OpenGIS Project Document 01-014r3

TITLE:	**Recommended Definition Data for Coordinate Reference Systems and Coordinate Transformations**	

AUTHOR:	**Name:**	**Arliss Whiteside**
	Address:	**BAE SYSTEMS Mission Solutions** **16250 Technology Drive** **San Diego, CA 92127**
	Phone:	**858-592-1608**
	Email:	**Arliss.Whiteside@baesystems.com**
	Name:	**John Bobbitt**
	Address:	**Petrotechnical Open Software Corporation** **9801 Westheimer, Suite 450** **Houston, Texas 77042**
	Phone:	**713-267-5174**
	Email:	**bobbitt@posc.org**
	Name:	**Roel Nicolai**
	Address:	**Shell International Exploration and Production** **Technology Applications and Research** **Volmerlaan 8, P.O. Box 60** **2280 AB Rijswijk, The Netherlands**
	Phone:	**+31 70 311 3652**
	Email:	**r.nicolai@siep.shell.com**

DATE:	**April 5th, 2001**
CATEGORY:	**OpenGIS© Recommendation Paper**

The attached document is a proposed OGC Recommendation Paper that specifies OGC standard definition data for Coordinate Reference Systems (CRSs) and Coordinate Transformations (CTs). This proposal is intended for consideration by the Coordinate Transformation (CT) Working Group (WG).

Those familiar with the format of OGC documents will notice that the attached paper uses a different format. This proposed Recommendation Paper uses the tentative new format for OGC Implementation Specifications that is similar to the ISO document format. This new format was developed by Martin Ford.

Document 01-014 was a revision of OGC document 00-040r8, with the same title. A new document number is used due to the new calendar year.

Open GIS Consortium Inc.

Date: 2001-03-08

Reference number of this OpenGIS© project document: **OGC 01-014r3**

Version: 1.0.1

Category: OpenGIS© Recommendation Paper

Editors: Arliss Whiteside
Roel Nicolai

OpenGIS© Specification — Recommended Definition Data for Coordinate Reference Systems and Coordinate Transformations

Document type: OpenGIS© Interface Specification
Document subtype: not applicable
Document stage: Final
Document language: English
File name: 01-014r3

Contents

i. Preface

This document is an OGC Recommendation Paper, specifying a Data Model to be adapted and used by multiple OGC Implementation Specifications. As for an abstract model, a variety of Implementation Specifications are expected to be based on this specified Data Model. This document is a Recommendation Paper because the specified Data Model is more general than an OGC Implementation Specification and more specific than the OGC Abstract Specification.

This Data Model was developed using object-oriented analysis and design thinking. This Data Model is recorded in XML format for a combination of several reasons, including:

a) CRS and CT definition data will very often be transferred in XML format, so this data must be defined such that subsets can be effectively encoded in XML, in various Implementation Specifications.

b) Although XML cannot represent interface classes or class operations, this is not a problem for representing the Data Model, since no operations are allowed.

c) Many interested OGC participants find XML easier to understand than UML (often because they are more familiar with XML).

If considered useful and worthwhile, a UML equivalent to this XML Data Model could be generated in a straight-forward manner. For example, the UML equivalent model might be generated by applying the correspondences described in Annex F.

This document does not now include two-letter prefixes to XML names, partially because it is not clear what prefixes to use. Obviously, two-letter prefixes can be added to all or selected XML names if considered desirable, after specific prefixes to be used are selected.

ii. Authors

Name:	Arliss Whiteside
Address:	BAE SYSTEMS Mission Solutions 16250 Technology Drive San Diego, CA 92127
Phone:	858-592-1608
Email:	Arliss.Whiteside@baesystems.com

Name: John Bobbitt

Address: Petrotechnical Open Software Corporation
10777 Westheimer, Suite 275
Houston, Texas 77042

Phone: 713-267-5174

Email: bobbitt@posc.org

Name: Roel Nicolai

Address: Shell International Exploration and Production B.V.
Technology Application and Research,
EPT-AGX
Volmerlaan 8
2288 GD Rijswijk
The Netherlands

Phone: +31 70 311 3652

Email: r.nicolai@siep.shell.com

Foreword

This specification harmonizes and improves the relevant XML work previously done by the Open GIS Consortium (OGC). The OGC-accepted CT Implementation Specification already specifies a XML DTD for CRSs and CTs. The approved GML Recommendation Paper now includes an informative annex that specifies a XML DTD for CRSs, including some CTs. These two XML DTDs are significantly different, and both appear to be incomplete. Use of an OGC standard for CRS and CT definition data is believed needed when the current GML Recommendation Paper is converted to an Implementation Specification.

This OGC standard data model for CRS and CT definition data is intended for initial use by the Geography Markup Language (GML) and in Coordinate Transformation (CT) Implementation Specifications. That is, each of these specifications is expected to use a subset and/or superset of the OGC standard definition data.

In future revisions of this specification, the current XML DTDs will be converted to XML Schema.

Many of the geospatial concepts used in this specification are from ISO DIS 19111 and Topic volume 2 of the OGC Abstract Specification (OGC document 01-102). However, whereas ISO DIS 19111 distinguishes between two types of Coordinate Operation, viz. Coordinate Conversion and Coordinate Transformation, this document, although implementing these concepts materially, refers to both collectively as “Coordinate Transformation”, thereby not using the term “Coordinate Operation”

NOTE This paper assumes that OGC Abstract Specification topic 2 (OGC document 99-102r1) is expanded to include the accepted additions in OGC document 00-026) and accepted editing in OGC document 00-046r1.

This document contains six annexes. Annex A is normative, and annexes B through F are informative.

Attention is drawn to the possibility that some elements of this document may be the subject of patent rights. The Open GIS Consortium Inc. shall not be held responsible for identifying any or all such patent rights.

Introduction

This document specifies a standard model for data defining Coordinate Reference Systems (CRSs) and Coordinate Transformations (CTs). This CRS and CT definition data will often be transferred between client and server software that implements various standardised interfaces and data formats. This specified general CRS and CT definition data is intended for use in multiple OGC Implementation Specifications. That is, each of these specifications is expected to use a subset and/or superset of this standard CRS and CT definition data.

This object-oriented Data Model supplements the Abstract Model now included in OGC Abstract Specification Topic 2: Spatial Reference Systems. That Abstract Model defines the concepts of coordinate reference systems and coordinate transformations in a manner that defines useful operations, in addition to defining most of the data needed to perform those operations. This Data Model just defines the data needed to perform operations, with emphasis on all the data that may need to be transferred between client and server software. This Data Model also supports transfer of more or less data, as may be needed in different applications. Specifically, this Data Model allows complete data to be transferred when complete data is needed, and allows only the minimum identifier data to be transferred when only an identifier is needed.

The position or location of a point can be described using coordinates. Such coordinates are unambiguous only when the coordinate reference system on which those coordinates are based is fully defined. Each position is described by a set of coordinates based on a specified coordinate reference system. Coordinates are often used in datasets in which all coordinates belong to the same coordinate reference system. This paper specifies data for defining coordinate reference systems for point coordinates.

The same point position will usually have different coordinates in different coordinate reference systems. There are a large number of different coordinate reference systems in current use. Coordinates from different datasets will thus often have different coordinate reference systems. In order to use together positions from different coordinate reference systems, known point coordinates often must be transformed into the corresponding coordinates in a different coordinate reference system. This paper specifies data for defining such coordinate transformations

OpenGIS© Specification — Recommended Definition Data for Coordinate Reference Systems and Coordinate Transformations

1 Scope

This OpenGIS© Specification defines standard definition data for Coordinate Reference Systems (CRSs) and Coordinate Transformations (CTs). This definition data is intended for transfer between client and server software that uses OGC standard interfaces, as specified in other OGC documents. The expected uses of CRS and CT definition data transfer include those described in Annex B (informative).

Definition data is specified herein for describing multiple types of CRSs, including:

a) Geocentric coordinate reference systems

b) Geographic coordinate reference systems

c) Projected coordinate reference systems

d) Local coordinate reference systems

e) Vertical coordinate reference systems

f) Temporal coordinate reference systems

g) Image coordinate reference systems

h) Compound coordinate reference systems, combining other coordinate reference systems

Definition data is specified herein for describing multiple types of CTs, including:

a) Concatenated coordinate transformations, combining coordinate transformations

b) Parameterized coordinate transformations

c) Pass-through coordinate transformations

d) Inverse coordinate transformations

The current scope of this Data Model does not include:

a) Vertical coordinate reference systems with time units

b) Possible changes in the handling of units

c) Scheduled upgrades to OGC Topic 2 Abstract model

This OpenGIS© standard specifies standard CRS and CT definition data to be used by multiple separate OGC interface Implementation Specifications. Each such Implementation Specification shall specify one or more subsets/supersets of the definition data specified herein. Those standard interfaces are implemented by service software that performs functions for separate client software. Those Implementation Specifications, or profiles of those Implementation Specifications, can use any Distributed Computing Platform (DCP).

2 Conformance

Conformance with this specification shall be checked using all the relevant tests specified in Annex A (normative). The framework, concepts, and methodology for testing, and the criteria to be achieved to claim conformance, are specified in ISO 19105: Geographic information — Conformance and Testing.

3 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this specification. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this specification are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies.

OGC 01-102, The OpenGIS™ Abstract Specification, Topic 2: Spatial Reference Systems.

NOTE This paper assumes that OGC Abstract Specification topic 2 (OGC document 99-102r1) is expanded to include the accepted additions in OGC document 00-026) and accepted editing in OGC document 00-046r1.

ISO/TC 211 DIS 19111, Geographic information – Spatial referencing by coordinates.

European Petroleum Survey Group: EPSG Geodesy Parameters V 6.0, available through EPSG: http://www.epsg.org/

4 Conventions

4.1 Symbols (and abbreviated terms)

API	Application Program Interface
COM	Component Object Model
CORBA	Common Object Request Broker Architecture
COTS	Commercial Off The Shelf
CRS	Coordinate Reference System

CT	Coordinate Transformation
DCP	Distributed Computing Platform
DCOM	Distributed Component Object Model
DTD	Document Type Definition
ISO	International Organization for Standardization
OGC	Open GIS Consortium
UML	Unified Modeling Language
WKT	Well Known Text
XML	eXtensible Markup Language
1D	One Dimensional
2D	Two Dimensional
3D	Three Dimensional

4.2 XML use conventions

This document uses a set of conventions for use of XML, that currently include:

a) In XML element and attribute names, use full English words (not non-common abbreviations), and combine multiple words without an underscore or other character between words. Capitalize the first letter of each combined word after the first.

b) Capitalize the first letter of each XML element and attribute name that would be a class in the corresponding UML object model (and thus have a capitalized name).

c) Do not capitalize the first letter of each XML element name that would be an attribute in the corresponding UML object model (and not have a capitalized name).

d) Do not capitalize the first letter of each alternate value of a XML attribute.

e) In XML element and attribute names, do not use two-letter prefixes.

f) Prefer use of XML element containing a single character string, over use of an XML attribute.

g) Not use the Entity construct in XML DTDs.

NOTE Note that this set of XML conventions might be changed in future versions of this document. Also, OGC Implementation Specifications might deviate from this set of XML conventions. For example, Implementation Specifications might use the Entity construct in XML DTDs. Specifically, the Entity construct might be used for Identifier and NameSet, instead using of the element construct.

5 Data model overview (informative)

5.1 Introduction

The specified Data Model includes definition data for both Coordinate Reference Systems (CRSs) and Coordinate Transformations (CTs). Multiple types of CRSs and CTs are supported, as summarized in Clause 1. This definition data includes identification data and specification data, both included as needed. When applicable, definition data for a CRS includes definition data for one or more CTs, and definition data for a CT includes definition data for two CRSs.

This Data Model is intended for use in multiple OGC Implementation Specifications. Each of these specifications is expected to transfer CRS and/or CT definition data between client and server software. This definition data will often be transferred in XML documents. However, alternative formats can be used, such as Well Known Text (WKT). The expected uses of CRS and CT data transfer include those described in Annex B (informative).

Each relevant Implementation Specification will transfer CRS and/or CT definition data for one or more purposes, using one or more types of XML or WKT documents. For each type of document transferred, the Implementation Specification will specify a subset or superset of the CRS and CT definition data specified by this Data Model. Each such Implementation Specification will thus specify one or more subsets/supersets of the general definition data specified here.

5.2 Data model structure

The specified Data Model includes multiple alternative top level XML elements that are intended to be used where needed. (That is, there is not a single top level element that will be the basis for all XML documents.) The alternative top level XML elements include at least:

a) CoordinateReferenceSystem. This element can be used to transfer the definition of one coordinate reference system.

b) CoordinateTransformationMetadata. This element can be used to transfer the definition of a coordinate transformation from one specified source coordinate reference system to one specified target coordinate reference system.

c) CoordinateTransformationDefinition. This element can be used to transfer the definition of one coordinate transformation, without the definitions of the associated source and target coordinate reference systems.

The CoordinateReferenceSystem XML element uses many lower level elements containing data structures, including the elements named:

a) CartesianCoordinateSystem

b) ObliqueCartesianCoordinateSystem

c) EllipsoidalCoordinateSystem

d) VerticalCoordinateSystem

e) PolarCoordinateSystem

f) CylindricalCoordinateSystem

g) LinearCoordinateSystem

h) UserDefinedCoordinateSystem

i) CoordinateAxis

j) CompoundCRS

k) GeocentricCRS

l) Geographic3dCRS

m) Geographic2dCRS

n) ProjectedCRS

o) LocalCRS

p) ImageCRS

q) VerticalCRS

r) TemporalCRS

s) VerticalDatum

t) LocalDatum

u) ImageDatum

v) GeodeticDatum

w) Ellipsoid

x) PrimeMeridian

The CoordinateTransformationDefinition XML element also uses many lower level elements containing data structures, including the elements named:

a) PositionErrorEstimates

b) CovarianceElement

c) ConcatenatedTransformation

d) InverseTransformation

e) PassThroughTransformation

f) ParameterizedTransformation

g) Parameter

h) TransformationMethod

The elements listed above share some lower level XML elements containing data structures, including the elements named:

a) Identifier

b) NameSet

c) LinearUnit

d) AngularUnit

e) TimeUnit

f) PixelSpacingUnit

g) ValidityRegion

5.3 Design patterns

This Data Model uses several “design patterns”. Each fundamental entity is represented by a “definition” data class. These entities include coordinate transformations, coordinate reference systems, datums, ellipsoids, prime meridians, and units. The various types of definitions are included in multiple packages. Most packages use (or reference) all or a portion of one or more other packages.

Selected portions of each package can be separately used by an OGC application, when only some portions are needed by that application. Also, each package contains a number of places where a data item or structure is optional in this Data Model. Most specific applications will never include selected optional data. In addition, some specific applications may always include selected optional data.

The “definition” of a fundamental entity is allowed to contain at least three parts: "identifier", “name set”, and "specification". The "identifier" and most of the "specification" parts are intended to be used by software. The “name set” part is human understandable, and is not intended to be interpreted by software. Each of these three parts is optional, but either the "identifier" or the "specification" part must be included. These alternatives allow a “definition” to be used in two ways:

a) Only the "identifier" part can be included for a geospatial entity that is “well-known” to the XML receiver. Well-known means that the corresponding "specification" information can be found elsewhere when needed, using only this "identifier". This “elsewhere” is never in the same XML document that contains this identifier-only definition, is usually not in any public XML document, and is never in a XML document whose name or URI is known to the XML sender.

b) Only the "specification" data can be included for a custom geospatial entity that is not uniquely or permanently identified.

Both the "identifier" and "specification" parts can be included for a fundamental entity, but this should generally not be done (except in a database not otherwise discussed herein). Specifically, the "specification" part should not be included with the "identifier" part of an entity that is “well-known” to the XML receiver, since these two forms of information are redundant and may conflict. If both are included, the included "specification" part should be used by the XML receiver, instead of the included "identifier" part.

In both of the ways listed above, the "name set" part can be included if useful, and it will often be useful to a human user. However, the "name set" part may not be useful when the included "identifier" or "specification" part is sufficiently human understandable.

Many XML elements need to include one or both of two parts. These two parts are often the “Identifier” element and a “Definition” set of elements. In these situations, this document uses a design pattern in the XML DTDs to require including either or both of two parts in an XML element. This XML DTD pattern avoids prohibited ambiguity when parsing the corresponding XML. The XML DTD design pattern used herein is:

```
<!ELEMENT ElementName (
        ...,
        (Identifier
        | ( (DefinitionPart1,
             DefinitionPart2,
             ...),
           Identifier?) ) ) >
```

6 XML data model (normative)

6.1 Introduction

This clause specifies the standard Data Model of CRS and CT definition data in eight parts, for representation of:

a) Coordinate Reference System definitions

b) Coordinate System definitions

c) Datum definitions

d) Ellipsoid and Prime Meridian definitions

e) Coordinate Transformation Definition

f) Parameterized Transformation definitions

g) Identification information

h) Unit definitions

i) Region of Validity definition

As previously stated, the specified Data Model includes multiple alternative top level XML elements that are intended to be used where needed. (That is, there is not a single top level element.) The alternative top level XML elements include at least:

a) CoordinateReferenceSystem

b) CoordinateTransformationMetadata

c) CoordinateTransformationDefinition

These XML DTDs are specified as external DTDs that are not included with actual XML data. Each DTD includes extensive comments to explain the actual XML data, using a format that we find useful. These DTDs are currently written such that they must all be included in one DTD file. These DTDs are written for XML version 1.0.

6.2 XML for coordinate reference system definition

This subclause presents the proposed XML DTD package for transfer of a Coordinate Reference System definition.

NOTE The CoordinateReferenceSystem element with all its contents is intended to allow including all the information now included in the XML DTD for CS_CoordinateSystem in the “low-level” CT interface specification. In that use, this CoordinateReferenceSystem is intended to replace CS_CoordinateSystem.)

```
<?xml version=“1.0” encoding=”UTF-8”?>
<!-- Version 0.0 of XML DTD for Coordinate Reference System
definition. This DTD uses XML elements that are specified in
other DTDs. -->

<!-- Coordinate Reference System: Definition of one Coordinate
Reference System (CRS), including the system identification and
possibly its complete specification. The Coordinate Reference
System defines a coordinate space having any number of
dimensions. The axes of the coordinate space can include from one
to three spatial axes, a temporal axis, and one or more other
axes.  However, descriptions of other than spatial/temporal
coordinate axes are not provided in this document.

The Coordinate Reference System data contains:

    Name Set: Set of one or more human understandable names for
    this Coordinate Reference System. This Name Set is optional,
    but is recommended to be included as discussed for the Name
    Set element.

    Identifier: Unique identifier used by software for this
    Coordinate Reference System. Either this Identifier or the
    following Coordinate Reference System Specification or both
    must be included.
```


Coordinate Reference System Specification: Detailed specification of this coordinate reference system. This specification data contains:

Validity Region: Description of the region of validity of this coordinate reference system. This data is optional, included whenever the region of validity is known and can be usefully represented.

Coordinate Reference System Type Specification: Specification of one of a set of alternative types of Coordinate Reference Systems.

The current coordinate reference system types are:

Compound CRS: Specification of a coordinate reference system that combines two or more simpler coordinate reference systems.

Geocentric CRS: Specification of a 3D Cartesian coordinate reference system with its origin at the (approximate) center of the Earth.

Geographic 3D CRS: Specification of a 3D coordinate reference system based on Latitude and Longitude position around an ellipsoid that approximates the shape of the Earth. The third dimension is height above or below the ellipsoid surface.

Geographic 2D CRS: Specification of a 2D coordinate reference system based on Latitude and Longitude position around an ellipsoid that approximates the shape of the Earth.

Projected CRS: Specification of a 2D Cartesian map coordinate reference system, derived from a Geographic 2D CRS by applying a “map projection” parameterized coordinate transformation.

Local CRS: Specification of a coordinate reference system that is defined for and usually used in a limited region. The origin of a Local CRS might or might not have a specified (or known) position in geodetic coordinates.

Image CRS: Specification of a coordinate reference system for an image. This document assumes an Image CRS to be two-dimensional, but 3D Image CRS’s are not excluded.

Vertical CRS: Specification of a 1D coordinate reference system used for elevation, depth, or height measurements.

Temporal CRS: Specification of a 1D coordinate reference system used for time measurements.

