Open Geospatial Consortium Inc.

Date: 2005-11-03

Reference number of this OGC® document: **OGC 05-117**

Version: 0.0.7

Category: OpenGIS® Discussion Paper

Editor: Clemens Portele

OWS-3 Schema Maintenance and Tailoring

Document type: OpenGIS® Discussion Paper
Document subtype: OWS IPR Experiment
Document stage: Approved
Document language: English

Contents

1 Scope....1
2 Conformance....1
3 Normative references....1
4 Terms and definitions....2
5 Conventions....2
5.1 Symbols (and abbreviated terms)....2
5.2 UML Notation....2
6 Metadata for application schemas....4
6.1 UML Application Schemas....4
6.1.1 Schema metadata elements....4
6.1.2 Schema metadata representation....8
6.2 GML Application Schemas....14
6.3 Schema metadata representation in the ebRIM Application Profile of CS-W 2.0....17
6.4 Examples for requests to the Catalog Service....18
6.4.1 Harvesting a GML Application Schema....18
6.4.2 Queryingan application schema....18
7 Application Schema creation....20
7.1 Overview....20
7.2 Application Schema for the NGA Feature Catalog....20
7.2.1 ISO 19109 Application Schema in UML....20
7.2.2 Dictionaries....28
7.2.3 GML Application Schema....31
7.2.3.1 Overview....31
7.2.3.2 Using the ShapeChange command line interface....31
7.2.3.3 Using the ShapeChange web interface and Catalog Service client....32
7.2.4 Known Issues....32
7.2.4.1 Conformance with the proposed GML Profile for Simple Features....32
7.2.4.2 NUNANPO....33
7.2.4.3 Properties with complex values....36
7.2.4.4 Enumerations....36
7.2.4.5 Non-linear ranges of allowed values....39
7.2.4.6 Moire 39
7.2.4.7 Multiple imports of Xlink schemas from different locations....39
7.2.4.8 Categorisations of feature types....39
7.3 Profile of ISO 19107....40
7.3.1 DGIWG Profile....40

7.3.2 GML Profile 40
7.4 Application Schemas for MSD Level 1-3 40
7.4.1 ISO 19109 Application Schema in UML 40
7.4.2 Coverages and Themes 43
7.4.3 Flattened schemas 47
7.4.4 Other known issues 48
7.4.5 GML Application Schema 48
8 Manipulating GML Application Schemas with the Schema Assembly Tool 48

i. Submitting organizations

The following organizations submitted this document to the Open Geospatial Consortium Inc.:

- interactive instruments GmbH

ii. Document Contributor Contact Points

All questions regarding this submission should be directed to the editor or the submitters:

CONTACT	COMPANY
Clemens Portele	interactive instruments
Paul Birkel	Mitre
Charles Roswell	NGA
Dave Wesloh	NGA
Wes Kubo	Galdos
Jody Hawkins	Intergraph
James Sullins	CAST
Panagiotis (Peter) Vretanos	Cubewerx
David Burggraf	Galdos
Craig Bruce	Cubewerx
Rafael Renkert	interactive instruments

iii. Revision history

Date	Release	Author	Paragraph modified	Description
2005-05-06	0.0.1	CP		First draft (Schema metadata)
2005-05-11	0.0.2	CP		Update (Schema metadata)
2005-05-31	0.0.3	CP		Update (Schema creation)
2005-06-10	0.0.4	CP		Schema metadata changes
2005-09-25	0.0.5	CP		Major additions

2005-10-30	0.0.6	CP		Final revision
2005-11-03	0.0.7	CP		Number of SAT IPR added (05-121)
2006-04-24	0.0.7	C. Reed	Various	Fix copyright, cover page, etc.

iv. Changes to the OpenGIS® Abstract Specification

The OpenGIS® Abstract Specification does not require changes to accommodate the technical contents of this document.

Foreword

Attention is drawn to the possibility that some of the elements of this part of OGC 05-117 may be the subject of patent rights. The Open Geospatial Consortium Inc. shall not be held responsible for identifying any or all such patent rights.

Introduction

This document contains a description of the schema tailoring process for the application schema development in the decision support services thread (GeoDSS) during the OWS-3 initiative. In particular it discusses:

- Metadata describing NGA application schemas to support their discovery and assessment using CS-W 2.0 services based on the ebXML Registry Information Model.

- Creation of ISO 19109 Application Schemas in UML for the NSG Feature Catalogue, MSD Level 1, MSD Level 2, and MSD Level 3

- Derivation of the GML Application Schemas for all four application schemas using the ShapeChange UML-to-GML-Application-Schema conversion tool

- Manipulation of the GML Application Schemas using a Schema Assembly Tool

GeoDSS — Schema Maintenance and Tailoring

1 Scope

This OpenGIS® document describes and discusses the process of creating application schemas for NGA data. The approach used to create the application schemas starts with the creation of ISO 19109 application schemas in UML and deriving GML application schemas from them. It also discusses schema metadata.

2 Conformance

Not required for an IP IPR, DIPR, or Discussion Paper.

3 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this document 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.

ISO/TS 19103:2005, *Geographic Information – Conceptual Schema Language*

ISO 19109:2004, *Geographic Information – Rules for Application Schemas*

ISO/CD 19136 (February 2004), *Geographic Information – Geography Markup Language (GML)*

ISO/TC211/N1784 (April 2005), *Geographic Information – Geography Markup Language (GML)*

Catalog Service 2.0, OGC Implementation Specification

ebXML Registry Information Model Application Profile of Catalog Service 2.0, OGC Recommendation Paper

FACC (NGA Profile), MS Access Database FACC_Manager.mdb, provided by NGA

NGA Feature Catalog, MS Access Database NGA_Feature_Catalog.mdb, provided by

NGA

MSD Levels 1-3, MS Access Database MSD_Profiles.mdb, provided by NGA

Department of Defense Discovery Metadata Specification (DDMS), Version 1.2, provided by NGA

DGIWG/TSMAD Profile, Profile(s) of ISO 19107 that support two-dimensional topology

4 Terms and definitions

Not applicable

5 Conventions

5.1 Symbols (and abbreviated terms)

ISO	International Organization for Standardization
OGC	Open Geospatial Consortium
GML	Geography Markup Language
UML	Unified Modeling Language
XML	eXtended Markup Language
1D	One Dimensional
2D	Two Dimensional
3D	Three Dimensional
NGA	National Geospatial Intelligence Agency
FACC	Feature and Attribute Coding Catalogue
DIGEST	Digital Geographic Information Exchange Standard
MSD	Mission Specific Data

5.2 UML Notation

The diagrams that appear in this document are presented using the Unified Modeling Language (UML) static structure diagram based on the rules of ISO/TS 19103 and GML Annex E.

The UML notations used in this document are described in the diagram below.

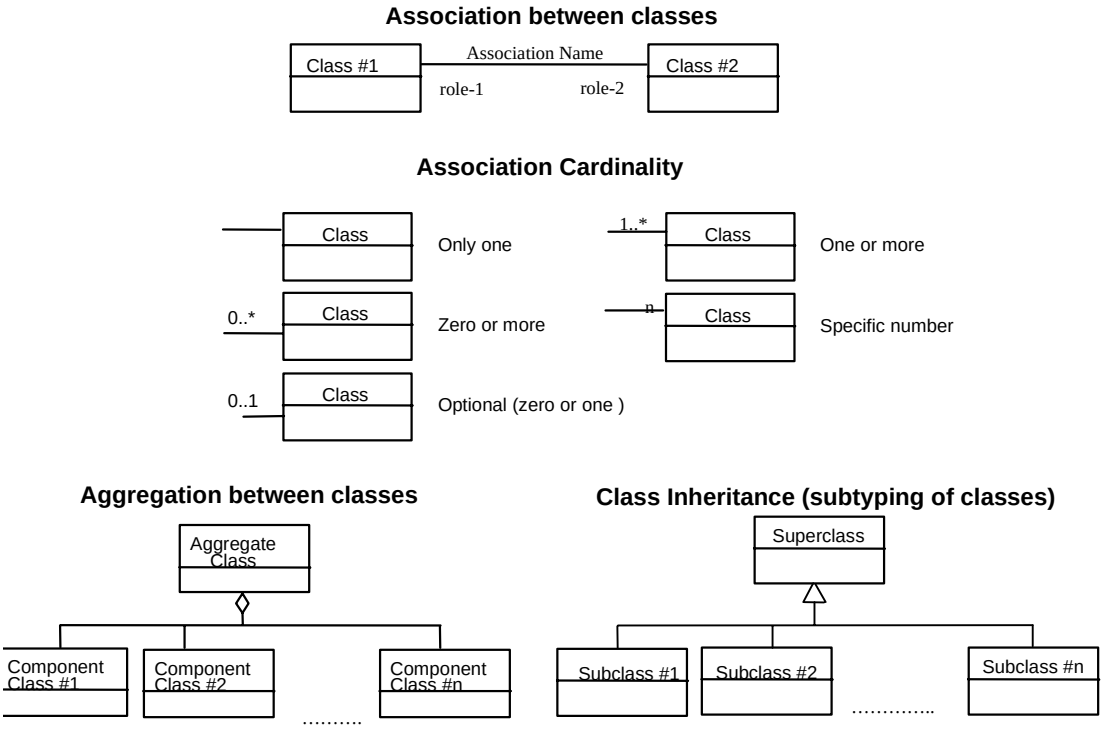


Figure 1 — UML notation

In the class diagrams, the following stereotypes of UML classes are used:

a) <<DataType>> A descriptor of a set of values that lack identity (independent existence and the possibility of side effects). A DataType is a class with no operations whose primary purpose is to hold the information.

b) <<BasicType>> A special type of a DataType that typically has a canonical representation in a computing platform, in this case XML Schema.

c) <<CodeList>> is a flexible enumeration that uses string values for expressing a list of potential values.

d) <<Enumeration>> is a fixed list of valid identifiers of named literal values. Attributes of an enumerated type can only take values from this list.

e) <<Union>> is a list of attributes. The semantics is that only one of the attributes can be present at any time.

In this document, the following standard basic data types are used:

a) CharacterString – A sequence of characters

b) Integer – An integer number

c) Real – A floating point number

d) Boolean – A boolean value (true or false)

6 Metadata for application schemas

6.1 UML Application Schemas

6.1.1 Schema metadata elements

The Catalog Service 2.0 Implementation Specification specifies the following list of "core queryable properties" (see Catalog Service 2.0, 6.3.2, for the definition of the properties). The following table specifies their interpretation in the context of OWS-3 application schemas and in particular the mapping to the DDMS metadata specification and the representation in a UML application schema:

Table 1 — Core queryable properties for UML Application Schemas

Property	DDMS Primary Category	Representation in UML
Subject	Subject (Mandatory)	Tagged value "Subject" of the package representing the application schema The content shall be a comma seperated list of values from either the default MD_TopicCategoryCode codelist from ISO 19115, i.e. - farming - biota - boundaries - climatologyMeteorologyAtmosphere - economy - elevation - environment - geoscientificInformation - health - imageryBaseMapsEarthCover - intelligenceMilitary - inlandWaters - location - oceans - planningCadastre - society - structure - transportation - utilitiesCommunication or the FACC category, i.e.

		- culture - hydrography - hypsography - physiography - vegetation - demarcation - aeronauticalInformation - cadastral - czechNational - routesSpecial - specialUse - general In addition, additional values shall be specified, where applicable, for the Level of the Applciation Schema: - MSD1 - MSD2 - MSD3 - MSD4 - MSD5 Note: These values shall be mapped in the Catalog Service into classifications. The three classifications (ISO 19115, FACC, MSD-Level) shall be maintained seperately in the Catalog Service. To distinguish between the three vocabularies, the following codespace quialifiers shall be used: - http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode - http://www.dgiwg.org/dfdd/category - http://www.nga.mil/
Title	Title (Mandatory)	Name of the package representing the application schema
Abstract	Description (Optional)	Documentation of the package representing the application schema
AnyText		(not used)
Format	Format (Optional)	Codelist values are - text/xml; subtype="xmi/1.0" This value is used for a UML model that does not follow the stricter rules specified by ISO 19109 and/or ISO 19136 - text/xml; subtype="xmi/1.0/iso19109" This value is used for a UML application schema that does not follow the rules specified by ISO 19109, but not the stricter rules specified by ISO 19136 Annex E - text/xml; subtype="xmi/1.0/iso19136" This value is used for a UML application schema that does follow the rules specified by ISO 19136 Annex E In OWS-3 all application schemas will be schemas of the format 'text/xml; subtype="xmi/1.0/iso19136"'.
Identifier	Identifier (Mandatory)	Tagged value "Identifier" of the package representing the application schema The content shall be a URN (see IETF RFC 2141) formed using a UUID and shall

		be the string "urn:uuid:" followed by the hexadecimal representation of a UUID. Example: "urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6" The Identifier value shall change with every new version of the application schema.
Modified	Date (Optional)	Tagged value "Modified" of the package representing the application schema The content shall be specified as "YYYY-MM-DD"
Type	Type (Optional)	Allowed codelist values are: - Schema This value is used for schema metadata for an application schema - FeatureType This value is used for schema metadata for a feature type In OWS-3 schema metadata is mandatory for all application schemas and feature types.
Envelope	Geospatial Coverage, Temporal Coverage	(not used)
CRS		(not used)
Association		The following associations are used: derivedFrom: *Schema/FeatureType resource* is derived from *Schema/FeatureType resource* (not used in UML) dependsOn: *Schema resource* depends on *Schema resource* (UML dependency relationsships between packages representing application schemas) resource: *Schema/FeatureType resource* is available at *URL* (location of the XMI document or the element in the XMI document representing the class resepctively) partOfApplicationSchema: *FeatureType resource* is part of *Schema resource* (containment of a feature type class in a package representing the application schema)

Table 2 — Additional queryable properties for UML Application Schemas

Property	DDMS Primary Category	Representation in UML
Security	Security (Mandatory)	Tagged values of the package representing the application schema; the tagged values are: - IC:classification - IC:ownerProducer - IC:SCIcontrols - IC:SARIdentifier - IC:disseminationControls - IC:FGIsourceOpen - IC:FGIsourceProtected - IC:releasableTo - IC:nonICmarkings - IC:classifiedBy

		- IC:classificationReason - IC:derivedFrom - IC:declassDate - IC:declassEvent - IC:declassException - IC:typeOfExemptedSource - IC:dateOfExemptedSource - IC:declassManualReview The values are used to populate XML attribute instances of the following XML Schema definitions from the Intelligence Community Information Security Marking (IC ISM) Version 2.0 as the authoritative implementation of CAPCO: - urn:us:gov:ic:ism:v2:SecurityAttributesOptionGroup The values therefore shall conform to the rules of this specification. The value of "IC:classification" shall be stored in the Catalog Service as a classification. The allowed values for this classification by IC ISM are: - U - C - S - TS - R - CTS - CTS-B - CTS-BALK - NU - NR - NC - NS - NS-S - NS-A - CTSA - NSAT - NCA
Creator	Creator (Mandatory)	Tagged Value "Creator" of the package representing the application schema
Publisher	Publisher (Optional)	Tagged Value "Publisher" of the package representing the application schema

Other optional DDMS categories exist (Contributor, Rights, Language, Source) and may be added if required.

