNaturalOrigin inDegrees="true">-
100.0</crs:longitudeOfNaturalOrigin>
  <crs:scaleFactor>1.0</crs:scaleFactor>
  <crs:falseEasting>0.0</crs:falseEasting>
  <crs:falseNorthing>0.0</crs:falseNorthing>
</crs:lambertAzimuthalEqualArea>
```

This projection element defines a <crs:latitudeOfNaturalOrigin> element which defines the location (in degrees) of the projection latitude. This value is also known as: central-latitude, latitude of natural origin, phi_1 (for azimuthal projections) or phi_0 (for other projections). If the inDegrees attribute is set to false, radians are expected.

The <crs:longitudeOfNaturalOrigin> element defines the location (in degrees) of the longitude of natural origin. This value is also known as: projection-meridian, central-meridian or lambda_0 (in Snyder).

The <crs:scaleFactor> element defines scale at the natural origin, often referred to as k_0.

To avoid negative coordinates due to the position of a natural origin (near a prime meridian for example) it is convenient to translate the projections origin with a special amount, resulting in pure positive projected coordinates. These translations can be defined with the <crs:falseEasting> and <crs:Northing> elements. The values defined with these elements are assumed to be in the projected crs' units.

Having defined the projection element and having stated that every <crs:projectedCRS>, is defined on a geographicCRS, it is now time to have a look at the projectedCRS element:

```
<crs:projectedCRS>
  <crs:id>EPSG:2163</crs:id>
  <crs:id>URN:OGC:DEF:CRS:EPSG::2163</crs:id>
  <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#2163</crs:id>
  <crs:id>URN:OPENGIS:DEF:CRS:EPSG::2163</crs:id>
  <crs:name>US National Atlas Equal Area</crs:name>
  <crs:version>2008-1-16T9:49</crs:version>
  <crs:Axis>
    <crs:name>x</crs:name>
    <crs:units>metre</crs:units>
    <crs:axisOrientation>east</crs:axisOrientation>
  </crs:Axis>
  <crs:Axis>
    <crs:name>y</crs:name>
    <crs:units>metre</crs:units>
    <crs:axisOrientation>north</crs:axisOrientation>
  </crs:Axis>
  <crs:axisOrder>x, y</crs:axisOrder>
  <crs:usedGeographicCRS>EPSG:4052</crs:usedGeographicCRS>
  <crs:projection>
    <crs:lambertAzimuthalEqualArea>
      <crs:latitudeOfNaturalOrigin
inDegrees="true">45.0</crs:latitudeOfNaturalOrigin>
```

```
      <crs:longitudeOfNaturalOrigin inDegrees="true">-
100.0</crs:longitudeOfNaturalOrigin>
      <crs:scaleFactor>1.0</crs:scaleFactor>
      <crs:falseEasting>0.0</crs:falseEasting>
      <crs:falseNorthing>0.0</crs:falseNorthing>
    </crs:lambertAzimuthalEqualArea>
  </crs:projection>
</crs:projectedCRS>
```

After the usual identifiable elements, he <crs:Axis> are defined (see Creating a geographic coordinate system for the explanation) stating that incoming coordinates are expected to be in metres on the 'x' and 'y' axis.

What follows is the reference to an **existing** geographic coordinate system, by supplying the <crs:usedGeographicCRS> element, pointing to one of the geographicCRS' identifiers.

After that, the <crs:projection> element describes the parameters of the map projection used with this projected coordinate system. It defines a Lambert azimuthal equal area projection with the above described parameters.

As stated above, the deegree crs library supports different projections, all derived from the ProjectionBase type. Additional parameters for the other projections are now described

Lambert conformal conic

A Lambert conformal conic projection needs the supplement information (like all conic projections) of the first and/or second projection parallels, e.g. the parallels which describe the intersection with the ellipsoid. If failing the projection latitude (defined in the ProjectionBase type) will be assumed. The necessary additional parameters (<crs:firstParallelLatitude> and <crs:secondParallelLatitude>) can be inserted in a standard projection element as following example demonstrates:

```
<crs:projection>
 <crs:lambertConformalConic>
  <crs:latitudeOfNaturalOrigin
inDegrees="true">36.66666666666666</crs:latitudeOfNaturalOrigin>
  <crs:longitudeOfNaturalOrigin inDegrees="true">-
111.5</crs:longitudeOfNaturalOrigin>
  <crs:scaleFactor>1.0</crs:scaleFactor>
  <crs:falseEasting>500000.0</crs:falseEasting>
  <crs:falseNorthing>3000000.0</crs:falseNorthing>
  <crs:firstParallelLatitude inDegrees="true">38.35</crs:firstParallelLatitude>
  <crs:secondParallelLatitude inDegrees="true">37.21</crs:secondParallelLatitude>
 </crs:lambertConformalConic>
</crs:projection>
```

Stereographic azimuthal

A stereographic azimuthal projection (as Snyder defines it, not the stereographic azimuthal alternative) needs the supplement information of the true scale latitude, which defines the latitude around the projection point for which the projection has a true scale. This can be configured by supplying a <crs:trueScaleLatitude> element. If this parameter is failing the projection latitude is assumed to have the true scale.

Transverse Mercator

A transverse Mercator (as well as the unified transvere mercator (UTM)) projection needs the supplement information about the northern/southern hemisphere on which it is applied. This parameter can be set by supplying the <crs:northernHemisphere> boolean element. If it is failing a northern hemisphere is assumed. Following example illustrates the definition of a transverse Mercator projection.

```
<crs:projection>
 <crs:transverseMercator>
  <crs:latitudeOfNaturalOrigin inDegrees="true">0.0</crs:latitudeOfNaturalOrigin>
  <crs:longitudeOfNaturalOrigin inDegrees="true">-75</crs:longitudeOfNaturalOrigin>
  <crs:scaleFactor>0.9996</crs:scaleFactor>
  <crs:falseEasting>500000.0</crs:falseEasting>
  <crs:falseNorthing>0.0</crs:falseNorthing>
  <crs:northernHemisphere>true</crs:northernHemisphere>
 </crs:transverseMercator>
</crs:projection>
```

4.2.2.3 Creating a compound coordinate system

Some coordinate systems are defined to have only two coordinates, like a geographic or projected crs. It is often convenient to have an idea of the height above or below the ellipsoid of coordinates defined in such a system. This (and in the future other possible parameters like a temporal ones) can be defined with a compound CRS.

For the moment it is assumed that a CompoundCRS' only adds a third height Axis to a two dimensional coordinate system. Therefore only two elements are required to be defined, the <crs:usedCRS> element references the coordinate system to which the height axis is added, and the actual <crs:heightAxis> element defining the units and the name of the additional axis. Interesting is the <crs:axisOrientation> which should always contain the value 'perpendicular' indicating, that the height axis is perpendicular to the ellipsoids surface.

The following example shows a <crs:compoundCRS> element adding a height axis using metres to the EPSG:4326 geographic crs.

```
<crs:compoundCRS>
 <crs:id>EPSG:4326_NEW</crs:id>
 <crs:name>CompoundCRS</crs:name>
 <crs:version>2008-1-16T9:49</crs:version>
 <crs:description>A definition of a compound crs</crs:description>
 <crs:usedCRS>EPSG:4326</crs:usedCRS>
 <crs:heightAxis>
  <crs:name>height</crs:name>
  <crs:units>metre</crs:units>
  <crs:axisOrientation>perpendicular</crs:axisOrientation>
 </crs:heightAxis>
 <crs:defaultHeight>208</crs:defaultHeight>
</crs:compoundCRS>
```

It occurred to us, that within a specific region some coordinates might have been associated with a measured height value, while at the same time other coordinates (in the same region) may not have been associated with such a value (for what ever reason). In such cases it can be convenient to be able to define a 'default' height[7] to those coordinates. For this purpose the <crs:defaultHeight> element can be used. If it is supplied, all of the incoming coordinates defined within a compoundCRS with the third coordinate set to NaN, will be handled as though they would have the default height.

4.2.3 Creating a geodetic datum element

The (ISO 19111) specification define a datum to be a set of parameters defining the position of the origin, the scale and the orientation of a coordinate reference system. A geodetic datum describes the relationship of a 3D or 2D coordinate system to the earth.

For this purpose a geodetic datum needs the axis of the ellipsoid approaching the earth and a prime meridian acting as the origin of the datum. Because the earth is not a perfect ellipsoid (far from it, it has more resemblance to a potato), the ellipsoid used to define the datum is often only accurate for a specific region of interest[8]. To make it possible to transform coordinates from a specific coordinate system based on some datum to another coordinate system defined on a different datum, it is necessary to do a datum shift.

There are several ways for doing this shift, a very common approach is the usage of a pivot datum usually the wgs 84 datum. This is the standard (but not the single) transformation method for datum shifts in the deegree crs library. Which is why every geodetic datum holds a reference to the toWGS84 parameters valid for a given datum.

Having specified the needs for a geodetic datum lets have a look at the necessities to create one:

```
<crs:geodeticDatum>
 <crs:id>EPSG:6258</crs:id>
 <crs:id>URN:OGC:DEF:CRS:EPSG::6258</crs:id>
 <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#6258</crs:id>
 <crs:id>URN:OPENGIS:DEF:CRS:EPSG::6258</crs:id>
 <crs:name>European Terrestrial Reference System 1989</crs:name>
 <crs:usedEllipsoid>EPSG:7019</crs:usedEllipsoid>
 <crs:usedPrimeMeridian>EPSG:8901</crs:usedPrimeMeridian>
 <crs:usedWGS84ConversionInfo>EPSG:1188</crs:usedWGS84ConversionInfo>
</crs:geodeticDatum>
```

Again a datum must be identifiable which is why it inherited all of the identifiable elements (see The identifiable elements for a description). All other elements are self explanatory. The only catch is, that the identifiers of the referenced

[7]For example the average height above sea level of a region.

[8]Another reason for the usage of different datums has an historic background. It was common that each country (or even region) had it's own prime meridian and ellipsoids.

components (ellipsoid, prime meridian and wgs84 parameters) must be defined in the configuration document.

4.2.4 Creating an ellipsoid element

An ellipsoid is commonly use to approximate the earth's surface. It can be defined by using different parameters. Common parameters are:

1. semi major axis, the largest of the two axis, often denoted with an 'a'
2. semi minor axis, the smallest of the two axis, aka 'b'
3. flattening, the flattening of the ellipsoid
4. inverse flattening, the calculated inverse of a given flattening
5. eccentricity, the measure of how much ellipsoid deviates from being circular.

If the semi major is given, all other values can be calculated from one another, which is why an ellipsoid can be defined by supplying a single one the latter four. To be able to use the ellipsoid it is also necessary to know in which units given values are given, which is why the units must be defined (for a description of the unit element see Creating coordinate system elements). Following example defines an ellipsoid by supplying a semiMajor axis and an inverseFlattening in metres:

```
<crs:ellipsoid>
 <crs:id>EPSG:7004</crs:id>
 <crs:id>URN:OGC:DEF:CRS:EPSG::7004</crs:id>
 <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#7004</crs:id>
 <crs:id>URN:OPENGIS:DEF:CRS:EPSG::7004</crs:id>
 <crs:name>Bessel 1841</crs:name>
 <crs:semiMajorAxis>6377397.155</crs:semiMajorAxis>
 <crs:inverseFlatting>299.1528128</crs:inverseFlatting>
 <crs:units>metre</crs:units>
</crs:ellipsoid>
```

4.2.5 Creating a prime meridian element

The prime meridian is used to define the origin of a datum. Over the years a lot of prime meridians have been defined. It is only for recent years, that the Greenwich prime meridian has become the 'quasi' standard. With the deegree crs library it is possible to define your own prime meridians, by supplying the longitude (in a given unit) from your prime meridian to Greenwich. This is necessary because deegree internally uses the Greenwich prime meridian as a standard. With the given values all necessary information for doing a translation is available. Following example shows the definition of a PrimeMeridian being 6.8 degrees away from the Greenwich prime meridian:

```
<crs:primeMeridian>
```

```
 <crs:id>COOL_PRIME</crs:id>
 <crs:name>hinter haching</crs:name>
 <crs:units>degree</crs:units>
 <crs:longitude>6.8</crs:longitude>
</crs:primeMeridian>
```

4.2.6 Creating a toWGS84 element

It is often required to convert coordinates defined in some spatial reference system based on some geodetic datum to another spatial reference system based on another geodetic datum. This is often referred to as a datumshift. There are various methods to do this conversion (also called a transformation). One of the most often used is the following:

1. Create Cartesian coordinates from the given (geographic/projected) coordinates.
2. Use the Helmert transformation to convert these coordinates to an euclidean space defined on a datum with known axis, origin and scale. This intermediate datum is sometimes referred to as a pivot datum
3. Use an invert Helmert transformation (of the target datum) to convert the euclidean coordinates defined on the 'pivot' datum to return to the euclidean space of the target datum.
4. Create geographic coordinates from the Cartesian coordinates.

The Helmert transformation uses seven parameters, three translation parameters, three rotation parameters and one scale parameter. These parameters are often called bursa/wolf parameters.

It is a common practice to use WGS84[9] as the intermediate space, which means, that the seven bursa/wolf parameters for the Helmert transformation must be supplied to align the source and target datum with the WGS84 datum. This is done by supplying the <crs:wgs84Transformation> element, which is again identifiable. The following example illustrates the definition of the bursa wolf parameters for the Bessel datum valid for Germany.