-->


```
<!ELEMENT CoordinateReferenceSystem (
        NameSet?,
        (Identifier
        | (ValidityRegion?,
          (  CompoundCRS
          |  GeocentricCRS
          |  Geographic3dCRS
          |  Geographic2dCRS
          |  ProjectedCRS
          |  LocalCRS
          |  ImageCRS
          |  VerticalCRS
          |  TemporalCRS  ),
          Identifier?) ) ) >

<!-- Compound CRS: Specification of a coordinate reference system
that combines two or more coordinate reference systems, none of
which can themselves be compound. In general, a compound
coordinate reference system may contain any number of axes.

For spatial coordinates, a number of constraints exist for the
construction of compound CRSs. For example, the coordinate
reference systems that are combined should not contain any
duplicate or redundant axes. Valid combinations include:
    Geographic 2D + Vertical
    Geographic 2D + Local 1D (near vertical)
    Projected + Vertical
    Projected + Local 1D (near vertical)
    Local (horizontal 2D or 1D linear) + Vertical

Any coordinate reference system or any of the above listed
combinations of coordinate reference systems can have a temporal
axis added. More than one temporal axis can be added if these
axes represent different time quantities. For example, the oil
industry sometimes uses “4D seismic”, by which is meant seismic
data with the vertical axis expressed in milliseconds (signal
travel time). A second time axis indicates how it changes with
time (years), e.g. as a reservoir is gradually exhausted of its
recoverable oil or gas).

The Compound Coordinate Reference System data contains:

    List of Coordinate Reference Systems: Ordered list of the two
    or more individual coordinate reference systems that are
    combined in this Compound Coordinate Reference System. The
    primary determinant for the order of the associated
    coordinates is the order of the constituting CRS’s, the
    secondary determinant is the order specified each CRS.
-->
<!ELEMENT CompoundCRS (
        CoordinateReferenceSystem+) >
```

```
<!-- Geocentric CRS: Specification of a coordinate reference
system with its origin at an approximation of the center of the
Earth.  This can only be associated with a Cartesian Coordinate
System, of which the axes and ordinates are commonly named:
Geocentric X, Geocentric Y and Geocentric Z. The X and Y axes lie
in the equatorial plane, with the positive X axis intersecting
the prime meridian. The positive Y-axis direction is obtained by
rotating the positive X-axis by 90 degrees counter-clockwise when
viewed from the northern hemisphere. The Z-axis is perpendicular
to the equatorial plane, with its positive half intersecting the
ellipsoid's North Pole and thus completing a right-handed 3D
Cartesian coordinate system. Every Coordinate Axis of the
associated Coordinate System must have the same Linear Unit.

The Geocentric CRS data contains:

    Geodetic Datum: The geodetic datum that specifies the origin,
    orientation, and scale of this Geocentric Coordinate
    Reference System.

    Coordinate System Definition: Definition of the set of
    coordinate axes with their metric used to record point
    coordinates in this Coordinate Reference System. For a
    Geocentric CRS only a Cartesian Coordinate System type is
    permitted.
-->
<!ELEMENT GeocentricCRS (
        GeodeticDatum,
        ( CartesianCoordinateSystem
        | SphericalCoordinateSystem) ) >

<!-- Geographic 3D CRS: Specification of a 3D coordinate
reference system that allows positions to be defined relative to
the surface of a reference ellipsoid.
    Note: In the associated OpenGIS Implementation Specification:
    Coordinate Transformation Services (OGC document 01-009) a
    Geographic 3D CRS is modeled as a Compound CRS, composed of a
    Geographic 2D CRS and a Vertical CRS, of which the latter is
    based on an Ellipsoidal Height Datum.  This solution is now
    actively discouraged; the current understanding is that
    ellipsoidal heights cannot exist on their own, but only as an
    inseparable part of 3D coordinate tuples.  This will be
    rectified in the next revision of the implementation
    specification.  See also the associated comment under
    Vertical Datum Type.

The Geographic 3D CRS data contains:

    Geodetic Datum: The geodetic datum that specifies the origin,
    orientation, and scale of this Geographic CRS.
```



```
    Coordinate System Definition: Definition of the set of
    coordinate axes with their metric used to record point
    coordinates in this Coordinate Reference System. For a
    Geographic 3D CRS only an Ellipsoidal Coordinate System type
    is permitted.
-->
<!ELEMENT Geographic3dCRS (
        GeodeticDatum,
        EllipsoidalCoordinateSystem) >

<!-- Geographic 2D CRS: Specification of a 2D coordinate
reference system that allows positions to be defined on the
surface of a reference ellipsoid.
    Note: Also see the related comment for Geographic 3D CRS
    above.

The Geographic 2D CRS data contains:

    Geodetic Datum: The geodetic (or horizontal) datum that
    specifies the origin, orientation, and scale of this
    Geographic 2D CRS.

    Coordinate System Definition: Definition of the set of
    coordinate axes with their metric, used to record point
    coordinates in this Coordinate Reference System. For a
    Geographic 2D CRS only an Ellipsoidal Coordinate System is
    permitted.
-->
<!ELEMENT Geographic2dCRS (
        GeodeticDatum,
        EllipsoidalCoordinateSystem) >

<!-- Projected CRS: Specification of a 2D map coordinate
reference system, derived from a Geographic 2D CRS by applying a
coordinate transformation, or rather: conversion to the
geographic coordinates of the source Geographic 2D CRS. This
conversion generates the map coordinates in the Projected CRS.
The Geodetic Datum of the Coordinate Reference system is not
affected by this coordinate transformation (conversion).  All
associated Coordinate Axes must have the same Linear Unit.

The Projected CRS data contains:

    Coordinate System Definition: Definition of the set of
    coordinate axes with their metric used to record point
    coordinates in this Coordinate Reference System. For a
    Projected CRS a 2D Cartesian Coordinate System is currently
    the only permitted coordinate system.

    Source Coordinate Reference System: The Geographic 2D CRS
    upon which this Projected CRS is based.
```



```
    Coordinate Transformation Definition: The Parameterized
    Transformation that defines this Projected CRS.  This
    coordinate transformation is a map projection.
-->
<!ELEMENT ProjectedCRS (
        CartesianCoordinateSystem,
        CoordinateReferenceSystem,
        CoordinateTransformationDefinition) >

<!-- Local CRS: Specification of a local coordinate reference
system that is defined for and usually used in a limited region,
significantly less than the complete surface of the earth.
Examples are local engineering or architectural coordinates, grids, and
drawings. A spatial Local Coordinate Reference System usually has
two characteristics:

    Does not account for the curvature of the earth’s surface. A
    Local Coordinate Reference System is not geocentric and is
    not a Projected Coordinate Reference System.

    The datum is within or on the edge of the region in which the
    Local Coordinate Reference System is expected to be used.
    (The origin is neither the center of the earth ellipsoid nor
    the point with zero Latitude and Longitude in any common
    Geographic Coordinate Reference System.)

The datum of a Local Coordinate Reference System might have a
specified (or known) position in a Geographic CRS, Geocentric
CRS, or Projected CRS. Similarly, the coordinate axes might have
specified directions in the same CRS.

    In general, if the origin position and/or axes directions are
    not specified in any geodetic CRS, point positions in a local
    CRS cannot be transformed to other CRSs. However, if two
    local CRSs have the same number of dimensions, axes names,
    axes units, and local datum, then software is permitted to
    assume that the two local CRSs are identical. This allows
    multiple datasets from a common source (e.g. a CAD system) to
    be overlaid. In addition, some implementations of a
    Coordinate Transformation software may have a mechanism for
    correlating Local Datums. (e.g. from a database of
    transformations, which is created and maintained from
    measurements.)

    One common use of a Local Coordinate Reference System is a
    grid system that defines the positions of a rectangular grid
    of points located in a source coordinate reference system.
    Such a grid CRS uses the grid origin for the origin of the
    axes and uses the grid spacing for the axis units. Such a
    grid of points is used in a grid coverage and for other
    purposes. Such a grid can be can be defined in advance of any
```

```
    data acquisition by an affine transformation from the source
    coordinate reference system.

The Local CRS data contains:

        Coordinate System Definition: Definition of the set of
        coordinate axes with their metric used to record point
        coordinates in this Coordinate Reference System. For a
        Local CRS all Coordinate Systems are permitted.

        Local Datum: The datum that specifies the reference used
        to measure positions in this local or engineering CRS.

        Coordinate reference system derivation: The definition of
        the coordinate transformation that defines this Local
        CRS. This information should be included only if the
        Local CRS is defined as originating from a source CRS
        through this transformation. This implies that only one
        such a coordinate transformation between the source CRS
        and the Local CRS can exist.

            Source Coordinate Reference System: The Coordinate
            Reference System from which this Local CRS has been
            derived. This source CRS can be of any type,
            including another Local Coordinate Reference System.
            However, care should be taken especially with the use
            of a Geographic CRS as source.

            Coordinate Transformation Definition: The
            Parameterized Transformation that specifies the
            transformation from the source coordinate reference
            system.
-->
<!ELEMENT LocalCRS (
        ( CartesianCoordinateSystem
        | ObliqueCartesianCoordinateSystem
        | EllipsoidalCoordinateSystem
        | VerticalCoordinateSystem
        | SphericalCoordinateSystem
        | PolarCoordinateSystem
        | CylindricalCoordinateSystem
        | LinearCoordinateSystem
        | UserDefinedCoordinateSystem),
        LocalDatum,
        ( CoordinateReferenceSystem,
          CoordinateTransformationDefinition)? ) >
```

<!-- Image CRS: Specification of the coordinate reference system for positions in an image. An Image CRS normally has two dimensions and can be seen as a special case of a Local CRS. All associated Coordinate Axes must have the same Unit and this should be a Pixel Spacing Unit or Linear Unit. There are thus three variations of an Image CRS:

Un-referenced Image CRS: An Image CRS with no known relationship to another CRS, inluding another Image CRS.

Georeferenced Image CRS: An Image CRS for a georeferenced image. Georeferenced means that the image position which corresponds to any ground position covered by that image can be computed using an image geometry model and implies that the camera or sensor position and pointing directions at the time of image collection are known, at least approximately. Most images being exploited are georeferenced.

For a two-dimensional image, the Image CRS is 2D. However, the ground coordinates are usually 3D, and are “projected” into 2D image coordinates using an image projection. The source CRS for a georeferenced image is thus a ground CRS, and the georeferenced Image CRS is defined by means of a coordinate transformation from a ground CRS.

The Coordinate Transformation that relates the 2D image coordinates to the 3D ground coordinates is normally a Parameterized Transformation. This Parameterized Transformation implements the image geometry model that relates the image coordinates to the corresponding ground coordinates. Different image geometry models will use different types of Parameterized Transformations.

Image Version CRS: An Image CRS for an image version produced by subsetting and/or resampling the pixels in an original image. A subset can be a selected patch or segment of the original image. Pixel resampling can be used to create a different image resolution (especially lower resolution) or to create a warped version of the original image (including rotated).

An Image Version CRS is defined by a coordinate transformation from another Image CRS, usually the original Image CRS. The relationship of Image Version CRS to the original Image CRS is usually defined by an affine coordinate transformation, which is a type of Parameterized Transformation. For a two-dimensional image, both the source and target Image CRS's are 2D.

```
The Image CRS data contains:

    Coordinate System Definition: Definition of the set of
    coordinate axes used to record point coordinates based on
    this Image CRS.

    Image Datum: The datum that specifies the reference used to
    measure positions in this original Image CRS.

    Coordinate reference system derivation: data that relates
    this Image CRS to a source CRS. This data should be included
    only if the Image CRS is defined as originating from a source
    CRS through this transformation. When the source CRS is
    another Image CRS, the result will be an “image version”
    Image CRS. The implication is that only one such a coordinate
    transformation between the source Image CRS and the resulting
    Image CRS can exist.

    This derivation data contains:

        Coordinate Reference System: The source CRS for this
        Image CRS. For a georeferenced image, defined through a
        transformation, the source ground CRS can be any CRS
        having three dimensions. For an image version, the source
        CRS can be any other 2D Image CRS that is not dependent
        of this Image CRS.

        Coordinate Transformation Definition: The definition of
        the coordinate transformation that defines this Image
        CRS.
-->
<!ELEMENT ImageCRS (
        ( CartesianCoordinateSystem
        | ObliqueCartesianCoordinateSystem ),
        ImageDatum,
        ( CoordinateReferenceSystem,
          CoordinateTransformationDefinition)? ) >

<!-- Vertical CRS: Specification of a 1D coordinate reference
system used for elevation, depth, or height measurements. The
Unit of the one associated Coordinate Axis must be a Linear Unit.

The Vertical CRS data contains:

    Coordinate System Definition: Definition of the coordinate
    axis used to record point coordinates based on this
    Coordinate Reference System. Only a Vertical Coordinate
    System is permitted.

    Vertical Datum: The datum that specifies the reference used
    to measure vertical heights or depths. This is often a named
    datum that is specified by a geodetic authority. However,
```



```
    “local vertical datums” also exist, but are likely to be
    specified as (part of) a Local Datum.
-->
<!ELEMENT VerticalCRS (
        VerticalCoordinateSystem,
        VerticalDatum) >

<!-- Temporal CRS: Specification of a 1D coordinate reference
system used for time measurements. A temporal CRS is based on
continuous time, using a time unit (or time interval) specified
in the one related Coordinate Axis. The one associated Coordinate
Axis must use a Time Unit.

The Temporal CRS data contains:

    Coordinate System Definition: Definition of the coordinate
    axis used to record data based on this Temporal CRS.  Only a
    Temporal Coordinate System is permitted.

    Origin: The DateTime origin used as a reference to measure
    time in this temporal coordinate reference system. This
    origin is specified in the Gregorian calendar with time of
    day in UTC. This DateTime is recorded as a character string
    in the format specified in ISO 19108 (which references ISO
    19103 and ISO 8601).
-->
<!ELEMENT TemporalCRS (
        TemporalCoordinateSystem,
        origin) >
<!ELEMENT origin   (#PCDATA) >

<!-- End of XML DTD for Coordinate Reference Systems -->
```

6.3 XML for coordinate system definitions

This subclause presents the proposed XML DTD package for transfer of a Coordinate System definition.

```
<!-- Version 0.0 of XML DTD for Coordinate System definition.
This DTD uses XML elements that are specified in other DTDs. -->

<!-- A Coordinate System is the set of coordinate axes, with the
implied metric of the coordinate space.  The Coordinate System is
defined as one of a series of alternative Coordinate System
Types.

Point coordinates in a Coordinate Reference System are recorded
in a Coordinate System. The metric of the coordinate space
determines how independent quantities (invariants) such as angles
and distances are calculated from coordinate values. Thus the
calculus required to derive angles and distances from point
coordinates is different in a map plane (Euclidean 2D space) and
```


on the surface of an ellipsoid (curved 2D space). The metric of the coordinate space cannot be specified in detail but should be derived from the properties of the space. Additional necessary information is comprised of the angles between the pair(s) of coordinate axes and the units of measure of the axes.

One Coordinate System may be used by multiple Coordinate Reference Systems. The coordinate axes can be spatial, temporal, or mixed. This definition includes the name, direction, unit, and sequence of each axis and the implied metric of the coordinate space. The number of axes should equal the number of dimensions in the Coordinate Reference System space and not contain duplicate or redundant axes.

Note that this Coordinate System Definition allows specification of an axis order, axis abbreviations and/or axis units other than normally used by the associated type of Coordinate System.

Currently the following types of Coordinate System are distinguished: -->

<!-- Cartesian Coordinate System: Set of orthogonal, straight coordinate axes sharing the same linear unit of measure. It can span a two- or three-dimensional spatial coordinate space and can be used with a Geocentric CRS (3d case), a Projected CRS (2D case), a Local CRS (2d or 3d case) or an Image CRS (2d or 3d case). Depending on its association with a coordinate reference system type the axes pairs/triplets may have specific names.
For a Geocentric CRS the recommended names for the permitted coordinate axes are: Geocentric X, Geocentric Y and Geocentric Z.
For a Projected CRS the recommended names for the permitted coordinate axes are: Easting, Northing, Southing or Westing.
For a Local CRS any name that provides an appropriate description of the axis is acceptable; however, it is strongly recommended to avoid names used for Geocentric CRS, Projected CRS (and Geographic CRS (2D and 3D)).

The Cartesian Coordinate System data contains:

> Name Set: Set of one or more human understandable names for this set of coordinate axes. This Name Set is optional, but is recommended to be included as discussed for the Name Set element.
>
> Identifier: Unique identifier used by software for this set of coordinate axes. Either this Identifier or the following Coordinate Axes Specification or both must be included.
>
> Dimensions: Positive integer number of coordinate axes (or dimensions) in this coordinate system.


```
    List of Coordinate Axes: Ordered list of one or more
    definitions of the axes of this coordinate system. These
    Coordinate Axis descriptions are ordered to match the order
    of the ordinates as used in describing a point location.
    Specific pairs or triplets of coordinate axes belong together
    and can only be used with specific coordinate reference
    systems.
-->
<!ELEMENT CartesianCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >
<!ELEMENT dimensions   (#PCDATA) >

<!-- Oblique Cartesian Coordinate System: Set of straight, but
not necessarily orthogonal coordinate axes.  All axes have the
same linear unit of measure. The Cartesian Coordinate System can
therefore be seen as a special case of the Oblique Cartesian
Coordinate System. An Oblique Cartesian Coordinate System can
span a one-, two- or three-dimensional spatial coordinate space
and can be used with a Local CRS (1d, 2d or 3d case) or an Image
CRS (2d or 3d case).

The Oblique Cartesian Coordinate System data contains:

    Name Set: Set of one or more human understandable names for
    this set of coordinate axes. This Name Set is optional, but
    is recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this set
    of coordinate axes. Either this Identifier or the following
    Coordinate Axes Specification or both must be included.

    Dimensions: Positive integer number of coordinate axes (or
    dimensions) in this coordinate system.

    List of Coordinate Axes: Ordered list of one or more
    definitions of the axes of this coordinate system. These
    Coordinate Axis descriptions are ordered to match the order
    of the ordinates as used in describing a point location.
-->
<!ELEMENT ObliqueCartesianCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >
```



```
<!-- Ellipsoidal Coordinate System: Coordinate system defined
relative to an reference ellipsoid that approximates the shape of
the earth. The position of a point on the ellipsoid is defined by
a) the angle between the line perpendicular to the ellipsoid's
   surface and the ellipsoid's equatorial plane (Ellipsoidal or
   Geodetic Latitude) and:
b) the angle between the meridian plane of the point and the
   prime meridian plane (Ellipsoidal or Geodetic Longitude).
The relevant coordinate axes, commonly indicated with the names
Geodetic Latitude and Geodetic Longitude respectively, must have
the same angular unit.
The 3D spatial position of a point near the surface of the
ellipsoid can be described by Geodetic Latitude and Longitude and
the height of the point above the ellipsoid, measured
perpendicular to the ellipsoid's surface. The Ellipsoidal
Coordinate System thus described can be used only with a
Geographic 3D CRS.  The height thus defined is termed Ellipsoidal
Height and can exist only as an element of the 3D spatial
position of point described in a Geographic 3D CRS and never on
its own (see the relevant note in Section 6.2 above under the
description of Geographic 3D CRS). A linear unit is associated
with the Ellipsoidal Height coordinate axis.

The Ellipsoidal Coordinate System data contains:

    Name Set: Set of one or more human understandable names for
    this set of coordinate axes. This Name Set is optional, but
    is recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this set
    of coordinate axes. Either this Identifier or the following
    Coordinate Axes Specification or both must be included.

    Dimensions: Positive integer number of coordinate axes (or
    dimensions) in this coordinate system.

    List of Coordinate Axes: Ordered list of one or more
    definitions of the axes of this coordinate system. These
    Coordinate Axis descriptions are ordered to match the order
    of the ordinates as used in describing a point location.
    Specific pairs or triplets of coordinate axes belong together
    and can only be used with specific coordinate reference
    systems.
-->
<!ELEMENT EllipsoidalCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >
```



```
<!-- Vertical Coordinate System: One dimensional coordinate
system used in association with a VerticalCRS. Heights are
measured along an axis that is related to the earth’s gravity
field and is therefore not a straight line in 3d space. The
details of the definition of height (and depth) is not trivial
and for that reason any attempt to be more precise in the
definition of the coordinate system (and its axis) is avoided
here. The associated coordinate axis has a linear unit.

The Coordinate System data contains:

    Name Set: Set of one or more human understandable names for
    this coordinate axis. This Name Set is optional, but is
    recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this
    coordinate axis. Either this Identifier or the following
    Coordinate Axis Specification or both must be included.

    Dimensions: Positive integer number of coordinate axes (or
    dimensions); in this case by definition: one.

    Coordinate Axis: Definition of the axis of this coordinate
    system.
-->
<!ELEMENT VerticalCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis,
          Identifier?) ) ) >

<!-- Spherical Coordinate System: Three dimensional coordinate
system in which a location is expressed by the following
coordinates:
a)  the distance of the point from the origin (radius) of the
    coordinate system
b)  the angle between the point vector and a defined reference
    plane (equatorial plane) – this ordinate is analogous to the
    concept of ‘latitude’.
c)  the angle between a base vector in the equatorial plane and
    the orthogonal projection of the point vector on the
    equatorial plane. The base vector defines the zero value of
    this ordinate, which is analogous to the concept ‘longitude’.
    The three coordinate axes have one associated linear and two
    associated angular units respectively.
The Spherical Coordinate System can be used in a geocentric and a
local context. It can be seen as the three-dimensional equivalent
of the Polar Coordinate System and is for that reason sometimes
referred to as a 3D polar coordinate system.
```

```
    Note: The Spherical Coordinate System should not be confused
    with the coordinate system often used for small scale mapping
    where the earth’s shape is approximated by a sphere rather
    than an ellipsoid. In spite of this spherical approximation
    the appropriate coordinate system for this case is the
    Ellipsoidal Coordinate System.  Its definition is still valid
    when the ellipsoid is allowed to ‘degenerate’ into a sphere.

The Spherical Coordinate System data contains:

    Name Set: Set of one or more human understandable names for
    this set of coordinate axes. This Name Set is optional, but
    is recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this set
    of coordinate axes. Either this Identifier or the following
    Coordinate Axes Specification or both must be included.

    Dimensions: Positive integer number of coordinate axes (or
    dimensions) in this coordinate system.

    List of Coordinate Axes: Ordered list of one or more
    definitions of the axes of this coordinate system. These
    Coordinate Axis descriptions are ordered to match the order
    of the axes values in point coordinates.
-->
<!ELEMENT SphericalCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >

<!-- Polar Coordinate System:  two-dimensional coordinate system
in which a location is expressed by the following coordinates:
    a)  the distance of the point from the origin (radius) of the
        coordinate system (linear unit)
    b)  the angle subtended by the point vector and a vector
        defining the zero direction (angular unit)
    The two coordinate axes have an associated linear and angular
    unit respectively.

The Polar Coordinate System data contains:

    Name Set: Set of one or more human understandable names for
    this set of coordinate axes. This Name Set is optional, but
    is recommended to be included as discussed for the Name Set
    element.
```

```
    Identifier: Unique identifier used by software for this set
    of coordinate axes. Either this Identifier or the following
    Coordinate Axes Specification or both must be included.

    Dimensions: Positive integer number of coordinate axes (or
    dimensions) in this coordinate system.

    List of Coordinate Axes: Ordered list of one or more
    definitions of the axes of this coordinate system. These
    Coordinate Axis descriptions are ordered to match the order
    of the axes values in point coordinates.
-->
<!ELEMENT PolarCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >

<!-- Cylindrical Coordinate System: a three-dimensional
coordinate system, which can be seen as composed of a 2d Polar
Coordinate System, extended by a coordinate axis, perpendicular
to the plane spanned by the Polar Coordinate System.  The third
ordinate is the distance along this axis, measured from the
origin of the system, with an associated linear unit.

The Cylindrical Coordinate System data contains:

    Name Set: Set of one or more human understandable names for
    this set of coordinate axes. This Name Set is optional, but
    is recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this set
    of coordinate axes. Either this Identifier or the following
    Coordinate Axes Specification or both must be included.

    Dimensions: Positive integer number of coordinate axes (or
    dimensions) in this coordinate system.

    List of Coordinate Axes: Ordered list of one or more
    definitions of the axes of this coordinate system. These
    Coordinate Axis descriptions are ordered to match the order
    of the axes values in point coordinates.
-->
<!ELEMENT CylindricalCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >
```

```
<!-- Linear Coordinate System: a one-dimensional coordinate
system used to quantify the locations of points along a curved or
straight coordinate axis, where the ordinate value equates to the
distance along the curve from a specified origin. The curved or
straight coordinate axis will usually coincide with a linear
feature such as a road, powerline or pipeline.  The coordinate
axis is usually located in the horizontal plane or topographic
surface. The associated unit is of type linear.

The Linear Coordinate System data contains:

    Name Set: Set of one or more human understandable names for
    this set of coordinate axes. This Name Set is optional, but
    is recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this set
    of coordinate axes. Either this Identifier or the following
    Coordinate Axes Specification or both must be included.

    Dimensions: Positive integer number of coordinate axes (or
    dimensions); in this case by definition one.

    Coordinate Axis: Definition of the axis of this coordinate
    system.
-->
<!ELEMENT LinearCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis,
          Identifier?) ) ) >

<!-- Temporal Coordinate System: a one-dimensional coordinate
system used to quantify time measurements.  This coordinate
system can only be associated with a Coordinate Axis of type
“time” and can only have a time unit.

The Temporal Coordinate System data contains:

    Name Set: Set of one or more human understandable names for
    this set of coordinate axes. This Name Set is optional, but
    is recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this set
    of coordinate axes. Either this Identifier or the following
    Coordinate Axes Specification or both must be included.

    Dimensions: Positive integer number of coordinate axes (or
    dimensions); in this case by definition one.
```

```
    Coordinate Axis: Definition of the axis of this coordinate
    system.
-->
<!ELEMENT TemporalCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis,
          Identifier?) ) ) >

<!-- User Defined Coordinate System: a set of coordinate axes
with their metric, not included in the list of types above and
left to the user's discretion to specify.

This Coordinate System data contains:

    Name Set: Set of one or more human understandable names for
    this set of coordinate axes. This Name Set is optional, but
    is recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this set
    of coordinate axes. Either this Identifier or the following
    Coordinate Axes Specification or both must be included.

    Dimensions: Positive integer number of coordinate axes (or
    dimensions) in this coordinate system.

    List of Coordinate Axes: Ordered list of one or more
    definitions of the axes of this coordinate system. These
    Coordinate Axis descriptions are ordered to match the order
    of the axes values in point coordinates.
-->
<!ELEMENT UserDefinedCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >
```

```
<!-- Coordinate Axis: Definition of one axis or ordinate of a
coordinate system. The Coordinate Axis data contains:

    Name Set: Set of one or more human understandable names for
    this Coordinate Axis or ordinate. A name should be a standard
    or widely used name whenever applicable and shall not specify
    or imply the units used; the units used shall be specified in
    the associated Unit element. This Name Set is optional, but
    is recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this
    Coordinate Axis. Either this Identifier or the following
    specification or both must be included.

    Axis Abbreviation: Abbreviated name of this axis or ordinate;
    this should be a human understandable text string. If unique
    abbreviations are used, this abbreviation can also be used by
    software, to match the same axis names. This abbreviation can
    be omitted if no useful axis name abbreviation is defined.

    Axis Direction: Text specifying the direction of increasing
    ordinate values along this axis. This text should be one of a
    standard set of direction names whenever applicable, but
    other values can be used when needed. The currently defined
    axis direction names are:

        North: Toward the North Pole, along the surface of the
        ellipsoid or the map plane, precisely or approximately.

        Northeast: In the quadrant between North and East, along
        the surface of the ellipsoid or the map plane.

        East: Toward the East, along the surface of the ellipsoid
        or the map plane, precisely or approximately.

        Southeast: In the quadrant between East and South, along
        the surface of the ellipsoid or the map plane.

        South: Toward the South Pole, along the surface of the
        ellipsoid or the map plane, precisely or approximately.

        Southwest: In the quadrant between South and West, along
        the surface of the ellipsoid or the map plane.

        West: Toward the West, along the surface of the ellipsoid
        or the map plane, precisely or approximately.

        Northwest: In the quadrant between West and North, along
        the surface of the ellipsoid or the map plane.
```

```
        Prime Meridian: Toward the Prime Meridian, to be used for
        the X-axis of a geocentric Cartesian Coordinate System.

        Future: Toward the future, to be used for a time
        coordinate.

        Past: Towards the past, to be used for a time coordinate.

        Up and Down: Directions to be used for a vertical
        coordinate. While the north, south, east, and west
        directions are intended to be close to the geodetic
        directions, the up and down directions is more loosely
        defined. An axis that begins obliquely downward can be
        said to be positive “down” (or positive “up”, depending
        on how the axis is defined). This is useful, for example,
        with wells. Depths of a well are often measured along the
        wellbore path. Since the wellbore path can become highly
        oblique – even horizontal, and at times, decreasing in
        true vertical depth – it is important that the concept of
        “down” not be restricted to increasing true vertical
        depth. In this case the concept “down” means that the
        measurement increases with increasing distance from the
        well reference point.
        If the direction of increasing increments makes an angle
        with the vector of gravity of less than 90 degrees in the
        reference point, the direction is defined as “down”.

        Out: Away from the origin point, to be used for the
        distance from the origin of a 2D polar, 3D spherical, or
        3D cylindrical coordinate system.

        Other: Any axis direction not described by one of the
        other standard Axis Direction names, and not described by
        using a non-standard Axis Direction name.

    Unit: Physical unit used for ordinate values of this axis.
    This unit can be a Linear, Angular, Time, or Pixel Spacing
    unit. Note that the Unit types must be consistent with the
    type of the associated coordinate reference system. For
    example, a Geographical 2D CRS uses only Angular Units, and a
    Geocentric CRS uses only Linear Units.
-->
<!ELEMENT CoordinateAxis (
        NameSet?,
        (Identifier
        | (axisAbbreviation?,
        axisDirection,
        (LinearUnit | AngularUnit | TimeUnit | PixelSpacingUnit),
        Identifier?) ) ) >
<!ELEMENT axisAbbreviation   (#PCDATA) >
<!ELEMENT axisDirection      (#PCDATA) >
```

```
<!-- End of XML DTD for Coordinate Systems -->
```