Then, an application schema in UML might be modelled as follows (see file OWSTest.mdl):

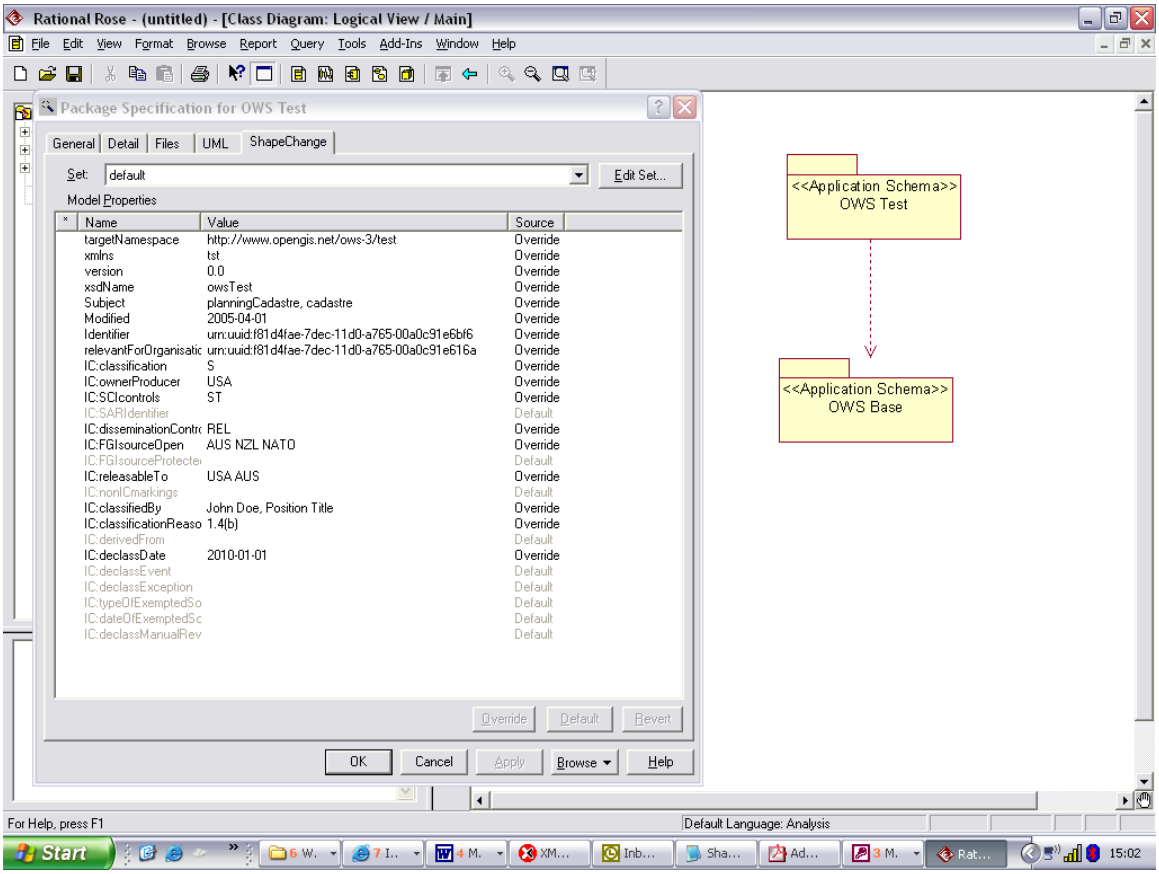


Figure 2 — Application Schema "OWS Test"

The XMI representation of this simple schema is (exported from Rational Rose) shown in file OWSTest.xml.

6.1.2 Schema metadata representation

To make the schema metadata properties available to the Catalog Service, three principal options exist:

1. Specification of Xpath expressions to the information within the XML resource (the XMI document)

2. Specification of a proxy XML resource containing these properties so they can be accessed by simple Xpath expressions

3. Maintaining the resource information in the Catalog "manually" using explicit transactions

Option 1 requires nested predicates and support for variables (here using "$asname" for the name of the application schema). Some examples for different core queryable properties shall illustrate the required level of Xpath support.

Subject:

```
//Foundation.Extension_Mechanisms.TaggedValue.value[
  ../Foundation.Extension_Mechanisms.TaggedValue.tag =
    "RationalRose$ShapeChange:Subject" and
  ../Foundation.Extension_Mechanisms.TaggedValue.modelElement/
Foundation.Core.ModelElement/@xmi.idref =
    //Model_Management.Package[./Foundation.Core.ModelElement.name=$asname]/@xmi.id
]
```

Title:

```
//Model_Management.Package[./Foundation.Core.ModelElement.name=$asname]/Foundation.Core.M
odelElement.name
```

Abstract

```
//Foundation.Extension_Mechanisms.TaggedValue.value[
  ../Foundation.Extension_Mechanisms.TaggedValue.tag = "documentation" and
  ../Foundation.Extension_Mechanisms.TaggedValue.modelElement/
Foundation.Core.ModelElement/@xmi.idref =
    //Model_Management.Package[./Foundation.Core.ModelElement.name=$asname]/@xmi.id
]
```

Format

```
Fixed value: text/xml; subtype="xmi/1.0/iso19136"
```

Identifier

```
//Foundation.Extension_Mechanisms.TaggedValue.value[
  ../Foundation.Extension_Mechanisms.TaggedValue.tag =
    "RationalRose$ShapeChange:Idenifier" and
  ../Foundation.Extension_Mechanisms.TaggedValue.modelElement/
Foundation.Core.ModelElement/@xmi.idref =
    //Model_Management.Package[./Foundation.Core.ModelElement.name=$asname]/@xmi.id
]
```

Modified

```
//Foundation.Extension_Mechanisms.TaggedValue.value[
  ../Foundation.Extension_Mechanisms.TaggedValue.tag =
    "RationalRose$ShapeChange:Modified" and
  ../Foundation.Extension_Mechanisms.TaggedValue.modelElement/
Foundation.Core.ModelElement/@xmi.idref =
    //Model_Management.Package[./Foundation.Core.ModelElement.name=$asname]/@xmi.id
]
```

Type

```
Fixed value: Schema
```

Association "dependsOn"

```
//Foundation.Extension_Mechanisms.TaggedValue.value[
  ../Foundation.Extension_Mechanisms.TaggedValue.tag =
    "RationalRose$ShapeChange:Identifier" and
  ../Foundation.Extension_Mechanisms.TaggedValue.modelElement/
Foundation.Core.ModelElement/@xmi.idref =
    //Model_Management.Package[
      ./@xmi.id=//Foundation.Core.Dependency.supplier[
        ../Foundation.Core.Dependency.client/ Foundation.Core.ModelElement/@xmi.idref =
          //Model_Management.Package[./Foundation.Core.ModelElement.name=$asname]/@xmi.id
      ]/Foundation.Core.ModelElement/@xmi.idref
```

```
    ]/@xmi.id
]
```

If such Xpath expressions can be supported by the Catalog Service(s), then this option would in principle be the preferred approach. However, taking into account that support for the various XMI dialects exported by different UML tools would require even more complex Xpath expressions, the risks associated with the approach do not seem justified. Therefore, an approach using a proxy resource (option 2) should be applied. An example resource file consisting of one schema resource and multiple feature type resources would be:

```
<Resources xmlns:ICISM="urn:us:gov:ic:ism:v2" xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:ddms="http://metadata.dod.mil/mdr/ns/DDMS/1.2/"
xmlns:ows3="http://www.opengis.net/ows3/geodss/schemametadata"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns="http://www.opengis.net/ows3/geodss/schemametadata"
xsi:schemaLocation="http://www.opengis.net/ows3/geodss/schemametadata
http://www.interactive-instruments.de/ows3/repository/xsd/metadata/sm.xsd">
   <Resource xmlns="http://metadata.dod.mil/mdr/ns/DDMS/1.2/">
      <identifier ddms:qualifier="urn:uuid" ddms:value="c03321f1-8129-404f-9d2c-
e1ebc2cc34bc"/>
      <title ICISM:classification="U" ICISM:ownerProducer="USA">MSD Level 3</title>
      <subtitle ICISM:classification="U" ICISM:ownerProducer="USA">Version 0.4_2005-04-
01</subtitle>
      <description ICISM:classification="U" ICISM:ownerProducer="USA">MSD Level 3
application schema. Automatically created from the NGA Access Databases (FACC/DFDD, NGA
Feature Catalogue and Mission Specific Data).</description>
      <language ddms:qualifier="ISO 639-1" ddms:value="en"/>
      <dates ddms:created="2005-07-27"/>
      <rights ddms:copyright="true"/>
      <type ddms:value="Schema"/>
      <creator ICISM:classification="U" ICISM:ownerProducer="USA">
         <Organization>
            <name>interactive instruments GmbH</name>
            <phone>+49-228-91410-70</phone>
            <email>ShapeChange@interactive-instruments.de</email>
         </Organization>
      </creator>
      <publisher ICISM:classification="U" ICISM:ownerProducer="USA">
         <Organization>
            <name>NGA</name>
            <phone>+1-703-814-4580</phone>
            <email>NCGIS-mail@nga.mil</email>
         </Organization>
      </publisher>
      <format>
         <Media>
            <mimeType>text/xml; subtype="xmi/1.0/iso19136"</mimeType>
            <extent/>
            <medium>digital</medium>
         </Media>
      </format>
      <subjectCoverage>
         <Subject>
            <category ddms:label="MSD3" ddms:qualifier="http://www.nga.mil/"/>
            <category ddms:label="culture"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
            <category ddms:label="hydrography"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
            <category ddms:label="hypsography"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
            <category ddms:label="physiography"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
            <category ddms:label="vegetation"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
            <category ddms:label="demarcation"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
```

```
            <category ddms:label="aeronauticalInformation"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
            <category ddms:label="specialUse"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
            <category ddms:label="general"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
            <category ddms:label="farming"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="biota"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="boundaries"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="economy"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="elevation"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="environment"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="geoscientificInformation"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="health"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="intelligenceMilitary"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="inlandWaters"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="location"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="oceans"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="society"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="structure"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="transportation"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
            <category ddms:label="utilitiesCommunication"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
         </Subject>
      </subjectCoverage>
      <geospatialCoverage>
         <Place/>
      </geospatialCoverage>
      <security ICISM:classification="U" ICISM:ownerProducer="USA"/>
      <relations xmlns="http://www.opengis.net/ows3/geodss/schemametadata">
         <relation xlink:href="urn:uuid:75cbe865-0453-4989-8aa0-
91504fa4116d">dependsOn</relation>
         <relation xlink:href="http://www.interactive-
instruments.de/ows3/repository/xmi/msd3.xml">resource</relation>
      </relations>
   </Resource>

   <Resource xmlns="http://metadata.dod.mil/mdr/ns/DDMS/1.2/">
      <identifier ddms:qualifier="urn:uuid" ddms:value="85bbc5c1-a387-4572-a76e-
bfd2c30aa249"/>
      <title ICISM:classification="U" ICISM:ownerProducer="USA">AFA012</title>
      <subtitle ICISM:classification="U" ICISM:ownerProducer="USA">Version 0.4_2005-04-
01</subtitle>
      <description ICISM:classification="U" ICISM:ownerProducer="USA">Contaminated Area:
An area which has been exposed to Nuclear, Biological or Chemical (NBC)
agents.</description>
      <language ddms:qualifier="ISO 639-1" ddms:value="en"/>
      <dates ddms:created="2005-07-27"/>
      <rights ddms:copyright="true"/>
      <type ddms:value="FeatureType"/>
      <creator ICISM:classification="U" ICISM:ownerProducer="USA">
         <Organization>
            <name>interactive instruments GmbH</name>
            <phone>+49-228-91410-70</phone>
            <email>ShapeChange@interactive-instruments.de</email>
         </Organization>
```

```
      </creator>
      <publisher ICISM:classification="U" ICISM:ownerProducer="USA">
         <Organization>
            <name>NGA</name>
            <phone>+1-703-814-4580</phone>
            <email>NCGIS-mail@nga.mil</email>
         </Organization>
      </publisher>
      <format>
         <Media>
            <mimeType>text/xml; subtype="xmi/1.0/iso19136"</mimeType>
            <extent/>
            <medium>digital</medium>
         </Media>
      </format>
      <subjectCoverage>
         <Subject>
            <keyword ddms:value="AFA012"/>
            <keyword ddms:value="Contaminated Area"/>
         </Subject>
      </subjectCoverage>
      <geospatialCoverage>
         <Place/>
      </geospatialCoverage>
      <security ICISM:classification="U" ICISM:ownerProducer="USA"/>
      <relations xmlns="http://www.opengis.net/ows3/geodss/schemametadata">
         <relation xlink:href="urn:uuid:c03321f1-8129-404f-9d2c-
e1ebc2cc34bc">partOfApplicationSchema</relation>
         <relation xlink:href="http://www.interactive-
instruments.de/ows3/repository/xmi/msd3.xml#element(//Foundation.Core.Class[Foundation.Co
re.ModelElement.name='AFA012'])">resource</relation>
      </relations>
   </Resource>
   <Resource xmlns="http://metadata.dod.mil/mdr/ns/DDMS/1.2/">
      <identifier ddms:qualifier="urn:uuid" ddms:value="27d50902-6ecd-4f78-ab5b-
6fcc764d95b7"/>
      <title ICISM:classification="U" ICISM:ownerProducer="USA">AAQ140</title>
      <subtitle ICISM:classification="U" ICISM:ownerProducer="USA">Version 0.4_2005-04-
01</subtitle>
      <description ICISM:classification="U" ICISM:ownerProducer="USA">US-Vehicle
Storage/Parking Area  UK-Vehicle Storage/Parking Area/Car Park/Boat Park: An open land
area used for storing or parking vehicles or vessels (including recreational vehicles).
(See also AI020 and AK060)</description>
      <language ddms:qualifier="ISO 639-1" ddms:value="en"/>
      <dates ddms:created="2005-07-27"/>
      <rights ddms:copyright="true"/>
      <type ddms:value="FeatureType"/>
      <creator ICISM:classification="U" ICISM:ownerProducer="USA">
         <Organization>
            <name>interactive instruments GmbH</name>
            <phone>+49-228-91410-70</phone>
            <email>ShapeChange@interactive-instruments.de</email>
         </Organization>
      </creator>
      <publisher ICISM:classification="U" ICISM:ownerProducer="USA">
         <Organization>
            <name>NGA</name>
            <phone>+1-703-814-4580</phone>
            <email>NCGIS-mail@nga.mil</email>
         </Organization>
      </publisher>
      <format>
         <Media>
            <mimeType>text/xml; subtype="xmi/1.0/iso19136"</mimeType>
            <extent/>
            <medium>digital</medium>
         </Media>
      </format>
      <subjectCoverage>
         <Subject>
            <keyword ddms:value="AAQ140"/>
```

```
            <keyword ddms:value="US-Vehicle Storage/Parking Area  UK-Vehicle
Storage/Parking Area/Car Park/Boat Park"/>
         </Subject>
      </subjectCoverage>
      <geospatialCoverage>
         <Place/>
      </geospatialCoverage>
      <security ICISM:classification="U" ICISM:ownerProducer="USA"/>
      <relations xmlns="http://www.opengis.net/ows3/geodss/schemametadata">
         <relation xlink:href="urn:uuid:c03321f1-8129-404f-9d2c-
e1ebc2cc34bc">partOfApplicationSchema</relation>
         <relation xlink:href="http://www.interactive-
instruments.de/ows3/repository/xmi/msd3.xml#element(//Foundation.Core.Class[Foundation.Co
re.ModelElement.name='AAQ140'])">resource</relation>
      </relations>
   </Resource>
   <!-- … more feature types -->
</Resources>
```