```
<crs:wgs84Transformation>
 <crs:id>EPSG:1777</crs:id>
 <crs:name>DHDN to WGS 84</crs:name>
 <crs:version>EPSG-Deu W 3m</crs:version>
 <crs:xAxisTranslation>598.1</crs:xAxisTranslation>
 <crs:yAxisTranslation>73.7</crs:yAxisTranslation>
 <crs:zAxisTranslation>418.2</crs:zAxisTranslation>
 <crs:xAxisRotation>0.202</crs:xAxisRotation>
 <crs:yAxisRotation>0.045</crs:yAxisRotation>
 <crs:zAxisRotation>-2.455</crs:zAxisRotation>
 <crs:scaleDifference>6.7</crs:scaleDifference>
</crs:wgs84Transformation>
```

[9]WGS84 is the Worlds Geodetic System, it was constructed in 1984 by satellite measurements and is valid to the year 2010.

5 Advanced Configuration

You are now able to define and use the deegree2 crs library by supplying the parameters for the standard spatial coordinates reference components. For most coordinate operations these definitions will be sufficient. However, it sometimes is necessary to use a projection which is not supplied (yet) by the deegree library, or to be able to do a transformation other than the helmert transformation for doing a datum shift.

For this purpose the deegree configuration allows you to supply your own projections and or your own transformation. The requirements of these user defined classes and the configuration of the deegree crs package will be briefly discussed in this chapter.

5.1 Automatic loading of projection/transformation classes

As described above, it is possible to create your own projection/transformation classes, which the DeegreeCRSprovider will try to load automatically.

You can achieve this loading by supplying the **class** attribute to a <crs:projection> or <crs:polynomialTransformation> elements in the 'deegree-crs-configuration.xml'. This attribute must contain the full class name (including package), for example to use and load your own supplied “Mercator” projection class you could define the following projection: <crs:projection class="org.mycompany.crs.projections.Mercator">. Because the instantiation of your defined class is done with reflections, the classes you provide, must sustain a number of criteria. They are explained in the following chapters.

5.2 Using your projection class

If you supply a projection class you have to take notice of the following implementation requirements:

1. Your class must be derive org.deegree.crs.projections.Projection.
2. A constructor with following signature must be supplied:

```
public MyProjection(
                org.deegree.crs.coordinatesystems.GeographicCRS underlyingCRS,
                double falseNorthing,
                double falseEasting,
                javax.vecmath.Point2d naturalOrigin,
                org.deegree.crs.components.Unit units,
                double scale,
                java.util.List<org.w3c.dom.Element> yourProjectionElements
);
```

The first six parameters of the constructor are common to all projections (for an explanation of their meaning take a look at Creating a projected coordinate system. The last list, will contain all xml-dom elements you supplied in the deegree configuration (child elements of crs:projection/crs:myProjection), thus relieving you of the parsing of the deegree-crs-configuration.xml document.

Following example illustrates the usage of a user defined projection, the element beneath the <crs:projection class="org.mycompany.crs.projections.Mercator> element (in this case <crs:testMercator>), can be named at your likings:

```
<crs:projectedCRS>
 <crs:id>EPSG:25832</crs:id>
 <crs:id>URN:OGC:DEF:CRS:EPSG::25832</crs:id>
 <crs:name>ETRS89 / UTM zone 32N</crs:name>
 <crs:version>2008-1-16T9:49</crs:version>
 <crs:Axis>
  <crs:name>x</crs:name>
  <crs:units>metre</crs:units>
  <crs:axisOrientation>east</crs:axisOrientation>
 </crs:Axis>
 <crs:Axis>
  <crs:name>y</crs:name>
  <crs:units>metre</crs:units>
  <crs:axisOrientation>north</crs:axisOrientation>
 </crs:Axis>
 <crs:axisOrder>x, y</crs:axisOrder>
 <crs:projection class="org.mycompany.crs.projections.Mercator">
  <crs:testMercator>
   <crs:latitudeOfNaturalOrigin inDegrees="true">0.0</crs:latitudeOfNaturalOrigin>
   <crs:longitudeOfNaturalOrigin
inDegrees="true">9.0</crs:longitudeOfNaturalOrigin>
   <crs:scaleFactor>0.9996</crs:scaleFactor>
   <crs:falseEasting>500000.0</crs:falseEasting>
   <crs:falseNorthing>0.0</crs:falseNorthing>
   <crs:northernHemisphere>true</crs:northernHemisphere>
   <crs:myParameter>yes</crs:myParameter>
  </crs:testMercator>
 </crs:projection>
</crs:projectedCRS>
```

5.3 Using a different datum shift transformation

If the Helmert transformation is to inaccurate for you, you may choose to use another transformation for the datum shift. This is why each deegree coordinate system can be defined to use a polynomial transformation instead of the standard wgs84 transformation.

The transformation direction is only one way though, this means, that if you supply the parameters for a polynomial transformation, for example to do a datum shift from EPSG:4258 to EPSG:31466 (thus going from ETRS89 to bessel), this polynomial transformation is not valid for coordinate transformations going from EPSG:31466 to EPSG:4258.

To enable this feature you can define the <crs:polynomialTransformation> element within each <crs:abstractCRS> element. To substantiate the previous example you could define the geographicCRS EPSG:4258 as following:

```
<crs:geographicCRS>
 <crs:id>EPSG:4258</crs:id>
 <crs:name>ETRS89</crs:name>
 <crs:Axis>
  <crs:name>longitude</crs:name>
```

```
  <crs:units>degree</crs:units>
  <crs:axisOrientation>east</crs:axisOrientation>
 </crs:Axis>
 <crs:Axis>
  <crs:name>latitude</crs:name>
  <crs:units>degree</crs:units>
  <crs:axisOrientation>north</crs:axisOrientation>
 </crs:Axis>
 <crs:axisOrder>longitude, latitude</crs:axisOrder>
 <crs:usedDatum>EPSG:6258</crs:usedDatum>
 <crs:polynomialTransformation>
  <crs:leastsquare>
   <crs:polynomialOrder>2</crs:polynomialOrder>
   <crs:xParameters>108.029884 0.9988207 -0.006757177 5.1565574E-5 -3.5190485E-6
-4.002526E-5</crs:xParameters>
   <crs:yParameters>1.2068206 0.012087965 0.99012643 -3.0174679E-5 1.2242106E-4
1.1114964E-5</crs:yParameters>
   <crs:targetCRS>EPSG:31466</crs:targetCRS>
   <crs:scaleX>4.597728E-5</crs:scaleX>
   <crs:scaleY>5.7211128E-5</crs:scaleY>
  </crs:leastsquare>
 </crs:polynomialTransformation>
</crs:geographicCRS>
```

This example uses the leasts square polynomial transformation method to supply the datum shift to EPSG:31466 (not inverse!) by defining the <crs:targetCRS>element. Note, that this transformation will use a deegree crs transformation method.

Common to all polynomial transformations are the x variables and the y variables describing the scaling factors for the constants in the polynomial. They can be defined by supplying the mandatory <crs:xParameters> and <crs:yParameters> elements. It is up to the user to calculate these parameters, but for this particular class the tool org.deegree.tools.srs.PolynomialParameterCreator can be of help. The <crs:scaleX> and <crs:scaleY> elements are used to map incoming coordinates to a [0-1] interval, thus letting the least squares method be more accurate in the approximation of the mapping. Again the above tool can be used to create these values.

5.3.1 Using your transformation class

As with the projections it is possible to define and use your own transformation class. To use it, it should meet following requirements:

1. You class must be derived from org.deegree.crs.transformations.polynomial.PolynomialTransformation
2. A constructor with following signature must be supplied:

```
public MyTransformation(
               java.util.list<Double> aValues,
               java.util.list<Double> bValues,
               org.deegree.crs.coordinatesystems.CoordinateSystem targetCRS,
               java.util.List<org.w3c.dom.Element> yourTransformationElements
);
```

The first three parameters are common to all polynomial transformations (explained above). As with the projections, the last list will contain all xml-dom elements you supplied in the deegree configuration (child elements of the crs:transformation/crs:MyTransformation), thus relieving you of the parsing of the deegree-crs-configuration.xml document.

As an example, to configure and use your own transformation class called “org.mycompany.transform.SplineTransform” in the above example (going from EPSG:4258 to EPSG:31466) your configuration should look like this:

```
<crs:geographicCRS>
 <crs:id>EPSG:4258</crs:id>
 <crs:name>ETRS89</crs:name>
 <crs:Axis>
  <crs:name>longitude</crs:name>
  <crs:units>degree</crs:units>
  <crs:axisOrientation>east</crs:axisOrientation>
 </crs:Axis>
 <crs:Axis>
  <crs:name>latitude</crs:name>
  <crs:units>degree</crs:units>
  <crs:axisOrientation>north</crs:axisOrientation>
 </crs:Axis>
 <crs:axisOrder>longitude, latitude</crs:axisOrder>
 <crs:usedDatum>EPSG:6258</crs:usedDatum>
 <crs:polynomialTransformation class="org.mycompany.transform.SplineTransform">
  <crs:spline>
   <crs:xParameters>108.029884 0.9988207 -0.006757177 5.1565574E-5 -3.5190485E-6
-4.002526E-5</crs:xParameters>
   <crs:yParameters>1.2068206 0.012087965 0.99012643 -3.0174679E-5 1.2242106E-4
1.1114964E-5</crs:yParameters>
   <crs:targetCRS>EPSG:31466</crs:targetCRS>
   <crs:splineParam>An optional parameter for your class</crs:splineParam>
  </crs:spline>
 </crs:polynomialTransformation>
</crs:geographicCRS>
```

Again the name of the element beneath the <crs:polynomialTransformation> element (in this case <crs:spline>) is of no relevance and may contain any character set.

In this example the <crs:splineParam> element will be presented to your spline class in the list of dom elements given to the constructor.

Appendix A Example deegree-crs-configuration.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<crs:definitions xmlns:crs="http://www.deegree.org/crs">
 <crs:ellipsoid>
  <crs:id>EPSG:7004</crs:id>
  <crs:id>URN:OGC:DEF:CRS:EPSG::7004</crs:id>
  <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#7004</crs:id>
  <crs:id>URN:OPENGIS:DEF:CRS:EPSG::7004</crs:id>
  <crs:name>Bessel 1841</crs:name>
  <crs:semiMajorAxis>6377397.155</crs:semiMajorAxis>
  <crs:inverseFlatting>299.1528128</crs:inverseFlatting>
  <crs:units>metre</crs:units>
 </crs:ellipsoid>
 <crs:ellipsoid>
  <crs:id>EPSG:7019</crs:id>
  <crs:id>URN:OGC:DEF:CRS:EPSG::7019</crs:id>
  <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#7019</crs:id>
  <crs:id>URN:OPENGIS:DEF:CRS:EPSG::7019</crs:id>
  <crs:name>GRS 1980(IUGG, 1980)</crs:name>
  <crs:semiMajorAxis>6378137.0</crs:semiMajorAxis>
  <crs:inverseFlatting>298.257222101</crs:inverseFlatting>
  <crs:units>metre</crs:units>
 </crs:ellipsoid>
 <crs:ellipsoid>
  <crs:id>EPSG:7030</crs:id>
  <crs:name>WGS84_Ellipsoid</crs:name>
  <crs:semiMajorAxis>6378137.0</crs:semiMajorAxis>
  <crs:inverseFlatting>298.257223563</crs:inverseFlatting>
  <crs:units>metre</crs:units>
 </crs:ellipsoid>
 <crs:geodeticDatum>
  <crs:id>EPSG:6314</crs:id>
  <crs:id>URN:OGC:DEF:CRS:EPSG::6314</crs:id>
  <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#6314</crs:id>
  <crs:id>URN:OPENGIS:DEF:CRS:EPSG::6314</crs:id>
  <crs:name>Deutsches Hauptdreiecksnetz</crs:name>
  <crs:version>2006-06-12</crs:version>
  <crs:description>Fundamental point: Rauenberg. Latitude: 52 deg 27 min 12.021 sec
N; Longitude: 13 deg 22 min 04.928 sec E (of Greenwich). This station was destroyed
in 1910 and the station at Potsdam substituted as the fundamental
point.</crs:description>
  <crs:usedEllipsoid>EPSG:7004</crs:usedEllipsoid>
  <crs:usedPrimeMeridian>EPSG:8901</crs:usedPrimeMeridian>
  <crs:usedWGS84ConversionInfo>EPSG:1777</crs:usedWGS84ConversionInfo>
 </crs:geodeticDatum>
 <crs:geodeticDatum>
  <crs:id>EPSG:6258</crs:id>
  <crs:id>URN:OGC:DEF:CRS:EPSG::6258</crs:id>
  <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#6258</crs:id>
  <crs:id>URN:OPENGIS:DEF:CRS:EPSG::6258</crs:id>
  <crs:name>European Terrestrial Reference System 1989</crs:name>
  <crs:usedEllipsoid>EPSG:7019</crs:usedEllipsoid>
  <crs:usedPrimeMeridian>EPSG:8901</crs:usedPrimeMeridian>
  <crs:usedWGS84ConversionInfo>EPSG:1188</crs:usedWGS84ConversionInfo>
 </crs:geodeticDatum>
 <crs:geodeticDatum>
  <crs:id>EPSG:6326</crs:id>
  <crs:name>WGS_1984</crs:name>
  <crs:usedEllipsoid>EPSG:7030</crs:usedEllipsoid>
  <crs:usedPrimeMeridian>EPSG:8901</crs:usedPrimeMeridian>
  <crs:usedWGS84ConversionInfo>EPSG:1188</crs:usedWGS84ConversionInfo>
 </crs:geodeticDatum>
 <crs:projectedCRS>
  <crs:id>EPSG:31466</crs:id>
  <crs:id>URN:OGC:DEF:CRS:EPSG::31466</crs:id>
  <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#31466</crs:id>
  <crs:id>URN:OPENGIS:DEF:CRS:EPSG::31466</crs:id>
  <crs:name>DHDN / Gauss-Kruger zone 2</crs:name>
```

```
  <crs:version>2008-1-16T9:49</crs:version>
  <crs:Axis>
   <crs:name>x</crs:name>
   <crs:units>metre</crs:units>
   <crs:axisOrientation>east</crs:axisOrientation>
  </crs:Axis>
  <crs:Axis>
   <crs:name>y</crs:name>
   <crs:units>metre</crs:units>
   <crs:axisOrientation>north</crs:axisOrientation>
  </crs:Axis>
  <crs:axisOrder>x, y</crs:axisOrder>
  <crs:usedGeographicCRS>EPSG:4314</crs:usedGeographicCRS>
  <crs:projection>
   <crs:transverseMercator>
    <crs:latitudeOfNaturalOrigin inDegrees="true">0.0</crs:latitudeOfNaturalOrigin>
    <crs:longitudeOfNaturalOrigin
inDegrees="true">6.0</crs:longitudeOfNaturalOrigin>
    <crs:scaleFactor>1.0</crs:scaleFactor>
    <crs:falseEasting>2500000.0</crs:falseEasting>
    <crs:falseNorthing>0.0</crs:falseNorthing>
    <crs:northernHemisphere>true</crs:northernHemisphere>
   </crs:transverseMercator>
  </crs:projection>
 </crs:projectedCRS>
 <crs:projectedCRS>
  <crs:id>EPSG:31467</crs:id>
  <crs:id>URN:OGC:DEF:CRS:EPSG::31467</crs:id>
  <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#31467</crs:id>
  <crs:id>URN:OPENGIS:DEF:CRS:EPSG::31467</crs:id>
  <crs:name>DHDN / Gauss-Kruger zone 3</crs:name>
  <crs:version>2008-1-16T9:49</crs:version>
  <crs:Axis>
   <crs:name>x</crs:name>
   <crs:units>metre</crs:units>
   <crs:axisOrientation>east</crs:axisOrientation>
  </crs:Axis>
  <crs:Axis>
   <crs:name>y</crs:name>
   <crs:units>metre</crs:units>
   <crs:axisOrientation>north</crs:axisOrientation>
  </crs:Axis>
  <crs:axisOrder>x, y</crs:axisOrder>
  <crs:usedGeographicCRS>EPSG:4314</crs:usedGeographicCRS>
  <crs:projection>
   <crs:transverseMercator>
    <crs:latitudeOfNaturalOrigin inDegrees="true">0.0</crs:latitudeOfNaturalOrigin>
    <crs:longitudeOfNaturalOrigin
inDegrees="true">9.0</crs:longitudeOfNaturalOrigin>
    <crs:scaleFactor>1.0</crs:scaleFactor>
    <crs:falseEasting>3500000.0</crs:falseEasting>
    <crs:falseNorthing>0.0</crs:falseNorthing>
    <crs:northernHemisphere>true</crs:northernHemisphere>
   </crs:transverseMercator>
  </crs:projection>
 </crs:projectedCRS>
 <crs:projectedCRS>
  <crs:id>EPSG:25832</crs:id>
  <crs:id>URN:OGC:DEF:CRS:EPSG::25832</crs:id>
  <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#25832</crs:id>
  <crs:id>URN:OPENGIS:DEF:CRS:EPSG::25832</crs:id>
  <crs:name>ETRS89 / UTM zone 32N</crs:name>
  <crs:version>2008-1-16T9:49</crs:version>
  <crs:Axis>
   <crs:name>x</crs:name>
   <crs:units>metre</crs:units>
   <crs:axisOrientation>east</crs:axisOrientation>
  </crs:Axis>
  <crs:Axis>
```

```xml
    <crs:name>y</crs:name>
    <crs:units>metre</crs:units>
    <crs:axisOrientation>north</crs:axisOrientation>
   </crs:Axis>
   <crs:axisOrder>x, y</crs:axisOrder>
   <crs:usedGeographicCRS>EPSG:4258</crs:usedGeographicCRS>
   <crs:projection>
    <crs:transverseMercator>
     <crs:latitudeOfNaturalOrigin inDegrees="true">0.0</crs:latitudeOfNaturalOrigin>
     <crs:longitudeOfNaturalOrigin
inDegrees="true">9.0</crs:longitudeOfNaturalOrigin>
     <crs:scaleFactor>0.9996</crs:scaleFactor>
     <crs:falseEasting>500000.0</crs:falseEasting>
     <crs:falseNorthing>0.0</crs:falseNorthing>
     <crs:northernHemisphere>true</crs:northernHemisphere>
    </crs:transverseMercator>
   </crs:projection>
  </crs:projectedCRS>
  <crs:compoundCRS>
   <crs:id>EPSG:31466_NEW</crs:id>
   <crs:name>CompoundCRS</crs:name>
   <crs:version>2008-1-16T9:49</crs:version>
   <crs:description>A definition of a compound crs</crs:description>
   <crs:usedCRS>EPSG:31466</crs:usedCRS>
   <crs:heightAxis>
    <crs:name>height</crs:name>
    <crs:units>metre</crs:units>
    <crs:axisOrientation>perpendicular</crs:axisOrientation>
   </crs:heightAxis>
  </crs:compoundCRS>
  <crs:compoundCRS>
   <crs:id>EPSG:25832_NEW</crs:id>
   <crs:name>CompoundCRS</crs:name>
   <crs:version>2008-1-16T9:49</crs:version>
   <crs:description>A definition of a compound crs</crs:description>
   <crs:usedCRS>EPSG:25832</crs:usedCRS>
   <crs:heightAxis>
    <crs:name>height</crs:name>
    <crs:units>metre</crs:units>
    <crs:axisOrientation>perpendicular</crs:axisOrientation>
   </crs:heightAxis>
  </crs:compoundCRS>
  <crs:compoundCRS>
   <crs:id>EPSG:4326_NEW</crs:id>
   <crs:name>CompoundCRS</crs:name>
   <crs:version>2008-1-16T9:49</crs:version>
   <crs:description>A definition of a compound crs</crs:description>
   <crs:usedCRS>EPSG:4326</crs:usedCRS>
   <crs:heightAxis>
    <crs:name>height</crs:name>
    <crs:units>metre</crs:units>
    <crs:axisOrientation>perpendicular</crs:axisOrientation>
   </crs:heightAxis>
  </crs:compoundCRS>
  <crs:geographicCRS>
   <crs:id>EPSG:4258</crs:id>
   <crs:id>URN:OGC:DEF:CRS:EPSG::4258</crs:id>
   <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#4258</crs:id>
   <crs:id>URN:OPENGIS:DEF:CRS:EPSG::4258</crs:id>
   <crs:name>ETRS89</crs:name>
   <crs:version>2008-1-16T9:49</crs:version>
   <crs:Axis>
    <crs:name>longitude</crs:name>
    <crs:units>degree</crs:units>
    <crs:axisOrientation>east</crs:axisOrientation>
   </crs:Axis>
   <crs:Axis>
    <crs:name>latitude</crs:name>
    <crs:units>degree</crs:units>
```

```
    <crs:axisOrientation>north</crs:axisOrientation>
   </crs:Axis>
   <crs:axisOrder>longitude, latitude</crs:axisOrder>
   <crs:usedDatum>EPSG:6258</crs:usedDatum>
  </crs:geographicCRS>
  <crs:geographicCRS>
   <crs:id>EPSG:4314</crs:id>
   <crs:id>URN:OGC:DEF:CRS:EPSG::4314</crs:id>
   <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#4314</crs:id>
   <crs:id>URN:OPENGIS:DEF:CRS:EPSG::4314</crs:id>
   <crs:name>DHDN</crs:name>
   <crs:version>2008-1-16T9:49</crs:version>
   <crs:description>Handmade proj4 geographic crs definition (parsed from
nad/epsg).</crs:description>
   <crs:areaOfUse>Unknown</crs:areaOfUse>
   <crs:Axis>
    <crs:name>longitude</crs:name>
    <crs:units>degree</crs:units>
    <crs:axisOrientation>east</crs:axisOrientation>
   </crs:Axis>
   <crs:Axis>
    <crs:name>latitude</crs:name>
    <crs:units>degree</crs:units>
    <crs:axisOrientation>north</crs:axisOrientation>
   </crs:Axis>
   <crs:axisOrder>longitude, latitude</crs:axisOrder>
   <crs:usedDatum>EPSG:6314</crs:usedDatum>
  </crs:geographicCRS>
  <crs:geographicCRS>
   <crs:id>EPSG:4326</crs:id>
   <crs:id>URN:OGC:DEF:CRS:EPSG::4326</crs:id>
   <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#4326</crs:id>
   <crs:id>URN:OPENGIS:DEF:CRS:EPSG::4326</crs:id>
   <crs:name>WGS 84</crs:name>
   <crs:version>2008-1-16T9:49</crs:version>
   <crs:description>Handmade proj4 geographic crs definition (parsed from
nad/epsg).</crs:description>
   <crs:areaOfUse>Unknown</crs:areaOfUse>
   <crs:Axis>
    <crs:name>longitude</crs:name>
    <crs:units>degree</crs:units>
    <crs:axisOrientation>east</crs:axisOrientation>
   </crs:Axis>
   <crs:Axis>
    <crs:name>latitude</crs:name>
    <crs:units>degree</crs:units>
    <crs:axisOrientation>north</crs:axisOrientation>
   </crs:Axis>
   <crs:axisOrder>longitude, latitude</crs:axisOrder>
   <crs:usedDatum>EPSG:6326</crs:usedDatum>
  </crs:geographicCRS>
  <crs:primeMeridian>
   <crs:id>EPSG:8901</crs:id>
   <crs:id>http://www.opengis.net/gml/srs/epsg.xml#8901</crs:id>
   <crs:id>URN:OPENGIS:DEF:CRS:EPSG::8901</crs:id>
   <crs:id>URN:OGC:DEF:CRS:EPSG::</crs:id>
   <crs:name>Greenwich</crs:name>
   <crs:version>1995-06-02</crs:version>
   <crs:units>degree</crs:units>
   <crs:longitude>0.0</crs:longitude>
  </crs:primeMeridian>
  <crs:wgs84Transformation>
   <crs:id>EPSG:1777</crs:id>
   <crs:id>URN:OGC:DEF:CRS:EPSG::1777</crs:id>
   <crs:id>HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#1777</crs:id>
   <crs:id>URN:OPENGIS:DEF:CRS:EPSG::1777</crs:id>
   <crs:name>DHDN to WGS 84</crs:name>
   <crs:version>EPSG-Deu W 3m</crs:version>
```

```
  <crs:description>Parameter values from DHDN to ETRS89 (2) (code 1776) assuming
that ETRS89 is equivalent to WGS 84 within the accuracy of the transformation.
Replaces DHDN to WGS 84 (1) (tfm code 1673).</crs:description>
  <crs:areaOfUse>Germany - states of former West Germany - Baden-Wurtemberg,
Bayern, Hessen, Niedersachsen, Nordrhein-Westfalen, Rheinland-Pfalz, Saarland,
Schleswig-Holstein.</crs:areaOfUse>
  <crs:xAxisTranslation>598.1</crs:xAxisTranslation>
  <crs:yAxisTranslation>73.7</crs:yAxisTranslation>
  <crs:zAxisTranslation>418.2</crs:zAxisTranslation>
  <crs:xAxisRotation>0.202</crs:xAxisRotation>
  <crs:yAxisRotation>0.045</crs:yAxisRotation>
  <crs:zAxisRotation>-2.455</crs:zAxisRotation>
  <crs:scaleDifference>6.7</crs:scaleDifference>
 </crs:wgs84Transformation>
 <crs:wgs84Transformation>
  <crs:id>EPSG:1188</crs:id>
  <crs:xAxisTranslation>0.0</crs:xAxisTranslation>
  <crs:yAxisTranslation>0.0</crs:yAxisTranslation>
  <crs:zAxisTranslation>0.0</crs:zAxisTranslation>
  <crs:xAxisRotation>0.0</crs:xAxisRotation>
  <crs:yAxisRotation>0.0</crs:yAxisRotation>
  <crs:zAxisRotation>0.0</crs:zAxisRotation>
  <crs:scaleDifference>0.0</crs:scaleDifference>
 </crs:wgs84Transformation>
</crs:definitions>
```