6.4 XML for datum definitions

This subclause presents the proposed XML DTD package for transfer of datum definitions.

```
<!-- Version 0.0 of XML DTD for Datum definitions. This DTD uses
XML elements that are specified in other DTDs. -->

<!-- Vertical Datum: The reference used to measure elevations,
depths, or heights in a vertical CRS.

The Vertical Datum data contains:

    Name Set: Set of one or more human understandable names for
    this Vertical Datum. This Name Set is optional, but is
    recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this
    Vertical Datum. Either this Identifier or the following
    vertical Datum Type must be included. Both can be included if
    useful, as discussed for the Identifier element.

    Datum Type: Text specifying the subtype of this vertical
    datum. The currently defined vertical datum subtypes and
    their meanings are:

        Geoidal: The zero point of the vertical axis is defined
        to approximately a constant potential surface, usually
        the geoid. Such a reference surface is usually determined
        by a national or scientific authority and is then a well
        known, named datum.

        Ellipsoidal: The ellipsoid surface defines the zero point
        of the datum. This datum type makes sense only if the
        associated Vertical CRS is combined with a 2D horizontal
        CRS. The ellipsoid is the same ellipsoid as for the
        horizontal CRS. For an ellipsoidal datum, the vertical
        axis is measured in a distance unit from the ellipsoid
        surface, and is usually named ‘height”.
            Note: This type of vertical datum is included in this
            document only for compatibility with the associated
            OpenGIS Implementation Specification: Coordinate
            Transformation Services (OGC document 01-009). In
            that Specification, a Geographic 3D CRS is modeled as
            a Compound CRS, composed of a Geographic 2D CRS and a
            Vertical CRS, of which the latter is based on an
            Ellipsoidal Height Datum.  This solution is now
            actively discouraged; the current understanding is
            that ellipsoidal heights cannot exist on their own,
```

```
            but only as an inseparable part of 3D coordinate
            tuples.  This will be rectified in the next revision
            of the Implementation Specification.  See also the
            definition of Geographic 3D CRS in Section 6.2 above.

        Depth: The zero point of the vertical axis is defined by
        a surface that has meaning for the purpose the associated
        vertical measurements are used for. For hydrographic
        charts this is often a predicted nominal sea surface
        (i.e. without waves or other wind and current effects)
        that occurs at low tide.  Examples are Lowest
        Astronomical Tide and Lowest Low Water Spring.  A
        different example is a sloping and undulating River Datum
        defined as the nominal river water surface occurring at a
        quantified river discharge.

        Barometric: The zero point of the vertical axis is
        defined by a surface of equal atmospheric pressure.
        Height is then measured by barometer.  This technique is
        routinely used in aircraft.  The zero level is an
        approximation of sea level, based on the atmospheric
        pressure on the ground at the time of take-off and the
        measured height value is commonly named “altitude.
-->
<!ELEMENT VerticalDatum (
        NameSet?,
        (Identifier
        | (datumType,
           Identifier?) ) ) >
<!ELEMENT datumType   (#PCDATA) >

<!-- Local Datum: The reference used to measure positions in a
local or engineering CRS. See the description of a Local CRS for
more information.

The Local Datum data contains:

    Name Set: Set of one or more human understandable names for
    this Local Datum. This Name Set is optional, but is
    recommended to be included as discussed for the Name Set
    element. Either this NameSet or the following Identifier or
    both must be included.
    The Remarks part of NameSet can be used for a description of
    the origin of the Local Datum.
        Note: The coordinate axes directions are defined by the
        associated Coordinate System element.

    Identifier: Unique identifier used by software for this Local
    Datum.
-->
 <!ELEMENT LocalDatum (
        NameSet
```

```
        | ( Identifier,
            NameSet?) ) >
```

<!-- Image Datum: The reference used to measure positions in an Image CRS. This concept is composed of two elements, the Image Origin and the Grid/Cell Association, both of which are required information in order to correctly transform the coordinates of image features from their Image CRS to a target CRS.

The image grid is defined as the set of lines of constant integer ordinate values. The term “image grid” is often used in other standards to describe the concept of Image CRS. However, care must be taken to correctly interpret this term in the context in which it is used. The term “grid cell” is often used as a substitute for the term “pixel”.

The grid lines of the image may be associated in two ways with the data attributes of the pixel or grid cell (ISO 19123). The data attributes of the image usually represent an average or integrated value that is associated with the entire pixel.

a) An image grid can be associated with this data in such a way that the grid lines run through the centers of the pixels. The cell centers will thus have integer coordinate values.
b) Alternatively the image grid may be defined such that the grid lines associate with the cell or pixel corners rather than the cell centers. The cell centers will thus have non-integer coordinate values, the fractional parts always being 0.5. ISO 19123 calls the grid points in this latter case “posts” and associated image data: “matrix data”.

This difference in perspective has no effect on the image interpretation, but is important for coordinate transformations involving this image.

The Image Datum data contains:

Name Set: Set of one or more human understandable names for this Image Datum. This Name Set is optional, but is recommended to be included as discussed for the Name Set element.

Identifier: Unique identifier used by software for this Image Datum. Either this Identifier or the following Image Origin and Pixel Reference must be included. Both can be included if useful, as discussed for the Identifier element.

Image Origin: Specification of the position of the origin (0,0) of the image coordinates. For a digital image, this origin position is within the pixel array. The currently defined image origins and their meanings are:

imageCenter: Center of this image. For a rectangular digital image, this is the center of the area covered by

```
        all the pixels of that image. If the image contains an
        even number of pixels, the image center will have integer
        coordinate values. However, if the image contains an odd
        number of pixels, the image center will have non-integer
        coordinate values (the fractional part being 0.5). The
        image center is rarely used as the grid origin for a
        digital image.

        imageCorner: The origin of the image grid. The image
        coordinates (or pixel indices) of this point are (0,0).
        This point may fall outside the data area of an image
        when it is associated with a null pixel (pixel with no
        data).

    Grid/Cell Association: Specification of the way the image
    grid is associated with the image data attributes. The
    current options and their meanings are:

        cellCenter: Grid lines run through the centers of the
        pixels or cells. The coordinates of the cell centers have
        integer values

        cellCorner: Grid lines run through the corners of cells,
        such that the grid point (post) that is associated with
        any pixel or cell is the corner of the cell in the
        direction of decreasing row and column indices. In the
        case the image origin is the corner of the image this is
        the cell corner closest to the grid origin.
-->
<!ELEMENT ImageDatum (
        NameSet?,
        (Identifier
        | (imageOrigin,
          gridCellAssociation,
          Identifier?) ) ) >
<!ELEMENT imageOrigin      (#PCDATA) >
<!ELEMENT gridCellAssociation (#PCDATA) >

<!-- Geodetic Datum: The specification of the origin,
    orientation, and scale of the associated Coordinate Reference
    System and thus the reference used to measure positions in a
    geocentric, geographic or projected CRS. Although a
    geocentric CRS does not explicitly use an ellipsoid in its
    definition, an ellipsoid specification as approximation of
    shape of the earth always underlies the definition of a
    geodetic datum. The geodetic datum of a projected CRS is
    ‘inherited’ from the geographic 2d CRS it is derived from and
    is therefore not specified in the relevant DTD.
        Note: The scale of the CRS is the deviation from the
        nominal unit of measure of the CRS.  This deviation is
        normally in the order of several part per million and is
```

```
        corrected for in coordinate transformations by the ‘scale
        difference’ parameter, specified for the transformation.

The Geodetic Datum data contains:

    Name Set: Set of one or more human understandable names for
    this Geodetic Datum. This Name Set is optional, but is
    recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this
    Geodetic Datum. Either this Identifier or the following
    Geodetic Datum Specification must be included. Both can be
    included if useful, as discussed for the Identifier element.

    Geodetic Datum Specification: Detailed specification of this
    geodetic datum. The Geodetic Datum Specification data
    contains:

        Ellipsoid: The approximation of the Earth's surface as a
        squashed sphere, used by this Geodetic Datum.

        Prime Meridian: Meridian toward which the X axis points.
        This data is optional, included only when the prime
        meridian is not the Greenwich meridian.
-->
<!ELEMENT GeodeticDatum (
        NameSet?,
        (Identifier
        | (Ellipsoid,
           PrimeMeridian?,
           Identifier?) ) ) >

<!-- End of XML DTD for Datum definitions -->
```

6.5 XML for ellipsoid and prime meridian definitions

This subclause presents the proposed XML DTD package for transfer of Ellipsoid and Prime Meridian definitions.

```
<!-- Version 0.0 of XML DTD for Ellipsoid and Prime Meridian
definitions. This DTD uses XML elements that are specified in
other DTDs. -->
```

```
<!-- Ellipsoid: Definition of an ellipsoid that approximates the
shape of the Earth.

The Ellipsoid data contains:

    Name Set: Set of one or more human understandable names for
    this Ellipsoid. This Name Set is optional, but is recommended
    to be included as discussed for the Name Set element.

    Identifier: Unique identifier used by software for this
    Ellipsoid. Either this Identifier or the following Ellipsoid
    Specification or both must be included.

    Ellipsoid Specification: Detailed specification of this
    ellipsoid. The Ellipsoid Specification data contains:

        Linear Unit: Definition of the linear unit used for the
        Semi-Major Axis and Semi-Minor Axis values of this
        ellipsoid.

        Semi-Major Axis: The numerical value of the equatorial
        radius of the ellipsoid, expressed in the related Linear
        Unit.

        Semi-Minor Axis: The numerical value of the distance from
        the center of the ellipsoid to either of its poles,
        expressed in the related Linear Unit.

        Inverse Flattening: The numerical value of the inverse
        flattening constant for this ellipsoid. The inverse
        flattening of a spheroid is related to the semi-major (a)
        and semi-minor (b) axes by the formula 1/f = a/(a-b). For
        perfect spheres, this formula breaks down, and the
        special inverse flattening value of zero is used.

        Flattening Definitive: The Boolean value that indicates
        if the Inverse Flattening value is definitive for this
        ellipsoid. Some ellipsoids use the inverse flattening as
        the defining value, and calculate the semi-minor axis
        when needed. Other ellipsoids use the semi-minor axis to
        calculate the Inverse Flattening value when needed. This
        distinction can be important to avoid floating point
        rounding errors.

        Ellipsoid Shape: The Boolean value that indicates if the
        reference surface is an ellipsoid. True means the
        reference surface is am ellipsoid, and False means it is
        a sphere. This data is optional, included when desired.
        When not included, the reference surface is a sphere when
        the values of the Semi-Major Axis and Semi-Minor Axis are
        equal
-->
```

```
<!ELEMENT Ellipsoid (
        NameSet?,
        (Identifier
        | (LinearUnit,
          semiMajorAxis,
          semiMinorAxis,
          inverseFlattening,
          Identifier?) ) ) >
    <!ATTLIST Ellipsoid
        flatteningDefinitive   (true | false)   #REQUIRED
        ellipsoidShape         (true | false)   #IMPLIED >
<!ELEMENT semiMajorAxis       (#PCDATA) >
<!ELEMENT semiMinorAxis       (#PCDATA) >
<!ELEMENT inverseFlattening   (#PCDATA) >

<!-- Prime Meridian: The meridian used as the reference for
Longitude measurements.

The Prime Meridian data contains:

    Name Set: Set of one or more human understandable names for
    this Prime Meridian. This Name Set is optional, but is
    recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this Prime
    Meridian. Either this Identifier or the following Prime
    Meridian Specification or both must be included.

    Prime Meridian Specification: Detailed specification of this
    prime meridian. The Prime Meridian Specification data
    contains:

        Greenwich Longitude: The Longitude of this Prime Meridian
        relative to the Greenwich Meridian, expressed in the
        related Angular Unit.

        Angular Unit: Definition of angular unit for the related
        Greenwich Longitude value.
-->
<!ELEMENT PrimeMeridian (
        NameSet?,
        (Identifier
        | (greenwichLongitude,
           AngularUnit,
           Identifier?) ) ) >
<!ELEMENT greenwichLongitude   (#PCDATA) >

<!-- End of XML DTD for Ellipsoid and Prime Meridian -->
```

6.6 XML for coordinate transformation definition

This subclause presents the proposed XML DTD package for transfer of a Coordinate Transformation definition.