The schema sm.xsd specifying the schema metadata used, i.e. the additions to DDMS is:

```
<xs:schema targetNamespace="http://www.opengis.net/ows3/geodss/schemametadata"
xmlns="http://www.opengis.net/ows3/geodss/schemametadata"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:dc="http://www.purl.org/dc/elements/1.1/"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:ddms="http://metadata.dod.mil/mdr/ns/DDMS/1.2/" elementFormDefault="qualified"
attributeFormDefault="unqualified">
   <xs:import namespace="http://metadata.dod.mil/mdr/ns/DDMS/1.2/" schemaLocation="DDMS-
v1_2.xsd"/>
   <xs:import namespace="http://www.purl.org/dc/elements/1.1/" schemaLocation="rec-
dcmes.xsd"/>
   <xs:import namespace="http://www.w3.org/1999/xlink"
schemaLocation="../xlink/xlinks.xsd"/>
   <xs:element name="relation" substitutionGroup="dc:relation">
      <xs:complexType>
         <xs:complexContent>
            <xs:extension base="dc:SimpleLiteral">
               <xs:attributeGroup ref="xlink:simpleLink"/>
            </xs:extension>
         </xs:complexContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="relations">
      <xs:complexType>
         <xs:sequence>
            <xs:element ref="relation" minOccurs="0" maxOccurs="unbounded"/>
         </xs:sequence>
      </xs:complexType>
   </xs:element>
   <xs:element name="Resources">
      <xs:complexType>
         <xs:sequence>
            <xs:element ref="ddms:Resource" minOccurs="0" maxOccurs="unbounded"/>
         </xs:sequence>
      </xs:complexType>
   </xs:element>
</xs:schema>
```

Option 3 should be avoided to make use of the capabilities of the Catalog Services to harvest the properties from well-known resources.

Note that once a resource is automatically harvested, some direct transactional operations may still be used for additional classifications or associations.

6.2 GML Application Schemas

The same schema information specified for the UML application schemas shall also be available for the GML Application Schemas. Information the original UML application schema shall be added.

The standard way of encoding the schema metadata in a GML Application Schema in OWS-3 is by embedding the metadata information in an appinfo annotation of the <schema> element of the root XML Schema document of the application schema. For example:

```
<schema targetNamespace="http://www.opengis.net/ows-3/nga/MSD3"
xmlns:sc="http://www.interactive-instruments.de/ShapeChange"
xmlns:gml="http://www.opengis.net/gml" xmlns:MSD3="http://www.opengis.net/ows-3/nga/MSD3"
xmlns="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
version="0.4_2005-04-01">
   <annotation>
      <appinfo>
         <Resources xmlns:ICISM="urn:us:gov:ic:ism:v2"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:ddms="http://metadata.dod.mil/mdr/ns/DDMS/1.2/"
xmlns:ows3="http://www.opengis.net/ows3/geodss/schemametadata"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns="http://www.opengis.net/ows3/geodss/schemametadata"
xsi:schemaLocation="http://www.opengis.net/ows3/geodss/schemametadata metadata/sm.xsd">
            <Resource xmlns="http://metadata.dod.mil/mdr/ns/DDMS/1.2/">
               <identifier ddms:qualifier="urn:uuid" ddms:value="02d45a51-bc39-49c2-9973-
03b427cdc17d"/>
               <title ICISM:classification="U" ICISM:ownerProducer="USA">MSD Level
3</title>
               <subtitle ICISM:classification="U" ICISM:ownerProducer="USA">Version
0.4_2005-04-01</subtitle>
               <description ICISM:classification="U" ICISM:ownerProducer="USA">MSD Level
3 application schema. Automatically created from the NGA Access Databases (FACC/DFDD, NGA
Feature Catalogue and Mission Specific Data).</description>
               <language ddms:qualifier="ISO 639-1" ddms:value="en"/>
               <dates ddms:created="2005-07-27"/>
               <rights ddms:copyright="true"/>
               <type ddms:value="Schema"/>
               <creator ICISM:classification="U" ICISM:ownerProducer="USA">
                  <Organization>
                     <name>interactive instruments GmbH</name>
                     <phone>+49-228-91410-70</phone>
                     <email>ShapeChange@interactive-instruments.de</email>
                  </Organization>
               </creator>
               <publisher ICISM:classification="U" ICISM:ownerProducer="USA">
                  <Organization>
                     <name>NGA</name>
                     <phone>+1-703-814-4580</phone>
                     <email>NCGIS-mail@nga.mil</email>
                  </Organization>
               </publisher>
               <format>
                  <Media>
                     <mimeType>text/xml; subtype="gmlas/3.1.1"</mimeType>
                     <extent/>
                     <medium>digital</medium>
                  </Media>
               </format>
               <subjectCoverage>
                  <Subject>
                     <category ddms:label="MSD3" ddms:qualifier="http://www.nga.mil/"/>
                     <category ddms:label="culture"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
                     <category ddms:label="hydrography"
```

```
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
                        <category ddms:label="hypsography"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
                        <category ddms:label="physiography"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
                        <category ddms:label="vegetation"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
                        <category ddms:label="demarcation"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
                        <category ddms:label="aeronauticalInformation"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
                        <category ddms:label="specialUse"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
                        <category ddms:label="general"
ddms:qualifier="http://www.dgiwg.org/dfdd/category"/>
                        <category ddms:label="farming"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="biota"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="boundaries"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="economy"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="elevation"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="environment"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="geoscientificInformation"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="health"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="intelligenceMilitary"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="inlandWaters"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="location"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="oceans"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="society"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="structure"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="transportation"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                        <category ddms:label="utilitiesCommunication"
ddms:qualifier="http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode"/>
                    </Subject>
                </subjectCoverage>
                <geospatialCoverage>
                    <Place/>
                </geospatialCoverage>
                <security ICISM:classification="U" ICISM:ownerProducer="USA"/>
                <relations xmlns="http://www.opengis.net/ows3/geodss/schemametadata">
                    <relation xlink:href="urn:uuid:96284A9E-8074-44E2-AF99-
012771B4963F">derivedFrom</relation>
                    <relation xlink:href="urn:uuid:c4a0f991-fe30-4e5f-9da3-
f7b986885aaf">dependsOn</relation>
                    <relation xlink:href="./MSD3.xsd">resource</relation>
                </relations>
            </Resource>
            <Resource xmlns="http://metadata.dod.mil/mdr/ns/DDMS/1.2/">
                <identifier ddms:qualifier="urn:uuid" ddms:value="7f269990-c0fb-4a3b-b51c-
3342a4607bb2"/>
                <title ICISM:classification="U" ICISM:ownerProducer="USA">AFA012</title>
                <subtitle ICISM:classification="U" ICISM:ownerProducer="USA">Version
0.4_2005-04-01</subtitle>
                <description ICISM:classification="U"
ICISM:ownerProducer="USA">Contaminated Area: An area which has been exposed to Nuclear,
Biological or Chemical (NBC) agents.</description>
                <language ddms:qualifier="ISO 639-1" ddms:value="en"/>
```

```
                <dates ddms:created="2005-07-27"/>
                <rights ddms:copyright="true"/>
                <type ddms:value="FeatureType"/>
                <creator ICISM:classification="U" ICISM:ownerProducer="USA">
                   <Organization>
                      <name>interactive instruments GmbH</name>
                      <phone>+49-228-91410-70</phone>
                      <email>ShapeChange@interactive-instruments.de</email>
                   </Organization>
                </creator>
                <publisher ICISM:classification="U" ICISM:ownerProducer="USA">
                   <Organization>
                      <name>NGA</name>
                      <phone>+1-703-814-4580</phone>
                      <email>NCGIS-mail@nga.mil</email>
                   </Organization>
                </publisher>
                <format>
                   <Media>
                      <mimeType>text/xml; subtype="gmlas/3.1.1"</mimeType>
                      <extent/>
                      <medium>digital</medium>
                   </Media>
                </format>
                <subjectCoverage>
                   <Subject>
                      <keyword ddms:value="AFA012"/>
                      <keyword ddms:value="Contaminated Area"/>
                   </Subject>
                </subjectCoverage>
                <geospatialCoverage>
                   <Place/>
                </geospatialCoverage>
                <security ICISM:classification="U" ICISM:ownerProducer="USA"/>
                <relations xmlns="http://www.opengis.net/ows3/geodss/schemametadata">
                   <relation xlink:href="urn:uuid:02d45a51-bc39-49c2-9973-
03b427cdc17d">partOfApplicationSchema</relation>
                   <relation xmlns:xs="http://www.w3.org/2001/XMLSchema"
xlink:href="./MSD3.xsd#element(//xs:element[@name='AFA012'])">resource</relation>
                </relations>
             </Resource>
             <Resource xmlns="http://metadata.dod.mil/mdr/ns/DDMS/1.2/">
                <identifier ddms:qualifier="urn:uuid" ddms:value="f8203a2b-1a89-4471-adbe-
804b0e3eba46"/>
                <title ICISM:classification="U" ICISM:ownerProducer="USA">AAQ140</title>
                <subtitle ICISM:classification="U" ICISM:ownerProducer="USA">Version
0.4_2005-04-01</subtitle>
                <description ICISM:classification="U" ICISM:ownerProducer="USA">US-Vehicle
Storage/Parking Area  UK-Vehicle Storage/Parking Area/Car Park/Boat Park: An open land
area used for storing or parking vehicles or vessels (including recreational vehicles).
(See also AI020 and AK060)</description>
                <language ddms:qualifier="ISO 639-1" ddms:value="en"/>
                <dates ddms:created="2005-07-27"/>
                <rights ddms:copyright="true"/>
                <type ddms:value="FeatureType"/>
                <creator ICISM:classification="U" ICISM:ownerProducer="USA">
                   <Organization>
                      <name>interactive instruments GmbH</name>
                      <phone>+49-228-91410-70</phone>
                      <email>ShapeChange@interactive-instruments.de</email>
                   </Organization>
                </creator>
                <publisher ICISM:classification="U" ICISM:ownerProducer="USA">
                   <Organization>
                      <name>NGA</name>
                      <phone>+1-703-814-4580</phone>
                      <email>NCGIS-mail@nga.mil</email>
                   </Organization>
                </publisher>
                <format>
                   <Media>
```

```
                     <mimeType>text/xml; subtype="gmlas/3.1.1"</mimeType>
                     <extent/>
                     <medium>digital</medium>
                  </Media>
               </format>
               <subjectCoverage>
                  <Subject>
                     <keyword ddms:value="AAQ140"/>
                     <keyword ddms:value="US-Vehicle Storage/Parking Area  UK-Vehicle
Storage/Parking Area/Car Park/Boat Park"/>
                  </Subject>
               </subjectCoverage>
               <geospatialCoverage>
                  <Place/>
               </geospatialCoverage>
               <security ICISM:classification="U" ICISM:ownerProducer="USA"/>
               <relations xmlns="http://www.opengis.net/ows3/geodss/schemametadata">
                  <relation xlink:href="urn:uuid:02d45a51-bc39-49c2-9973-
03b427cdc17d">partOfApplicationSchema</relation>
                  <relation xmlns:xs="http://www.w3.org/2001/XMLSchema"
xlink:href="./MSD3.xsd#element(//xs:element[@name='AAQ140'])">resource</relation>
               </relations>
            </Resource>
            <!-- … more feature types -->
         </Resources>
      </appinfo>
      <documentation>MSD Level 3 application schema. Automatically created from the NGA
Access Databases (FACC/DFDD, NGA Feature Catalogue and Mission Specific
Data).</documentation>
   </annotation>
   <import namespace="http://www.interactive-instruments.de/ShapeChange"
schemaLocation="ShapeChangeAppinfo.xsd"/>
   <import namespace="http://www.opengis.net/gml"
schemaLocation="base/gml3NGAProfile.xsd"/>
   <!-- ... -->
</schema>
```

In principle, in cases where the schema cannot be edited to add the schema metadata, a proxy resource may be used like in the UML/XMI case. In either case, a relation "resource" shall be part of the resource description pointing to the root schema document of the GML application schema or the element definition of the feature type.