Appendix B Supported SRS

Following spatial reference systems are supported by the standard deegree2 crs library:

EPSG:2000	EPSG:2333	EPSG:26753	EPSG:2905	EPSG:32104	EPSG:3298
EPSG:20004	EPSG:2334	EPSG:26754	EPSG:2906	EPSG:32107	EPSG:3299
EPSG:20005	EPSG:2335	EPSG:26755	EPSG:2907	EPSG:32108	EPSG:3300
EPSG:20006	EPSG:2336	EPSG:26756	EPSG:2908	EPSG:32109	EPSG:3301
EPSG:20007	EPSG:2337	EPSG:26757	EPSG:2909	EPSG:3211	EPSG:3302
EPSG:20008	EPSG:2338	EPSG:26758	EPSG:2910	EPSG:32110	EPSG:3303
EPSG:20009	EPSG:2339	EPSG:26759	EPSG:2911	EPSG:32111	EPSG:3304
EPSG:2001	EPSG:2340	EPSG:2676	EPSG:29118	EPSG:32112	EPSG:3305
EPSG:20010	EPSG:2341	EPSG:26760	EPSG:29119	EPSG:32113	EPSG:3306
EPSG:20011	EPSG:2342	EPSG:26766	EPSG:2912	EPSG:32114	EPSG:3307
EPSG:20012	EPSG:2343	EPSG:26767	EPSG:29120	EPSG:32115	EPSG:3308
EPSG:20013	EPSG:23433	EPSG:26768	EPSG:29121	EPSG:32116	EPSG:3312
EPSG:20014	EPSG:2344	EPSG:26769	EPSG:29122	EPSG:32117	EPSG:3313
EPSG:20015	EPSG:2345	EPSG:2677	EPSG:2913	EPSG:32118	EPSG:3314
EPSG:20016	EPSG:2346	EPSG:26770	EPSG:2914	EPSG:32119	EPSG:3315
EPSG:20017	EPSG:2347	EPSG:26771	EPSG:2915	EPSG:3212	EPSG:3316
EPSG:20018	EPSG:2348	EPSG:26772	EPSG:2916	EPSG:32120	EPSG:3317
EPSG:20019	EPSG:2349	EPSG:26773	EPSG:29168	EPSG:32121	EPSG:3318
EPSG:2002	EPSG:2350	EPSG:26774	EPSG:29169	EPSG:32122	EPSG:3319
EPSG:20020	EPSG:2351	EPSG:26775	EPSG:2917	EPSG:32123	EPSG:3320
EPSG:20021	EPSG:2352	EPSG:26776	EPSG:29170	EPSG:32124	EPSG:3321
EPSG:20022	EPSG:2353	EPSG:26777	EPSG:29171	EPSG:32125	EPSG:3322

EPSG:20023	EPSG:2354	EPSG:26778	EPSG:29172	EPSG:32126	EPSG:3323
EPSG:20024	EPSG:2355	EPSG:26779	EPSG:29177	EPSG:32127	EPSG:3324
EPSG:20025	EPSG:2356	EPSG:2678	EPSG:29178	EPSG:32128	EPSG:3325
EPSG:20026	EPSG:2357	EPSG:26780	EPSG:29179	EPSG:32129	EPSG:3326
EPSG:20027	EPSG:2358	EPSG:26781	EPSG:2918	EPSG:3213	EPSG:3327
EPSG:20028	EPSG:2359	EPSG:26782	EPSG:29180	EPSG:32130	EPSG:3328
EPSG:20029	EPSG:2360	EPSG:26783	EPSG:29181	EPSG:32133	EPSG:3329
EPSG:2003	EPSG:2361	EPSG:26784	EPSG:29182	EPSG:32134	EPSG:3330
EPSG:20030	EPSG:2362	EPSG:26785	EPSG:29183	EPSG:32135	EPSG:3331
EPSG:20031	EPSG:2363	EPSG:26786	EPSG:29184	EPSG:32136	EPSG:3332
EPSG:20032	EPSG:2364	EPSG:26787	EPSG:29185	EPSG:32137	EPSG:3333
EPSG:2004	EPSG:2365	EPSG:2679	EPSG:29187	EPSG:32138	EPSG:3334
EPSG:2005	EPSG:2366	EPSG:26791	EPSG:29188	EPSG:32139	EPSG:3335
EPSG:2006	EPSG:2367	EPSG:26792	EPSG:29189	EPSG:3214	EPSG:3336
EPSG:20064	EPSG:2368	EPSG:26793	EPSG:2919	EPSG:32140	EPSG:3337
EPSG:20065	EPSG:2369	EPSG:26794	EPSG:29190	EPSG:32141	EPSG:3339
EPSG:20066	EPSG:2370	EPSG:26795	EPSG:29191	EPSG:32142	EPSG:3340
EPSG:20067	EPSG:2371	EPSG:26796	EPSG:29192	EPSG:32143	EPSG:3341
EPSG:20068	EPSG:2372	EPSG:26797	EPSG:29193	EPSG:32144	EPSG:3342
EPSG:20069	EPSG:2373	EPSG:26798	EPSG:29194	EPSG:32145	EPSG:3343
EPSG:2007	EPSG:2374	EPSG:26799	EPSG:29195	EPSG:32146	EPSG:3344
EPSG:20070	EPSG:2375	EPSG:2680	EPSG:2920	EPSG:32147	EPSG:3345
EPSG:20071	EPSG:2376	EPSG:26801	EPSG:2921	EPSG:32148	EPSG:3346
EPSG:20072	EPSG:2377	EPSG:26802	EPSG:2922	EPSG:32149	EPSG:3347

EPSG:20073	EPSG:2378	EPSG:26803	EPSG:29220	EPSG:3215	EPSG:3348
EPSG:20074	EPSG:2379	EPSG:2681	EPSG:29221	EPSG:32150	EPSG:3350
EPSG:20075	EPSG:2380	EPSG:26811	EPSG:2923	EPSG:32151	EPSG:3351
EPSG:20076	EPSG:2381	EPSG:26812	EPSG:2924	EPSG:32152	EPSG:3352
EPSG:20077	EPSG:2382	EPSG:26813	EPSG:2925	EPSG:32153	EPSG:3353
EPSG:20078	EPSG:2383	EPSG:2682	EPSG:2926	EPSG:32154	EPSG:3354
EPSG:20079	EPSG:2384	EPSG:2683	EPSG:2927	EPSG:32155	EPSG:3355
EPSG:2008	EPSG:23846	EPSG:2684	EPSG:2928	EPSG:32156	EPSG:3356
EPSG:20080	EPSG:23847	EPSG:2685	EPSG:2929	EPSG:32157	EPSG:3357
EPSG:20081	EPSG:23848	EPSG:2686	EPSG:2930	EPSG:32158	EPSG:3358
EPSG:20082	EPSG:23849	EPSG:2687	EPSG:2931	EPSG:3216	EPSG:3359
EPSG:20083	EPSG:2385	EPSG:2688	EPSG:2932	EPSG:32161	EPSG:3360
EPSG:20084	EPSG:23850	EPSG:2689	EPSG:2933	EPSG:32164	EPSG:3361
EPSG:20085	EPSG:23851	EPSG:2690	EPSG:29333	EPSG:32165	EPSG:3362
EPSG:20086	EPSG:23852	EPSG:26901	EPSG:2935	EPSG:32166	EPSG:3363
EPSG:20087	EPSG:23853	EPSG:26902	EPSG:2936	EPSG:32167	EPSG:3364
EPSG:20088	EPSG:2386	EPSG:26903	EPSG:2937	EPSG:3217	EPSG:3365
EPSG:20089	EPSG:23866	EPSG:26904	EPSG:2938	EPSG:3218	EPSG:3367
EPSG:2009	EPSG:23867	EPSG:26905	EPSG:2939	EPSG:32180	EPSG:3368
EPSG:20090	EPSG:23868	EPSG:26906	EPSG:2940	EPSG:32181	EPSG:3369
EPSG:20091	EPSG:23869	EPSG:26907	EPSG:2941	EPSG:32182	EPSG:3370
EPSG:20092	EPSG:2387	EPSG:26908	EPSG:2942	EPSG:32183	EPSG:3371
EPSG:2010	EPSG:23870	EPSG:26909	EPSG:2943	EPSG:32184	EPSG:3372
EPSG:2011	EPSG:23871	EPSG:2691	EPSG:2944	EPSG:32185	EPSG:3373

EPSG:2012	EPSG:23872	EPSG:26910	EPSG:2945	EPSG:32186	EPSG:3374
EPSG:2013	EPSG:23877	EPSG:26911	EPSG:2946	EPSG:32187	EPSG:3386
EPSG:20135	EPSG:23878	EPSG:26912	EPSG:2947	EPSG:32188	EPSG:3387
EPSG:20136	EPSG:23879	EPSG:26913	EPSG:2948	EPSG:32189	EPSG:3389
EPSG:20137	EPSG:2388	EPSG:26914	EPSG:2949	EPSG:3219	EPSG:3390
EPSG:20138	EPSG:23880	EPSG:26915	EPSG:2950	EPSG:32190	EPSG:3391
EPSG:2014	EPSG:23881	EPSG:26916	EPSG:2951	EPSG:32191	EPSG:3392
EPSG:2015	EPSG:23882	EPSG:26917	EPSG:2952	EPSG:32192	EPSG:3393
EPSG:2016	EPSG:23883	EPSG:26918	EPSG:2953	EPSG:32193	EPSG:3394
EPSG:2017	EPSG:23884	EPSG:26919	EPSG:2954	EPSG:32194	EPSG:3396
EPSG:2018	EPSG:23886	EPSG:2692	EPSG:2955	EPSG:32195	EPSG:3397
EPSG:2019	EPSG:23887	EPSG:26920	EPSG:2956	EPSG:32196	EPSG:3398
EPSG:2020	EPSG:23888	EPSG:26921	EPSG:2957	EPSG:32197	EPSG:3399
EPSG:2021	EPSG:23889	EPSG:26922	EPSG:2958	EPSG:32198	EPSG:3400
EPSG:2022	EPSG:2389	EPSG:26923	EPSG:2959	EPSG:32199	EPSG:3401
EPSG:2023	EPSG:23890	EPSG:26929	EPSG:2960	EPSG:3220	EPSG:3402
EPSG:2024	EPSG:23891	EPSG:2693	EPSG:2961	EPSG:32201	EPSG:3403
EPSG:20248	EPSG:23892	EPSG:26930	EPSG:2962	EPSG:32202	EPSG:3404
EPSG:20249	EPSG:23893	EPSG:26932	EPSG:29635	EPSG:32203	EPSG:3405
EPSG:2025	EPSG:23894	EPSG:26933	EPSG:29636	EPSG:32204	EPSG:3406
EPSG:20250	EPSG:2390	EPSG:26934	EPSG:2965	EPSG:32205	EPSG:3439
EPSG:20251	EPSG:2391	EPSG:26935	EPSG:2966	EPSG:32206	EPSG:3440
EPSG:20252	EPSG:2392	EPSG:26936	EPSG:2967	EPSG:32207	EPSG:3561
EPSG:20253	EPSG:2393	EPSG:26937	EPSG:2968	EPSG:32208	EPSG:3562