NOTE The CoordinateTransformationDefinition is intended to allow including all the information now included in the XML DTD for CT_MathTransform in the "low-level" Coordinate Transformation Services interface specification. In that use, this CoordinateTransformationDefinition is intended to replace CT_MathTransform.

```
<!-- Version 0.0 of XML DTD for Coordinate Transformations. This
DTD uses XML elements that are specified in other DTDs. -->

<!-- Expected usage: Either the Coordinate Transformation
Metadata or Coordinate Transformation Definition can be directly
used under different circumstances:

    The Coordinate Transformation Metadata element should be used
    when definitions of both the source and target coordinate
    reference systems are included.  In that case, the Coordinate
    Transformation Definition component element is optional. This
    element is used in the current draft "high level" (ground)
    Coordinate Transformation interface specification.

    The Coordinate Transformation Definition element should be
    directly used when the source and target coordinate reference
    systems are NOT included. This element could thus be used in
    the current "low-level" Coordinate Transformation Services
    interface specification.
-->

<!-- Coordinate Transformation Metadata: Metadata for one
specific coordinate transformation, including definitions of the
source coordinate reference system, target coordinate reference
system, and coordinate transformation parameters and/or
transformation choice. All of this metadata can be recorded in
this XML format. The source and target coordinate reference
systems can each be for any number of axes or ordinates included
in the coordinates of a point, from one to four or more axes.

The Coordinate Transformation Metadata data contains:

    Source Coordinate Reference System: The source coordinate
    reference system for this transformation, including the
    coordinate reference system identification and/or its
    complete specification. This Source Coordinate Reference
    System is used by the input point coordinates to a coordinate
    transformation.

    Target Coordinate Reference System: The target coordinate
    reference system for this transformation, including the
    coordinate reference system identification and/or its
    complete specification. This Target Coordinate Reference
```



```
    System is used by the output point coordinates from a
    coordinate transformation.

    Coordinate Transformation Definition: One coordinate
    transformation, including specification of transformation
    parameters and/or selection from transformation alternatives.
    This definition includes the transformation identification
    and/or its specification, but not the definitions of the
    source and target coordinate reference systems. This
    Transformation Definition is optional, included only when
    needed or useful. For example, for input to a coordinate
    transformation server, this Coordinate Transformation
    Definition can be omitted when the server software does not
    require this information.
-->
<!ELEMENT CoordinateTransformationMetadata (
        CoordinateReferenceSystem,
        CoordinateReferenceSystem,
        CoordinateTransformationDefinition?) >

<!—- Coordinate Transformation Definition: Definition of one
coordinate transformation, including the coordinate
transformation identification and/or its specification, but not
including the definitions of the source and target CRSs. This
transformation can be for any number of axes or ordinates
included in the coordinates of a point, from one to four or more
axes. The related source and target CRSs can be spatial,
temporal, or mixed.

The Coordinate Transformation Definition data contains:

    Name Set: Set of one or more human understandable names for
    this coordinate transformation. This Name Set is optional,
    but is recommended to be included as discussed for the Name
    Set element.

    Identifier: Unique identifier used by software for this
    coordinate transformation. Either this Identifier or the
    following Coordinate Transformation Specification must be
    included. Both can be included if useful, as discussed for
    the Identifier element.

    Coordinate Transformation Specification: Detailed
    specification of this specific coordinate transformation,
    between the related source and target coordinate reference
    systems. This Coordinate Transformation Specification data
    includes:

        Source Dimensions: Positive integer number of dimensions
        in a source coordinate reference system to which this
        transformation can be applied.
```

```
        Target Dimensions: Positive integer number of dimensions
        in a target coordinate reference system that this
        transformation can produce.

        Validity Region: Description of the region of validity of
        this coordinate transformation, in the source coordinate
        reference system. This data is optional, included
        whenever the region of validity is known and can be
        usefully represented.

        Position Error Estimates: Definition of error estimates
        for the position errors introduced by this coordinate
        transformation, in the target coordinate reference
        system. This data is optional and can be included when
        estimates are available for the position errors
        introduced by this coordinate transformation.

        Transformation Type Specification: Specification of one
        of a set of alternative types of coordinate
        transformation.

The current coordinate transformation types are:

    Concatenated Transformation: Specification of a coordinate
    transformation that sequentially applies two or more other
    transformations.

    Inverse Transformation: Specification of a coordinate
    transformation that is the inverse of another transformation.

    Pass Through Transformation: Specification of a coordinate
    transformation that transforms only a subset of the
    ordinates, and passes the remaining ordinates unchanged. The
    modified ordinates are transformed by a referenced Coordinate
    Transformation.

    Parameterized Transformation: Specification of a coordinate
    transformation that is defined by a specified transformation
    type and set of parameters.
-->
```

```
<!ELEMENT CoordinateTransformationDefinition (
        NameSet?,
        (Identifier
        | (sourceDimensions,
           targetDimensions,
           ValidityRegion?,
           PositionErrorEstimates?,
           ( ConcatenatedTransformation
           | InverseTransformation
           | PassThroughTransformation
           | ParameterizedTransformation),
           Identifier?) ) ) >
<!ELEMENT sourceDimensions   (#PCDATA) >
<!ELEMENT targetDimensions   (#PCDATA) >

<!-- Position Error Estimates: Definition of error estimates for
the position errors introduced by this coordinate transformation,
in the target coordinate reference system. These error estimates
are recorded in the form of a covariance matrix, sometimes called
a variance-covariance matrix. The matrix elements are the
expected average values of the product of the error in the row
coordinate times the simultaneous error in the column coordinate.
In a complete covariance matrix, the diagonal elements are the
error variances of the corresponding coordinates, or the squares
of the standard deviations. The off-diagonal elements are the
covariances between the errors in the corresponding coordinates;
these covariances will be zero when the errors in different
coordinates are not statistically correlated.

    A complete covariance matrix is always square and
    symmetrical, meaning that the same matrix element values
    appear on both sides of the diagonal elements. Only one of
    each pair of symmetrical matrix elements needs to be
    recorded. Other matrix elements may also be omitted, meaning
    the value of that element is unknown.

    For the three coordinates of one point, a covariance matrix
    is a 3 by 3 matrix, with the matrix rows and columns each
    corresponding to the three coordinates. For just the two
    horizontal coordinates, a covariance matrix is a 2 by 2
    matrix, with the matrix rows and columns each corresponding
    to the two horizontal coordinates. Similarly, for two image
    coordinates, a covariance matrix is a 2 by 2 matrix, with the
    matrix rows and columns each corresponding to the two image
    coordinates.

The Position Error Estimates data contains:

    Matrix Size: Positive integer specifying the number of rows
    and columns in the square covariance matrix.
```

```
    List of Units: Ordered list of the physical units used for
    each position ordinate in this covariance matrix. Each
    Covariance Element uses the product of the units associated
    with the row and column ordinates of the covariance matrix.
    Each unit can be either a Linear, Pixel Spacing, or Time
    unit. For digital image coordinates, the units are usually
    Pixel Spacings. For ground coordinates, the units are always
    Linear units and are almost always metres. Notice that these
    metre units can be different than the units of the
    corresponding ground coordinates, which may be Angular units.

    List of Covariance Elements: Unordered list of all the known
    and unique elements in this covariance matrix. Since a
    complete covariance matrix is symmetrical, only one of each
    pair of symmetrical matrix elements needs to be recorded. Any
    other omitted elements shall be interpreted as having unknown
    values.
-->
<!ELEMENT PositionErrorEstimates (
        matrixSize,
        (LinearUnit | PixelSpacingUnit | TimeUnit)+,
        CovarianceElement+) >
<!ELEMENT matrixSize   (#PCDATA) >

<!-- Covariance Element: Specifies the value of an element of a
covariance matrix. If this element is off the matrix diagonal,
specifies the values of both symmetrical matrix elements. The
Covariance Element data contains:

    Row: Positive integer value of the row position of this
    element in the covariance matrix. The value can range from 1
    up to the value of the “matrixSize” of the containing
    Covariance Matrix.

    Column: Positive integer value of the column position of this
    element in the covariance matrix. The value can range from 1
    up to the value of the “matrixSize” of the containing
    Covariance Matrix.

    Covariance: Floating point number that specifies the value of
    this element of a covariance matrix. A zero value means the
    expected error is zero or negligible. The units are the
    product of the units associated with the row and column
    ordinates of this element in the covariance matrix.  Those
    units are specified in the Position Error Estimates XML
    element.
-->
```

```
<!ELEMENT CovarianceElement (
        row,
        column,
        covariance) >
<!ELEMENT row          (#PCDATA) >
<!ELEMENT column       (#PCDATA) >
<!ELEMENT covariance   (#PCDATA) >

<!-- Concatenated Transformation: Specification of a coordinate
transformation that sequentially applies two or more other
coordinate transformations. The target coordinate reference
system from the first coordinate transformation must be the
source coordinate reference system to the second coordinate
transformation, and so forth. This restriction holds whether or
not those target and source coordinate reference systems are
explicitly defined. The number of dimensions of the output
coordinate reference system from the first transformation must
match the number of dimensions of the input coordinate reference
system to the second transformation, and so forth.

The Concatenated Transformation data contains:

    List of Coordinate Transformation Definitions: Ordered list
    of the two or more individual coordinate transformations that
    are combined in this Concatenated Transformation. The order
    of listing is the order of transformation application, to go
    from the associated source to the associated target
    coordinate reference systems.
-->
<!ELEMENT ConcatenatedTransformation (
        CoordinateTransformationDefinition+) >

<!-- Inverse Transformation: Specification of a coordinate
transformation that is the inverse of another coordinate
transformation. That is, this inverse transformation transforms
coordinates in the opposite direction between the same two
coordinate reference systems. (Note: The Inverse Transform will
not exist if the referenced coordinate transformation is not one
to one. All map projections should succeed.)

The Inverse Transformation data contains:

    Coordinate Transformation Definition: Definition of the
    coordinate transformation that is inverted (or reversed) by
    this Inverse Transformation.
-->
<!ELEMENT InverseTransformation (
        CoordinateTransformationDefinition) >
```

```
<!-- Pass Through Transformation: Specification of a coordinate
transformation that transforms only a subset of the ordinates,
and passes the remaining ordinates unchanged. The ordinates that
are modified are transformed by a referenced Coordinate
Transformation. One major use of a Pass Through Transformation is
transforming one component of a compound coordinate reference
system.

    Example: Given (Latitude, Longitude, Height) coordinates in
    either a Geographic 3D CRS or a Compound CRS, this
    transformation allows converting the Height values from
    metres to feet, without affecting the Latitude and Longitude
    values. Alternately, this transformation allows converting
    the Latitude and Longitude values from degrees to radians,
    without affecting the Height value. Of course, either of
    these physical Unit conversions could alternately be done
    using a degenerate affine Parameterized Transformation. Note
    that in such a unit conversion also two coordinate reference
    systems are involved: the source CRS (with the original unit)
    and the target CRS (with the target unit).  Both of these
    CRSs shall be referenced in the Coordinate Transformation
    Metadata, whenever that XML element is used.

The Pass Through Transformation data contains:

    Coordinate Transformation Definition: Definition of the
    source coordinate transformation that is used by this Pass
    Through Transformation to transform the coordinate axes that
    are modified.

    First Modified Ordinate: Positive integer index to the first
    ordinate to be transformed by the identified coordinate
    transformation.

    Number Modified Ordinates: Positive integer number of
    dimensions to be transformed by the identified coordinate
    transformation.
-->
<!ELEMENT PassThroughTransformation (
        CoordinateTransformationDefinition,
        firstModifiedOrdinate,
        numberModifiedOrdinates) >
<!ELEMENT firstModifiedOrdinate     (#PCDATA) >
<!ELEMENT numberModifiedOrdinates   (#PCDATA) >

<!-- End of XML DTD for Coordinate Transformation Definition -->
```

6.7 XML for parameterized transformation definition

This subclause presents the proposed XML DTD package for transfer of a Parameterized Transformation definition.


```
<!-- Version 0.0 of XML DTD for Parameterized Transformation
definition. This DTD uses XML elements that are specified in
other DTDs. -->

<!-- Parameterized Transformation: Specification of a coordinate
transformation that is defined by a specified transformation
method plus the set of parameter values needed by that
transformation method. Each parameter is recorded with its name,
value, and physical unit when applicable.
    Notes: The number of dimensions of the target CRS is always
    equal to or less than the number of dimensions of the source
    CRS. The specified coordinate transformation should be one-
    to-one within the domain of the source CRS.

The Parameterized Transformation data contains:

    Transformation Method: Identification or specification of the
    method used by this Parameterized Transformation.

    List of Parameters: Unordered list of the zero or more
    parameters that define the specific coordinate transformation
    of the associated transformation method.
-->
<!ELEMENT ParameterizedTransformation (
        TransformationMethod,
        Parameter*) >

<!-- Transformation Method: Definition of the method used by a
Parameterized Transformation, including the set of parameters
used by that method. This data structure always identifies the
transformation method, and optionally includes a more complete
description and specification of this method.

The Transformation Method data contains:

    Name Set: Set of one or more human understandable names for
    this Transformation Method. This Name Set is optional, but is
    recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this
    Transformation Method. Either this Identifier or the
    following Transformation Method Specification or both must be
    included.

    Transformation Method Specification: Specification of this
    parameterized coordinate transformation method. This
    Transformation Method Specification data includes:

        Description: Human understandable text describing this
        transformation method.
```



```
        Reference: Character string referring to a source of more
        information about this transformation method. This data
        is optional, included only when considered useful. This
        element can be repeated when useful, if multiple
        references exist for one transformation method.

        Formula: Human understandable text giving formulas that
        can be used to implement this transformation method. This
        data is optional, included only when considered useful.

        Parameters: Positive integer number of the parameters
        defined for this transformation method. This data is
        optional, included only when considered useful. This data
        can be repeated when useful.

        List of Parameter Definitions: Unordered list of the
        definitions of the zero or more parameters defined for
        the identified transformation type. This data is
        optional, included only when considered useful.

        Transformation Type: Text identifying whether this
        transformation is actually a transformation or a
        conversion, as those two terms are defined by ISO 19111.
        This data is optional, included only when considered
        useful, because it can also be deduced from the context
        and structure provided in this DTD: when a datum change
        is involved, it is a transformation, when not, it is a
        conversion.
-->
<!ELEMENT TransformationMethod (
        NameSet?,
        (Identifier
        | (description,
           reference*,
           formula?,
           parameters?,
           Identifier?) ) ) >
    <!ATTLIST TransformationMethod
        transformationType   (transformation | conversion)
                         #IMPLIED >
<!ELEMENT description   (#PCDATA) >
<!ELEMENT reference     (#PCDATA) >
<!ELEMENT formula       (#PCDATA) >
<!ELEMENT parameters    (#PCDATA) >

<!-- Parameter: Data structure containing the name and value of
one coordinate transformation or map projection parameter. The
Parameter data contains:

    Name Set: Set of one or more human understandable names for
    this coordinate transformation parameter. This Name Set is
    optional, but usually should be included as discussed for the
```



```
    Name Set element.

    Identifier: Unique identifier used by software for this
    coordinate transformation parameter. Either this Identifier
    or the following Parameter Specification or both must be
    included.

    Parameter Specification: Detailed specification of this
    specific coordinate transformation parameter. This Parameter
    Specification data includes:

         Value: Value of this transformation parameter.

         Unit: Physical unit used for this parameter value.  This
         unit can be a Linear, Angular, Time, or Pixel Spacing
         unit. This data is mandatory when applicable to the
         parameter values.
               Note: In the case of polynomial coefficients (in
               polynomial transformations) the unit does not need
               to be supplied, although, strictly speaking, each
               polynomial coefficient does have a unit. Generally
               such transformations are provided together with
               the algorithm, which practically eliminates the
               probability of “unit errors”.
-->
<!ELEMENT Parameter (
        NameSet?,
        (Identifier
        | ( ( value,
            ( LinearUnit | AngularUnit | TimeUnit
            | PixelSpacingUnit)? ),
        Identifier?) ) ) >
<!ELEMENT value      (#PCDATA) >

<!-- End of XML DTD for Parameterized Transformation -->
```

6.8 XML for identification information

This subclause presents the proposed XML DTD package for transfer of Identification data.

```
<!-- Version 0.0 of XML DTD for Identification information. -->

<!-- Identifier: Unique identifier used by software for a
specific coordinate reference system, coordinate transformation,
or other set of data. This identifier does not contain
specification information, but contains the minimum information
needed to allow finding specification data stored elsewhere.

The Identifier data contains:

    Code: Character string code, name, or other unique identifier
    of the related set of data. The meaning of this Code is
```

defined by the related Code Space, and the range and format of these codes are defined by the Code Space authority. This code is usually not human understandable, but it could be.

Code Space: Character string identifier of a code space within which one or more Codes are defined. This code space is often defined by some authority organization, where one organization may define multiple code spaces. The range and format of each Code Space identifier is defined by that code space authority. When the Code Space identifier is long, an all-capitals abbreviation of that identifier may be used (such as "EPSG" for European Petroleum Survey Group).

Edition: Character string identifier of one edition or version of the related Code Space or "Code". This edition identifier applies to either the Code Space or the "Code" as defined by the Code Space authority. This data is optional, included whenever multiple editions exist or are expected to exist. When appropriate, the edition is identified by the effective date, coded using ISO 19108 date format.

Note 1: One authority (code space and edition) can define more than one set of unique codes for identifying one item from a set of alternatives. For example, the EPSG currently defines three different sets of unique codes for identifying a Length Unit and for an Angular Unit. When an authority defines more than one set of unique codes, any of these sets of codes could be used.

Note 2: Whenever an Identifier element can be included in another element, certain “specification” data can alternately be included in the same other element. Either the Identifier element or the specification data element(s) must be included, allowing a “definition” XML element to be used in two ways:

1. Only the Identifier can be included for a geospatial entity that is “well-known” to the XML receiver. Well-known means that the corresponding "specification" data can be found elsewhere when needed, using only this Identifier. This “elsewhere” is never in the same XML file that contains this Identifier-only definition element, is usually not in any public XML file, and is never in a XML file whose name or URI is known to the XML sender.

2. Only the "specification" data can be included for a custom geospatial entity that is not uniquely or permanently identified.

Both the "identifier" data and "specification" data can be included in one XML element if useful, but this should generally not be done. Specifically, the "specification" data should not be included with an Identifier for any geospatial entity that is “well-known” to the XML receiver, since these two forms of

```
information are redundant and may conflict. If both are included,
the included "specification" data should be used by the XML
receiver, instead of the included Identifier.
-->
<!ELEMENT Identifier (
        code,
        codeSpace,
        edition? ) >
<!ELEMENT code          (#PCDATA) >
<!ELEMENT codeSpace     (#PCDATA) >
<!ELEMENT edition       (#PCDATA) >

<!-- Name Set: Set of human understandable names for a specific
coordinate reference system, coordinate transformation, or
related set of data. These names usually do not uniquely identify
the set of data (or metadata), but provide human understandable
names for that data. The Name Set data contains at least one of:

    Name: The primary human understandable name for this set of
    data. This data is optional, included only when a human
    readable name is defined and useful.

    Abbreviation: An abbreviated human understandable name for
    this set of data. This data is optional, included only when
    an abbreviation is defined and useful. More than one
    abbreviation can be included if useful.

    Alias: An alternate human understandable name for this set of
    data. This data is optional, included only when an alias is
    defined and useful. More than one alias can be included if
    useful.

    Remarks: Human understandable remarks or comments about this
    coordinate reference system, coordinate transformation, or
    related set of data. More than one remark can be included if
    useful.

Note: Whenever a Name Set element can be included in another
element, the Identifier element and/or “specification” data can
also be included in the same other element. Whatever other data
is included in an element, the Name Set element should be
included if useful, and it will often be useful to a human user.
However, the Name Set element may not be useful when the included
Identifier element or "specification" data is sufficiently human
understandable.
-->
```

```
<!ELEMENT NameSet (
        name?,
        abbreviation*,
        alias*,
        remarks*) >
<!ELEMENT name           (#PCDATA) >
<!ELEMENT abbreviation   (#PCDATA) >
<!ELEMENT alias          (#PCDATA) >
<!ELEMENT remarks        (#PCDATA) >

<!-- End of XML DTD for Identification information -->
```

6.9 XML for unit definitions

This subclause presents the proposed XML DTD package for transfer of Unit definitions.

```
<!-- Version 0.0 of XML DTD for Unit definitions. -->

<!-- Linear Unit: Definition of a linear unit of measure.

The Linear Unit data contains:

    Name Set: Set of one or more human understandable names for
    this Linear Unit. This Name Set is optional, but is
    recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this
    Linear Unit. Either this Identifier or the following Linear
    Unit Specification must be included. Both can be included if
    useful, as discussed for the Identifier element.

    Linear Unit Specification: Specification of this linear unit
    relative to the international standard metre. The Linear Unit
    Specification data contains either or both:

        Metres Per Unit: Number of metres per linear unit.

        Units Per Metre: Number of linear units per metre.
-->
<!ELEMENT LinearUnit (
        NameSet?,
        (Identifier
        | ( ( (metresPerUnit, unitsPerMetre?)
            |  unitsPerMetre),
            Identifier?) ) ) >
<!ELEMENT metresPerUnit   (#PCDATA) >
<!ELEMENT unitsPerMetre   (#PCDATA) >
```

```
<!-- Angular Unit: Definition of an angular unit of measure.

The Angular Unit data contains:

    Name Set: Set of one or more human understandable names for
    this Angular Unit. This Name Set is optional, but is
    recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this
    Angular Unit. Either this Identifier or the following Angular
    Unit Specification must be included. Both can be included if
    useful, as discussed for the Identifier element.

    Angular Unit Specification: Specification of this angular
    unit relative to radians. The Angular Unit Specification data
    contains either or both:

        Radians Per Unit: Number of radians per angular unit.

        Units Per Radian: Number of angular units per radian.
-->
<!ELEMENT AngularUnit (
        NameSet?,
        (Identifier
        | ( ( (radiansPerUnit, unitsPerRadian?)
            |  unitsPerRadian),
            Identifier?) ) ) >
<!ELEMENT radiansPerUnit   (#PCDATA) >
<!ELEMENT unitsPerRadian   (#PCDATA) >

<!-- Time Unit: Definition of a time unit of measure.

The Time Unit data contains:

    Name Set: Set of one or more human understandable names for
    this Time Unit. This Name Set is optional, but is recommended
    to be included as discussed for the Name Set element.

    Identifier: Unique identifier used by software for this Time
    Unit. Either this Identifier or the following Time Unit
    Specification must be included. Both can be included if
    useful, as discussed for the Identifier element.

    Time Unit Specification: Specification of this time unit
    relative to international standard seconds. The Time Unit
    Specification data contains either or both:

        Seconds Per Unit: Number of seconds per time unit.

        Units Per Second: Number of time units per seconds.
-->
```

```
<!ELEMENT TimeUnit (
        NameSet?,
        (Identifier
        | ( ( (secondsPerUnit, unitsPerSecond?)
            |  unitsPerSecond),
            Identifier?) ) ) >
<!ELEMENT secondsPerUnit   (#PCDATA) >
<!ELEMENT unitsPerSecond   (#PCDATA) >

<!-- Pixel Spacing Unit: Definition of a pixel spacing based unit
of measure, used for positions in the image coordinate reference
system of a digital image.

The Pixel Spacing Unit data contains:

    Name Set: Set of one or more human understandable names for
    this Pixel Spacing Unit. This Name Set is optional, but is
    recommended to be included as discussed for the Name Set
    element.

    Identifier: Unique identifier used by software for this Pixel
    Spacing Unit. Either this Identifier or the following Pixel
    Spacing Unit Specification must be included. Both can be
    included if useful, as discussed for the Identifier element.

    Pixel Spacing Unit Specification: Specification of this pixel
    spacing unit relative to one pixel spacing. The Pixel Spacing
    Unit Specification data contains either or both:

        Pixel Spacings Per Unit: Number of pixel spacings per
        unit. This number is usually 1.0.

        Units Per Pixel Spacing: Number of units per pixel
        spacing. This number is usually omitted.
-->
<!ELEMENT PixelSpacingUnit (
        NameSet?,
        (Identifier
        | ( ( (pixelSpacingsPerUnit, unitsPerPixelSpacing?)
            |  unitsPerPixelSpacing),
            Identifier?) ) ) >
<!ELEMENT pixelSpacingsPerUnit   (#PCDATA) >
<!ELEMENT unitsPerPixelSpacing   (#PCDATA) >

<!-- End of XML DTD for Unit definitions -->
```

6.10 XML for validity region definition

This subclause presents the proposed XML DTD package for transfer of a Validity Region definition.

```
<!-- Version 0.0 of XML DTD for Validity Region definition. -->

<!-- Validity Region: Description of the region of validity of a
coordinate reference system, coordinate transformation, or
related set of data. For a coordinate transformation, this region
of validity applies to the source coordinate reference system.
The Validity Region data contains either or both:

    Validity Area: Human-understandable text defining the region
    of validity.

    Validity Envelope: Specification of the region of validity as
    a rectangle in two dimensions, or a rectangular volume in
    three dimensions. The Validity Envelope contains:

        Minimum Coordinates: Ordered list of numbers, one for
        each coordinate system axis, that specifies the minimum
        coordinates of this region of validity. Each number is
        normally the algebraically minimum axis value over the
        region of validity. A minimum Longitude is therefore the
        West-most value over the region of validity.

        Maximum Coordinates: Ordered list of numbers, one for
        each coordinate system axis, that specifies the maximum
        coordinates of this region of validity. Each number is
        normally the algebraically maximum axis value over the
        region of validity. A maximum Longitude is therefore the
        East-most value over the region of validity.

        Notes: The numbers in the Minimum and Maximum Coordinates
        lists are separated by spaces, and can be either real
        (floating point) or integer numbers. A point with these
        Minimum Coordinates or Maximum Coordinates may be outside
        the valid domain of the related coordinate reference
        system or coordinate transformation.

        Validity Type: Character string code indicating the
        precision or meaning of this envelope of validity. For a
        CRS, the possible Validity Type values and meanings are:

            Exact: Any point within this envelope is valid, and
            no point outside this envelope is valid.

            Typical: Most points within this envelope are valid,
            and most points outside this envelope are not valid.

            Minimum: Any point within this envelope is valid, and
            some points outside this envelope are also valid.

            Maximum: Most points inside this envelope are valid,
            and no point outside this envelope is valid.
```

```
            Unlimited: The region of validity is limited only by
            the valid ranges of all the individual axes values.
            In this case, the minimum and maximum envelope values
            contain the axes minimum and maximum values. Any
            point with all axes values within that axis value
            range is valid.

        For a coordinate transformation, the possible values and
        meanings of the Validity Type are:

            Exact: Any point within this envelope can be
            accurately transformed from the source coordinate
            reference system and no point outside this envelope
            can be accurately transformed.

            Typical: Most points within this envelope can be
            accurately transformed from the source coordinate
            reference system and most points outside this
            envelope can not be accurately transformed.

            Minimum: Any point within this envelope can be
            accurately transformed from the source coordinate
            reference system and some points outside this
            envelope will also be accurately transformed.

            Maximum: Most points within this envelope can be
            accurately transformed from the source coordinate
            reference system, but no points outside this envelope
            will be accurately transformed.

            Unlimited: The envelope of validity is limited only
            by the valid ranges of all the individual axes values
            for the source coordinates. Any point with all axes
            values within that axis value range can be accurately
            transformed from the source coordinate reference
            system.
-->
<!ELEMENT ValidityRegion (
        validityArea
        | (minimumCoordinates,
           maximumCoordinates,
           validityArea?) ) >
    <!ATTLIST ValidityRegion
        validityType   (exact | typical | minimum | maximum
            | unlimited)          #IMPLIED >
<!ELEMENT   validityArea         (#PCDATA) >
<!ELEMENT   minimumCoordinates   (#PCDATA) >
<!ELEMENT   maximumCoordinates   (#PCDATA) >

<!-- End of XML DTD for Validity Region definition -->
```

7 Supporting information

This clause introduces information supporting the Data Model specified in Clauses 5 and 6. Supporting data is provided in Annexes A through F:

– Annex A (normative) specifies the requirements for OGC Implementation Specification conformance to this specification, and an abstract test suite for checking conformance.

– Annex B (informative) describes the expected OGC uses (or use cases) for transferring CRS and CT definition data.

– Annex C (informative) repeats the XML DTDs from Clause 6 with all the comments removed, for easier scanning.

– Annex F (informative) provides example XML documents using the XML DTDs specified in Clause 6.