The code list of allowed format values is:

- text/xml; subtype="gmlas/2.1.2"
- text/xml; subtype="gmlas/3.0.0"
- text/xml; subtype="gmlas/3.0.1"
- text/xml; subtype="gmlas/3.1.0"
- text/xml; subtype="gmlas/3.1.1"
- text/xml; subtype="gmlas/3.1.1/sfgml"

6.3 Schema metadata representation in the ebRIM Application Profile of CS-W 2.0

See document OGC document 05-109 (Catalog 2.0 Profile for OWS).

6.4 Examples for requests to the Catalog Service

6.4.1 Harvesting a GML Application Schema

```
<Harvest xmlns="http://www.opengis.net/csw" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.opengis.net/csw
http://schemas.cubewerx.com/schemas/csw/2.0.0/CSW-discovery.xsd" version="2.0.0"
service="CSW" outputFormat="text/xml">
   <Source>http://www.interactive-
instruments.de/ows3/repository/out/151263621636/msd3.xsd</Source>
</Harvest>
```

6.4.2 Queryingan application schema

The following query searches the catalog for all schemas and feature types that have the word "Mine" in their title, subtitle or description. Note that the OWS-3 extension "matchCase='false'" is used to make the search not case-sensitive.

```
<GetRecords xmlns="http://www.opengis.net/csw" xmlns:ogc="http://www.opengis.net/ogc"
xmlns:dc="http://www.purl.org/dc/elements/1.1/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/csw http://schemas.opengis.net/csw/2.0.0/CSW-
discovery.xsd http://www.opengis.net/ogc
http://schemas.opengis.net/filter/1.0.0/filter.xsd http://www.purl.org/dc/elements/1.1/
http://schemas.opengis.net/csw/2.0.0/rec-dcmes.xsd" version="2.0.0" service="CSW"
outputFormat="text/xml" outputSchema="http://schemas.opengis.net/csw/2.0.0/record.xsd">
   <Query typeNames="csw:Record">
      <ElementSetName>full</ElementSetName>
      <Constraint>
         <ogc:Filter>
            <ogc:Or>
               <ogc:PropertyIsLike wildCard="%" singleChar="_" escape="\"
matchCase="false">
                  <ogc:PropertyName>/csw:Record/dc:alternative</ogc:PropertyName>
                  <ogc:Literal>%Mine%</ogc:Literal>
               </ogc:PropertyIsLike>
               <ogc:PropertyIsLike wildCard="%" singleChar="_" escape="\"
matchCase="false">
                  <ogc:PropertyName>/csw:Record/dc:description</ogc:PropertyName>
                  <ogc:Literal>%Mine%</ogc:Literal>
               </ogc:PropertyIsLike>
               <ogc:PropertyIsLike wildCard="%" singleChar="_" escape="\"
matchCase="false">
                  <ogc:PropertyName>/csw:Record/dc:title</ogc:PropertyName>
                  <ogc:Literal>%Mine%</ogc:Literal>
               </ogc:PropertyIsLike>
            </ogc:Or>
         </ogc:Filter>
      </Constraint>
   </Query>
</GetRecords>
```

A query for requesting a *brief* result set (other supported values are *summary* and *full*):

```
<GetRecords xmlns="http://www.opengis.net/csw" xmlns:ogc="http://www.opengis.net/ogc"
xmlns:dc="http://www.purl.org/dc/elements/1.1/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation=
"http://www.opengis.net/csw http://schemas.opengis.net/csw/2.0.0/CSW-discovery.xsd
http://www.opengis.net/ogc http://schemas.opengis.net/filter/1.0.0/filter.xsd
http://www.purl.org/dc/elements/1.1/ http://schemas.opengis.net/csw/2.0.0/rec-dcmes.xsd"
version="2.0.0" service="CSW" outputFormat="text/xml"
outputSchema="http://schemas.opengis.net/csw/2.0.0/record.xsd">
 <Query typeNames="csw:Record">
   <ElementSetName>brief</ElementSetName>
   <Constraint>
```

```
      <ogc:Filter>
         <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/csw:Record/dc:publisher</ogc:PropertyName>
            <ogc:Literal>NGA</ogc:Literal>
         </ogc:PropertyIsEqualTo>
      </ogc:Filter>
   </Constraint>
 </Query>
</GetRecords>
```

The following query illustrates queries on associations. Here, it searches for all resources that have an association with a resource that is located at a URL that contains "interactive-instruments.de":

```
<csw:GetRecords xmlns="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5"
xmlns:ebrim="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5"
xmlns:csw="http://www.opengis.net/cat/csw" xmlns:ogc="http://www.opengis.net/ogc"
xmlns:dc="http://www.purl.org/dc/elements/1.1/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/csw
http://schemas.opengis.net/csw/2.0.0/CSW-discovery.xsd
http://www.opengis.net/ogc http://schemas.opengis.net/filter/1.0.0/filter.xsd
urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5
http://schemas.cubewerx.com/ebrim/2.5/rim.xsd" version="2.0.0" service="CSW"
outputFormat="text/xml" outputSchema="ebRIM">
   <csw:Query typeNames="Association ExternalLink">
      <csw:Constraint>
         <ogc:Filter>
            <ogc:And>
               <ogc:PropertyIsEqualTo>
                  <ogc:PropertyName>/Association/@targetObject</ogc:PropertyName>
                  <ogc:PropertyName>/ExternalLink/@id</ogc:PropertyName>
               </ogc:PropertyIsEqualTo>
               <ogc:PropertyIsLike wildCard="%" singleChar="_" escape="\">
                  <ogc:PropertyName>/ExternalLink/@externalURI</ogc:PropertyName>
                  <ogc:Literal>%interactive-instruments.de%</ogc:Literal>
               </ogc:PropertyIsLike>
            </ogc:And>
         </ogc:Filter>
      </csw:Constraint>
   </csw:Query>
</csw:GetRecords>
```

There are issues with the schema metadata and their CS-W mapping. While the schema metadata is harvested as DDMS metadata elements, the CS-W returns the metadata elements as mapped to the CS-W and ebXML RIM elements. This is not without side effects. Two examples:

- The <ddms:createor> has three sub elements for an organization: name, phone, and email. However, the CS-W core queryable property is just a text string so

```
               <creator ICISM:classification="U" ICISM:ownerProducer="USA">
                  <Organization>
                     <name>interactive instruments GmbH</name>
                     <phone>+49-228-91410-70</phone>
                     <email>ShapeChange@interactive-instruments.de</email>
                  </Organization>
               </creator>
```

  ends up in a CS-W response as

```
               <creator>interactive instruments GmbH+49-228-91410-
70ShapeChange@interactive-instruments.de</creator>
```

- While DDMS has an element <ddms:subtitle> this is mapped to <dc:alternative> because there is no subtitle element in the CS-W core queryable properties which may not be what a CS-W client user expects.

7 Application Schema creation

7.1 Overview

Based on existing data specifications used by NGA, a number of application schemas were created. These application schemas were developed as ISO 19109 application schemas in UML and as GML application schemas (using GML version 3.1, ISO CD 19136). These are:

- a NGA profile of the FACC (NGA Feature Catalog)
- product application schemas for MSD Levels 1, 2 and 3

Since the application schemas are large in terms of the number of feature type and attribute definitions, the schemas were created automatically based on the processes developed in the OWS-2 initiative:

Step (1) was executed using a script creating a Rational Rose UML model from the input descriptions provided by NGA. This script was implemented in Visual Basic.

Step (2) was executed using the Open Source UGAS tool “ShapeChange” of interactive instruments. More information about the tool including documentation can be found in document 05-117 and at http://www.interactive-instruments.de/ugas/. The tool documentation includes one document describing the mapping rules from UML to GML as implemented by the tool plus a second document describing the implementation of the ShapeChange tool, its installation and guidelines for using the tool.

7.2 Application Schema for the NGA Feature Catalog

7.2.1 ISO 19109 Application Schema in UML

The NGA Feature Catalog application schema is created from the MS Access databases maintained by Rex Buckley at NGA (NGA_Feature_Catalog.mdb, using the master table dated April 1st, 2005 and FACC_Manager.mdb in the version 2003-02).

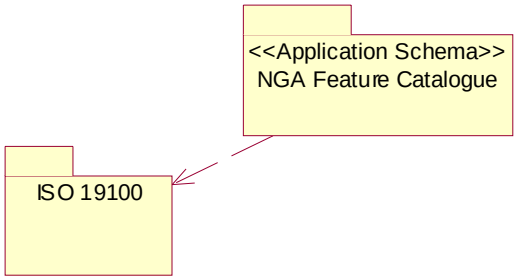
The Master Table in the NGA Feature Catalog database references all components of the

FACC data dictionary that are used in the NGA Feature Catalog.

In order to process these input databases a Visual Basic program was developed implementing the required steps to convert the NGA Feature Catalog to a UML Application Schema in Rational Rose. The DAO interface was used to acccess the MS Access databases and the Rational Rose COM Interface was used to create the objects in Rational Rose. At the same time, the GML Dictionaries for the Feature Type names, the Attribute Type names and the Attribute Code descriptions were created by the program.

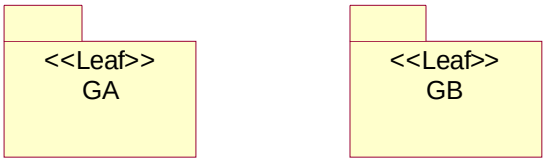
In its first step, the following UML packages were created by the program:

- A root package for the NGA Feature Catalog (stereotype <<Application Schema>>). Pre-defined, tagged values were added to this package to associate the application schema with a namespace ("http://www.opengis.net/ows-3/nga"), a namespace prefix ("nga") and a version number; see the ShapeChange documentation for details. Also, the package is marked as dependant on the Harmonized Model of the ISO 19100 series.



- Within the root package, one package per FACC category (e.g. "A", "B", etc.).

- Within each FACC category package another package for each FACC sub-category (e.g. “AA”, “AB”, etc.). As packages without sub-packages these were tagged with the stereotype <<Leaf>>.



In the second step, the classes were created by the program:

- One class per feature type in the NGA Feature Catalog.

  It was decided to use the DIGEST FACC feature codes as the name of the feature type classes, e.g. “GA065”. For the name of the attributes also the FACC naming scheme was used, too.

  The alternative approach would have been to use descriptive names, e.g. instead of the opaque name "AT005" (which is opaque at least for those not working with the FACC on a regular basis) the name "Cable" would be used and the attribute type name "acc" would be replaced by "accuracyCategory".

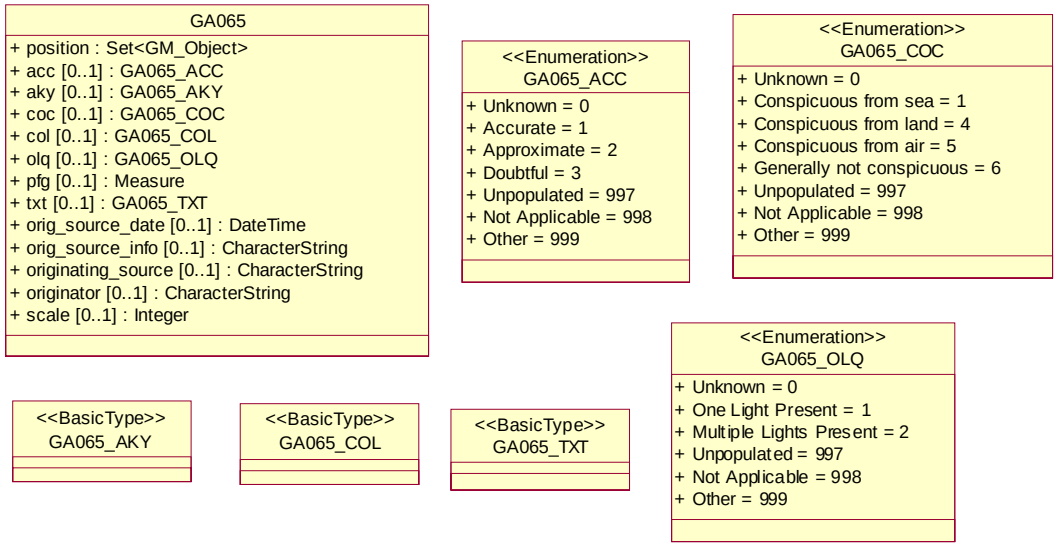
To use the codes instead of the descriptive names has been selected for the following reasons:

- The codes are currently the identifiers for the feature and attribute types, in particular for the exchange of data. While in principle the descriptive names should be unique and stable, too, this is not necessarily true at the moment. In particular, the names are not fully stable; an example are changes from US to British English.

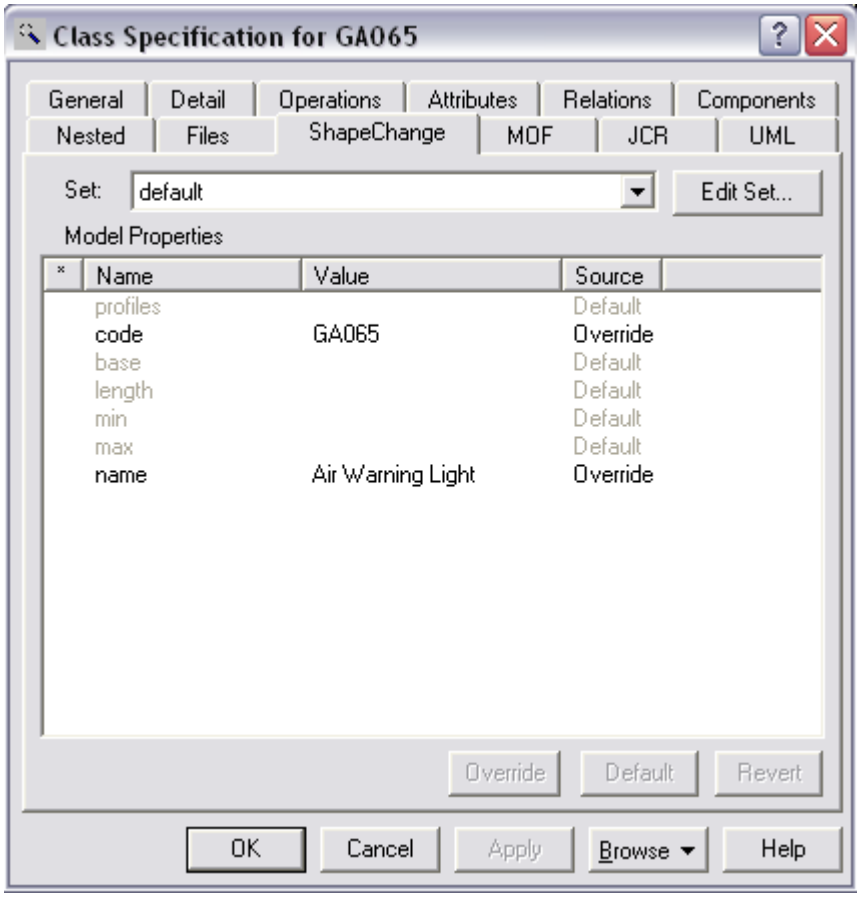
- Since the codes are used in existing data, this makes the transformation and processing easier as it does not require access to the dictionaries.