EPSG:20254	EPSG:2394	EPSG:26938	EPSG:2969	EPSG:32209	EPSG:3563
EPSG:20255	EPSG:23946	EPSG:26939	EPSG:2970	EPSG:3221	EPSG:3564
EPSG:20256	EPSG:23947	EPSG:2694	EPSG:2971	EPSG:32210	EPSG:3565
EPSG:20257	EPSG:23948	EPSG:26940	EPSG:2972	EPSG:32211	EPSG:3920
EPSG:20258	EPSG:2395	EPSG:26941	EPSG:2973	EPSG:32212	EPSG:3991
EPSG:2026	EPSG:2396	EPSG:26942	EPSG:29738	EPSG:32213	EPSG:3992
EPSG:2027	EPSG:2397	EPSG:26943	EPSG:29739	EPSG:32214	EPSG:4001
EPSG:2028	EPSG:2398	EPSG:26944	EPSG:2975	EPSG:32215	EPSG:4002
EPSG:2029	EPSG:2399	EPSG:26945	EPSG:2976	EPSG:32216	EPSG:4003
EPSG:2030	EPSG:2400	EPSG:26946	EPSG:2977	EPSG:32217	EPSG:4004
EPSG:2031	EPSG:2401	EPSG:26948	EPSG:2978	EPSG:32218	EPSG:4005
EPSG:2032	EPSG:2402	EPSG:26949	EPSG:2979	EPSG:32219	EPSG:4006
EPSG:2033	EPSG:2403	EPSG:2695	EPSG:2980	EPSG:3222	EPSG:4007
EPSG:2034	EPSG:2404	EPSG:26950	EPSG:2981	EPSG:32220	EPSG:4008
EPSG:20348	EPSG:24047	EPSG:26951	EPSG:2982	EPSG:32221	EPSG:4009
EPSG:20349	EPSG:24048	EPSG:26952	EPSG:2983	EPSG:32222	EPSG:4010
EPSG:2035	EPSG:2405	EPSG:26953	EPSG:2984	EPSG:32223	EPSG:4011
EPSG:20350	EPSG:2406	EPSG:26954	EPSG:29849	EPSG:32224	EPSG:4012
EPSG:20351	EPSG:2407	EPSG:26955	EPSG:29850	EPSG:32225	EPSG:4013
EPSG:20352	EPSG:2408	EPSG:26956	EPSG:2987	EPSG:32226	EPSG:4014
EPSG:20353	EPSG:2409	EPSG:26957	EPSG:2988	EPSG:32227	EPSG:4015
EPSG:20354	EPSG:2410	EPSG:26958	EPSG:2989	EPSG:32228	EPSG:4016
EPSG:20355	EPSG:24100	EPSG:26959	EPSG:2990	EPSG:32229	EPSG:4018
EPSG:20356	EPSG:2411	EPSG:2696	EPSG:29900	EPSG:3223	EPSG:4019

EPSG:20357	EPSG:2412	EPSG:26960	EPSG:29901	EPSG:32230	EPSG:4020
EPSG:20358	EPSG:2413	EPSG:26961	EPSG:29902	EPSG:32231	EPSG:4021
EPSG:2036	EPSG:2414	EPSG:26962	EPSG:29903	EPSG:32232	EPSG:4022
EPSG:2037	EPSG:2415	EPSG:26963	EPSG:2991	EPSG:32233	EPSG:4024
EPSG:2038	EPSG:2416	EPSG:26964	EPSG:2992	EPSG:32234	EPSG:4025
EPSG:2039	EPSG:2417	EPSG:26965	EPSG:2993	EPSG:32235	EPSG:4027
EPSG:2040	EPSG:2418	EPSG:26966	EPSG:2994	EPSG:32236	EPSG:4028
EPSG:2041	EPSG:2419	EPSG:26967	EPSG:2995	EPSG:32237	EPSG:4029
EPSG:2042	EPSG:2420	EPSG:26968	EPSG:2996	EPSG:32238	EPSG:4030
EPSG:2043	EPSG:24200	EPSG:26969	EPSG:2997	EPSG:32239	EPSG:4031
EPSG:20436	EPSG:2421	EPSG:2697	EPSG:2998	EPSG:3224	EPSG:4032
EPSG:20437	EPSG:2422	EPSG:26970	EPSG:2999	EPSG:32240	EPSG:4033
EPSG:20438	EPSG:2423	EPSG:26971	EPSG:3003	EPSG:32241	EPSG:4034
EPSG:20439	EPSG:2424	EPSG:26972	EPSG:3004	EPSG:32242	EPSG:4035
EPSG:2044	EPSG:2425	EPSG:26973	EPSG:3006	EPSG:32243	EPSG:4036
EPSG:20440	EPSG:2426	EPSG:26974	EPSG:3007	EPSG:32244	EPSG:4041
EPSG:2045	EPSG:2427	EPSG:26975	EPSG:3008	EPSG:32245	EPSG:4042
EPSG:20499	EPSG:2428	EPSG:26976	EPSG:3009	EPSG:32246	EPSG:4043
EPSG:20538	EPSG:2429	EPSG:26977	EPSG:3010	EPSG:32247	EPSG:4044
EPSG:20539	EPSG:2430	EPSG:26978	EPSG:3011	EPSG:32248	EPSG:4045
EPSG:2058	EPSG:24305	EPSG:26979	EPSG:3012	EPSG:32249	EPSG:4047
EPSG:2059	EPSG:24306	EPSG:2698	EPSG:3013	EPSG:3225	EPSG:4052
EPSG:2060	EPSG:2431	EPSG:26980	EPSG:3014	EPSG:32250	EPSG:4120
EPSG:2061	EPSG:24311	EPSG:26981	EPSG:3015	EPSG:32251	EPSG:4121

EPSG:2062	EPSG:24312	EPSG:26982	EPSG:3016	EPSG:32252	EPSG:4122
EPSG:2063	EPSG:24313	EPSG:26983	EPSG:30161	EPSG:32253	EPSG:4123
EPSG:2064	EPSG:2432	EPSG:26984	EPSG:30162	EPSG:32254	EPSG:4124
EPSG:2067	EPSG:2433	EPSG:26985	EPSG:30163	EPSG:32255	EPSG:4125
EPSG:2068	EPSG:2434	EPSG:26986	EPSG:30164	EPSG:32256	EPSG:4127
EPSG:2069	EPSG:24342	EPSG:26987	EPSG:30165	EPSG:32257	EPSG:4128
EPSG:2070	EPSG:24343	EPSG:26988	EPSG:30166	EPSG:32258	EPSG:4129
EPSG:2071	EPSG:24344	EPSG:26989	EPSG:30167	EPSG:32259	EPSG:4130
EPSG:2072	EPSG:24345	EPSG:2699	EPSG:30168	EPSG:3226	EPSG:4131
EPSG:2073	EPSG:24346	EPSG:26990	EPSG:30169	EPSG:32260	EPSG:4132
EPSG:2074	EPSG:24347	EPSG:26991	EPSG:3017	EPSG:3227	EPSG:4133
EPSG:2075	EPSG:2435	EPSG:26992	EPSG:30170	EPSG:3228	EPSG:4134
EPSG:2076	EPSG:2436	EPSG:26993	EPSG:30171	EPSG:3229	EPSG:4135
EPSG:2077	EPSG:2437	EPSG:26994	EPSG:30172	EPSG:3230	EPSG:4136
EPSG:2078	EPSG:24370	EPSG:26995	EPSG:30173	EPSG:32301	EPSG:4137
EPSG:2079	EPSG:24371	EPSG:26996	EPSG:30174	EPSG:32302	EPSG:4138
EPSG:20790	EPSG:24372	EPSG:26997	EPSG:30175	EPSG:32303	EPSG:4139
EPSG:20791	EPSG:24373	EPSG:26998	EPSG:30176	EPSG:32304	EPSG:4140
EPSG:2080	EPSG:24374	EPSG:2700	EPSG:30177	EPSG:32305	EPSG:4141
EPSG:2081	EPSG:24375	EPSG:2701	EPSG:30178	EPSG:32306	EPSG:4142
EPSG:2082	EPSG:24376	EPSG:2702	EPSG:30179	EPSG:32307	EPSG:4143
EPSG:20822	EPSG:24377	EPSG:2703	EPSG:3018	EPSG:32308	EPSG:4144
EPSG:20823	EPSG:24378	EPSG:27037	EPSG:3019	EPSG:32309	EPSG:4145
EPSG:20824	EPSG:24379	EPSG:27038	EPSG:3020	EPSG:3231	EPSG:4146

EPSG:2083	EPSG:2438	EPSG:27039	EPSG:3021	EPSG:32310	EPSG:4147
EPSG:2084	EPSG:24380	EPSG:2704	EPSG:3022	EPSG:32311	EPSG:4148
EPSG:2085	EPSG:24381	EPSG:27040	EPSG:3023	EPSG:32312	EPSG:4149
EPSG:2086	EPSG:24382	EPSG:2705	EPSG:3024	EPSG:32313	EPSG:4150
EPSG:2087	EPSG:24383	EPSG:2706	EPSG:3025	EPSG:32314	EPSG:4151
EPSG:2088	EPSG:2439	EPSG:2707	EPSG:3026	EPSG:32315	EPSG:4152
EPSG:2089	EPSG:2440	EPSG:2708	EPSG:3027	EPSG:32316	EPSG:4153
EPSG:2090	EPSG:2441	EPSG:2709	EPSG:3028	EPSG:32317	EPSG:4154
EPSG:2091	EPSG:2442	EPSG:2710	EPSG:3029	EPSG:32318	EPSG:4155
EPSG:2092	EPSG:2443	EPSG:2711	EPSG:3030	EPSG:32319	EPSG:4156
EPSG:2093	EPSG:2444	EPSG:2712	EPSG:3033	EPSG:3232	EPSG:4157
EPSG:20934	EPSG:2445	EPSG:27120	EPSG:30339	EPSG:32320	EPSG:4158
EPSG:20935	EPSG:2446	EPSG:2713	EPSG:3034	EPSG:32321	EPSG:4159
EPSG:20936	EPSG:2447	EPSG:2714	EPSG:30340	EPSG:32322	EPSG:4160
EPSG:2094	EPSG:2448	EPSG:2715	EPSG:3035	EPSG:32323	EPSG:4161
EPSG:2095	EPSG:2449	EPSG:2716	EPSG:3036	EPSG:32324	EPSG:4162
EPSG:2096	EPSG:2450	EPSG:2717	EPSG:3037	EPSG:32325	EPSG:4163
EPSG:2097	EPSG:2451	EPSG:2718	EPSG:3038	EPSG:32326	EPSG:4164
EPSG:2098	EPSG:2452	EPSG:2719	EPSG:3039	EPSG:32327	EPSG:4165
EPSG:2100	EPSG:2453	EPSG:2720	EPSG:3040	EPSG:32328	EPSG:4166
EPSG:2101	EPSG:2454	EPSG:27205	EPSG:3041	EPSG:32329	EPSG:4167
EPSG:2102	EPSG:24547	EPSG:27206	EPSG:3042	EPSG:3233	EPSG:4168
EPSG:2103	EPSG:24548	EPSG:27207	EPSG:3043	EPSG:32330	EPSG:4169
EPSG:21035	EPSG:2455	EPSG:27208	EPSG:3044	EPSG:32331	EPSG:4170