– Annex E (informative) provides a table showing how the names used in this specification correspond to the names used in previous OGC and ISO documents.

– Annex F (informative) summarizes the correspondence between the XML DTDs herein and possible UML class diagrams for recording the specified Data Model.

8 Terms and definitions

The following terms and definitions apply to this specification. For that reason they may deviate from ISO 19111, where more universally valid definitions are provided.

8.1

Cartesian coordinate system
coordinate system which gives the position of points relative to *N* mutually-perpendicular straight axes

NOTE In the context of geospatial coordinates the maximum value of N is three.

8.2

compound coordinate reference system
coordinate system describing the position of points through two or more independent coordinate reference systems

EXAMPLE One coordinate reference system can be a two-dimensional horizontal coordinate system, and the other coordinate reference system can be a vertical gravity-related height system.

8.3

concatenated transformation
sequential application of multiple transformations

8.4
coordinate
one of a sequence of *N* numbers designating the position of a point in *N*-dimensional space

NOTE In a coordinate reference system, the coordinate numbers must be qualified by units.

8.5
coordinate conversion
change of coordinates, based on a one-to-one relationship, from one coordinate reference system to another based on the same datum

EXAMPLE Between geodetic and Cartesian coordinate systems or between geodetic coordinates and projected coordinates, or change of units such as from radians to degrees or feet to metres.

NOTE A conversion uses parameters which have specified values, not empirically determined values.

8.6
coordinate reference system
coordinate system which is related to the real world by a datum

NOTE For geodetic and vertical datums, it will be related to the Earth.

8.7
coordinate system
the set of coordinate axes, with defined or implied metric, used to record point coordinates in a coordinate reference system

NOTE 2 One coordinate system may be used in many coordinate reference systems.

8.8
coordinate transformation
computational process of converting a position given in one coordinate reference system into the corresponding position in another coordinate reference system

NOTE 1 A coordinate transformation can require and use the parameters of the ellipsoids associated with the source and target coordinate reference systems, in addition to the parameters explicitly associated with the transformation.

NOTE 2 The term ‘transformation’ is used only when the parameter values associated with the transformation have been determined empirically from a measurement / calculation process. This is typically the case when a change of datum is involved.

8.9
covariance matrix
matrix of elements (or cells) that contain the expected average values of the product of the error in the matrix row coordinate times the simultaneous error in the matrix column coordinate

NOTE 1 A covariance matrix is a form of detailed error estimate data. Covariance matrices are sometimes called variance-covariance matrices.

NOTE 2 All complete covariance matrices are symmetrical, meaning that the same element values appear on both sides of the diagonal elements.

NOTE 3 Covariance matrices contain information about the absolute and/or relative accuracy of the data elements (e.g. coordinates). The absolute accuracy information is contained in the diagonal matrix elements. Relative accuracy is a function of multiple diagonal and off-diagonal elements. A complete covariance matrix for N specific points in 3D space would contain 3N rows by 3N columns

EXAMPLE For three coordinates, a covariance matrix is a 3 by 3 matrix, with the matrix rows and columns each corresponding to the three coordinates. For just two horizontal coordinates, a covariance matrix is a 2 by 2 matrix, with the matrix rows and columns each corresponding to the two horizontal coordinates. Similarly, for two image coordinates, a covariance matrix is a 2 by 2 matrix, with the matrix rows and columns each corresponding to the two image coordinates.

8.10
cylindrical coordinates
3-dimensional coordinates with two distance and one angular coordinate

8.11
datum
parameter or set of parameters that determine the location of the origin, the orientation and the scale of a coordinate reference system

8.12
depth
distance of a point below a chosen reference surface usually measured along the local vertical (gravity vector).

NOTE 1 Depth is sometimes measured along a line that does not follow the vector of gravity locally. An example is depth in an oil or gas well. These are generally measured along the wellbore path, which may vary significantly from the local vertical. Some sections of a wellbore path may even run horizontally or slope upwards. Nevertheless the distance along the wellbore path is referred to as 'depth'.

NOTE 2 See elevation, ellipsoidal height, and gravity-related height.

8.13
dimension
number of ordinates needed to describe a position in a coordinate system

8.14
elevation
distance of a point from a chosen reference surface along the direction of the gravity vector from the point to that surface.

NOTE 1 See ellipsoidal height and gravity-related height. It should be noted that ellipsoidal height is defined w.r.t. an ellipsoidal model of the shape of the earth. Ellipsoidal height is measured from the point along the line perpendicular to the ellipsoid's surface.

NOTE 2 Height of a point outside the surface treated as positive; negative height is also named as depth.

8.15
ellipsoid
surface formed by the rotation of an ellipse about an axis

NOTE 1 In this document the axis of rotation is always the minor axis.

NOTE 2 Sometimes the alternative word 'spheroid' is used in geodetic or survey practice to express the same concept. Although mathematically speaking incorrect the more common term in geodetic or survey practice is 'ellipsoid'.

NOTE 3 An alternative term used in geodetic practice is 'reference ellipsoid'

8.16
ellipsoidal coordinate system
geodetic coordinate system
coordinate system in which position is specified by geodetic latitude, geodetic longitude and (in the three-dimensional case) ellipsoidal height, associated with one or more geographic coordinate reference systems.

8.17
ellipsoidal height
geodetic height
distance of a point from the ellipsoid measured along the perpendicular from the ellipsoid to this point, positive if upwards or outside of the ellipsoid

NOTE Only used as part of a three-dimensional geodetic coordinate system and never on its own.

8.18
flattening
ratio of the difference between the semi-major (a) and semi-minor axis (b) of an ellipsoid to the semi-major axis; $f = (a-b)/a$

NOTE Sometimes inverse flattening $1/f = a/(a-b)$ is given instead of flattening; $1/f$ is also known as reciprocal flattening.

8.19
geocentric coordinate reference system
3-dimensional coordinate reference system with its origin at the (approximate) center of the Earth. It is commonly associated with a 3D Cartesian coordinate system.

8.20
geodetic coordinates
coordinates defined in a Geocentric, Geographic (2D or 3D) or Projected coordinate reference system.

8.21
geodetic datum
datum describing the relationship of a 3D or 2D coordinate system to the Earth

NOTE The geodetic datum includes an ellipsoid definition.

8.22
geographic coordinate reference system
coordinate reference system using an ellipsoidal coordinate system and based on an ellipsoid that approximates the shape of the Earth

NOTE A geographic coordinate system can be 2D or 3D. In a 3D geographic coordinate system, the third dimension is height above the ellipsoid surface

NOTE A geographic coordinate reference system can be 2D or 3D. In a 3D geographic coordinate reference system, the third dimension is height above or below the ellipsoid surface

8.23
geographic dataset
dataset with a spatial aspect

8.24
geoid
level surface which best fits mean sea level either locally or globally

NOTE "Level surface" means an equipotential surface of the Earth's gravity field that is everywhere perpendicular to the direction of gravity.

8.25
georeferenced image
image where the image position that corresponds to any ground position covered by that image can be computed using an image geometry model

NOTE Most images being exploited are georeferenced, at least approximately. Georeferenced implies that the camera position and pointing directions at the time of image collection are known, with some degree of accuracy. For a two-dimensional image (the normal type), the image coordinate reference system is 2D. However, the ground coordinates are always 3D. The 3D ground coordinates are "projected" into 2D image coordinates using an image projection.

8.26
gravity-related height
height dependent on the Earth's gravity field

NOTE In particular, orthometric height or normal height, which are both approximations of the distance of a point above the geoid.

8.27
Gregorian calendar
calendar in general use first introduced in 1582 to correct an error in the Julian calendar

NOTE In the Gregorian calendar, common years have 365 days and leap years 366 days divided into 12 sequential months .

8.28
Greenwich meridian
meridian passing through Greenwich, United Kingdom

NOTE Most geodetic datums use the Greenwich meridian as the prime meridian.

8.29
ground coordinates
earth referenced coordinates
terrestrial coordinates
coordinates of points measured in a non-image coordinate reference system

NOTE The term ground coordinates is used herein to distinguish such coordinates from image coordinates. Even when an image is collected by a near vertical camera, image coordinates are different from ground coordinates!

8.30
image
record of the likeness of any features, objects, and activities

NOTE An image can be acquired through the sensing of visual or any other segment of the electromagnetic spectrum by sensors, such as thermal infrared, and high resolution radar.

8.31
image coordinates
definition of position within an image, expressed in image row and column coordinates

8.32
image geometry model
mathematical model that specifies the mapping (or projection) from 3D ground position coordinates to the corresponding 2D image position coordinates

NOTE 1 An image geometry model is alternately called an image sensor model, sensor model, imaging model, or image mathematical model. The term "sensor" is often used when the image is generated by a digital camera and is thus originally digital. The word "camera" is usually used when the image is recorded in analog form, normally on film. Of course, film images can be later scanned or digitized and are then "digital".

NOTE 2 An image geometry model can also be used to determine the correct ground position for an image position, if used with additional data. When a single (or monoscopic) image is used, this additional data normally defines the shape and position of the visible ground (or object) surface. For example, this additional data is often a single elevation or is grid elevation data, sometimes called a Digital Terrain Model (DTM). Alternately, two stereoscopic images or multiple overlapping images can be used, that show the same ground point viewed from different directions. In this case, the two (or more) image geometry mathematical models can also be used, with the point coordinates in each individual image, to determine the corresponding 3D ground position.

8.33
image version
new image produced by subsetting and/or resampling the pixels in an original image

8.34
interface
shared boundary between two functional entities

NOTE An interface standard specifies the services in terms of the functional characteristics and behavior observed at the interface. The standard is a contract in the sense that it documents a mutual obligation between the service user and provider and assures stable definition of that obligation.

8.35
latitude
geodetic latitude
ellipsoidal latitude
angle from the equatorial plane to the perpendicular to the ellipsoid through a given point, northwards treated as positive

8.36
local coordinate reference system
a coordinate reference system that is defined for and usually used in a limited region, significantly less than the complete surface of the earth

EXAMPLES Local engineering and architectural coordinates, grids, and drawings.

NOTE The datum of a Local Coordinate Reference System might or might not have a specified (or known) position in geodetic coordinates. Similarly, the coordinate axes might or might not have specified directions in geodetic coordinates. Such geodetic coordinates include geographic coordinates, geocentric coordinates, and map projected coordinates.

8.37
local datum
engineering datum
datum with a local reference, used as a basis for a local coordinate system

NOTE Engineering datum excludes both geodetic and vertical datums.

8.38
longitude
geodetic longitude
ellipsoidal longitude
angle from the prime meridian plane to the meridian plane of the given point, eastward treated as positive

8.39
map projection
conversion from a geodetic coordinate system to a planar surface

8.40
mean sea level
average level of the surface of the sea over all stages of tide

NOTE Mean sea level in a local context normally means mean sea level for the region as measured by tide gauge measurements at one or more points over a given period of time. Mean sea level in a global context differs from a global geoid by not more than 2 metres.

8.41
meridian
intersection of an ellipsoid by a plane containing the semi-minor axis of the ellipsoid

NOTE This term is often used for the pole-to-pole arc rather than the complete closed figure.

8.42
metric (of a coordinate system)
the set of mathematical rules that determines how independent quantities (invariants) such as angles and distances are calculated from coordinate values

8.43
Oblique Cartesian coordinate system
coordinate system with straight axes that are not necessarily mutually-perpendicular

8.44

pixel

2-dimensional picture element that is the smallest nondivisible element of a digital image. In image processing, the smallest element of a digital image that can be assigned a gray level.

NOTE This term originated as a contraction for "picture element".

8.45

polar coordinates

2-dimensional coordinates in which position is specified by distance to the origin and the direction angle

NOTE ISO 19111 does not specify the number of dimensions and therefore implicitly permits a 3-dimensional polar coordinate system to exist. The equivalent of the latter is termed "spherical coordinate system" in this document.

8.46

position

spatial reference of a point or an object

8.47

prime meridian

zero meridian

meridian from which the longitudes of other meridians are quantified

8.48

projected coordinate reference system

two-dimensional coordinate system system resulting from a map projection.

NOTE A projected coordinate reference system is derived from a 2D geographic coordinate reference system by applying a parameterized coordinate transformation known as a 'map projection'.

NOTE A projected coordinate reference system commonly uses a Cartesian coordinate system.

8.49

reference ellipsoid

ellipsoid used as the best local or global approximation of the surface of the geoid.

semi-major axis

semi-diameter of the longest axis of a reference ellipsoid.

NOTE This equates to the semi-diameter of the reference ellipsoid measured in its equatorial plane

8.50

semi-minor axis

semi-diameter of the shortest axis of a reference ellipsoid

NOTE The shortest axis coincides with the rotation axis of the reference ellipsoid and therefore contains both poles.

8.51
spherical coordinate system
3-dimensional coordinate system with one distance, measured from the origin and two angular coordinates; not to be confused with an ellipsoidal coordinate system based on an ellipsoid 'degenerated' into a sphere.

8.52
temporal coordinate
distance from the origin of the interval time scale used as the basis for a temporal reference system

8.53
temporal coordinate reference system
1-dimensional temporal coordinate reference system based on an interval scale defined in terms of a single standard interval

8.54
transformation
change of coordinates from one coordinate reference system to another coordinate reference system based on a different datum through a one-to-one relationship

NOTE A transformation uses parameter values which may have to be derived empirically by a set of points common to both coordinate reference systems. See coordinate conversion and coordinate transformation.

8.55
unit
defined quantity in which dimensioned parameters are expressed

NOTE In this document, the subtypes of units are length units, angular units, time units, and pixel spacing units .

8.56
UTC
coordinated Universal Time
time scale maintained by the Bureau International des Poids et Mesures (International Bureau of Weights and Measures) and the International Earth Rotation Service (IERS) that forms the basis of a coordinated dissemination of standard frequencies and time

8.57
vertical coordinate system
1-dimensional coordinate reference system used for elevation, height, or depth measurements

8.58
vertical datum
datum describing the relation of gravity-related heights to the Earth

NOTE In most cases the vertical datum will be related to sea level. Ellipsoidal heights are treated as related to a three-dimensional ellipsoidal coordinate system referenced to a geodetic datum. Vertical datums include sounding datums (used for hydrographic purposes), in which case the heights may be negative heights or depths.

8.59
well-known
shared
data that has been completely specified and published by some recognized authority, and is accessible through use of an identifier

NOTE 1 Well-known data does not always have to be transferred between client and server software.

NOTE 2 Well-known usually implies that this data is very trustworthy. Well-known does NOT imply that this data is less trustworthy or of lesser importance.

Annex A
(normative)

Conformance

A.1 Conformance requirements

Each OGC Implementation Specification that transfers data defining coordinate reference systems and/or coordinate transformations shall transfer definition data that conforms to this Recommendation Paper. Each relevant data transfer situation specified by such an OGC Implementation Specification shall transfer data that contains the entire applicable subset of the definition data specified in this Paper. Each such data transfer may include any additional data needed. Each relevant OGC Implementation Specification shall clearly specify the contents and format of the data transferred in each specified data transfer situation.

NOTE In many cases, an Implementation Specification will specify the Application Programming Interface (API) to service software. In those cases, each specified data transfer situation is each input and output argument of each operation in a UML model of the service interface. Of course, multiple operation arguments will often transfer the same possible data, and thus use the same subset of the definition data specified here. Also, multiple arguments may use the same subset of the definition data specified here, although different arguments use different subsets of that data.

Whenever coordinate reference system and/or coordinate transformation definition data is transferred, the data contents and format specified herein shall be used wherever applicable. The data contents and format specified herein that shall be used include:

a) Name of each specified data structure and individual data item

b) Meaning of each specified name

c) Contents of each specified data structure

d) Sequence of components within each specified data structure

e) Multiplicity and optionality of each data item in each specified data structure

f) Data type of each specified individual data item

g) Set of specified alternative values, if any, specified for each individual data item

h) XML tag specified for each XML element and attribute

When an OGC Implementation Specification data transfer situation requires a subset or superset of the definition information specified here, the permissible changes are:

a) Omit a specified data structure or individual data item, when not needed

b) Restrict the meaning of a specified name, to match a restriction of an Implementation Specification

c) Remove some of a set of alternative contents in a specified data structure

d) Add additional contents to a specified data structure, containing different information

e) Make an optional data item required or excluded (always included and not allowed)

f) Reduce specified range of number-of-repetitions of a data item in a data structure

g) Change data type to a more restrictive type, of an individual data item

h) Remove some specified alternative values for an individual data item

Many possible changes to the definition data specified here are not allowed, such as:

a) Completely change the definition of a specified name

b) Expand the set of alternative contents in a specified data structure

c) Make optional a data item required in a specified data structure

d) Increase allowed range of number-of-repetitions of a data item in a data structure

e) Change data type to a less restrictive type, of a specified individual data item

f) Change the required order of data items in a specified data structure

A.2 Abstract test suite

Conformance of an OGC Implementation Specification shall be tested by inspecting the specification of each transferred data structure and individual data item. This shall be done for each data structure and item transferred in each specified data transfer situation. For each transferred data structure or item, the following questions shall be answered:

a) What is the (complete) meaning of this data structure or item?

b) Is this meaning part of the definition of a coordinate reference system or coordinate transformation? If not related, this data structure or item IS conformant.

c) Is this meaning similar to the meaning of any data structure or item specified here? If not similar, this data structure or item IS conformant.

d) Is this meaning the same as, or a restriction of, the most-similar meaning specified here? If not the same or restricted, this data structure or item is NOT conformant.

e) Is the name of this data structure or item the same as the name of the corresponding structure or item specified here? If not the same, this data structure or item is NOT conformant.

f) Is the XML tag for this data structure or item the same as the tag of the corresponding structure or item specified here? If not the same, this data structure or item is NOT conformant.

If a data structure is being inspected, the questions continue:

a) Do the contents of this data structure include all or a subset of the contents of the corresponding structure specified here? If don't include all or a subset, this data structure is NOT conformant.

b) Do the contents of this data structure include the corresponding components in the same required order? If not the same order, this data structure is NOT conformant.

c) For each component of the contents of this data structure that has a corresponding component in the structure specified here, is the component multiplicity and optionality compatible?

    1) If the multiplicity and optionality are the same, this data structure IS conformant.

    2) If this component is now required instead of optional, this data structure IS conformant.

    3) If this component is now optional instead of required, this data structure is NOT conformant.

    4) If this component is now not repeated instead of being repeated one or more times, this data structure IS conformant.

    5) If this component can now be repeated instead of not being repeated, this data structure is NOT conformant.

If an individual data item is being inspected, the questions continue:

a) Is the data type of this item the same as, or a subset of, the data type of the corresponding item specified here? If not the same or a subset, this data item is NOT conformant.

b) Does this item have no specified set of alternative values and the corresponding item specified here has a specified set of alternative values? If no alternative values are specified, this data item is NOT conformant.

c) If this item has a specified set of alternative values, are any of those value not allowed values of the corresponding item specified here? If additional values are allowed, this data item is NOT conformant. If not, this data item IS conformant.

d) Does this item have a specified range of values and the corresponding item specified here have a smaller allowed range? If a larger range is allowed, this data item is NOT conformant.

e) Does this item have a specified range of values and the corresponding item specified here have the same or larger allowed range? If so, this data item IS conformant.

Annex B
(informative)

Expected Uses of CRS and CT Definition Data

B.1 Introduction

The expected OGC uses (or use cases) for transferring CRS and CT definition data produce requirements to be satisfied by this data model. Three broad uses of such definition data have been identified:

a) Use to transfer definition of Coordinate Reference System (CRS), used by other geospatial data

b) Use to transfer definition of Coordinate Transformation (CT), that can be performed by a coordinate transformation service

c) Use to transfer lineage or history of other geospatial data

Additional OGC uses of the proposed CRS and CT definition data might be identified in the future. These expected OGC uses assume the OGC is standardizing software-to-software interfaces (or Application Programming Interfaces, APIs) and associated data transfer formats, but is not (currently) trying to standardize human-computer interfaces.

Each OGC standard interface must allow both server and client software to be written that implements that interface. Simplicity of the interfaces is highly desirable, to simplify client software. Simplicity of server implementation software is somewhat important, but not as important as interface simplicity.

The following subclauses first discuss the multiple uses of definition data, the alternate ways in which definition data can be transferred, and then the three data transfer uses listed above.

B.2 Multiple uses of definition data

CRS and CT definition data can be used in at least three broad overlapping ways, to:

a) Describe a specific CRS or CT to a human user. Note that different human users need different degrees of definition detail.

b) Uniquely identify a specific CRS or CT to software. For a CRS, such identification can be used by software to check if different geospatial datasets are recorded using the same CRS, or to find additional data about a CRS that is stored else where.

c) Provide data for performing coordinate transformations, useful to transformation software. For a CT, such data might be used to perform that transformation. For a CRS, such data might be used in transforming point positions to or from that CRS.

B.3 Alternative ways to transfer definition data

CRS and CT definition data can be transferred in three alternative ways:

a) Transfer only identifier, of each CRS or CT

b) Transfer complete specification, of each CRS or CT

c) Transfer partial specification with identifiers for remaining parts, of each CRS or CT

Only an identifier needs to be transferred for a complete CRS or CT, or for any part thereof, for any part that is well-known to the receiving software. Well-known usually means that some recognized authority has produced and published a complete specification of that part. Server software could have those well-known definitions coded into the software. Alternately, server software could be coded to access those definitions when needed from a separate service known to the server. Such a separate service may be maintained by the authority or by a third party.

In general, there will be several CRS and CT specification authorities that one server or client might use, separately and together. Of course, some software implementations may use no such authority, or only one such authority. One widely-used, publicly-available authority is the European Petroleum Survey Group (EPSG), and use of this authority is currently specified in several OGC Implementation Specifications. However, other and more private authorities exist and must be supported by some OGC standard interfaces, including military affiliated authorities (e.g., DIGEST). Somewhat private authorities are expected to be used for image and local coordinate systems.

A complete CRS or CT specification must be transferred for a CRS or CT, or for any part thereof, for any part that is <u>not</u> well-known to the receiving software. Some server or client implementations could support no well-known CRS and/or CT definitions. More likely, some applications are expected to use some non-well-known CRS or CT definitions. For example, a grid coverage can use a CRS specific to that coverage. Also, an Image CRS or image coordinate transformation is likely to be specific to one image, and may not be supported by any authority which makes that CRS or CT well-known.