This reflects the current situation. In the future there may be changes to the role of the names/codes and the decision to use the codes should be reevaluted in that case.

However, the semanticly more meaningful name used in the FACC database was used in the name of the class diagram, e.g. “GA065 - Air Warning Light”.



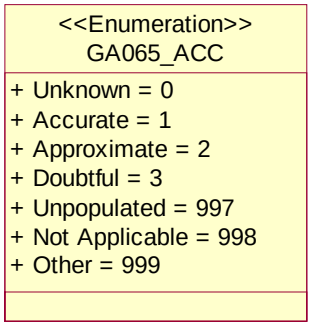
Both the descriptive name and the FACC code were stored in tagged values of the class (“name” and “code”) which are processed by the UGAS tool and therefore represented also in the GML Application Schema, too.



For every attribute of the feature type in the NGA Feature Catalog, one attribute was added to the class. Again, the three letter FACC code was used, e.g. “acc”. All attributes are optional.

The following attribute types can be distinguished:

- Attributes with coded values: In this case, the allowed coded values for the respective feature type are captured in a enumeration class (stereotype <<Enumeration>>). Since the coded values are different for each class, the uniqueness is guaranteed by naming the class by a concatenation of the feature type name and the attribute name. Each coded value is represented with its textual description and code value.

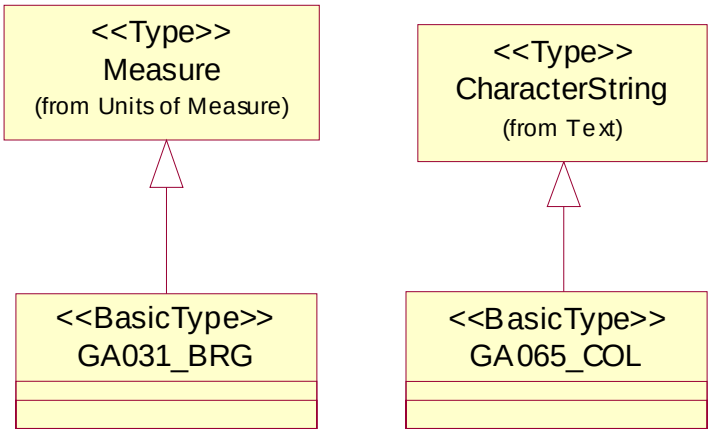


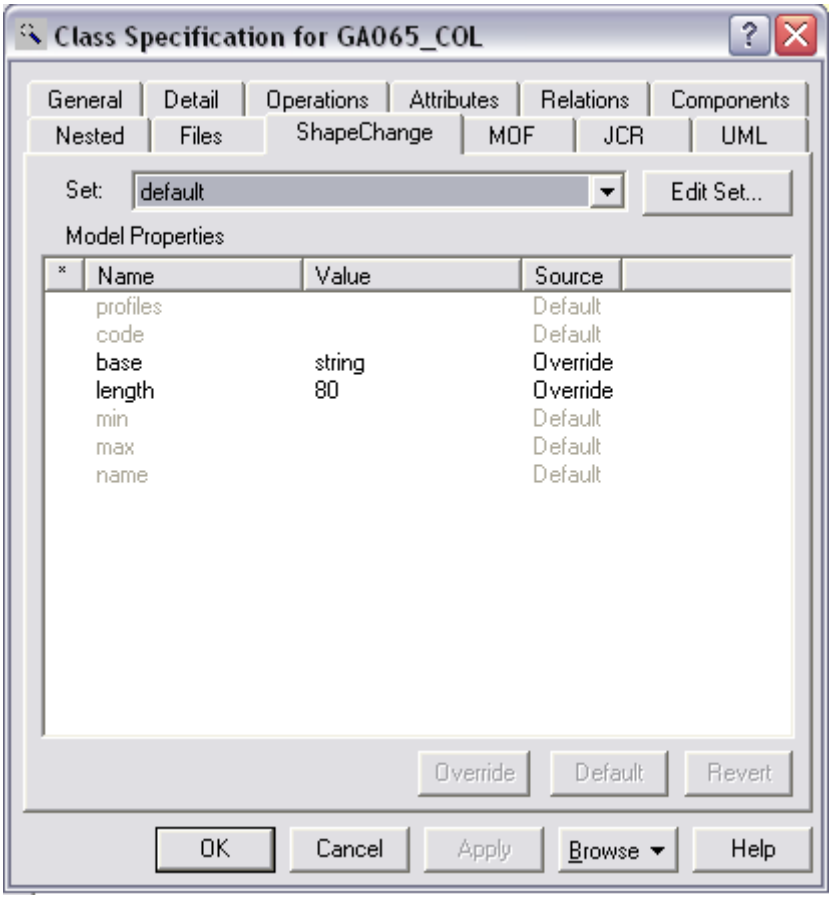
ISO/TS 19103 distinguishes enumerations (fixed lists of values) and code lists (lists of values that can be maintained seperately from the schema).

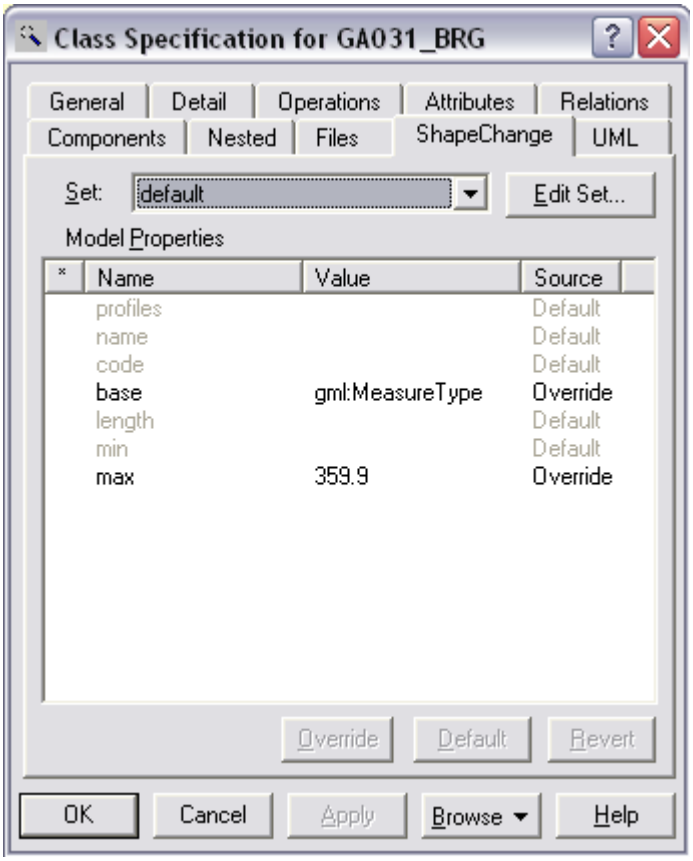
Since the allowed values are fixed for a certain version of the NGA Feature Catalog, the coded values are modelled as enumerations and not as code lists; as a result they are "tied" to the schema (consequently, an XML parser can validate if a coded value is allowed or not for a specific feature instance – something which would not have been true for a code list approach).

Still all coded values are also documented in a separate GML dictionary to allow for the translation from code value to textual description (and vice versa) by applications.

- Other Attributes: These attributes are attributes with either a string or numeric value. String values are often associated with a maximum length. Numeric values are often values associated with a unit of measurement, a minimum and maximum value. If existing, this information is captured in tagged values of classes for the attribute type, which are subtypes of the respective ISO/TS 19103 basic type and where the <<BasicType>> stereotype indicates a canonical representation in XML Schema as a simple type. The XML Schema or GML simple type to be used is stored in the tagged value “base”, in case of a string value, the maximum length is stored in a tagged value “length”. Tagged names “min” and “max” are used to store minumum and maximum values in case of a numeric type.


<<Type>>
Measure
(from Units of Measure)
<<Type>>
CharacterString
(from Text)
<<BasicType>>
GA031_BRG
<<BasicType>>
GA065_COL


Class Specification for GA065_COL
General
Detail
Operations
Attributes
Relations
Components
Nested
Files
ShapeChange
MOF
JCR
UML
Set:
default
Edit Set...
Model Properties
Name
Value
Source
profiles
Default
code
Default
base
string
Override
length
80
Override
min
Default
max
Default
name
Default
Override
Default
Revert
OK
Cancel
Apply
Browse
Help



In addition to the attributes specified in the NGA Feature Catalog, the following attributes were added to all feature types:

- "position": A set of geometry objects describing the spatial position of the feature. This attribute is not strictly required for features according to the NGA Feature Catalog; however, since all feature types are intended to be associated with a spatial location and to make this application schema directly usable, this property has been added.

- In addition, several optional feature-level metadata properties. These are often provided in current datasets and this information should not be lost in a GML representation:

    - "orig_source_date": Date/time of the original source that was used for the extraction/ingest/creation of a feature

    - "orig_source_info": Additonal information about source of the information

    - "originating_source": Identification of the source of the information

- "originator": Originator of the information
- "scale": Scale at which the information was captured

Note 1: In order to support simpler modelling schemes for application schemas that prohibit inheritance/derivation, namely the proposed Simple Feature Profile of GML (GML-SF), these common properties are not represented in an abstract supertype from which then all instantiable feature types are derived.

Note 2: Another simplification – for the reason of supporting GML-SF - is that it is not possible to determine the properties providing feature-level metadata as such. This will however be easier with GML 3.2 where properties can be tagged as such in the application schema. It remains to be seen, if this will be supported in a revised GML-SF , too.

In general, the feature type definitions of the NGA Feature Catalog Application Schema are fairly simple. All properties are simple valued and cannot occur multiple times, i.e. a feature type can be represented as a single table in a relational database.

7.2.2 Dictionaries

During the creation process, three GML dictionaries are created which can be used to map codes to long names and descriptions. These can be used by applications:

FeatureTypeDictionary.xml:

```
<Dictionary xmlns="http://www.opengis.net/gml" xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/gml
http://schemas.opengis.net/gml/3.1.1/base/gml.xsd" gml:id="NGAFeatureTypes">
   <description>The list of feature types in the NGA Feature Catalog</description>
   <name>NGA Feature Types</name>
   <dictionaryEntry>
      <Definition gml:id="AA010">
         <description>An excavation made in the earth for the purpose of extracting
natural deposits.  (See also AQ090)</description>
         <name>Mine</name>
      </Definition>
   </dictionaryEntry>
   <dictionaryEntry>
      <Definition gml:id="AA011">
         <description>The wall facing of the excavation within a
quarry/mine.</description>
         <name>Quarry/Mine Shear Wall</name>
      </Definition>
   </dictionaryEntry>
   ...
</Dictionary>
```

AttributeTypeDictionary.xml:

```
<Dictionary xmlns="http://www.opengis.net/gml" xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/gml
http://schemas.opengis.net/gml/3.1.1/base/gml.xsd" gml:id="NGAAttributeTypes">
   <description>The list of attribute types in the NGA Feature Catalog</description>
   <name>NGA Attribute Types</name>
   <dictionaryEntry>
```

```
      <Definition gml:id="ACC">
         <description>Accuracy of geographic position.</description>
         <name>Accuracy Category</name>
      </Definition>
   </dictionaryEntry>
   <dictionaryEntry>
      <Definition gml:id="AOO">
         <description>The angular distance measured from true north (0 deg) clockwise to
the major axis of the feature. If the feature is square, the axis 0 through 89 deg shall
be recorded. If the feature is circular, 360 deg shall be recorded.</description>
         <name>Angle of Orientation</name>
      </Definition>
   </dictionaryEntry>
   …
</Dictionary>
```

CodeValueDictionary.xml:

```
<Dictionary xmlns="http://www.opengis.net/gml" xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/gml
http://schemas.opengis.net/gml/3.1.1/base/gml.xsd" gml:id="NGACodeValues">
   <description>The list of coded values in the NGA Feature Catalog</description>
   <name>NGA Coded Values</name>
   <dictionaryEntry>
      <Definition gml:id="ACC_1">
         <description><![CDATA[Accurate]]></description>
         <name codeSpace="http://www.opengis.net/ows3/nga/attributes/ACC">1</name>
      </Definition>
   </dictionaryEntry>
   <dictionaryEntry>
      <Definition gml:id="ACC_2">
         <description><![CDATA[Approximate]]></description>
         <name codeSpace="http://www.opengis.net/ows3/nga/attributes/ACC">2</name>
      </Definition>
   </dictionaryEntry>
   <dictionaryEntry>
      <Definition gml:id="ACC_3">
         <description><![CDATA[Doubtful]]></description>
         <name codeSpace="http://www.opengis.net/ows3/nga/attributes/ACC">3</name>
      </Definition>
   </dictionaryEntry>
   <dictionaryEntry>
      <Definition gml:id="ACC_997">
         <description><![CDATA[Unpopulated]]></description>
         <name codeSpace="http://www.opengis.net/ows3/nga/attributes/ACC">997</name>
      </Definition>
   </dictionaryEntry>
   <dictionaryEntry>
      <Definition gml:id="ACC_998">
         <description><![CDATA[Not Applicable]]></description>
         <name codeSpace="http://www.opengis.net/ows3/nga/attributes/ACC">998</name>
      </Definition>
   </dictionaryEntry>
   <dictionaryEntry>
      <Definition gml:id="ACC_999">
         <description><![CDATA[Other]]></description>
         <name codeSpace="http://www.opengis.net/ows3/nga/attributes/ACC">999</name>
      </Definition>
   </dictionaryEntry>
   …
</Dictionary>
```