EPSG:21036	EPSG:2456	EPSG:27209	EPSG:3045	EPSG:32332	EPSG:4171
EPSG:21037	EPSG:2457	EPSG:2721	EPSG:3046	EPSG:32333	EPSG:4172
EPSG:2104	EPSG:2458	EPSG:27210	EPSG:3047	EPSG:32334	EPSG:4173
EPSG:2105	EPSG:2459	EPSG:27211	EPSG:3048	EPSG:32335	EPSG:4174
EPSG:2106	EPSG:2460	EPSG:27212	EPSG:3049	EPSG:32336	EPSG:4175
EPSG:2107	EPSG:24600	EPSG:27213	EPSG:30491	EPSG:32337	EPSG:4176
EPSG:2108	EPSG:2461	EPSG:27214	EPSG:30492	EPSG:32338	EPSG:4178
EPSG:2109	EPSG:2462	EPSG:27215	EPSG:30493	EPSG:32339	EPSG:4179
EPSG:21095	EPSG:2463	EPSG:27216	EPSG:30494	EPSG:3234	EPSG:4180
EPSG:21096	EPSG:2464	EPSG:27217	EPSG:3050	EPSG:32340	EPSG:4181
EPSG:21097	EPSG:2465	EPSG:27218	EPSG:3051	EPSG:32341	EPSG:4182
EPSG:2110	EPSG:2466	EPSG:27219	EPSG:3054	EPSG:32342	EPSG:4183
EPSG:2111	EPSG:2467	EPSG:2722	EPSG:3055	EPSG:32343	EPSG:4184
EPSG:2112	EPSG:2468	EPSG:27220	EPSG:3056	EPSG:32344	EPSG:4185
EPSG:2113	EPSG:2469	EPSG:27221	EPSG:3057	EPSG:32345	EPSG:4188
EPSG:2114	EPSG:2470	EPSG:27222	EPSG:3058	EPSG:32346	EPSG:4189
EPSG:21148	EPSG:2471	EPSG:27223	EPSG:3059	EPSG:32347	EPSG:4190
EPSG:21149	EPSG:24718	EPSG:27224	EPSG:3060	EPSG:32348	EPSG:4191
EPSG:2115	EPSG:24719	EPSG:27225	EPSG:3061	EPSG:32349	EPSG:4192
EPSG:21150	EPSG:2472	EPSG:27226	EPSG:3062	EPSG:3235	EPSG:4193
EPSG:2116	EPSG:24720	EPSG:27227	EPSG:3063	EPSG:32350	EPSG:4194
EPSG:2117	EPSG:2473	EPSG:27228	EPSG:3064	EPSG:32351	EPSG:4195
EPSG:2118	EPSG:2474	EPSG:27229	EPSG:3065	EPSG:32352	EPSG:4196
EPSG:2119	EPSG:2475	EPSG:2723	EPSG:3066	EPSG:32353	EPSG:4197

EPSG:2120	EPSG:2476	EPSG:27230	EPSG:3067	EPSG:32354	EPSG:4198
EPSG:2121	EPSG:2477	EPSG:27231	EPSG:3069	EPSG:32355	EPSG:4199
EPSG:2122	EPSG:2478	EPSG:27232	EPSG:3070	EPSG:32356	EPSG:4200
EPSG:2123	EPSG:2479	EPSG:2724	EPSG:3071	EPSG:32357	EPSG:4201
EPSG:2124	EPSG:2480	EPSG:2725	EPSG:3072	EPSG:32358	EPSG:4202
EPSG:2125	EPSG:2481	EPSG:27258	EPSG:30729	EPSG:32359	EPSG:4203
EPSG:2126	EPSG:24817	EPSG:27259	EPSG:3073	EPSG:3236	EPSG:4204
EPSG:2127	EPSG:24818	EPSG:2726	EPSG:30730	EPSG:32360	EPSG:4205
EPSG:2128	EPSG:24819	EPSG:27260	EPSG:30731	EPSG:3237	EPSG:4206
EPSG:2129	EPSG:2482	EPSG:2727	EPSG:30732	EPSG:3238	EPSG:4207
EPSG:21291	EPSG:24820	EPSG:2728	EPSG:3074	EPSG:3239	EPSG:4208
EPSG:21292	EPSG:24821	EPSG:2729	EPSG:3075	EPSG:3240	EPSG:4209
EPSG:2130	EPSG:2483	EPSG:27291	EPSG:3076	EPSG:32401	EPSG:4210
EPSG:2131	EPSG:2484	EPSG:27292	EPSG:3077	EPSG:32402	EPSG:4211
EPSG:2132	EPSG:2485	EPSG:2730	EPSG:30791	EPSG:32403	EPSG:4212
EPSG:2133	EPSG:2486	EPSG:2731	EPSG:30792	EPSG:32404	EPSG:4213
EPSG:2134	EPSG:2487	EPSG:2732	EPSG:3080	EPSG:32405	EPSG:4214
EPSG:2135	EPSG:24877	EPSG:2733	EPSG:30800	EPSG:32406	EPSG:4215
EPSG:2136	EPSG:24878	EPSG:2734	EPSG:3081	EPSG:32407	EPSG:4216
EPSG:2137	EPSG:24879	EPSG:2735	EPSG:3082	EPSG:32408	EPSG:4218
EPSG:2138	EPSG:2488	EPSG:2736	EPSG:3084	EPSG:32409	EPSG:4219
EPSG:2139	EPSG:24880	EPSG:2737	EPSG:3088	EPSG:3241	EPSG:4220
EPSG:2140	EPSG:24881	EPSG:2738	EPSG:3089	EPSG:32410	EPSG:4221
EPSG:2141	EPSG:24882	EPSG:2739	EPSG:3090	EPSG:32411	EPSG:4222

EPSG:21413	EPSG:2489	EPSG:27391	EPSG:3091	EPSG:32412	EPSG:4223
EPSG:21414	EPSG:24891	EPSG:27392	EPSG:3092	EPSG:32413	EPSG:4224
EPSG:21415	EPSG:24892	EPSG:27393	EPSG:3093	EPSG:32414	EPSG:4225
EPSG:21416	EPSG:24893	EPSG:27394	EPSG:3094	EPSG:32415	EPSG:4226
EPSG:21417	EPSG:2490	EPSG:27395	EPSG:3095	EPSG:32416	EPSG:4227
EPSG:21418	EPSG:2491	EPSG:27396	EPSG:3096	EPSG:32417	EPSG:4228
EPSG:21419	EPSG:2492	EPSG:27397	EPSG:3097	EPSG:32418	EPSG:4229
EPSG:2142	EPSG:2493	EPSG:27398	EPSG:3098	EPSG:32419	EPSG:4230
EPSG:21420	EPSG:2494	EPSG:2740	EPSG:3099	EPSG:3242	EPSG:4231
EPSG:21421	EPSG:2495	EPSG:2741	EPSG:3100	EPSG:32420	EPSG:4232
EPSG:21422	EPSG:2496	EPSG:2742	EPSG:3101	EPSG:32421	EPSG:4233
EPSG:21423	EPSG:2497	EPSG:27429	EPSG:3102	EPSG:32422	EPSG:4234
EPSG:2143	EPSG:2498	EPSG:2743	EPSG:31028	EPSG:32423	EPSG:4235
EPSG:2144	EPSG:2499	EPSG:2744	EPSG:3103	EPSG:32424	EPSG:4236
EPSG:2145	EPSG:2500	EPSG:2745	EPSG:3104	EPSG:32425	EPSG:4237
EPSG:21453	EPSG:25000	EPSG:2746	EPSG:3105	EPSG:32426	EPSG:4238
EPSG:21454	EPSG:2501	EPSG:2747	EPSG:3106	EPSG:32427	EPSG:4239
EPSG:21455	EPSG:2502	EPSG:2748	EPSG:3107	EPSG:32428	EPSG:4240
EPSG:21456	EPSG:2503	EPSG:2749	EPSG:3108	EPSG:32429	EPSG:4241
EPSG:21457	EPSG:2504	EPSG:27492	EPSG:3109	EPSG:3243	EPSG:4242
EPSG:21458	EPSG:2505	EPSG:2750	EPSG:3110	EPSG:32430	EPSG:4243
EPSG:21459	EPSG:2506	EPSG:27500	EPSG:3111	EPSG:32431	EPSG:4244
EPSG:2146	EPSG:2507	EPSG:2751	EPSG:3112	EPSG:32432	EPSG:4245
EPSG:21460	EPSG:2508	EPSG:2752	EPSG:31121	EPSG:32433	EPSG:4246

EPSG:21461	EPSG:2509	EPSG:2753	EPSG:3113	EPSG:32434	EPSG:4247
EPSG:21462	EPSG:2510	EPSG:2754	EPSG:3114	EPSG:32435	EPSG:4248
EPSG:21463	EPSG:2511	EPSG:2755	EPSG:3115	EPSG:32436	EPSG:4249
EPSG:2147	EPSG:2512	EPSG:2756	EPSG:31154	EPSG:32437	EPSG:4250
EPSG:21473	EPSG:2513	EPSG:27561	EPSG:3116	EPSG:32438	EPSG:4251
EPSG:21474	EPSG:2514	EPSG:27562	EPSG:3117	EPSG:32439	EPSG:4252
EPSG:21475	EPSG:2515	EPSG:27563	EPSG:31170	EPSG:3244	EPSG:4253
EPSG:21476	EPSG:2516	EPSG:27564	EPSG:31171	EPSG:32440	EPSG:4254
EPSG:21477	EPSG:2517	EPSG:2757	EPSG:3118	EPSG:32441	EPSG:4255
EPSG:21478	EPSG:2518	EPSG:27571	EPSG:3119	EPSG:32442	EPSG:4256
EPSG:21479	EPSG:2519	EPSG:27572	EPSG:3120	EPSG:32443	EPSG:4257
EPSG:2148	EPSG:2520	EPSG:27573	EPSG:3121	EPSG:32444	EPSG:4258
EPSG:21480	EPSG:2521	EPSG:27574	EPSG:3122	EPSG:32445	EPSG:4259
EPSG:21481	EPSG:2522	EPSG:2758	EPSG:3123	EPSG:32446	EPSG:4260
EPSG:21482	EPSG:2523	EPSG:27581	EPSG:3124	EPSG:32447	EPSG:4261
EPSG:21483	EPSG:25231	EPSG:27582	EPSG:3125	EPSG:32448	EPSG:4262
EPSG:2149	EPSG:2524	EPSG:27583	EPSG:3126	EPSG:32449	EPSG:4263
EPSG:2150	EPSG:2525	EPSG:27584	EPSG:31265	EPSG:3245	EPSG:4264
EPSG:21500	EPSG:2526	EPSG:2759	EPSG:31266	EPSG:32450	EPSG:4265
EPSG:2151	EPSG:2527	EPSG:27591	EPSG:31267	EPSG:32451	EPSG:4266
EPSG:2152	EPSG:2528	EPSG:27592	EPSG:31268	EPSG:32452	EPSG:4267
EPSG:2153	EPSG:2529	EPSG:27593	EPSG:3127	EPSG:32453	EPSG:4268
EPSG:2154	EPSG:2530	EPSG:27594	EPSG:31275	EPSG:32454	EPSG:4269
EPSG:2155	EPSG:2531	EPSG:2760	EPSG:31276	EPSG:32455	EPSG:4270