B.4 Use to transfer coordinate reference system definition

XML can be used to transfer the definition of a Coordinate Reference System (CRS) used by other geospatial data, especially data encoded using XML. In most cases, one geospatial dataset will use only one CRS. The definition of a CRS for a geospatial dataset can be transferred (usually through an OGC interface) for several different purposes, including:

a) Coordinate Transformation (CT) interface. The current Coordinate Transformation (CT) server interface uses XML to transfer definitions of CRSs. That is, a client can send to server software the definition of one CRS that the client is requesting access to. Similarly, the server software can send to a client the definition of one CRS that the client currently has access to.

b) Geography Markup Language (GML). The encoding of simple features in XML, now specified in the GML Recommendation Paper, is expected to use a subset of this CRS XML to define the CRS of a feature collection or feature. In the current GML paper, only well-known CRSs are supported, so only the identifier of that well-known CRS is transferred with a feature collection.

c) Web Map Server (WMS) interface. A future version of the WMS interface might use XML to define the CRS that the client is requesting "map" data in. (The WMS interface does not currently use XML for this purpose.) In the current WMS interface, only well-known CRSs are supported, so only the identifier of that well-known CRS is transferred in a Get Map request.

d) Grid Coverage (GC) interface. A future version of the GC (access) interface might use XML to define the CRS used by a coverage, when requested by a client. (The current GC interface does not use XML for this purpose.) In the current draft GC interface, the CRS can be defined for one specific grid coverage, requiring transfer of a more complete specification of that CRS. That grid CRS is usually specified as a custom affine coordinate transformation from another CRS.

B.4.1 Current CRS transfer by low-level CT interface

The current low-level Coordinate Transformation (CT) Implementation Specification (OGC document 00-007r4) supports XML transfer of a CRS definition in one operation and in one attribute.

The createFromXML operation of the CS_CoordinateSystemFactory <<Interface>> class creates a CS_CoordinateSystem object from a XML character string. One server will implement one CS_CoordinateSystemFactory object. The signature of this UML operation is:

```
createFromXML (xml:CharacterString) : CS_CoordinateSystem
```

The "XML" read-only UML attribute of the CS_Info class allows a client to get an XML character string representation of an object, which can be either a CS_CoordinateSystem or a CS_Unit UML object. Objects of both the CS_CoordinateSystem and CS_Unit UML classes can be instantiated by multiple other objects visible to one client. Note that a CS_CoordinateSystem object can be created in several ways other than use of the createFromXML operation. The signature of this UML attribute is:

```
XML : CharacterString
```

In the createFromXML operation, and when the UML attribute is used to get XML for a CS_CoordinateSystem object, the XML DTD for CS_CoordinateSystem provided in Section 15.1.1 (pages 113 through 115) of 00-007r4 is used. Example XML using that XML DTD is provided in Section 15.1.2 (pages 116 and 117) of 00-007r4. (Because that XML DTD and example XML are each more than one page, they are not copied here.)

When the UML attribute is used to get XML for a CS_LinearUnit or CS_AngularUnit object, the XML DTD is not clearly specified in 00-007r4. I assume that a subset of the CS_CoordinateSystem XML DTD would be used, namely:

```
<!ELEMENT CS_Info EMPTY>
<!ATTLIST CS_Info
    Code                CDATA    #IMPLIED
    Abbreviation        CDATA    #IMPLIED
    Alias               CDATA    #IMPLIED
    Authority           CDATA    #IMPLIED
    Name                CDATA    #IMPLIED
    >

<!ELEMENT CS_LinearUnit (CS_Info) >
<!ATTLIST CS_LinearUnit
    MetresPerUnit       CDATA    #REQUIRED
    >

<!ELEMENT CS_AngularUnit (CS_Info) >
<!ATTLIST CS_AngularUnit
    RadiansPerUnit      CDATA    #REQUIRED
    >
```

NOTE A future version of the Coordinate Transformation (CT) Implementation Specification could use XML in additional places, especially where use of WKT is now supported but use of XML is not now supported.

B.4.2 Current CRS transfer by ICT and high-level CT interfaces

The current draft high-level Coordinate Transformation (CT) Implementation Specification (OGC document 01-013) supports XML transfer of a CRS definition by two operations. The same two operations are supported by the draft Image Coordinate Transformation (ICT) interface (OGC document 00-045r1). These two operations are provided to clients by the one Ground Coordinate Transformation Service <<Interface>> object. The UML operation signatures of these two operations are:

```
addTransformation (metadata : TransformationMetadata,
    format : TextFormat) : TransformationID
transformationMetadata (transformation : TransformationID,
    format : TextFormat) : TransformationMetadata
```

XML is one possible format used by the Transformation Metadata <<DataType>> class that is used by these two operations. The Transformation Metadata class contains three UML attributes, with the signatures:

```
sourceCS : CoordinateSystemDefinition
targetCS : CoordinateSystemDefinition
transformation [0..1]: TransformationDefinition
```

The CoordinateReferenceSystem XML element, with all its contents, specified in this document is intended to be used for each Coordinate System Definition instance in Transformation Metadata. Similarly, the CoordinateTransformationMetadata XML element specified in this document is intended to be used for a complete Transformation Metadata instance.

B.4.3 Current CRS transfer by GML

The current GML Recommendation Paper (OGC document 00-029) supports XML transfer of a CRS definition with each feature transferred in XML. The current GML is limited to use of only well-known CRSs, so this XML currently uses a XML DTD fragment that contains a single attribute name and value:

```
srsName CDATA #IMPLIED (Note: sometimes #REQUIRED)
```

The srsName text must contain both the CRS name-space and name, separated by a colon. An example XML fragment is:

```
srsName="EPSG:4326"
```

The GML paper recommends that implementations consider the name-space to be case insensitive.

B.5 Use to transfer coordinate transformation (CT) definition

XML can be used to transfer the definition of a Coordinate Transformation (CT) in the interfaces to Coordinate Transformation (CT) services. These interfaces include the accepted (low level) CT Implementation Specification, OGC document 00-007r4. These interfaces also include a high-level CT interface now proposed in document 01-013, plus Image Coordinate Transformation interfaces now being developed by an RFC team and documented in 00-045r1. These CT interfaces use XML to transfer CT definitions with and without associated source and target CRSs.

B.5.1 Current CT transfer by low-level CT Interface

The current low-level Coordinate Transformation (CT) Implementation Specification (OGC document 00-007r4) supports XML transfer of a CT definition in one operation and in one attribute.

The createFromXML operation of the CT_MathTransformFactory <<Interface>> class creates a CT_MathTransform object from a XML character string. One server will implement one CT_MathTransformFactory object. The UML signature of this operation is:

```
createFromXML (xml:CharacterString) : CT_MathTransform
```

The "XML" read-only UML attribute of the CT_MathTransform class allows a client to get an XML character string representation of this UML object. Objects of the CT_MathTransform class can be instantiated by multiple UML objects visible to one client. Note that a CT_MathTransform object can be created in several ways other than use of the createFromXML operation. The signature of this UML attribute is:

```
XML : CharacterString
```

For both of these XML uses, the XML DTD for CT_MathTransform provided in Section 15.1.1 (page 113) of 00-007r4 is used. Example XML using that XML DTD is not provided in 00-007r4. The XML DTD for CT_MathTransform now in Section 15.1.1 is:

```
<!DOCTYPE CT_MathTransform [
<!ELEMENT CT_MathTransform (
    CT_ConcatenatedTransform |
    CT_InverseTransform |
    CT_ParameterizedMathTransform |
    CT_PassThroughTransform) >

<!ELEMENT CT_ParameterizedMathTransform (CT_Parameter*)>
<!ATTLIST CT_ParameterizedMathTransform
    ClassName           CDATA    #REQUIRED
    >

<!ELEMENT CT_PassThroughTransform (CT_MathTransform)>
<!ATTLIST CT_PassThroughTransform
    FirstAffectedOrdinate   CDATA    #REQUIRED
    >

<!ELEMENT CT_ConcatenatedTransform (CT_MathTransform*)>
<!ELEMENT CT_InverseTransform (CT_MathTransform)>

<!ELEMENT CT_Parameter EMPTY>
<!ATTLIST CT_Parameter
    Name                CDATA    #REQUIRED
    Value               CDATA    #REQUIRED
    >
]>
```

NOTE This CT_MathTransform does not include any information on the source and target coordinate systems. Also, a future version of the low-level Coordinate Transformation (CT) Implementation Specification could use XML in additional places, especially where use of WKT is now supported but use of XML is not now supported.)

B.5.2 Current CT transfer by ICT and high-level CT interfaces

The current draft high-level Coordinate Transformation (CT) interface specification (OGC document 01-013) supports XML transfer of a CT definition by two operations. The same two operations are supported by the draft Image Coordinate Transformation (ICT) interface (OGC document 00-045r1). These two operations are provided to clients by the one Ground Coordinate Transformation Service <<Interface>> object. The signatures of these two UML operations are:

```
addTransformation (metadata : TransformationMetadata,
    format : TextFormat) : TransformationID
transformationMetadata (transformation : TransformationID,
    format : TextFormat) : TransformationMetadata
```


XML is one possible format used by the Transformation Metadata <<DataType>> class that is used by these two operations. The Transformation Metadata class contains three UML attributes, with the signatures:

```
sourceCS : CoordinateSystemDefinition
targetCS : CoordinateSystemDefinition
transformation [0..1]: TransformationDefinition
```

The CoordinateSystemDefinition XML element, with all its contents, now specified in this proposal is intended to be used for each Coordinate System Definition instance in Transformation Metadata. Similarly, the CoordinateTransformationMetadata XML element now specified in this proposal is intended to be used for a complete Transformation Metadata instance.

B.6 Use to transfer dataset lineage or history

XML can be used to transfer the lineage or history of geospatial data, especially data encoded using XML. Such lineage information is specified by ISO 19115: Geographic information – Metadata to be part of the useful metadata about a dataset, and that metadata could be recorded in XML. (The OGC intends to allow use of ISO 19115 Metadata, but there are no uses of XML to transfer dataset lineage or history in current draft or accepted OGC Implementation Specifications.)

The lineage of a dataset, or of a part of a larger dataset, is likely to include the original CRS of the positions in that data, plus the sequence of coordinate transformations used to change these positions into the CRS in which the data is now recorded. Alternately, one concatenated coordinate transformation could be recorded that includes the original CRS and the sequence of coordinate transformations used. Similar metadata might be recorded for data still in the original CRS, but planned to be converted into a different CRS.

Annex C
(informative)

Transformation Methods

C.1 Introduction

This annex provides a non-exhaustive description of transformation methods. It lists a number of transformation methods defined in the EPSG v 6 well-known data set, extended by a few other often used methods and some transformation methods applicable to the transformation of image coordinates.
The transformation method should uniquely specify the algorithm to be used for a coordinate transformation. This specification may comprise two algorithms, one forward and inverse, as is customary for map projection algorithms. It is recommended to make use of well-known data sets in the referencing of such algorithms, although this naturally does not imply that the targeted coordinate transformation software has implemented these algorithms correctly or has implemented them at all.

C.2 EPSG defined transformation methods

The well-known data set from EPSG (version 6) specifies the following list of transformation methods. Please note that this list is neither prescriptive, nor exhaustive. It also includes some transformation methods that make use of data files rather than parameters (EPSG codes 9613, 9614, 9615, 9620, 9634). Such transformations are currently not covered by this DTD as they cannot be described as a parameterized transformation.

EPSG code	Transformation Method Name (EPSG)
9601	Longitude rotation
9602	Geographic/geocentric conversions
9603	Geocentric translations
9604	Molodenski
9605	Abridged Molodenski
9606	Position Vector 7-param. transformation
9607	Coordinate Frame rotation
9613	NADCON
9614	NTv1
9615	NTv2
9616	Vertical Offset
9617	Madrid to ED50
9618	Geographical and Height Offsets
9619	Geographical Offsets
9620	Norway Offshore Interpolation
9621	Similarity transformation
9622	Affine orthogonal geometric transformation
9623	Affine general geometric transformation
9624	Affine general parametric transformation
9625	General polynomial (2nd-order)
9626	General polynomial (3rd-order)
9627	General polynomial (4th-order)

9628	Reversible polynomial (2nd-order)
9629	Reversible polynomial (3rd-order)
9630	Reversible polynomial (4th-order)
9631	Complex polynomial (3rd-order)
9632	Complex polynomial (4th-order)
9633	Ordnance Survey National Transformation
9634	Maritime Provinces polynomial interpolation
9801	Lambert Conic Conformal (1SP)
9802	Lambert Conic Conformal (2SP)
9803	Lambert Conic Conformal (2SP Belgium)
9804	Mercator (1SP)
9805	Mercator (2SP)
9806	Cassini-Soldner
9807	Transverse Mercator
9808	Transverse Mercator (South Orientated)
9809	Oblique Stereographic
9810	Polar Stereographic
9811	New Zealand Map Grid
9812	Hotine Oblique Mercator
9813	Laborde Oblique Mercator
9814	Swiss Oblique Cylindrical
9815	Oblique Mercator
9816	Tunisia Mining Grid
9817	Lambert Conic Near-Conformal
9818	American Polyconic
9819	Krovak Oblique Conic Conformal

These transformation methods are defined making use of well defined transformation parameters. When a coordinate transformation is specified referencing an EPSG defined transformation method, the detailed specification in the XML file should adhere to the EPSG definition of transformation parameters.

The detailed definition of these transformation algorithms, with their formulas and a worked example is included in the EPSG data set, available on the Web, through the reflector:
http://www.epsg.org/
These and other information on coordinate transformations and –methods is available on:
http://www.remotesensing.org/geotiff/proj_list/
See also:
http://www.posc.org/Epicentre.2_2/DataModel/ExamplesofUsage/eu_cs.html.

C.3. Other transformation methods

The following transformation methods are not specified in any well-known data set but are implied in the specification of Coordinate Systems and Coordinate Reference Systems.

Transformation Method Name
Polar / Cartesian (2D) conversion
Spherical / Cartesian (3D) conversion
Cylindrical / Cartesian conversion
Geographic3D to Geographic2D/GravRelatedHeight
Unit of Measure conversion
Linear to Cartesian (2D)
Linear to Cartesian (3D)

Polar / Cartesian (2D) conversion
This method converts two-dimensional polar coordinates to plane Cartesian coordinates and vice versa (the inverse algorithm is implied in this transformation method).

Spherical / Cartesian (3D) conversion
This method converts spherical coordinates, sometimes referred to polar 3D coordinates, to 3D Cartesian coordinates and vice versa (the inverse algorithm is again implied in this transformation method). Please bear in mind that when the spherical coordinate system is used in a local context (i.e. with a Local CRS) the converted Cartesian coordinate system can only be used in the same local context: there is no datum change involved. Transformation to a Geocentric Cartesian system is possible only when the spherical coordinate system is also geocentric.

Cylindrical / Cartesian conversion
This method converts (3D) cylindrical coordinates to 3D Cartesian coordinates and vice versa. As with spherical coordinate systems the area of use of the cylindrical coordinate system is unlikely to be other than local. The Cartesian coordinate system will therefore also be associated with a Local CRS. A further (or indeed a one-step) transformation to a Geocentric system will be incorrect, although it may seem mathematically correct.
Geographic3D to Geographic2D/GravRelatedHeight: Transformation from a Geographic 3D CRS to a Compound CRS consisting of a Geographic 2D CRS and a Vertical CRS, or vice versa. The Vertical CRS has a Vertical Datum of type: "geoidal". The Geographic 3D and the Geographic 2D CRS must be based on the same Geodetic Datum.

> (This transformation cannot be described by this version of the XML DTD for Spatial Referencing and Coordinate Transformations but will be included in a future revision. A short description is provided here mainly to prevent confusion.)

This is a complex, multi-step transformation, involving the application of a "geoid model". The geoid model should be available as a regular grid on latitude and longitude with the height of the geoid above the ellipsoid in each grid node. This set of coordinates of points on the geoid is defined in a specified Geographic 3D CRS, normally different from the CRS the user is working in.
Only the vertical component of the coordinates is affected by this transformation; the geodetic latitude and longitude are not.
The transformation involves the following sequence of steps:

- Selection of a subset of the geoid file covering the extent of the points to be transformed.
- Transformation of all three coordinates of the grid nodes in this subset to the target Geographic 3D CRS (regardless whether this Geographic 3D CRS is itself the source or target CRS of the entire coordinate transformation).
- Calculation of the height of the geoid above the ellipsoid ("geoid undulation") in the relevant point(s). This is achieved through a bi-linear interpolation of the geoid undulation, using the latitude and longitude to locate the point in the transformed sub-grid. This step results in the

height of the geoid above the reference ellipsoid of the Geographic 3D CRS that is the source or target CRS in this transformation for each point subject to the transformation.
- Application of the calculated geoid undulation(s) to the heights to be transformed.

C.4. Image coordinate transformation methods

Additionally the following non-exhaustive list of coordinate transformations applicable to Image coordinates is provided:

Transformation Method Name
Ground-to-image transformation
Single-image-to-ground transformation
Elevation coverage with single-image-to-ground transformation
Elevation coverage for 2D to 3D coordinate transformation
Separate coordinate transformation server implementation

Ground-to-image transformation
This method describes parameterized transformations used for ground-to-image coordinate transformations, with one transformation type used for each image geometry model type. The input to such a transformation is 3D ground coordinates, and the output is 2D image coordinates. These parameterized transformations will usually use parameters for directly transforming 3D ground coordinates to 2D image coordinates.

Single-image-to-ground transformation
This method describes parameterized transformations used for single (or monoscopic) image-to-ground transformations, with one transformation type used for each image geometry model type. These parameterized transformations might use parameters for directly transforming 2D image coordinates to multiple 3D ground coordinates. However, these parameterized transformations could use parameters for directly transforming 3D ground coordinates to 2D image coordinates (like ground-to-image transformations).
The output from such a transformation is 3D ground coordinates, and the input is compound 2D image plus 3D ground coordinates. The 3D ground coordinates are used to provide a fixed elevation value, to be used with the 2D image input coordinates. The horizontal position is included with each elevation to allow ground coordinate transformations. Inclusion of the horizontal position also allows the client to provide an approximate horizontal ground position, when known. The server software is allowed to make use of this approximate horizontal position, to speed computation.

Elevation coverage with single-image-to-ground transformation
This method describes a parameterized transformation using an elevation coverage with a single-image-to-ground transformation. One parameterized transformation type is used for any elevation coverage that uses the same OGC specified (grid) coverage interface. The transformation parameters reference one elevation coverage from one external elevation coverage service. This assumes that interface implementations will first use a separate single-image-to-ground transformation, to compute two or more points along the image ray. The implementation will then intersect this ray with ground shape defined by elevation coverage.
The output from this transformation is 3D ground coordinates, and the input is compound 2D image plus 3D ground coordinates. The 3D ground coordinates are used to provide an approximate elevation value, to be used with the 2D image input coordinates. The horizontal position is included with each elevation to allow ground coordinate transformations. Inclusion of the horizontal position also allows the client to provide an approximate horizontal ground

position, when known. The server software is allowed to make use of this approximate horizontal position, to speed computation.

Elevation coverage for 2D to 3D coordinate transformation

This method describes a parameterized transformation using an elevation coverage for 2D to 3D coordinate transformations. One parameterized transformation type is used for any elevation coverage that uses the same OGC specified (grid) coverage interface. The transformation parameters reference one elevation coverage from one external elevation coverage service. The input to this transformation is 2D horizontal coordinates (plus the separately specified elevation coverage), and the output is 3D coordinates. This transformation can be used in a Concatenated Transformation with a ground-to-image transformation, to go from 2D ground coordinates to 2D image coordinates.

Elevation coverage with single ground-to-image transformation

This method describes a parameterized transformation using an elevation coverage with a single ground-to-image transformation. One parameterized transformation type is used for any elevation coverage that uses the same OGC specified (grid) coverage interface. The transformation parameters reference one elevation coverage from one external elevation coverage service. The input to this transformation is 2D horizontal ground coordinates (plus the separately specified elevation coverage), and the output is 2D image coordinates.

An implementation of this parameterized transformation is expected to first use the elevation coverage to obtain the third ground coordinate, and then use a separate implementation of a ground-to-image parameterized transformation.

Separate coordinate transformation server implementation

This method describes a parameterized transformation using a separate Coordinate Transformation server implementation to perform a coordinate transformation. This capability allows use of existing Coordinate Transformation server implementations to perform some transformations, including well-known ground-to-ground transformations. One type of parameterized transformation is used for all external CT servers that use the same OGC specified CT interface. The transformation parameters reference one TransformationID in one external Coordinate Transformation server implementation.

Annex D
(informative)

Uncommented XML DTDs

D.1 Introduction

Since the extensive comments included in the XML DTDs make it difficult to quickly scan these DTDs, the DTDs from Clause 6 are repeated in this annex with all comments removed.