In addition, a units dictionary has been used, based on the dictionary created for OWS-2:

UnitsDictionary.xml:

```
<Dictionary xmlns="http://www.opengis.net/gml" xmlns:gml="http://www.opengis.net/gml"
```

```
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xlink="http://www.w3.org/1999/xlink" xsi:schemaLocation="http://www.opengis.net/gml
http://schemas.opengis.net/gml/3.1.1/base/gml.xsd" gml:id="UnitsDictionary">
   <description>A dictionary of units of measure for the NGA FACC Profile for OWS-
2.</description>
   <name>NGA FACC Profile Units Dictionary</name>
   <dictionaryEntry>
      <Dictionary gml:id="SIBaseUnits">
         <description>The Base Units from the SI units system.</description>
         <name>SI Base Units</name>
         <dictionaryEntry>
            <BaseUnit gml:id="m">
               <description>The metre is the length of the path travelled by light in
vacuum during a time interval of 1/299 792 458 of a second.</description>
               <name codeSpace="http://www.bipm.fr/en/si">metre</name>
               <name>meter</name>
               <quantityType>length</quantityType>
               <catalogSymbol codeSpace="http://www.bipm.fr/en/si">m</catalogSymbol>
               <unitsSystem xlink:href="http://www.bipm.fr/en/si"/>
            </BaseUnit>
         </dictionaryEntry>
         …
      </Dictionary>
   </dictionaryEntry>
   <dictionaryEntry>
      <Dictionary gml:id="SIDerivedUnits">
         <description>The Derived Units from the SI units system. These are all derived
as a product of SI Base Units, except degrees Celsius in which the conversion formaula to
the SI Base Unit (kelvin) involves an offset.</description>
         <name>SI Derived Units</name>
         <dictionaryEntry>
            <DerivedUnit gml:id="rad">
               <name codeSpace="http://www.bipm.fr/en/si"> radian</name>
               <quantityType>plane angle</quantityType>
               <catalogSymbol codeSpace="http://www.bipm.fr/en/si">rad</catalogSymbol>
               <derivationUnitTerm uom="#m" exponent="1"/>
               <derivationUnitTerm uom="#m" exponent="-1"/>
            </DerivedUnit>
         </dictionaryEntry>
         …
      </Dictionary>
   </dictionaryEntry>
   <dictionaryEntry>
      <Dictionary gml:id="ConventionalUnitsDictionary">
         <description>A collection of Conventional Units.  These are units of measure
which are either widely used or important within a specific community.  For most of these
there is
1. a known derivation from more primitive units, which may or may not be SI Base Units,
or
2. a known conversion to a preferred unit, which may or may not be an SI Base or Derived
unit, through rescaling and offset,
or both.  </description>
         <name>Conventional units.</name>
         <dictionaryEntry>
            <ConventionalUnit gml:id="GHz">
               <name>GigaHertz</name>
               <quantityType>frequency</quantityType>
               <catalogSymbol>GHz</catalogSymbol>
               <conversionToPreferredUnit uom="#Hz">
                  <factor>1.e9</factor>
               </conversionToPreferredUnit>
            </ConventionalUnit>
         </dictionaryEntry>
         …
      </Dictionary>
   </dictionaryEntry>
   <dictionaryEntry>
      <Dictionary gml:id="UnitlessUnits">
         <description>A collection of unitless units.</description>
         <name>unitless units</name>
         <dictionaryEntry>
```

```
            <BaseUnit gml:id="unity">
               <name>unitless</name>
               <quantityType>scale factor</quantityType>
               <unitsSystem xlink:href="http://www.opengis.net/ows-2/nga"/>
            </BaseUnit>
         </dictionaryEntry>
         …
      </Dictionary>
   </dictionaryEntry>
</Dictionary>
```

7.2.3 GML Application Schema

7.2.3.1 Overview

The GML Application Schema is automatically derived from the UML application schema using the ShapeChange UGAS (UML-to-GML-Application-Schema conversion) tool. This can be done in two ways. First, by using the command line interface which requires that the UML model is already available. Second, using the web interface and its integrated Catalog Service client by discovering the previously unknown model by querying a registry.

7.2.3.2 Using the ShapeChange command line interface

After exporting the UML model as an XMI 1.0 file "ngafc.xml", the ShapeChange tool is executed with the following parameters:

```
java -Xms128m -Xmx1024m -jar ShapeChange.jar -C -D -E -M "OWS3" -
o "out" "xmi\ngafc.xml"
```

See the ShapeChange documentation for details about the parameters.

As a result of the operation, the GML Application Schema NGAFC.xsd is created.

ShapeChange reports the following warnings which all refer to model elements from the ISO 19100 model and not from the NGA Feature Catalog:

```
<ShapeChangeResult xmlns="http://www.interactive-instruments.de/ShapeChange"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" resultCode="0"
xsi:schemaLocation="http://www.interactive-instruments.de/ShapeChange
http://www.interactive-instruments.de/ShapeChange/ShapeChangeResult.xsd">
   <Messages>
      <Warning>Stereotype &lt;&lt;MetaClass&gt;&gt; not supported for UML model elements
of type Class. The Class will be ignored.</Warning>
      <Warning>Stereotype &lt;&lt;bind&gt;&gt; not supported for UML model elements of
type Association. The Association will be ignored.</Warning>
      <Warning>Stereotype &lt;&lt;parameter&gt;&gt; not supported for UML model elements
of type Class. The Class will be ignored.</Warning>
      <Warning>Stereotype &lt;&lt;normative&gt;&gt; not supported for UML model elements
of type Package. The Package will be ignored.</Warning>
      <Warning>Stereotype &lt;&lt;normative&gt;&gt; not supported for UML model elements
of type Package. The Package will be ignored.</Warning>
      <Warning>Stereotype &lt;&lt;Complete&gt;&gt; not supported for UML model elements
of type Generalization. The Generalization will be ignored.</Warning>
      <Warning>Stereotype &lt;&lt;realize&gt;&gt; not supported for UML model elements of
type Abstraction. The Abstraction will be ignored.</Warning>
      <Warning>The multiplicity value of 'size' is neither a number nor a known string.
'*' is used instead.</Warning>
      <Warning>The multiplicity value of 'dimension' is neither a number nor a known
```

```
string. '*' is used instead.</Warning>
   </Messages>
   <XSDFiles>
      <XSDFile appSchema="true" appSchemaId="S.206.1217.37.129"
href="out/NGAFC.xsd">NGAFC.xsd</XSDFile>
   </XSDFiles>
</ShapeChangeResult>
```

The GML Application Schema has been verified with Xerces.

7.2.3.3 Using the ShapeChange web interface and Catalog Service client

This alternative path uses the same approach as in the previous subclause, however, the steps are exectuted using a graphical, webbrowser-based user interface.

In addition, UML models published in Catalog Services registries following the guidelines for schema metadata specified in Clause 7 can be discovered through that interface. After the GML Application Schema has been created, the document is uploaded to a http-accesible ftp repository and the newly created schema can be published published in a Catalog Service registry following the guidelines for schema metadata specified in Clause 7.

See the ShapeChange documentation for details about using the web interface and the Catalog Service client.

7.2.4 Known Issues

7.2.4.1 Conformance with the proposed GML Profile for Simple Features

The application schemas in this initiative have been constructed with the goal to meet the requirements of the proposed GML Profile for Simple Features (GML-SF; the public comment period for this draft profile took place during the OWS-3 initiative) whenever possible. With a few exceptions this has been possible.

- Inheritance was avoided. However it is assumed that this will not be possible in general. It should also be noted that this restriction results in a loss of expressiveness in the model as common properties have to be copied to every feature type instead of capturing them once in an abstract supertype.

- All numeric properties associated with a unit of measurement have been mapped to "gml:MeasureType" instead of using the more specific types like "gml:LengthType", because the general type is allowed in the profile, while the more specific types are not.

  The allowed unit of measurement is fixed by the FACC for a specific attribute. Therefore the mandatory "uom" attribute value is practically fixed for every property (although it has to be provided explicitly in every instance). The XML Schema rules unfortunately do not allow to fix the value in the schema in a way so that it does not need to be repeated in every instance of a property.

- There was a discussion whether the profile rules have to be interpreted in a way that the element of a feature type has to be directly preceeding the associated complex type definition. This was not the case in some OWS-3 schema documents and has been corrected. In the meantime the discussion in the GML-SF Revision Working Group has clarified that such a rule was not intended.

- While GML-SF supports the use of "gml:CodeType" it does so only for references to (well-known) code lists. Restrictions of gml:CodeType are not allowed.

  As a result of the discussion that it would desirable to be able to specify the namespace (or the naming authority) of certain name attributes, e.g. "nam", gml:CodeType was used to provide that functionality. It could be argued that this is beyond the scope of where gml:CodeTyoe is allowed in GML-SF: GML-SF promotes that the dictionary of the controlled values is specified in the schema. In the NGA Feature Catalog GML Application Schema the gml:CodeType is used for values that are names specified by a naming authority, but this naming authority is provided in the instances and not in the schema itself (since there is no single naming authority for the name attributes in NGA data) as proposed by the GML-SF specification.

  Also, in GML-SF it is not allowed to restrict the length of such an attribute value.

  As a consequence, in a GML-SF version of the schema the type would probably need to be replaced by xs:string.

  No other aspects are known, where the NGA Feature Catalog Application Schema is not conformant with GML-SF.

At the same time is has to be noted that according to discussions it can be expected that in the future NGA application schemas will become more complex and will make use of inheritance and other modelling language constructs that cannot be represented with the proposed Simple Feature GML Profile.

7.2.4.2 NUNANPO

DIGEST specifies rules how certain special values are to be encoded using reserved values within the domain of the different data types. NGA has extended support for these so-called NUNANPO values (Null, Unknown, Not Applicable, Not Populated, Other).

This has to be taken into account, when ranges for values are restricted. For example, some attribute values are restricted to values between, say 0 and 359.9. In order to support the NUNANPO values which are using large, negative values restrictions for minimum values have been removed for such attributes.

An example. Assuming that a feature attribute in the context of a feature type in the application schema is specified as:

```
<element name="aProperty" minOccurs="0">
 <simpleType>
  <restriction base="integer">
   <minInclusive value="-1200"/>
   <maxInclusive value="10000"/>
  </restriction>
 </simpleType>
</element>
```

Mimicing the full NUNANPO values (in the range of –32768 for "null" to –32764 for "other") as they are currently used by NGA one would need to change the type definiton to[1]

```
<element name="aProperty" minOccurs="0">
 <simpleType>
  <union>
   <simpleType>
    <restriction base="integer">
     <minInclusive value="-1200"/>
     <maxInclusive value="10000"/>
    </restriction>
   </simpleType>
   <simpleType>
    <restriction base="integer">
     <minInclusive value="-32678"/>
     <maxInclusive value="-32674"/>
    </restriction>
   </simpleType>
  </union>
 </simpleType>
</element>
```

As this restriction is not supported by GML-SF, the property definition would be changed to collapse the multiple ranges (accepting that this schema would validate inccorect values) to:

```
<element name="aProperty" minOccurs="0">
 <simpleType>
  <restriction base="integer">
   <minInclusive value="-32768"/>
   <maxInclusive value="10000"/>
  </restriction>
 </simpleType>
</element>
```

This is the representation which has been choosen for OWS-3 to ease the encoding of the existing test data into GML data.

This approach works for value types that are numbers, but fails for NUNANPO values for date-valued properties.

It has to be noted that representing this NUNANPO values in GML by reducing the value domain of the data types by a few values and associating these with a specific meaning is a "hack". This is particularily true, if an implementation offers another solution for the NUNANPO requirement. GML does this, and so the GML way of doing this would be – using the most recent, to-be-published version GML 3.2.0 = ISO/DIS 19136 as this

---

[1] Note that this representation may not only be useful for NUNANPO representations, but also where the "true" values may be different disjunct intervals.

mechanism has been changed in this version – the following:

Define the property element representing the feature attribute as "nillable" and add the pre-defined nilReason attribute from GML:

```
<simpleType name="APropertyType">
  <restriction base="integer">
   <minInclusive value="-1200"/>
   <maxInclusive value="10000"/>
  </restriction>
</simpleType>

<element name="aProperty" minOccurs="0" nillable="true">
 <complexType>
  <simpleType>
   <extension base="ngafc:APropertyType">
    <attribute ref="gml:nilReason"/>
   </extension>
  </simpleType>
 </complexType>
</element>
```

nilReason allows the following values

- "inapplicable" (there is no value)
- "missing" (the correct value is not readily available to the sender of this data; furthermore, a correct value may not exist)
- "template" (the value will be available later)
- "unknown" (the correct value is not known to, and not computable by, the sender of this data; however, a correct value probably exists)
- "withheld" (the value is not divulged)
- "other:…" where "…" is a string of two or more characters with no included spaces (other brief explanation)
- anyURI (which should refer to a resource which describes the reason for the exception)

So, if a value is populated (i.e. "Value Specified") then this is encoded as

```
<XY000>
   …
   <aProperty>1000</aProperty>
   …
</XY000>
```

If the value is, for example, "Not applicable", then this would be encoded as

```
<XY000>
   …
   <aProperty xsi:nil="true" nilReason="inapplicable"/>
   …
</XY000>
```

If done consistently then this would also be applied to the coded values where currently the commonly used values –32768, 0, 997, 998, and 999 are representing the NUNANPO values.

A possible mapping of the NUNANPO values to the predefined nilReason values of GML could be:

"Null" → nilReason missing

"Unknown" → nilReason="unknown"

"Not Applicable" → nilReason="inapplicable"

"Not Populated" → nilReason="missing"

"Other" → nilReason="other:other"

While this makes the data type of the attribute a complex type, it still works as a simple content type and does not make the encoding more bulky.