EPSG:2156	EPSG:2532	EPSG:2761	EPSG:31277	EPSG:32456	EPSG:4271
EPSG:2157	EPSG:2533	EPSG:2762	EPSG:31278	EPSG:32457	EPSG:4272
EPSG:2158	EPSG:2534	EPSG:2763	EPSG:31279	EPSG:32458	EPSG:4273
EPSG:2159	EPSG:2535	EPSG:2764	EPSG:3128	EPSG:32459	EPSG:4274
EPSG:2160	EPSG:2536	EPSG:2765	EPSG:31281	EPSG:3246	EPSG:4275
EPSG:2161	EPSG:2537	EPSG:2766	EPSG:31282	EPSG:32460	EPSG:4276
EPSG:2162	EPSG:2538	EPSG:2767	EPSG:31283	EPSG:3247	EPSG:4277
EPSG:2163	EPSG:2539	EPSG:2768	EPSG:31284	EPSG:3248	EPSG:4278
EPSG:2164	EPSG:25391	EPSG:2769	EPSG:31285	EPSG:3249	EPSG:4279
EPSG:2165	EPSG:25392	EPSG:2770	EPSG:31286	EPSG:3250	EPSG:4280
EPSG:2166	EPSG:25393	EPSG:27700	EPSG:31287	EPSG:32501	EPSG:4281
EPSG:2167	EPSG:25394	EPSG:2771	EPSG:31288	EPSG:32502	EPSG:4282
EPSG:2168	EPSG:25395	EPSG:2772	EPSG:31289	EPSG:32503	EPSG:4283
EPSG:2169	EPSG:2540	EPSG:2773	EPSG:3129	EPSG:32504	EPSG:4284
EPSG:2170	EPSG:2541	EPSG:2774	EPSG:31290	EPSG:32505	EPSG:4285
EPSG:2171	EPSG:2542	EPSG:2775	EPSG:31291	EPSG:32506	EPSG:4286
EPSG:2172	EPSG:2543	EPSG:2776	EPSG:31292	EPSG:32507	EPSG:4287
EPSG:2173	EPSG:2544	EPSG:2777	EPSG:31293	EPSG:32508	EPSG:4288
EPSG:2174	EPSG:2545	EPSG:2778	EPSG:31294	EPSG:32509	EPSG:4289
EPSG:2175	EPSG:2546	EPSG:2779	EPSG:31295	EPSG:3251	EPSG:4291
EPSG:2176	EPSG:2547	EPSG:2780	EPSG:31296	EPSG:32510	EPSG:4292
EPSG:2177	EPSG:2548	EPSG:2781	EPSG:31297	EPSG:32511	EPSG:4293
EPSG:2178	EPSG:2549	EPSG:2782	EPSG:3130	EPSG:32512	EPSG:4294
EPSG:2179	EPSG:2550	EPSG:2783	EPSG:31300	EPSG:32513	EPSG:4295

EPSG:2180	EPSG:2551	EPSG:2784	EPSG:3131	EPSG:32514	EPSG:4296
EPSG:21817	EPSG:2552	EPSG:2785	EPSG:3132	EPSG:32515	EPSG:4297
EPSG:21818	EPSG:2553	EPSG:2786	EPSG:3133	EPSG:32516	EPSG:4298
EPSG:2188	EPSG:2554	EPSG:2787	EPSG:3134	EPSG:32517	EPSG:4299
EPSG:2189	EPSG:2555	EPSG:2788	EPSG:3135	EPSG:32518	EPSG:4300
EPSG:21891	EPSG:2556	EPSG:2789	EPSG:3136	EPSG:32519	EPSG:4301
EPSG:21892	EPSG:2557	EPSG:2790	EPSG:3137	EPSG:3252	EPSG:4302
EPSG:21893	EPSG:2558	EPSG:2791	EPSG:31370	EPSG:32520	EPSG:4303
EPSG:21894	EPSG:2559	EPSG:2792	EPSG:3138	EPSG:32521	EPSG:4304
EPSG:21896	EPSG:2560	EPSG:2793	EPSG:3141	EPSG:32522	EPSG:4306
EPSG:21897	EPSG:2561	EPSG:2794	EPSG:3142	EPSG:32523	EPSG:4307
EPSG:21898	EPSG:2562	EPSG:2795	EPSG:3143	EPSG:32524	EPSG:4308
EPSG:21899	EPSG:2563	EPSG:2796	EPSG:3146	EPSG:32525	EPSG:4309
EPSG:2190	EPSG:2564	EPSG:2797	EPSG:31461	EPSG:32526	EPSG:4310
EPSG:2191	EPSG:2565	EPSG:2798	EPSG:31462	EPSG:32527	EPSG:4311
EPSG:2192	EPSG:2566	EPSG:2799	EPSG:31463	EPSG:32528	EPSG:4312
EPSG:2193	EPSG:2567	EPSG:2800	EPSG:31464	EPSG:32529	EPSG:4313
EPSG:2194	EPSG:2568	EPSG:2801	EPSG:31465	EPSG:3253	EPSG:4314
EPSG:2195	EPSG:2569	EPSG:2802	EPSG:31466	EPSG:32530	EPSG:4315
EPSG:2196	EPSG:2570	EPSG:2803	EPSG:31467	EPSG:32531	EPSG:4316
EPSG:2197	EPSG:2571	EPSG:2804	EPSG:31468	EPSG:32532	EPSG:4317
EPSG:2198	EPSG:2572	EPSG:2805	EPSG:31469	EPSG:32533	EPSG:4318
EPSG:2199	EPSG:2573	EPSG:2806	EPSG:3147	EPSG:32534	EPSG:4319
EPSG:2200	EPSG:2574	EPSG:2807	EPSG:3148	EPSG:32535	EPSG:4322

EPSG:2201	EPSG:2575	EPSG:2808	EPSG:3149	EPSG:32536	EPSG:4324
EPSG:2202	EPSG:2576	EPSG:2809	EPSG:3150	EPSG:32537	EPSG:4326
EPSG:2203	EPSG:2577	EPSG:2810	EPSG:3151	EPSG:32538	EPSG:4600
EPSG:22032	EPSG:2578	EPSG:2811	EPSG:3152	EPSG:32539	EPSG:4601
EPSG:22033	EPSG:2579	EPSG:2812	EPSG:31528	EPSG:3254	EPSG:4602
EPSG:2204	EPSG:2580	EPSG:2813	EPSG:31529	EPSG:32540	EPSG:4603
EPSG:2205	EPSG:2581	EPSG:2814	EPSG:3154	EPSG:32541	EPSG:4604
EPSG:2206	EPSG:2582	EPSG:2815	EPSG:3155	EPSG:32542	EPSG:4605
EPSG:2207	EPSG:25828	EPSG:2816	EPSG:3156	EPSG:32543	EPSG:4606
EPSG:2208	EPSG:25829	EPSG:2817	EPSG:3157	EPSG:32544	EPSG:4607
EPSG:2209	EPSG:2583	EPSG:2818	EPSG:3158	EPSG:32545	EPSG:4608
EPSG:22091	EPSG:25830	EPSG:2819	EPSG:3159	EPSG:32546	EPSG:4609
EPSG:22092	EPSG:25831	EPSG:28192	EPSG:3160	EPSG:32547	EPSG:4610
EPSG:2210	EPSG:25832	EPSG:2820	EPSG:31600	EPSG:32548	EPSG:4611
EPSG:2211	EPSG:25833	EPSG:2821	EPSG:3161	EPSG:32549	EPSG:4612
EPSG:2212	EPSG:25834	EPSG:2822	EPSG:3162	EPSG:3255	EPSG:4613
EPSG:2213	EPSG:25835	EPSG:2823	EPSG:3163	EPSG:32550	EPSG:4614
EPSG:2214	EPSG:25836	EPSG:28232	EPSG:3164	EPSG:32551	EPSG:4615
EPSG:2215	EPSG:25837	EPSG:2824	EPSG:3165	EPSG:32552	EPSG:4616
EPSG:2216	EPSG:25838	EPSG:2825	EPSG:3166	EPSG:32553	EPSG:4617
EPSG:2217	EPSG:2584	EPSG:2826	EPSG:3169	EPSG:32554	EPSG:4618
EPSG:22171	EPSG:2585	EPSG:2827	EPSG:3170	EPSG:32555	EPSG:4619
EPSG:22172	EPSG:2586	EPSG:2828	EPSG:31700	EPSG:32556	EPSG:4620
EPSG:22173	EPSG:2587	EPSG:2829	EPSG:3171	EPSG:32557	EPSG:4621

EPSG:22174	EPSG:2588	EPSG:2830	EPSG:3172	EPSG:32558	EPSG:4622
EPSG:22175	EPSG:25884	EPSG:2831	EPSG:3176	EPSG:32559	EPSG:4623
EPSG:22176	EPSG:2589	EPSG:2832	EPSG:3177	EPSG:3256	EPSG:4624
EPSG:22177	EPSG:2590	EPSG:2833	EPSG:3178	EPSG:32560	EPSG:4625
EPSG:22181	EPSG:2591	EPSG:2834	EPSG:3179	EPSG:3257	EPSG:4626
EPSG:22182	EPSG:2592	EPSG:28348	EPSG:3180	EPSG:3258	EPSG:4627
EPSG:22183	EPSG:2593	EPSG:28349	EPSG:3181	EPSG:325833	EPSG:4628
EPSG:22184	EPSG:25932	EPSG:2835	EPSG:3182	EPSG:3259	EPSG:4629
EPSG:22185	EPSG:2594	EPSG:28350	EPSG:3183	EPSG:3260	EPSG:4630
EPSG:22186	EPSG:2595	EPSG:28351	EPSG:31838	EPSG:32601	EPSG:4631
EPSG:22187	EPSG:2596	EPSG:28352	EPSG:31839	EPSG:32602	EPSG:4632
EPSG:2219	EPSG:2597	EPSG:28353	EPSG:3184	EPSG:32603	EPSG:4633
EPSG:22191	EPSG:2598	EPSG:28354	EPSG:3185	EPSG:32604	EPSG:4634
EPSG:22192	EPSG:2599	EPSG:28355	EPSG:3186	EPSG:32605	EPSG:4635
EPSG:22193	EPSG:2600	EPSG:28356	EPSG:3187	EPSG:32606	EPSG:4636
EPSG:22194	EPSG:2601	EPSG:28357	EPSG:3188	EPSG:32607	EPSG:4637
EPSG:22195	EPSG:2602	EPSG:28358	EPSG:3189	EPSG:32608	EPSG:4638
EPSG:22196	EPSG:2603	EPSG:2836	EPSG:3190	EPSG:32609	EPSG:4639
EPSG:22197	EPSG:2604	EPSG:2837	EPSG:31900	EPSG:3261	EPSG:4640
EPSG:2220	EPSG:2605	EPSG:2838	EPSG:31901	EPSG:32610	EPSG:4641
EPSG:2222	EPSG:2606	EPSG:2839	EPSG:3191	EPSG:32611	EPSG:4642
EPSG:2223	EPSG:2607	EPSG:2840	EPSG:3192	EPSG:32612	EPSG:4643
EPSG:22234	EPSG:2608	EPSG:28402	EPSG:3193	EPSG:32613	EPSG:4644
EPSG:22235	EPSG:2609	EPSG:28403	EPSG:3194	EPSG:32614	EPSG:4645

EPSG:22236	EPSG:2610	EPSG:28404	EPSG:3195	EPSG:32615	EPSG:4646
EPSG:2224	EPSG:2611	EPSG:28405	EPSG:3196	EPSG:32616	EPSG:4657
EPSG:2225	EPSG:2612	EPSG:28406	EPSG:31965	EPSG:32617	EPSG:4658
EPSG:2226	EPSG:2613	EPSG:28407	EPSG:31966	EPSG:32618	EPSG:4659
EPSG:2227	EPSG:2614	EPSG:28408	EPSG:31967	EPSG:32619	EPSG:4660
EPSG:2228	EPSG:2615	EPSG:28409	EPSG:31968	EPSG:3262	EPSG:4661
EPSG:2229	EPSG:2616	EPSG:2841	EPSG:31969	EPSG:32620	EPSG:4662
EPSG:2230	EPSG:2617	EPSG:28410	EPSG:3197	EPSG:32621	EPSG:4663
EPSG:2231	EPSG:2618	EPSG:28411	EPSG:31970	EPSG:32622	EPSG:4664
EPSG:2232	EPSG:2619	EPSG:28412	EPSG:31971	EPSG:32623	EPSG:4665
EPSG:2233	EPSG:26191	EPSG:28413	EPSG:31972	EPSG:32624	EPSG:4666
EPSG:22332	EPSG:26192	EPSG:28414	EPSG:31973	EPSG:32625	EPSG:4667
EPSG:2234	EPSG:26193	EPSG:28415	EPSG:31974	EPSG:32626	EPSG:4668
EPSG:2235	EPSG:26194	EPSG:28416	EPSG:31975	EPSG:32627	EPSG:4669
EPSG:2236	EPSG:26195	EPSG:28417	EPSG:31976	EPSG:32628	EPSG:4670
EPSG:2237	EPSG:2620	EPSG:28418	EPSG:31977	EPSG:32629	EPSG:4671
EPSG:2238	EPSG:2621	EPSG:28419	EPSG:31978	EPSG:3263	EPSG:4672
EPSG:2239	EPSG:2622	EPSG:2842	EPSG:31979	EPSG:32630	EPSG:4673
EPSG:22391	EPSG:2623	EPSG:28420	EPSG:3198	EPSG:32631	EPSG:4674
EPSG:22392	EPSG:26237	EPSG:28421	EPSG:31980	EPSG:32632	EPSG:4675
EPSG:2240	EPSG:2624	EPSG:28422	EPSG:31981	EPSG:32633	EPSG:4676
EPSG:2241	EPSG:2625	EPSG:28423	EPSG:31982	EPSG:32634	EPSG:4677
EPSG:2242	EPSG:2626	EPSG:28424	EPSG:31983	EPSG:32635	EPSG:4678
EPSG:2243	EPSG:2627	EPSG:28425	EPSG:31984	EPSG:32636	EPSG:4679