D.2 XML for coordinate reference system definition

```
<?xml version=”1.0” encoding=”UTF-8”?>
<!ELEMENT CoordinateReferenceSystem (
        NameSet?,
        (Identifier
        | (ValidityRegion?,
          (  CompoundCRS
          |  GeocentricCRS
          |  Geographic3dCRS
          |  Geographic2dCRS
          |  ProjectedCRS
          |  LocalCRS
          |  ImageCRS
          |  VerticalCRS
          |  TemporalCRS  ),
          Identifier?) ) ) >

<!ELEMENT CompoundCRS (
        CoordinateReferenceSystem+) >

<!ELEMENT GeocentricCRS (
        GeodeticDatum,
        ( CartesianCoordinateSystem
        | SphericalCoordinateSystem) ) >

<!ELEMENT Geographic3dCRS (
        GeodeticDatum,
        EllipsoidalCoordinateSystem) >

<!ELEMENT Geographic2dCRS (
        GeodeticDatum,
        EllipsoidalCoordinateSystem) >
```

```
<!ELEMENT ProjectedCRS (
        CartesianCoordinateSystem,
        CoordinateReferenceSystem,
        CoordinateTransformationDefinition) >

<!ELEMENT LocalCRS (
        ( CartesianCoordinateSystem
        | ObliqueCartesianCoordinateSystem
        | EllipsoidalCoordinateSystem
        | VerticalCoordinateSystem
        | PolarCoordinateSystem
        | CylindricalCoordinateSystem
        | LinearCoordinateSystem
        | UserDefinedCoordinateSystem),
        LocalDatum,
        ( CoordinateReferenceSystem,
          CoordinateTransformationDefinition)? ) >

<!ELEMENT ImageCRS (
        ( CartesianCoordinateSystem
        | ObliqueCartesianCoordinateSystem ),
        ImageDatum,
        ( CoordinateReferenceSystem,
          CoordinateTransformationDefinition)? ) >

<!ELEMENT VerticalCRS (
        VerticalCoordinateSystem,
        VerticalDatum) >

<!ELEMENT TemporalCRS (
        TemporalCoordinateSystem,
        origin) >
<!ELEMENT origin   (#PCDATA) >
```

D.3 XML for coordinate system definitions

```
<!ELEMENT CartesianCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >
<!ELEMENT dimensions   (#PCDATA) >


<!ELEMENT ObliqueCartesianCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >
```

```
<!ELEMENT EllipsoidalCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >

<!ELEMENT VerticalCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis,
          Identifier?) ) ) >

<!ELEMENT SphericalCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >

<!ELEMENT PolarCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >

<!ELEMENT CylindricalCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >

<!ELEMENT LinearCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis,
          Identifier?) ) ) >

<!ELEMENT TemporalCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis,
          Identifier?) ) ) >
```

```
<!ELEMENT UserDefinedCoordinateSystem (
        NameSet?,
        (Identifier
        | (dimensions,
          CoordinateAxis+,
          Identifier?) ) ) >

<!ELEMENT CoordinateAxis (
        NameSet?,
        (Identifier
        | (axisAbbreviation?,
        axisDirection,
        (LinearUnit | AngularUnit | TimeUnit | PixelSpacingUnit),
        Identifier?) ) ) >
<!ELEMENT axisAbbreviation   (#PCDATA) >
<!ELEMENT axisDirection      (#PCDATA) >
```

D.4 XML for datum definitions

```
<!ELEMENT VerticalDatum (
        NameSet?,
        (Identifier
        | (datumType,
           Identifier?) ) ) >
<!ELEMENT datumType   (#PCDATA) >

<!ELEMENT LocalDatum (
        NameSet
        | ( Identifier,
            NameSet?) ) >

<!ELEMENT ImageDatum (
        NameSet?,
        (Identifier
        | (imageOrigin,
          gridCellAssociation,
          Identifier?) ) ) >
<!ELEMENT imageOrigin      (#PCDATA) >
<!ELEMENT gridCellAssociation (#PCDATA) >

<!ELEMENT GeodeticDatum (
        NameSet?,
        (Identifier
        | (Ellipsoid,
           PrimeMeridian?,
           Identifier?) ) ) >
```

D.5 XML for ellipsoid and prime meridian definitions

```
<!ELEMENT Ellipsoid (
        NameSet?,
        (Identifier
        | (LinearUnit,
          semiMajorAxis,
          semiMinorAxis,
          inverseFlattening,
          Identifier?) ) ) >
    <!ATTLIST Ellipsoid
        flatteningDefinitive   (true | false)   #REQUIRED
        ellipsoidShape         (true | false)   #IMPLIED >
<!ELEMENT semiMajorAxis       (#PCDATA) >
<!ELEMENT semiMinorAxis       (#PCDATA) >
<!ELEMENT inverseFlattening   (#PCDATA) >

<!ELEMENT PrimeMeridian (
        NameSet?,
        (Identifier
        | (greenwichLongitude,
          AngularUnit,
          Identifier?) ) ) >
<!ELEMENT greenwichLongitude   (#PCDATA) >
```

D.6 XML for coordinate transformation definition

```
<!ELEMENT CoordinateTransformationMetadata (
        CoordinateReferenceSystem,
        CoordinateReferenceSystem,
        CoordinateTransformationDefinition?) >

<!ELEMENT CoordinateTransformationDefinition (
        NameSet?,
        (Identifier
        | (sourceDimensions,
          targetDimensions,
          ValidityRegion?,
          PositionErrorEstimates?,
          ( ConcatenatedTransformation
          | InverseTransformation
          | PassThroughTransformation
          | ParameterizedTransformation),
          Identifier?) ) ) >
<!ELEMENT sourceDimensions   (#PCDATA) >
<!ELEMENT targetDimensions   (#PCDATA) >
```

```
<!ELEMENT PositionErrorEstimates (
        matrixSize,
        (LinearUnit | PixelSpacingUnit | TimeUnit)+,
        CovarianceElement+) >
<!ELEMENT matrixSize   (#PCDATA) >

<!ELEMENT CovarianceElement (
        row,
        column,
        covariance) >
<!ELEMENT row          (#PCDATA) >
<!ELEMENT column       (#PCDATA) >
<!ELEMENT covariance   (#PCDATA) >

<!ELEMENT ConcatenatedTransformation (
        CoordinateTransformationDefinition+) >

<!ELEMENT InverseTransformation (
        CoordinateTransformationDefinition) >

<!ELEMENT PassThroughTransformation (
        CoordinateTransformationDefinition,
        firstModifiedOrdinate,
        numberModifiedOrdinates) >
<!ELEMENT firstModifiedOrdinate     (#PCDATA) >
<!ELEMENT numberModifiedOrdinates   (#PCDATA) >
```

D.7 XML for parameterized transformation definition

```
<!ELEMENT ParameterizedTransformation (
        TransformationMethod,
        Parameter*) >

<!ELEMENT TransformationMethod (
        NameSet?,
        (Identifier
        | (description,
           reference*,
           formula?,
           parameters?,
           Identifier?) ) ) >
    <!ATTLIST TransformationMethod
        transformationType   (transformation | conversion)
                         #IMPLIED >
<!ELEMENT description   (#PCDATA) >
<!ELEMENT reference     (#PCDATA) >
<!ELEMENT formula       (#PCDATA) >
<!ELEMENT parameters    (#PCDATA) >
```

```
<!ELEMENT Parameter (
        NameSet?,
        (Identifier
        | ( ( value,
            ( LinearUnit | AngularUnit | TimeUnit
            | PixelSpacingUnit)? ),
        Identifier?) ) ) >
<!ELEMENT value      (#PCDATA) >
```

D.8 XML for identification information

```
<!ELEMENT Identifier (
        code,
        codeSpace,
        edition? ) >
<!ELEMENT code          (#PCDATA) >
<!ELEMENT codeSpace     (#PCDATA) >
<!ELEMENT edition       (#PCDATA) >

<!ELEMENT NameSet (
        name,
        abbreviation*,
        alias*,
        remarks*) >
<!ELEMENT name          (#PCDATA) >
<!ELEMENT abbreviation  (#PCDATA) >
<!ELEMENT alias         (#PCDATA) >
<!ELEMENT remarks       (#PCDATA) >
```

D.8 XML for unit definitions

```
<!ELEMENT LinearUnit (
        NameSet?,
        (Identifier
        | ( ( (metresPerUnit, unitsPerMetre?)
            |  unitsPerMetre),
            Identifier?) ) ) >
<!ELEMENT metresPerUnit   (#PCDATA) >
<!ELEMENT unitsPerMetre   (#PCDATA) >

<!ELEMENT AngularUnit (
        NameSet?,
        (Identifier
        | ( ( (radiansPerUnit, unitsPerRadian?)
            |  unitsPerRadian),
            Identifier?) ) ) >
<!ELEMENT radiansPerUnit   (#PCDATA) >
<!ELEMENT unitsPerRadian   (#PCDATA) >
```

```
<!ELEMENT TimeUnit (
        NameSet?,
        (Identifier
        | ( ( (secondsPerUnit, unitsPerSecond?)
            |  unitsPerSecond),
            Identifier?) ) ) >
<!ELEMENT secondsPerUnit   (#PCDATA) >
<!ELEMENT unitsPerSecond   (#PCDATA) >

<!ELEMENT PixelSpacingUnit (
        NameSet?,
        (Identifier
        | ( ( (pixelSpacingsPerUnit, unitsPerPixelSpacing?)
            |  unitsPerPixelSpacing),
            Identifier?) ) ) >
<!ELEMENT pixelSpacingsPerUnit   (#PCDATA) >
<!ELEMENT unitsPerPixelSpacing   (#PCDATA) >
```

D.9 XML for validity region definition

```
<!ELEMENT ValidityRegion (
        validityArea
        | (minimumCoordinates,
           maximumCoordinates,
           validityArea?) ) >
    <!ATTLIST ValidityRegion
        validityType   (exact | typical | minimum | maximum
            | unlimited)          #IMPLIED >
<!ELEMENT   validityArea         (#PCDATA) >
<!ELEMENT   minimumCoordinates   (#PCDATA) >
<!ELEMENT   maximumCoordinates   (#PCDATA) >
```

Annex E
(informative)

XML examples

E.1 Introduction

This annex provides example XML documents using the XML DTDs specified in Clause 6. These XML examples omit some optional elements and attributes that might be included, and include some optional elements and attributes that might be omitted. This omission or inclusion was partially based on whether reasonable values were known for optional elements and attributes.

E.2 Example XML for well-known CRS

This subclause provides example XML using the Coordinate Reference System definition XML element containing only the Identifier data element. This example XML is applicable to a well-known CRS included in XML formatted feature data, such as specified in the GML Recommendation Paper, document 00-029.

```
<CoordinateReferenceSystem>
   <Identifier>
      <code>4326</code>
      <codeSpace>EPSG</codeSpace>
      <edition>6.0</edition>
   </Identifier>
</CoordinateReferenceSystem>
```

Note that the current GML Recommendation Paper is limited to use of only well-known CRSs, so this XML is currently abbreviated by a single attribute name and value:

```
srsName=”EPSG:4326”
```

E.3 Example XML for geographic 2D coordinate reference system

This subclause provides example XML using the Coordinate Reference System definition XML element with its’ contained elements, applied to a geographic 2D coordinate system. This example XML is applicable to the current low-level Coordinate Transformation (CT) Implementation Specification, document 00-007r4.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE CoordinateReferenceSystem SYSTEM
      "http://opengis.org/xmldtds/transformations.dtd">
<CoordinateReferenceSystem>
   <NameSet>
      <name>OSGB 1936</name>
   </NameSet>
   <ValidityRegion validityType=“typical”>
```

```
    <validityArea>United Kingdom (UK) - Great Britain - England
    Scotland  Wales - onshore; Isle of Man.
    </validityArea>
  </ValidityRegion>
  <Geographic2dCRS>
    <GeodeticDatum>
      <NameSet>
        <name>OSGB 1936</name>
      </NameSet>
      <Ellipsoid flatteningDefinitive="true">
        <NameSet>
          <name>Airy 1830</name>
        </NameSet>
        <LinearUnit>
          <NameSet>
            <name>metre</name>
            <abbreviation>m</abbreviation>
          </NameSet>
          <metresPerUnit>1</metresPerUnit>
          <Identifier>
            <code>9001</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
          </Identifier>
        </LinearUnit>
        <semiMajorAxis>6377563.396</semiMajorAxis>
        <semiMinorAxis>6356256.90923729</semiMinorAxis>
        <inverseFlattening>299.3249646</inverseFlattening>
        <Identifier>
          <code>7001</code>
          <codeSpace>EPSG</codeSpace>
          <edition>6.0</edition>
        </Identifier>
      </Ellipsoid>
      <PrimeMeridian>
        <NameSet>
          <name>Greenwich</name>
        </NameSet>
        <Identifier>
          <code>8901</code>
          <codeSpace>EPSG</codeSpace>
          <edition>6.0</edition>
        </Identifier>
      </PrimeMeridian>
      <Identifier>
        <code>6277</code>
        <codeSpace>EPSG</codeSpace>
        <edition>6.0</edition>
      </Identifier>
    </GeodeticDatum>
    <EllipsoidalCoordinateSystem>
      <NameSet>
```



```
      <name>ellipsoidal</name>
      <alias>geodetic</alias>
   </NameSet>
   <dimensions>2</dimensions>
   <CoordinateAxis>
      <NameSet>
         <name>Geodetic latitude</name>
         <alias>Latitude</alias>
      </NameSet>
      <axisAbbreviation>Lat</axisAbbreviation>
      <axisDirection>north</axisDirection>
      <AngularUnit>
         <NameSet>
            <name>degree</name>
         </NameSet>
         <radiansPerUnit>1.74532925199433E-
               02</radiansPerUnit>
         <Identifier>
            <code>9108</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </AngularUnit>
      <Identifier>
         <code>9901</code>
         <codeSpace>EPSG</codeSpace>
         <edition>6.0</edition>
      </Identifier>
   </CoordinateAxis>
   <CoordinateAxis>
      <NameSet>
         <name>Geodetic longitude</name>
         <alias>Longitude</alias>
      </NameSet>
      <axisAbbreviation>Lon</axisAbbreviation>
      <axisDirection>east</axisDirection>
      <AngularUnit>
         <NameSet>
            <name>degree</name>
         </NameSet>
         <radiansPerUnit>1.74532925199433E-
               02</radiansPerUnit>
         <Identifier>
            <code>9108</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </AngularUnit>
      <Identifier>
         <code>9902</code>
         <codeSpace>EPSG</codeSpace>
         <edition>6.0</edition>
```

```
          </Identifier>
        </CoordinateAxis>
        <Identifier>
          <code>6402</code>
          <codeSpace>EPSG</codeSpace>
          <edition>6.0</edition>
        </Identifier>
      </EllipsoidalCoordinateSystem>
   </Geographic2dCRS>
   <Identifier>
      <code>4277</code>
      <codeSpace>EPSG</codeSpace>
      <edition>6.0</edition>
   </Identifier>
</CoordinateReferenceSystem>
```

The above example XML assumes that the DTDs specified in Clause 6 are stored in a file available from the future URL "http://opengis.org/xmldtds/transformations.dtd". This example includes both an Identifier element and associated specification data for most geospatial entities, although inclusion of both is discouraged when these entities are well-known to the XML receiver.

The optional elements and attributes that are omitted in the above example XML include:

a) The “alias”, “abbreviation” and “remarks” elements in most NameSet elements

b) The unitsPerMetre element in each LinearUnit element.

c) The unitsPerRadian element in each AngularUnit element.

E.4 Example XML for compound coordinate reference system

This subclause provides example XML using the Coordinate Reference System definition XML element with its contained elements, applied to a 3D compound coordinate reference system that combines a Projected and a Vertical CRS. Note that this Compound CRS Definition contains three CRS definitions, for a Vertical CRS, a Geographic 2D CRS, and a Projected CRS.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE CoordinateReferenceSystem SYSTEM
    "http://opengis.org/xmldtds/transformations.dtd">
<CoordinateReferenceSystem>
   <NameSet>
      <name>OSGB36 /British National Grid + ODN</name>
      <abbreviation>GB National Grid + ODN</abbreviation>
   </NameSet>
   <ValidityRegion validityType=“typical”>
      <validityArea>United Kingdom (UK) - Great Britain - England
      Scotland  Wales - onshore; Isle of Man.
      </validityArea>
   </ValidityRegion>
```



```
<CompoundCRS>
   <CoordinateReferenceSystem>
      <NameSet>
         <name>OSGB 1936 / British National Grid</name>
         <abbreviation>British National Grid</abbreviation>
      </NameSet>
      <ProjectedCRS>
         <CartesianCoordinateSystem>
            <NameSet>
               <name>Cartesian</name>
            </NameSet>
            <dimensions>2</dimensions>
            <CoordinateAxis>
               <NameSet>
                  <name>Easting</name>
               </NameSet>
               <axisAbbreviation>E</axisAbbreviation>
               <axisDirection>east</axisDirection>
               <LinearUnit>
                  <NameSet>
                     <name>metre</name>
                  </NameSet>
                  <metresPerUnit>1</metresPerUnit>
                  <Identifier>
                     <code>9001</code>
                     <codeSpace>EPSG</codeSpace>
                     <edition>6.0</edition>
                  </Identifier>
               </LinearUnit>
               <Identifier>
                  <code>9906</code>
                  <codeSpace>EPSG</codeSpace>
                  <edition>6.0</edition>
               </Identifier>
            </CoordinateAxis>
            <CoordinateAxis>
               <NameSet>
                  <name>Northing</name>
               </NameSet>
               <axisAbbreviation>Lon</axisAbbreviation>
               <axisDirection>east</axisDirection>
               <LinearUnit>
                  <NameSet>
                     <name>degree</name>
                  </NameSet>
                  <metresPerUnit>1</metresPerUnit>
                  <Identifier>
                     <code>9001</code>
                     <codeSpace>EPSG</codeSpace>
                     <edition>6.0</edition>
                  </Identifier>
               </LinearUnit>
```

```
          <Identifier>
            <code>9907</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
          </Identifier>
        </CoordinateAxis>
        <Identifier>
          <code>4400</code>
          <codeSpace>EPSG</codeSpace>
          <edition>6.0</edition>
        </Identifier>
      </CartesianCoordinateSystem>
      <CoordinateReferenceSystem>
        <NameSet>
          <name>OSGB 1936</name>
        </NameSet>
        <Geographic2dCRS>
          <GeodeticDatum>
            <NameSet>
              <name>OSGB 1936</name>
            </NameSet>
            <Ellipsoid flatteningDefinitive="true">
              <NameSet>
                <name>Airy 1830</name>
              </NameSet>
              <LinearUnit>
                <NameSet>
                  <name>metre</name>
                  <abbreviation>m</abbreviation>
                </NameSet>
                <metresPerUnit>1</metresPerUnit>
                <Identifier>
                  <code>9001</code>
                  <codeSpace>EPSG</codeSpace>
                  <edition>6.0</edition>
                </Identifier>
              </LinearUnit>
              <semiMajorAxis>6377563.396</semiMajorAxis>
              <semiMinorAxis>6356256.90923729</semiMinorAxis>
              <inverseFlattening>299.3249646</inverseFlattening>
              <Identifier>
                <code>7001</code>
                <codeSpace>EPSG</codeSpace>
                <edition>6.0</edition>
              </Identifier>
            </Ellipsoid>
            <PrimeMeridian>
              <NameSet>
                <name>Greenwich</name>
              </NameSet>
              <Identifier>
                <code>8901</code>
```

```
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </PrimeMeridian>
      <Identifier>
         <code>6277</code>
         <codeSpace>EPSG</codeSpace>
         <edition>6.0</edition>
      </Identifier>
   </GeodeticDatum>
   <EllipsoidalCoordinateSystem>
      <NameSet>
         <name>ellipsoidal</name>
         <alias>geodetic</alias>
      </NameSet>
      <dimensions>2</dimensions>
      <CoordinateAxis>
         <NameSet>
            <name>Geodetic latitude</name>
         </NameSet>
         <axisAbbreviation>Lat</axisAbbreviation>
         <axisDirection>north</axisDirection>
         <AngularUnit>
            <NameSet>
               <name>degree</name>
            </NameSet>
            <radiansPerUnit>1.74532925199433E-02
            </radiansPerUnit>
            <Identifier>
               <code>9108</code>
               <codeSpace>EPSG</codeSpace>
               <edition>6.0</edition>
            </Identifier>
         </AngularUnit>
         <Identifier>
            <code>9901</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </CoordinateAxis>
      <CoordinateAxis>
         <NameSet>
            <name>Geodetic longitude</name>
            <alias>Longitude</alias>
         </NameSet>
         <axisAbbreviation>Lon</axisAbbreviation>
         <axisDirection>east</axisDirection>
         <AngularUnit>
            <NameSet>
               <name>degree</name>
            </NameSet>
            <radiansPerUnit>1.74532925199433E-02
```



```
               </radiansPerUnit>
               <Identifier>
                  <code>9108</code>
                  <codeSpace>EPSG</codeSpace>
                  <edition>6.0</edition>
               </Identifier>
               </AngularUnit>
               <Identifier>
                  <code>9902</code>
                  <codeSpace>EPSG</codeSpace>
                  <edition>6.0</edition>
               </Identifier>
         </CoordinateAxis>
         <Identifier>
            <code>6402</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </EllipsoidalCoordinateSystem>
   </Geographic2dCRS>
   <Identifier>
      <code>4277</code>
      <codeSpace>EPSG</codeSpace>
      <edition>6.0</edition>
   </Identifier>
</CoordinateReferenceSystem>
<CoordinateTransformationDefinition>
   <sourceDimensions>2</sourceDimensions>
   <targetDimensions>2</targetDimensions>
   <ParameterizedTransformation>
      <TransformationMethod>
         <NameSet>
            <name>Transverse Mercator</name>
         </NameSet>
         <description>conformal cylindrical projection,
               also known as Gauss-Kruger
         </description>
         <reference>EPSG Guidance Note 7, December 2000
         </reference>
         <parameters>5</parameters>
         <Identifier>
            <code>9807</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </TransformationMethod>
      <Parameter>
         <NameSet>
            <name>Latitude of Natural Origin</name>
         </NameSet>
         <value>49</value>
         <AngularUnit>
```

```
         <NameSet>
            <name>degree</name>
            <abbreviation>deg</abbreviation>
         </NameSet>
         <radiansPerUnit>1.74532925199433E-02
         </radiansPerUnit>
         <Identifier>
            <code>9108</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </AngularUnit>
      <Identifier>
         <code>8801</code>
         <codeSpace>EPSG</codeSpace>
         <edition>6.0</edition>
      </Identifier>
   </Parameter>
   <Parameter>
      <NameSet>
         <name>Longitude of Natural Origin</name>
      </NameSet>
      <value>-2</value>
      <AngularUnit>
         <NameSet>
            <name>degree</name>
            <abbreviation>deg</abbreviation>
         </NameSet>
         <radiansPerUnit>1.74532925199433E-02
         </radiansPerUnit>
         <Identifier>
            <code>9108</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </AngularUnit>
      <Identifier>
         <code>8802</code>
         <codeSpace>EPSG</codeSpace>
         <edition>6.0</edition>
      </Identifier>
   </Parameter>
   <Parameter>
      <NameSet>
         <name>Scale Factor at Natural Origin</name>
      </NameSet>
      <value>0.999601272</value>
      <Identifier>
         <code>8805</code>
         <codeSpace>EPSG</codeSpace>
         <edition>6.0</edition>
      </Identifier>
```



```
      </Parameter>
      <Parameter>
         <NameSet>
            <name>False Easting</name>
         </NameSet>
         <value>400000</value>
         <LinearUnit>
            <NameSet>
               <name>metre</name>
               <abbreviation>m</abbreviation>
            </NameSet>
            <metresPerUnit>1</metresPerUnit>
            <Identifier>
               <code>9001</code>
               <codeSpace>EPSG</codeSpace>
               <edition>6.0</edition>
            </Identifier>
         </LinearUnit>
         <Identifier>
            <code>8806</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </Parameter>
      <Parameter>
         <NameSet>
            <name>False Northing</name>
         </NameSet>
         <value>-100000</value>
         <LinearUnit>
            <NameSet>
               <name>metre</name>
               <abbreviation>m</abbreviation>
            </NameSet>
            <metresPerUnit>1</metresPerUnit>
            <Identifier>
               <code>9001</code>
               <codeSpace>EPSG</codeSpace>
               <edition>6.0</edition>
            </Identifier>
         </LinearUnit>
         <Identifier>
            <code>8807</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </Parameter>
   </ParameterizedTransformation>
   <Identifier>
      <code>19916</code>
      <codeSpace>EPSG</codeSpace>
      <edition>6.0</edition>
```

```
        </Identifier>
      </CoordinateTransformationDefinition>
    </ProjectedCRS>
    <Identifier>
      <code>27700</code>
      <codeSpace>EPSG</codeSpace>
      <edition>6.0</edition>
    </Identifier>
  </CoordinateReferenceSystem>
  <CoordinateReferenceSystem>
    <NameSet>
      <name>Newlyn</name>
      <abbreviation>ODN</abbreviation>
    </NameSet>
    <VerticalCRS>
      <VerticalCoordinateSystem>
        <NameSet>
          <name>Gravity-related</name>
        </NameSet>
        <dimensions>1</dimensions>
        <CoordinateAxis>
          <NameSet>
            <name>Gravity-related height</name>
          </NameSet>
          <axisAbbreviation>H</axisAbbreviation>
          <axisDirection>up</axisDirection>
          <LinearUnit>
            <NameSet>
              <name>metre</name>
              <abbreviation>m</abbreviation>
            </NameSet>
            <metresPerUnit>1</metresPerUnit>
            <Identifier>
              <code>9001</code>
              <codeSpace>EPSG</codeSpace>
              <edition>6.0</edition>
            </Identifier>
          </LinearUnit>
          <Identifier>
            <code>9904</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
          </Identifier>
        </CoordinateAxis>
        <Identifier>
          <code>6499</code>
          <codeSpace>EPSG</codeSpace>
          <edition>6.0</edition>
        </Identifier>
      </VerticalCoordinateSystem>
      <VerticalDatum>
        <NameSet>
```



```
                <name>Ordnance Datum Newlyn</name>
                <abbreviation>ODN</abbreviation>
              </NameSet>
              <datumType>geoidal</datumType>
              <Identifier>
                <code>5101</code>
                <codeSpace>EPSG</codeSpace>
                <edition>6.0</edition>
              </Identifier>
            </VerticalDatum>
          </VerticalCRS>
          <Identifier>
            <code>5701</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
          </Identifier>
        </CoordinateReferenceSystem>
      </CompoundCRS>
      <Identifier>
        <code>7405</code>
        <codeSpace>EPSG</codeSpace>
        <edition>6.0</edition>
      </Identifier>
</CoordinateReferenceSystem>
```

The above example XML assumes that the DTDs specified in Clause 6 are stored in a file available from the future URL "http://opengis.org/xmldtds/transformations.dtd". This example includes both an Identifier element and associated specification data for most geospatial entities, although inclusion of both is discouraged when these entities are well-known to the XML receiver.