Instead of using the pre-defined enumerants, a URI could be used to reference a definition in a dictionary (possibly held in a registry), if this would provide additional benefits. For example:

```
<XY000>
   …
   <aProperty xsi:nil="true" nilReason="http://www.nga.mil/dfdd/nunanpo.xml#Unknown"/>
   …
</XY000>
```

The approach to use XML attributes to encode metadata for information that is encoded as an XML element (which includes feature properties) is also used by the security/releasability information in the IC ISM specification.

This approach, however, has its limitations whenever the metadata properties are either complex or may occur multiple times. This cannot be represented in XML attributes and requires that the values become complex types with complex content.

7.2.4.3 Properties with complex values

It has been noted in the discussions about the NGA Feature Catalog(s) and the corresponding application schemas that in the future it is expected that future DFDD-based feature catalogs will include feature attributes with complex values. If this happens, ISO 19109 application schemas as well as GML are designed to being able to express such complex valued properties and, therefore, are in principle capable of representing such values.

7.2.4.4 Enumerations

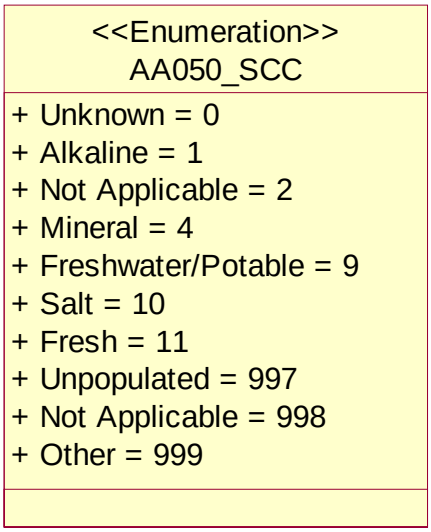
The FACC contains – for legacy reasons – in some lists of coded values multiple codes

for the same meaning. Only one of the values is supposed to be used in feature catalogues, however, the NGA Feature Catalog also includes deprecated code values and a number of enumerations contain the same enumerant twice with different code values. This seems to be an error in the NGA Feature Catalog.

These multiple values create inconsistencies in the UML model as they result in multiple class attributes with the same name which is not allowed in UML.

As a result, the XMI document exported from the UML model maps both values to the same code.

Example: AA050_SCC::Not Applicable



If we look at the "scc" property in feature type "AA050" we find:

```
<element name="AA050" substitutionGroup="gml:_Feature" type="NGAFC:AA050Type"/>
<complexType name="AA050Type">
 ...
 <complexContent>
  <extension base="gml:AbstractFeatureType">
   <sequence>
    ...
    <element minOccurs="0" name="scc">
     ...
     <simpleType>
      ...
      <restriction base="string">
       ...
       <enumeration value="2">
        <annotation>
         <documentation>Not Applicable</documentation>
         ...
        </annotation>
       </enumeration>
       ...
       <enumeration value="997">
        <annotation>
         <documentation>Unpopulated</documentation>
         ...
```

```
        </annotation>
       </enumeration>
       <enumeration value="2">
        <annotation>
         <documentation>Not Applicable</documentation>
         ...
        </annotation>
       </enumeration>
       <enumeration value="999">
        <annotation>
         <documentation>Other</documentation>
         ...
        </annotation>
       </enumeration>
      </restriction>
     </simpleType>
    </element>
    ...
   </sequence>
  </extension>
 </complexContent>
</complexType>
```

There are two enumerations with "2" and "not applicable". The second "2" would need to be changed to "998" as can be identified by inspecting the "Attribute values" table in the FACC.

ATT_CODE	ATT_VAL	ATT_VAL_DESCRIPTION
SCC	000	Unknown
...		
SCC	002	Not Applicable
...		
SCC	997	Unpopulated
SCC	998	Not Applicable
SCC	999	Other

These issues have been corrected with the Schema Assembly Tool (see Clause 8).

An alterative way to deal with these issues would be to change the Visual Basic script creating the UML model. This, however, would require either a name change in the text one of the two enumerants or an algorithm to determine the value that is outdated and to be dropped.

It is recommended that these deprecated enumerants are removed from the NGA Feature Cataogue.

7.2.4.5 Non-linear ranges of allowed values

The attributes "mhh" and "mhl" use a non-linear range of allowed values (MIN 180.5, MAX 0.5) which creates a problem in the validation as no value does fit both requirements and it is unclear how the cyclic nature of the values (and the length of a period) could be determined automatically. As a simple fix the range for these two attributes is ignored. In the long run, either the concept of periodic alues needs to be introduced or the values in the FACC need to be adapted.

7.2.4.6 Moire

The feature types BC020 and CB040 have a property called 'cha' that has an enumerated value 43 which has the description "Directional Moiré".

There was a known problem in ShapeChange that if the XMI file is not in UTF-8 (Rose exports the NGA files in an ISO-8859-1 character encoding), special characters may create encoding errors in the generated GML Application Schema. This error was fixed by creating the GML Application Schema in the same encoding as the original XMI file.

7.2.4.7 Multiple imports of Xlink schemas from different locations

In early drafts of the schema, XML validation reported errors which were traced back to the fact that the Xlink namespace was imported multiple times, but with different schema locations. This obviously confuses some XML parsers, so the schema documents were changed to avoid this behaviour.

This is a general issue also noted by others at the same time and a change proposal for a reorganisation of http://schemas.opengis.net/ has already been submitted.

7.2.4.8 Categorisations of feature types

In the schema metadata for the individual feature types, each feature type should be categorized according to the individually fitting classifications. However, while this is possible for the MSDx classification scheme (same classification as the schema) and the DGIWG/FACC classification scheme (every feature type is assigned to a category and this is expressed in the first letter of the name of the feature type), this is not always evident for the ISO 19115/MD_TopicCategory classification scheme. Such assingments have not been done as part of this initiative, but since ISO 19115 metadata plays an important role for all types of geographic information metadata, it would be beneficial, if every organisation responsible for the description of a conceptual feature type (be it in a data dictionary, a feature catalogue or application schema) would provide such a categorisation.

7.3 Profile of ISO 19107

7.3.1 DGIWG Profile

The spatial schema to be used in connecion with NGA Feature Catalog feature types is restricted to the profile described by the DGIWG/TSMAD Profile of ISO 19107.

Note that this profile does not include geometric aggregates, therefore, in the application schema the data type of spatial geometries are in general specified as "Set<GM_(Orientable)XYZ>" instead of "GM_MultiXYZ".

7.3.2 GML Profile

The DGIWG Profile goes beyond the spatial geometry concepts implemented by GML-SF. Required extensions include:

- Composite geometries
- Non-linear interpolation types
- Topologies

Thus, a GML Profile[2] was specified to be used in conjunction with the application schemas described in this document. This was based on the GML Profile described in OGC document 04-100 (NGA Application Schema deveopment).

The resulting GML profile is captured in the schema document “gml3NGAProfile.xsd”. It is restricted to the GML schema components that are required by the GML application schemas in this DIPR.

The use of topology in such features is discussed in the GML Topology Investigations DIPR (OGC document 05-112).

In GML, the "Set<GM_(Orientable)XYZ>" values are implemented by gml:MultiXYZ objects.

7.4 Application Schemas for MSD Level 1-3

7.4.1 ISO 19109 Application Schema in UML

The MSD application schemas are created from the NGA Feature Catalog application schema and the MSD MS Access database (MSD_Profiles.mdb, using the master table

---

[2] Since the GML 3.1 schema offers a wide range of components that may be used by geospatial application schemas, almost all application schemas will use only a very limited part of the full GML schema. To specify and document the usage of GML by an application schema typically a profile of the GML schema is created and used instead of the full schema.

for Levels 1, 2 and 3).

The contents of the table is shown below:

Coverage	Theme	FAC_CODE	ATT_CODE	ATT_VAL	Area	Line	Point	Text	Default Value	Priority	Comments	Pilot?	Prototype?
atn		GA035	AAI	12	0	0	-1	0				Nein	Nein
atn		GA035	AAI	13	0	0	-1	0				Nein	Nein
atn		GA035	AAI	14	0	0	-1	0				Nein	Nein
atn		GA035	AAI	15	0	0	-1	0				Nein	Nein
atn		GA035	AAI	16	0	0	-1	0				Nein	Nein
...													

In order to process the input database the Visual Basic program described above was enhanced to implement the following steps:

- A root package for every MSD application schema (stereotype <<Application Schema>>). Pre-defined, tagged values were added to this package to associate the application schema with a namespace (e.g. “http://www.opengis.net/ows-3/nga/MSD1”), a namespace prefix (“MSD1”) and a version number; see the ShapeChange documentation for details.

- For every feature type in the profile, one or more feature type classes are created based on the dictionary in the NGA Feature Catalog. One class is created for every NGA Feature Catalog feature type per spatial dimension. Therefore, a maximum of four feature types are possible per feature type in the NGA Feature Catalog: surface (2D), curve (1D), point (0D) or text (0D + text). To distinguish the subtypes, a prefix is added to the name of the feature types, e.g. “A” for a feature with 2D geometry, “L” for 1D, “P” for 0D, and "T" is used for texts.

  An alternative approach would have been to keep one single feature type as in the NGA Feature Catalog, but to define a single spatial property that allows one of the four spatial representations. The reason to stick to the splitting into separate feature types on the MSD level for now are similar to the discussion of the use of codes vs names above. The basic reasons are:

  - The current definition of the MSD level does make the disctinction.

  - In addition, the allowed attributes and attribute values sometimes do depend on

the spatial dimension.

It is recommended to validate this decision in the future, i.e. check whether there are actually good reasons for another approach. Test cases could include:

- Does the encoding of SLDs get significantly easier or more complicated?

- Does the encoding of queries using standard WFS clients get significantly easier or more complicated? For example, is it a problem that he has to query multiple feature types if he wants to select all buildings? Does he want to select features of a feature type based on the sptial dimension? Etc.

- Is there any loss of information in the schemas or instances?

- Is the user interested in selecting only feature instances of the certain spatial dimension?

See Annex A for an overview of the alternatives.

It should be noted that the NGA Feature Catalog (which does not restrict the spatial representation of a feature in any way and does not mix feature classification and spatial characteristics) as a potential basis for a conflated representation of data is different from the MSD point of view. MSD data is usually created with a specific view in terms of detail/scale and themes. Therefore feature instances encoded according to a MSD application schema are views of the underlying feature that may and typically will contain additional information not relevant for the specific scope of the MSD. That is, there will usually be a (distributed) representation of the feature at NGA that would be the most complete and detailed representation that is available to NGA.

Feature Access services typically allow to select the information of interest (spatial and non-spatial properties) of a feature for the particular use case. In this sense, a MSD schema is a manifestation of such a collection of feature types and properties that are of interest for a particular use.

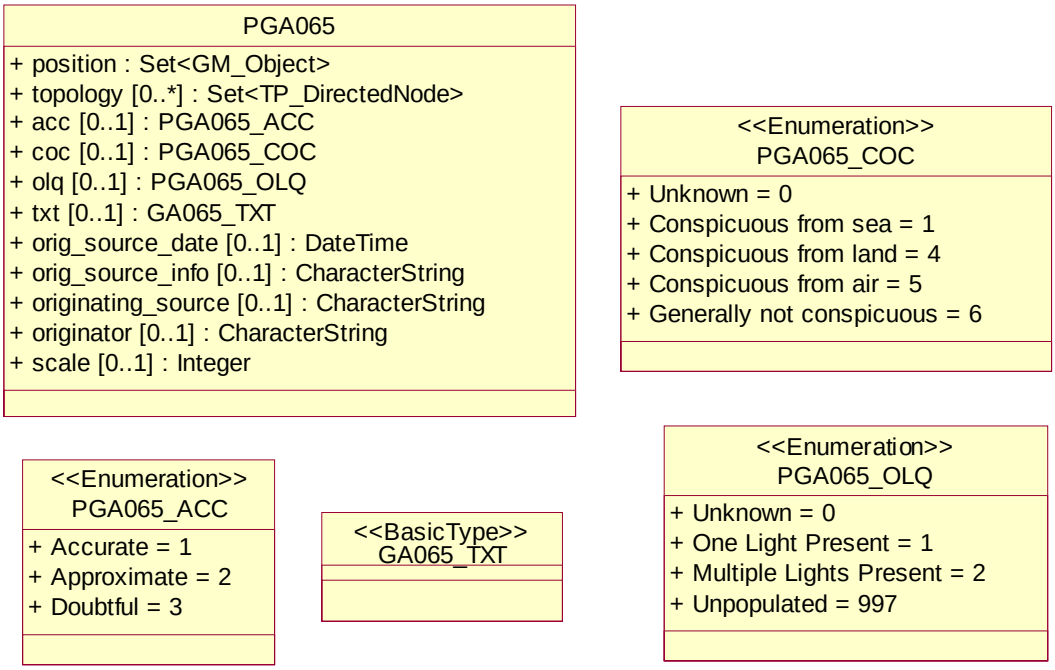
The current MSD descriptions do not require (or support) multiple spatial representations for the same feature instance (e.g. the areal river used for sailing-the-boat and the linear river centerline for planning the route). While data sets will usually have different geometry types on instances of the same feature type (e.g., some Building instances may be Points and some Building instances may be Areas), but no single feature instance will have more than one spatial representation. Therefore, this is not supported in the MSD schemas derived from the current MSD descriptions. That said, it would be possible (and easy) to add such multiple representations to feature types in a MSD schema if such a requirement is identified.

In addition, it is also possible to create explicit associations between different feature instances relating the features to each other. Note that currently explicit feature associations are not used in the NGA schemas and are therefore not part of the

schemas derived from the Access databases.

The rules for application schemas (ISO 19109) explicitly support all such modelling patterns (the same is true for GML / ISO 19136).

- Basic types are directly used from the NGA Feature Catalog and coded values are restricted to the values defined in the profile for the particular spatial dimension.



Thus, the new feature type classes are specific implementations of the feature types defined in the NGA Feature Catalog.

7.4.2 Coverages and Themes

The MSD feature catalogs associate feature types, attribute types and coded values with so called coverages and themes. Themes are not relevant any longer, but the concept of coverages – although under discussion in NGA and DGIWG – was decided to be relevant for OWS-3.