EPSG:2244	EPSG:2628	EPSG:28426	EPSG:31985	EPSG:32637	EPSG:4680
EPSG:2245	EPSG:2629	EPSG:28427	EPSG:31986	EPSG:32638	EPSG:4681
EPSG:2246	EPSG:2630	EPSG:28428	EPSG:31987	EPSG:32639	EPSG:4682
EPSG:2247	EPSG:2631	EPSG:28429	EPSG:31988	EPSG:3264	EPSG:4683
EPSG:2248	EPSG:2632	EPSG:2843	EPSG:31989	EPSG:32640	EPSG:4684
EPSG:2249	EPSG:2633	EPSG:28430	EPSG:3199	EPSG:32641	EPSG:4685
EPSG:2250	EPSG:26331	EPSG:28431	EPSG:31990	EPSG:32642	EPSG:4686
EPSG:2251	EPSG:26332	EPSG:28432	EPSG:31991	EPSG:32643	EPSG:4687
EPSG:2252	EPSG:2634	EPSG:2844	EPSG:31992	EPSG:32644	EPSG:4688
EPSG:22521	EPSG:2635	EPSG:2845	EPSG:31993	EPSG:32645	EPSG:4689
EPSG:22522	EPSG:2636	EPSG:2846	EPSG:31994	EPSG:32646	EPSG:4690
EPSG:22523	EPSG:2637	EPSG:28462	EPSG:31995	EPSG:32647	EPSG:4691
EPSG:22524	EPSG:2638	EPSG:28463	EPSG:31996	EPSG:32648	EPSG:4692
EPSG:22525	EPSG:2639	EPSG:28464	EPSG:31997	EPSG:32649	EPSG:4693
EPSG:2253	EPSG:26391	EPSG:28465	EPSG:31998	EPSG:3265	EPSG:4694
EPSG:2254	EPSG:26392	EPSG:28466	EPSG:31999	EPSG:32650	EPSG:4695
EPSG:2255	EPSG:26393	EPSG:28467	EPSG:3200	EPSG:32651	EPSG:4696
EPSG:2256	EPSG:2640	EPSG:28468	EPSG:32000	EPSG:32652	EPSG:4697
EPSG:2257	EPSG:2641	EPSG:28469	EPSG:32001	EPSG:32653	EPSG:4698
EPSG:2258	EPSG:2642	EPSG:2847	EPSG:32002	EPSG:32654	EPSG:4699
EPSG:2259	EPSG:2643	EPSG:28470	EPSG:32003	EPSG:32655	EPSG:4700
EPSG:2260	EPSG:26432	EPSG:28471	EPSG:32005	EPSG:32656	EPSG:4701
EPSG:2261	EPSG:2644	EPSG:28472	EPSG:32006	EPSG:32657	EPSG:4702
EPSG:2262	EPSG:2645	EPSG:28473	EPSG:32007	EPSG:32658	EPSG:4703

EPSG:2263	EPSG:2646	EPSG:28474	EPSG:32008	EPSG:32659	EPSG:4704
EPSG:2264	EPSG:2647	EPSG:28475	EPSG:32009	EPSG:3266	EPSG:4705
EPSG:2265	EPSG:2648	EPSG:28476	EPSG:3201	EPSG:32660	EPSG:4706
EPSG:2266	EPSG:2649	EPSG:28477	EPSG:32010	EPSG:32664	EPSG:4707
EPSG:2267	EPSG:2650	EPSG:28478	EPSG:32011	EPSG:32665	EPSG:4708
EPSG:2268	EPSG:2651	EPSG:28479	EPSG:32012	EPSG:32666	EPSG:4709
EPSG:2269	EPSG:2652	EPSG:2848	EPSG:32013	EPSG:32667	EPSG:4710
EPSG:2270	EPSG:2653	EPSG:28480	EPSG:32014	EPSG:3267	EPSG:4711
EPSG:22700	EPSG:2654	EPSG:28481	EPSG:32015	EPSG:3268	EPSG:4712
EPSG:2271	EPSG:2655	EPSG:28482	EPSG:32016	EPSG:3269	EPSG:4713
EPSG:2272	EPSG:2656	EPSG:28483	EPSG:32017	EPSG:3270	EPSG:4714
EPSG:2273	EPSG:2657	EPSG:28484	EPSG:32018	EPSG:32701	EPSG:4715
EPSG:2274	EPSG:2658	EPSG:28485	EPSG:32019	EPSG:32702	EPSG:4716
EPSG:2275	EPSG:2659	EPSG:28486	EPSG:3202	EPSG:32703	EPSG:4717
EPSG:2276	EPSG:26591	EPSG:28487	EPSG:32020	EPSG:32704	EPSG:4718
EPSG:2277	EPSG:26592	EPSG:28488	EPSG:32021	EPSG:32705	EPSG:4719
EPSG:22770	EPSG:2660	EPSG:28489	EPSG:32022	EPSG:32706	EPSG:4720
EPSG:2278	EPSG:2661	EPSG:2849	EPSG:32023	EPSG:32707	EPSG:4721
EPSG:22780	EPSG:2662	EPSG:28490	EPSG:32024	EPSG:32708	EPSG:4722
EPSG:2279	EPSG:2663	EPSG:28491	EPSG:32025	EPSG:32709	EPSG:4723
EPSG:2280	EPSG:26632	EPSG:28492	EPSG:32026	EPSG:3271	EPSG:4724
EPSG:2281	EPSG:2664	EPSG:2850	EPSG:32027	EPSG:32710	EPSG:4725
EPSG:2282	EPSG:2665	EPSG:2851	EPSG:32028	EPSG:32711	EPSG:4726
EPSG:2283	EPSG:2666	EPSG:2852	EPSG:32029	EPSG:32712	EPSG:4727

EPSG:22832	EPSG:2667	EPSG:2853	EPSG:3203	EPSG:32713	EPSG:4728
EPSG:2284	EPSG:2668	EPSG:2854	EPSG:32030	EPSG:32714	EPSG:4729
EPSG:2285	EPSG:2669	EPSG:2855	EPSG:32031	EPSG:32715	EPSG:4730
EPSG:2286	EPSG:26692	EPSG:2856	EPSG:32033	EPSG:32716	EPSG:4731
EPSG:2287	EPSG:2670	EPSG:2857	EPSG:32034	EPSG:32717	EPSG:4732
EPSG:2288	EPSG:26701	EPSG:2858	EPSG:32035	EPSG:32718	EPSG:4733
EPSG:2289	EPSG:26702	EPSG:2859	EPSG:32036	EPSG:32719	EPSG:4734
EPSG:2290	EPSG:26703	EPSG:2860	EPSG:32037	EPSG:3272	EPSG:4735
EPSG:2291	EPSG:26704	EPSG:28600	EPSG:32038	EPSG:32720	EPSG:4736
EPSG:2292	EPSG:26705	EPSG:2861	EPSG:32039	EPSG:32721	EPSG:4737
EPSG:2294	EPSG:26706	EPSG:2862	EPSG:3204	EPSG:32722	EPSG:4738
EPSG:2295	EPSG:26707	EPSG:2863	EPSG:32040	EPSG:32723	EPSG:4739
EPSG:22991	EPSG:26708	EPSG:2864	EPSG:32041	EPSG:32724	EPSG:4740
EPSG:22992	EPSG:26709	EPSG:2865	EPSG:32042	EPSG:32725	EPSG:4741
EPSG:22993	EPSG:2671	EPSG:2866	EPSG:32043	EPSG:32726	EPSG:4742
EPSG:22994	EPSG:26710	EPSG:2867	EPSG:32044	EPSG:32727	EPSG:4743
EPSG:23028	EPSG:26711	EPSG:2868	EPSG:32045	EPSG:32728	EPSG:4744
EPSG:23029	EPSG:26712	EPSG:2869	EPSG:32046	EPSG:32729	EPSG:4745
EPSG:23030	EPSG:26713	EPSG:2870	EPSG:32047	EPSG:3273	EPSG:4746
EPSG:23031	EPSG:26714	EPSG:2871	EPSG:32048	EPSG:32730	EPSG:4747
EPSG:23032	EPSG:26715	EPSG:2872	EPSG:32049	EPSG:32731	EPSG:4748
EPSG:23033	EPSG:26716	EPSG:2873	EPSG:3205	EPSG:32732	EPSG:4749
EPSG:23034	EPSG:26717	EPSG:2874	EPSG:32050	EPSG:32733	EPSG:4750
EPSG:23035	EPSG:26718	EPSG:2875	EPSG:32051	EPSG:32734	EPSG:4751

EPSG:23036	EPSG:26719	EPSG:2876	EPSG:32052	EPSG:32735	EPSG:4752
EPSG:23037	EPSG:2672	EPSG:2877	EPSG:32053	EPSG:32736	EPSG:4753
EPSG:23038	EPSG:26720	EPSG:2878	EPSG:32054	EPSG:32737	EPSG:4754
EPSG:2308	EPSG:26721	EPSG:2879	EPSG:32055	EPSG:32738	EPSG:4755
EPSG:2309	EPSG:26722	EPSG:2880	EPSG:32056	EPSG:32739	EPSG:4756
EPSG:23090	EPSG:26729	EPSG:2881	EPSG:32057	EPSG:3274	EPSG:4801
EPSG:23095	EPSG:2673	EPSG:2882	EPSG:32058	EPSG:32740	EPSG:4802
EPSG:2310	EPSG:26730	EPSG:2883	EPSG:3206	EPSG:32741	EPSG:4803
EPSG:2311	EPSG:26732	EPSG:2884	EPSG:32061	EPSG:32742	EPSG:4804
EPSG:2312	EPSG:26733	EPSG:2885	EPSG:32062	EPSG:32743	EPSG:4805
EPSG:2313	EPSG:26734	EPSG:2886	EPSG:32064	EPSG:32744	EPSG:4806
EPSG:2315	EPSG:26735	EPSG:2887	EPSG:32065	EPSG:32745	EPSG:4807
EPSG:2316	EPSG:26736	EPSG:2888	EPSG:32066	EPSG:32746	EPSG:4808
EPSG:2317	EPSG:26737	EPSG:2889	EPSG:32067	EPSG:32747	EPSG:4809
EPSG:2318	EPSG:26738	EPSG:2890	EPSG:3207	EPSG:32748	EPSG:4810
EPSG:2319	EPSG:26739	EPSG:2891	EPSG:32074	EPSG:32749	EPSG:4811
EPSG:2320	EPSG:2674	EPSG:2892	EPSG:32075	EPSG:32750	EPSG:4813
EPSG:2321	EPSG:26740	EPSG:2893	EPSG:32076	EPSG:32751	EPSG:4814
EPSG:2322	EPSG:26741	EPSG:2894	EPSG:32077	EPSG:32752	EPSG:4815
EPSG:2323	EPSG:26742	EPSG:2895	EPSG:3208	EPSG:32753	EPSG:4816
EPSG:23239	EPSG:26743	EPSG:2896	EPSG:32081	EPSG:32754	EPSG:4817
EPSG:2324	EPSG:26744	EPSG:2897	EPSG:32082	EPSG:32755	EPSG:4818
EPSG:23240	EPSG:26745	EPSG:2898	EPSG:32083	EPSG:32756	EPSG:4819
EPSG:2325	EPSG:26746	EPSG:2899	EPSG:32084	EPSG:32757	EPSG:4820

EPSG:2326	EPSG:26747	EPSG:28991	EPSG:32085	EPSG:32758	EPSG:4821
EPSG:2327	EPSG:26748	EPSG:28992	EPSG:32086	EPSG:32759	EPSG:4901
EPSG:2328	EPSG:26749	EPSG:2900	EPSG:3209	EPSG:32760	EPSG:4902
EPSG:2329	EPSG:2675	EPSG:2901	EPSG:32098	EPSG:32766	EPSG:4903
EPSG:2330	EPSG:26750	EPSG:2902	EPSG:32099	EPSG:3294	EPSG:4904
EPSG:2331	EPSG:26751	EPSG:2903	EPSG:3210	EPSG:3296	GEO_CRS_19
EPSG:2332	EPSG:26752	EPSG:2904	EPSG:32100	EPSG:3297	GEO_CRS_204