The optional elements and attributes that are omitted in the above example XML include:

a) The “alias” element in each NameSet element

b) The ValidityRegion element in a CoordinateTransformationDefinition element

c) The unitsPerMetre element in each LinearUnit element.

d) The unitsPerRadian element in each AngularUnit element.

E.5 Simple example XML for compound coordinate reference system

The example XML in the previous subclause includes many elements and attributes that are optional, and can thus be omitted when not needed. This subclause provides a simplified example XML for the same 3D compound coordinate system, which omits most of the optional elements and attributes. This simplified example XML is intended to be adequate when the Projected CRS and the Vertical CRS are both well-known, as input to server software implementing the high-level Coordinate Transformation interface. That is, this simplified example assumes that the server software can obtain the coordinate reference system specification and other data for these two coordinate reference systems using only the Identifier that is input. This simplified example XML also assumes the coordinate axes from the two constituting coordinate systems are sequentially combined.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE CoordinateReferenceSystem SYSTEM
    "http://opengis.org/xmldtds/transformations.dtd">
<CoordinateReferenceSystem>
   <NameSet>
      <name>OSGB36 /British National Grid + ODN</name>
      <abbreviation>GB National Grid + ODN</abbreviation>
   </NameSet>
   <ValidityRegion validityType="typical">
      <validityArea>United Kingdom (UK) - Great Britain - England
      Scotland  Wales - onshore; Isle of Man.
      </validityArea>
   </ValidityRegion>
   <CompoundCRS>
      <CoordinateReferenceSystem>
         <NameSet>
            <name>OSGB 1936 / British National Grid</name>
            <abbreviation>British National Grid</abbreviation>
         </NameSet>
         <Identifier>
            <code>27700</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </CoordinateReferenceSystem>
      <CoordinateReferenceSystem>
         <NameSet>
            <name>Newlyn</name>
            <abbreviation>ODN</abbreviation>
         </NameSet>
         <Identifier>
            <code>5701</code>
            <codeSpace>EPSG</codeSpace>
            <edition>6.0</edition>
         </Identifier>
      </CoordinateReferenceSystem>
   </CompoundCRS>
   <Identifier>
      <code>7405</code>
      <codeSpace>EPSG</codeSpace>
      <edition>6.0</edition>
   </Identifier>
</CoordinateReferenceSystem>
```

E.6 Example XML for coordinate transformation

This subclause provides example XML using a Coordinate Transformation specification using the Coordinate Metadata XML element with its contained elements. To keep the example short and to avoid unnecessary repetition the source and target coordinate reference systems involved with this transformation example are specified by Identifier only (both are well-known coordinate reference systems). The example can therefore been seen to apply to a server implementation where the server software 'knows' the referenced well-known data set, in this case the EPSG data set, edition 6.0. The Coordinate Transformation definition contains both the identifier and the complete specification for a number of contained elements, but for other contained elements, such as units, only definition by identifier is provided. A number of optional elements have been omitted.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE CoordinateTransformationMetadata SYSTEM
   "http://opengis.org/xmldtds/transformations.dtd">
<CoordinateTransformationMetadata>
   <CoordinateReferenceSystem>
      <NameSet>
         <name>OSGB 1936</name>
      </NameSet>
      <Identifier>
         <code>4277</code>
         <codeSpace>EPSG</codeSpace>
         <edition>6.0</edition>
      </Identifier>
   </CoordinateReferenceSystem>
   <CoordinateReferenceSystem>
      <NameSet>
         <name>WGS 84</name>
      </NameSet>
      <Identifier>
         <code>4326</code>
         <codeSpace>EPSG</codeSpace>
         <edition>6.0</edition>
      </Identifier>
   </CoordinateReferenceSystem>
   <CoordinateTransformationDefinition>
      <NameSet>
         <name>OSGB 1936 to WGS 84 (Petroleum)</name>
      </NameSet>
      <sourceDimensions>2</sourceDimensions>
      <targetDimensions>2</targetDimensions>
      <ValidityRegion validityType="typical">
         <validityArea>United Kingdom (UKCS) - Great Britain (GB)
         - England; Scotland; Wales; - North Sea
         </validityArea>
      </ValidityRegion>
      <ParameterizedTransformation>
```

```
<TransformationMethod>
   <NameSet>
      <name>Position Vector 7-param. transformation
      </name>
   </NameSet>
   <description>Transformation of coordinates from one
   geographic CRS into another (also known as a "datum
   transformation").  Implicit concatenation of three
   transformations, therefore not specified as a
   concatenated transformation:
   [geographical to geocentric >> geocentric to
   geocentric >> geocentric to geographic]
   The middle part of this transformation, from
   geocentric to geocentric, is a simplified 7-parameter
   Helmert transformation. Formally the transformation is
   between two geographic 3d CRS’s.  However, in this
   case the transformation method is applied to two
   geographic 2d CRS’s by assuming an input ellipsoidal
   height of zero and ignoring the output ellipsoidal
   height
   </description>
   <reference>EPSG Guidance Note 7, December 2000
   </reference>
   <parameters>7</parameters>
   <Identifier>
      <code>9606</code>
      <codeSpace>EPSG</codeSpace>
      <edition>6.0</edition>
   </Identifier>
</TransformationMethod>
<Parameter>
   <NameSet>
      <name>X-axis translation</name>
   </NameSet>
   <value>+446.448</value>
   <LinearUnit>
      <NameSet>
         <name>metre</name>
         <abbreviation>m</abbreviation>
      </NameSet>
      <Identifier>
         <code>9001</code>
         <codeSpace>EPSG</codeSpace>
         <edition>6.0</edition>
      </Identifier>
   </LinearUnit>
   <Identifier>
      <code>8605</code>
      <codeSpace>EPSG</codeSpace>
      <edition>6.0</edition>
   </Identifier>
</Parameter>
```



```
<Parameter>
   <NameSet>
      <name>Y-axis translation</name>
   </NameSet>
   <value>-125.157</value>
   <LinearUnit>
      <NameSet>
         <name>metre</name>
         <abbreviation>m</abbreviation>
      </NameSet>
      <Identifier>
         <code>9001</code>
         <codeSpace>EPSG</codeSpace>
         <edition>6.0</edition>
      </Identifier>
   </LinearUnit>
   <Identifier>
      <code>8606</code>
      <codeSpace>EPSG</codeSpace>
      <edition>6.0</edition>
   </Identifier>
</Parameter>
<Parameter>
   <NameSet>
      <name>Z-axis translation</name>
   </NameSet>
   <value>+542.060</value>
   <LinearUnit>
      <NameSet>
         <name>metre</name>
         <abbreviation>m</abbreviation>
      </NameSet>
      <Identifier>
         <code>9001</code>
         <codeSpace>EPSG</codeSpace>
         <edition>6.0</edition>
      </Identifier>
   </LinearUnit>
   <Identifier>
      <code>8607</code>
      <codeSpace>EPSG</codeSpace>
      <edition>6.0</edition>
   </Identifier>
</Parameter>
<Parameter>
   <NameSet>
      <name>X-axis rotation</name>
   </NameSet>
   <value>+0.150</value>
   <AngularUnit>
      <NameSet>
         <name>arc-second</name>
```

```
          <abbreviation>sec</abbreviation>
        </NameSet>
        <Identifier>
          <code>9104</code>
          <codeSpace>EPSG</codeSpace>
          <edition>6.0</edition>
        </Identifier>
      </AngularUnit>
      <Identifier>
        <code>8608</code>
        <codeSpace>EPSG</codeSpace>
        <edition>6.0</edition>
      </Identifier>
    </Parameter>
    <Parameter>
      <NameSet>
        <name>Y-axis rotation</name>
      </NameSet>
      <value>+0.247</value>
      <AngularUnit>
        <NameSet>
          <name>arc-second</name>
          <abbreviation>sec</abbreviation>
        </NameSet>
        <Identifier>
          <code>9104</code>
          <codeSpace>EPSG</codeSpace>
          <edition>6.0</edition>
        </Identifier>
      </AngularUnit>
      <Identifier>
        <code>8609</code>
        <codeSpace>EPSG</codeSpace>
        <edition>6.0</edition>
      </Identifier>
    </Parameter>
    <Parameter>
      <NameSet>
        <name>Z-axis rotation</name>
      </NameSet>
      <value>+0.842</value>
      <AngularUnit>
        <NameSet>
          <name>arc-second</name>
          <abbreviation>sec</abbreviation>
        </NameSet>
        <Identifier>
          <code>9104</code>
          <codeSpace>EPSG</codeSpace>
          <edition>6.0</edition>
        </Identifier>
      </AngularUnit>
```



```
          <Identifier>
             <code>8610</code>
             <codeSpace>EPSG</codeSpace>
             <edition>6.0</edition>
          </Identifier>
       </Parameter>
       <Parameter>
          <NameSet>
             <name>Scale difference</name>
          </NameSet>
          <value>-0.000020489</value>
          <Identifier>
             <code>8611</code>
             <codeSpace>EPSG</codeSpace>
             <edition>6.0</edition>
          </Identifier>
       </Parameter>
    </ParameterizedTransformation>
  </CoordinateTransformationDefinition>
</CoordinateTransformationMetadata>
```

E.7 Example XML for server coordinate transformation

This subclause provides example XML that would be used to request coordinate transformation from a coordinate transformation server that ‘knows’ the well-known data set that is referenced in the example, in this case the EPSG v 6.0 data set. The Coordinate Metadata XML element is ‘stripped’ to the bare minimum and only contains the definitions by Identifier of the source and target CRS’s. The server software is expected to find the complete specification of the appropriate coordinate transformation itself. Most optional elements have been omitted in this example.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE CoordinateTransformationMetadata SYSTEM
  "http://opengis.org/xmldtds/transformations.dtd">
<CoordinateTransformationMetadata>
  <CoordinateReferenceSystem>
     <NameSet>
        <name>OSGB 1936</name>
     </NameSet>
     <Identifier>
        <code>4277</code>
        <codeSpace>EPSG</codeSpace>
        <edition>6.0</edition>
     </Identifier>
  </CoordinateReferenceSystem>
  <CoordinateReferenceSystem>
     <NameSet>
        <name>WGS 84</name>
     </NameSet>
     <Identifier>
```

```
        <code>4326</code>
        <codeSpace>EPSG</codeSpace>
        <edition>6.0</edition>
     </Identifier>
   </CoordinateReferenceSystem>
</CoordinateTransformationMetadata>
```

Annex F
(informative)

Corresponding names

Some of the names (or tags) used herein for different data structures and data items are different than used in previous OGC and ISO documents. To support understanding these difference, the left column of Table E.1 lists most of the names now used in this document. The other table columns list the corresponding names now used in the:

a) CT Implementation Specification (OGC document 01-009)

b) GML Recommendation Paper (OGC document 01-029)

c) OGC Abstract Specification Topic 2: Spatial Reference Systems (OGC document 99-102r1)

d) ISO/TC 211 DIS 19111 Spatial Referencing by Coordinates

This table omits those data structures and items used herein only for capabilities not included in any of the other documents, including for:

a) Image coordinate reference systems

b) Temporal coordinate systems

c) Position error estimates

A dash in a table cell means that a corresponding name is not now used. A question mark means that I am not sure. This table includes the two-letter prefixes for some names that are used in ISO/TC 211 DIS 19111, based on the UML schemas in Annex B of that document. (Most of those prefixes do not appear in Clause 6 of 19111.) The above table also includes the two-letter prefixes for element names now used in the low-level CT Implementation Specification (01-009).

Table F.1 Corresponding Names Now Used in Different Documents

This Document (01-014)	CT IS (00-009)	GML (00-029)	OGC AS Topic 2 (99-102r1)	ISO/TC 211 DIS 19111
CoordinateReferenceSystem	CS_CoordinateSystem	EBCS	CoordinateReferenceSystem	SC_CoordinateReferenceSystem, SC_CompoundCRS, SC_CRS
CoordinateSystemDefinition	CS_Info (part of)	—	(included in CoordinateReferenceSystem)	SC_CoordinateSystem
dimensions	Dimension	Dimension	dimension	dimension
Identifier, NameSet	—	—	—	CS ID (= RS_Identifier)
—	—	?	coordinateSystemType	SC_CoordinateSystemType
CoordinateAxis	CS_AxisInfo	CoordinateAxis	CoordinateSystemAxis	SC_CoordinateSystemAxis
axisName	Name	ID (?)	axisName	axisName
axisAbbreviation	—	—	—	—
axisDirection	Orientation	—	axisDefinition (?)	axisDirection
axisNumber	—	—	—	—
LinearUnit, AngularUnit, etc.	CS_LinearUnit, CS_AngularUnit	Unit (?)	axisUnits	axisUnitID (= UnitOfMeasure)
CompoundCoordinateSystem	CS_CompoundCoordinateSystem	—	?	SC_CompoundCRS with SC_KindCode = 1
GeocentricCartesianCoordinate-System	CS_GeocentricCoordinateSystem	—	GeocentricCoordinateReference-System	SC_GeodeticDatum with SC_CoordinateSystemType = Cartesian
Geographic3dCoordinateSystem	—	—	?	?
Geographic2dCoordinateSystem	CS_GeographicCoordinateSystem	Geographic2D (?)	EllipsoidalCoordinateReference-System	SC_GeodeticDatum with SC_CoordinateSystemType = geodetic
ProjectedCoordinateSystem	CS_ProjectedCoordinateSystem	Projected2D	ProjectedCoordinateReference-System	SC_GeodeticDatum with SC_CoordinateSystemType = projected
CoordinateReferenceSystem	CS_GeographicCoordinate-System	Geographic2D-Used	?	CC_Operation:targetID (= RS_Identifier)
LocalCoordinateSystem	CS_LocalCoordinateSystem, CS_FittedCoordinateSystem	—	LocalCoordinateSystem	(SC_EngineeringDatum)
CoordinateTransformation-Definition	ToBase	—	—	?
VerticalCoordinateSystem	CS_VerticalCoordinateSystem	—	?	(SC_VerticalDatum)

This Document (01-014)	CT IS (00-009)	GML (00-029)	OGC AS Topic 2 (99-102r1)	ISO/TC 211 DIS 19111
VerticalDatum	CS_VerticalDatum	—	VerticalDatum	SC_VerticalDatum
datumType	DatumType	—	VerticalDatumType	SC_Datum:type = vertical
LocalDatum	CS_LocalDatum	—	NonGeodeticDatum	SC_EngineeringDatum
datumType	DatumType	—	NonGeodeticDatumType	SC_Datum:type = engineering
GeodeticDatum	CS_HorizontalDatum	GeodeticDatum	GeodeticDatum	SC_GeodeticDatum
—	DatumType	—	GeodeticDatumType	SC_Datum:type = geodetic
—	—	Origin	?	SC_Datum:point (= CharacterString)
—	CS_WGS84ConversionInfo	—	—	—
Ellipsoid	CS_Ellipsoid	—	Ellipsoid	SC_Ellipsoid
semiMajorAxis	SemiMajorAxis	—	semiMajorAxis	semiMajorAxis
semiMinorAxis	SemiMinorAxis	—	semiMinorAxis	—
inverseFlattening	InverseFlattening	—	inverseFlattening	SC_InverseFlattening
flatteningDefinitive	IvfDefinitive	—	definingParameterType (?)	—
ellipsoidShape	—	?	?	ellipsoidShape (=Boolean)
PrimeMeridian	CS_PrimeMeridian	PrimeMeridian	Meridian	SC_PrimeMeridian (is conditional on not 0)
greenwichLongitude	Longitude	—	greenwichLongitude	GreenwichLongitude
CoordinateTransformation-Metadata	—	—	—	CC_Operation
CoordinateTransformation-Definition	CT_MathTransform included	—	CoordinateTransformation	CC_Operation
CoordinateTransformation-Definition using TransformationMethod of any map projection type	CS_Projection	ProjectionClass (?)	CartographicProjection-Transformation	CC_Operation
—	ClassName	—	?	CC_Operation:methodName (= CharacterString)
—	—	—	sphericalRadius	—
ConcatenatedTransformation	CT_ConcatenatedTransform	—	ConcatenatedTransformation	CC_ConcatenatedOperation
InverseTransformation	CT_InverseTransform	—	?	—

This Document (01-014)	CT IS (00-009)	GML (00-029)	OGC AS Topic 2 (99-102r1)	ISO/TC 211 DIS 19111
PassThroughTransformation	CT_PassThroughTransform	—	—	—
firstModifiedAxis	FirstAffectedOrdinate	—	—	—
numberModifiedAxes	— (implied)	—	—	—
ParameterizedTransformation	CT_ParameterizedMathTransform	—	MathTransform (?)	CC_Operation
TransformationMethod	ClassName	—	?	CC_Operation:methodName (= CharacterString)
reference	—	—	—	—
description	—	—	—	remarks
formula	—	—	—	formula
Parameter	CS_ProjectionParameter, CT_Parameter	Parameter	Parameter	CC_OperationParameters
codeName	Name	ID	parameterName	name
value	Value	Units (?)	parameterValue	value (= Measure)
Identifier	CS_Info (part)	ID (part)	?	RS_Identifier (part)
code	Code	Name	CoordinateReferenceSystem-Name, transformationName	identifier
codeSpace	Authority	Authority	—	CI_Citation:title (?)
edition	—	—	—	CI_Citation:edition
NameSet	CS_Info (part)	ID (part)	?	RS_Identifier (part)
name	Name	—	?	alias (?)
abbreviation	Abbreviation	Abbreviation	—	alias (one use)
alias	Alias	Alias	—	alias (one use)
remarks	—	—	—	?
LinearUnit	CS_LinearUnit	—	?	?
metresPerUnit	MetersPerUnit	—	?	?
unitsPerMetre	—	—	—	—
AngularUnit	CS_AngularUnit	—	?	?
radiansPerUnit	RadiansPerUnit	—	?	?
unitsPerRadian	—	—	—	—
ValidityRegion	—	—	DomainOfValidity (?)	validArea (= CharacterString)
validityArea	—	—	—	(validArea = CharacterString)
minimumCoordinates,	PT_Envelope,	—	—	—

This Document (01-014)	CT IS (00-009)	GML (00-029)	OGC AS Topic 2 (99-102r1)	ISO/TC 211 DIS 19111
maximumCoordinates	areaOfUse			
validityType	—	—	—	—

Annex G
(informative)

Correspondence between XML DTD and UML

This annex summarizes the correspondence between the XML DTDs here and possible UML class diagrams for recording this specified Data Model. The corresponding XML DTD parts and UML class diagram parts are listed in Table F.1.

In addition to the above table, potentially useful information on the correspondence between of XML DTDs and UML models is available in the documents:

[1} Mapping from UML Generalised Message Descriptions to XML DTDs, ISIS XML/EDI Project, October 1999 (http://www.tieke.fi/isis-xmledi/D2/UmlToDtdMapping05.doc)

[2] Geographic information - Encoding, ISO/TC 211 CD 19118.2 (N 917), 2000-05-11

Table G.1 Correspondence Between XML DTDs and UML Class Diagrams

XML DTD Part	UML Class Diagram Part	Part Names
DTD (part)	Package	UML package name = XML DTD part name
Element with multiple element and attribute contents	<<Data>> class	UML class name = XML element name
Element with single #PCDATA contents	Attribute, of class containing this XML element	UML attribute name = XML element name Use corresponding UML and XML multiplicities UML attribute type = UML basic type as identified by XML element notes
Attribute with (true | false) values	Attribute, of class containing this XML attribute	UML attribute name = XML attribute name UML attribute type = Boolean
Attribute with more than two alternative values	<<CodeList>> class	UML class name = XML attribute name
Multiple alternative elements included in another element (e.g., Unit, Coordinate System elements)	<<Abstract>> class serving as generalization of alternative subclasses, moving any common attributes and relationships into <<Abstract>> class, referencing <<Abstract>> class in containing class	UML class name = XML note name
Multiple elements and attributes considered parts of a specification of containing element	<<Data>> class containing all specification XML elements and attributes	UML class name = XML note name
Non-alternative and non-specification element included in another element	Attribute of class containing XML element, with dependency on the class for this XML element (Alternative: Aggregation or general association from containing XML element)	UML attribute type = XML element name Use corresponding UML and XML multiplicities UML attribute name = Shortened XML element name
Multiple similar elements (e.g., Datum elements)	<<Abstract>> class serving as generalization of similar subclasses, moving common attributes and relationships into <<Abstract>> class	UML class name = Common part of XML element names (e.g., Datum, Coordinate System)

Bibliography

[1] OGC Recommendation Paper: Geography Markup Language, OGC document 00-029

[2] OpenGIS Implementation Specification: Coordinate Transformation Services, OGC document 00-007r4

[3] OpenGIS Proposal 01-013: High-Level Ground Coordinate Transformation Interface

[4} Draft OGC RFC 00-045r1: Draft RFC on Image Coordinate Transformations

[5] OGC Abstract Specification Topic 0: Overview, OGC document 99-100r1

[6] OpenGIS Accepted Proposal 00-026: Standardize Spatial Reference System (SRS) Identifiers

[7] OpenGIS Accepted Proposal 00-046r1: Clarify Meaning of Local Coordinates System, in Topic 2

[8] Guidelines for Successful OGC Interface Specifications, OGC document 00-014r1

[9] ISO 31 (all parts), *Quantities and units.*