In general, a coverage represents a "layer", where the geometry/topology of the layer can be expressed as a geometry/topology complex, i.e. the geometry of any feature in a coverage is part of a geometry complex representing the coverage. As a result, the coverage concept could therefore be represented in the data by using geometry or topology complexes. However, this was not choosen for the OWS-3 application schemas for the following reasons:

- Topology information is modelled as optional, based on the results of the OWS-2 initiative and pending the results of the OWS-3 GML investigantions on topology

(see OGC document 05-112). In fact, for most applications interested only in the features and their geometrical position, requiring them to deal with complexes is any unnecessary burden.

- Such an approach would not work with GML-SF as this profile does not support complexes.

Any feature instance may be part of multiple layers – although this will in general be realized by cloning the feature instance.

Whether a particular instance is part of a coverage or not depends only on its properties and their values, it does not involve any decision in the data capturing process. Thus, it is not appropriate to explicitly represent this information in the dataset (for example, by using explicit feature collections to represent coverages or by adding a property to all feature types). The information, to which coverage a feature belongs, can be derived at any time and may not be altered.

The stated requirement by NGA is, that it should be possible to "pull" data based on a coverage, it is not intended to store data based on a coverage. Since the GML representation is intended to be used to transfer such "pull" requests of data an approach was required supporting this.

This is solved by the following mechanism: As the property pattern of a feature determines whether a feature belongs to a profile or not, a Web Feature Service GetFeature request document per coverage was created by the ShapeChange UGAS tool[3]. Submitting one of these requests to a Web Feature Service hosting data according to one of the GML Application Schemas will return all features in that particular coverage.

The following is a GetFeature operation for a very simple coverage in MSD3, the "slp" coverage:

```
<GetFeature xmlns:MSD3="http://www.opengis.net/ows-3/nga/MSD3"
xmlns:fes="http://www.opengis.net/ogc" xmlns:wfs="http://www.opengis.net/wfs"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.opengis.net/wfs"
service="WFS" version="1.0.0" xsi:schemaLocation="http://www.opengis.net/wfs
http://schemas.opengis.net/wfs/1.0.0/WFS-basic.xsd">
   <Query typeName="MSD3:ASA050">
      <Filter xmlns="http://www.opengis.net/ogc">
         <Or>
            <PropertyIsEqualTo>
               <PropertyName>MSD3:spr</PropertyName>
               <Literal>0</Literal>
            </PropertyIsEqualTo>
            <PropertyIsEqualTo>
               <PropertyName>MSD3:spr</PropertyName>
               <Literal>1</Literal>
            </PropertyIsEqualTo>
            <PropertyIsEqualTo>
```

[3] The information about the property pattern of feature types within the different coverages as expressed in the table specifying the feature catalog are stored in tagged values in the UML model. These tagged values are evaluated by ShapeChange.

```
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>2</Literal>
         </PropertyIsEqualTo>
         <PropertyIsEqualTo>
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>3</Literal>
         </PropertyIsEqualTo>
         <PropertyIsEqualTo>
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>4</Literal>
         </PropertyIsEqualTo>
         <PropertyIsEqualTo>
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>5</Literal>
         </PropertyIsEqualTo>
         <PropertyIsEqualTo>
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>6</Literal>
         </PropertyIsEqualTo>
         <PropertyIsEqualTo>
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>7</Literal>
         </PropertyIsEqualTo>
         <PropertyIsEqualTo>
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>8</Literal>
         </PropertyIsEqualTo>
         <PropertyIsEqualTo>
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>9</Literal>
         </PropertyIsEqualTo>
         <PropertyIsEqualTo>
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>10</Literal>
         </PropertyIsEqualTo>
         <PropertyIsEqualTo>
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>997</Literal>
         </PropertyIsEqualTo>
         <PropertyIsEqualTo>
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>998</Literal>
         </PropertyIsEqualTo>
         <PropertyIsEqualTo>
            <PropertyName>MSD3:spr</PropertyName>
            <Literal>999</Literal>
         </PropertyIsEqualTo>
      </Or>
   </Filter>
</Query>
<Query typeName="MSD3:PZD019">
   <Filter xmlns="http://www.opengis.net/ogc">
      <Or>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:acp</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:hgp</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:leg</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:nam</PropertyName>
```

```
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:nfi</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:nfn</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:txt</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:wgp</PropertyName>
            </PropertyIsNull>
         </Not>
      </Or>
   </Filter>
</Query>
<Query typeName="MSD3:LZD019">
   <Filter xmlns="http://www.opengis.net/ogc">
      <Or>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:acp</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:hgp</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:leg</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:nam</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:nfi</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:nfn</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:txt</PropertyName>
            </PropertyIsNull>
         </Not>
         <Not>
            <PropertyIsNull>
               <PropertyName>MSD3:wgp</PropertyName>
            </PropertyIsNull>
         </Not>
      </Or>
   </Filter>
</Query>
```

```
   <Query typeName="MSD3:AZD019">
      <Filter xmlns="http://www.opengis.net/ogc">
         <Or>
            <Not>
               <PropertyIsNull>
                  <PropertyName>MSD3:acp</PropertyName>
               </PropertyIsNull>
            </Not>
            <Not>
               <PropertyIsNull>
                  <PropertyName>MSD3:hgp</PropertyName>
               </PropertyIsNull>
            </Not>
            <Not>
               <PropertyIsNull>
                  <PropertyName>MSD3:leg</PropertyName>
               </PropertyIsNull>
            </Not>
            <Not>
               <PropertyIsNull>
                  <PropertyName>MSD3:nam</PropertyName>
               </PropertyIsNull>
            </Not>
            <Not>
               <PropertyIsNull>
                  <PropertyName>MSD3:nfi</PropertyName>
               </PropertyIsNull>
            </Not>
            <Not>
               <PropertyIsNull>
                  <PropertyName>MSD3:nfn</PropertyName>
               </PropertyIsNull>
            </Not>
            <Not>
               <PropertyIsNull>
                  <PropertyName>MSD3:txt</PropertyName>
               </PropertyIsNull>
            </Not>
            <Not>
               <PropertyIsNull>
                  <PropertyName>MSD3:wgp</PropertyName>
               </PropertyIsNull>
            </Not>
         </Or>
      </Filter>
   </Query>
</GetFeature>
```

7.4.3 Flattened schemas

The NGA Feature Catalog and the MSD schemas are "flat" in the sense that all feature types are more or less independent of each other. There are no explicit feature associations defined (only the implicit spatial relationship) and there are no semantic hierarchies except for the loose grouping of feature types into categories, sub-categories, coverages and themes.

For the application schemas no supertype has currently been defined although it would be helpful to avoid copying some common properties to all the instantiable feature types. This has been done in order to allow a simpler mapping to GML-SF. In general, semantic information is lost by flattening a type hierarchy. Since the NGA Feature Catalog does not contain a well-developed hierarchy, this loss can be considered acceptable, considering the potential gains assuming that the software vendors will support the GML-SF Profile in their products soon. It should be takien into account, too, that the MSD

schema is mostly designed for data exchange.

If, however, the conceptual model requires a non-flattend schema, then it will be required to use such a schema instead. It is expected that this will occur in the future and that future application schemas will need to deal with the following:

- a hierarchy of classes (including abstract classes) making use of attribute inheritance (including, for example, intial values);

- collections of parts belonging to or being associated with a complex feature (e.g., an aerodrome and all of its "associated features");

- feature associations that deal with a variety of semantics explicitly on the feature level including adjacency, control, connectivity, and over/under;

- complex values of attributes (e.g., a 1D vector of temperature vs. depth or surface reflectivity vs. frequency);

- properties with a multiplicities greater than one (e.g., the sectors of a beacon, where each sector is a complex value and there are many-per-beacon).

All of these requirements are supported by the general approach using ISO 19109 (and ISO 19136 for the XML encoding), therefore it is expected that the general approach outlined above will also work if the requirements change in the above mentioned way.

7.4.4 Other known issues

Some inconsistencies in the MSD master table were identified. Not all possible combinations of NGA Feature Catalog feature types with spatial dimensions were specified. This was solved by taking also the inclusion conditions into account.

Also, some differences between the MSD definition and the test data were identified (usually coded values that are not allowed according to the MSD definition). Alternate values for the invalid values were identified in close co-operation between Intergraph (encoding the test data in GML) and NGA.

7.4.5 GML Application Schema

The MSD1, MSD2, and MSD3 GML Application Schemas are created the exact same way as the schema for the NGA Feature Catalog was created.

8 Manipulating GML Application Schemas with the Schema Assembly Tool

After the schemas had been created by the ShapeChange UGAS tool, two additional changes discussed above have been applied in the schemas using the Schema Assembly Tool (see OGC document 05-121) to create the final GML Application Schemas:

- The constraining facets of the attributes "mhh" and "mhl" have been removed.
- The code values in the cases of identical enumerants with different code values have been fixed.

Annex A: Spatial representation in the GML instances and the effects on WFS queries

There are several alternatives for the encoding of the spatial property (or properties) in the instances. The examples use the oil/gas field feature type (AA052), which can carry either an area or a point as the value of its spatial property.

It was decided to use alternative 1b in the GML-SF compatible form in the OWS-3 schemas.

Alternative 1a: As in OWS-2 (simplified)

Two feature types are created, one for the point and one for the area representation. To allow primitives and aggregations as values alternatively, a choice of two properties is used.

Note that the point representation carries another property ("txt") which is not allowed for the area representation, but is mandatory for a point.

```
   <element name="AAA052" type="vmap0:AAA052Type" substitutionGroup="gml:_Feature"/>
   <complexType name="AAA052Type">
      <complexContent>
         <extension base="gml:AbstractFeatureType">
            <sequence>
               <choice>
                  <element name="surfaceGeometry" type="gml:SurfacePropertyType"/>
                  <element name="multiSurfaceGeometry"
type="gml:MultiSurfacePropertyType"/>
               </choice>
            </sequence>
         </extension>
      </complexContent>
   </complexType>

   <element name="PAA052" type="vmap0:PAA052Type" substitutionGroup="gml:_Feature"/>
   <complexType name="PAA052Type">
      <complexContent>
         <extension base="gml:AbstractFeatureType">
            <sequence>
               <choice>
                  <element name="pointGeometry" type="gml:PointPropertyType"/>
                  <element name="multiPointGeometry" type="gml:MultiPointPropertyType"/>
               </choice>
               <element name="txt">
                  <simpleType>
                     <restriction base="string">
                        <maxLength value="256"/>
                     </restriction>
                  </simpleType>
               </element>
            </sequence>
         </extension>
      </complexContent>
   </complexType>
```

Instance examples:

```
<AAA052 gml:id="_0">
```

```
   <surfaceGeometry>
      <gml:Surface/>
   </surfaceGeometry>
</AAA052>

<PAA052 gml:id="_1">
   <pointGeometry>
      <gml:Point/>
   </pointGeometry>
   </txt>My text …</txt>
</PAA052>
```

Alternative 1b: As in Alternative 1a, but with a single spatial property per feature

This representation was proposed at the end of OWS-2, because the separate properties were irritating in the user interface of the client:

```
   <element name="AAA052" type="vmap0:AAA052Type" substitutionGroup="gml:_Feature"/>
   <complexType name="AAA052Type">
      <complexContent>
         <extension base="gml:AbstractFeatureType">
            <sequence>
               <element name="position">
                  <complexType>
                     <choice>
                        <element ref="gml:_Surface"/>
                        <element ref="gml:MultiSurface"/>
                     </choice>
                  </complexType>
               </element>
            </sequence>
         </extension>
      </complexContent>
   </complexType>

   <element name="PAA052" type="vmap0:PAA052Type" substitutionGroup="gml:_Feature"/>
   <complexType name="PAA052Type">
      <complexContent>
         <extension base="gml:AbstractFeatureType">
            <sequence>
               <element name="position">
                  <complexType>
                     <choice>
                        <element ref="gml:Point"/>
                        <element ref="gml:MultiPoint"/>
                     </choice>
                  </complexType>
               </element>
               <element name="txt">
                  <simpleType>
                     <restriction base="string">
                        <maxLength value="256"/>
                     </restriction>
                  </simpleType>
               </element>
            </sequence>
         </extension>
      </complexContent>
   </complexType>
```

Instance examples:

```
<AAA052 gml:id="_0">
   <position>
      <gml:Surface/>
   </position>
</AAA052>
```

```
<PAA052 gml:id="_1">
   <position>
      <gml:Point/>
   </position>
   </txt>My text …</txt>
</PAA052>
```

Alternative 2a: Feature classification independent of the spatial representation

A single feature type is created, the different spatial object value types are distinguished by different mutually exclusive properties.

Note that in this representation an XML parser can not identify that the properties "surfaceGeometry" and "txt" are mutually exculsive. In addition, it is not clear the existance of the "pointGeometry" property implies that the "txt" property must exist, too. (Schematron could in principle be used to represent such constraints, but this is not included in this example right now nor is this part of the UML-to-GML-mapping-rules at this point.)

```
   <element name="AA052" type="app:AA052Type" substitutionGroup="gml:_Feature"/>
   <complexType name="AA052Type">
      <complexContent>
         <extension base="gml:AbstractFeatureType">
            <sequence>
               <choice>
                  <element name="pointGeometry" type="gml:PointPropertyType"/>
                  <element name="multiPointGeometry" type="gml:MultiPointPropertyType"/>
                  <element name="surfaceGeometry" type="gml:SurfacePropertyType"/>
                  <element name="multiSurfaceGeometry"
type="gml:MultiSurfacePropertyType"/>
               </choice>
               <element name="txt" minOccurs="0">
                  <simpleType>
                     <restriction base="string">
                        <maxLength value="256"/>
                     </restriction>
                  </simpleType>
               </element>
            </sequence>
         </extension>
      </complexContent>
   </complexType>
```

Instance examples:

```
<AA052 gml:id="_0">
   <surfaceGeometry>
      <gml:Surface/>
   </surfaceGeometry>
</AA052>

<AA052 gml:id="_1">
   <pointGeometry>
      <gml:Point/>
   </pointGeometry>
   </txt>My text …</txt>
</AA052>
```

Alternative 2b: As in Alternative 2a, but with a single spatial property per feature

Again, note that in this representation an XML parser can not identify that the constraints with respect to the "txt" property as in 2a.

In addition, a (conceptually derived) property "spatialDimension" has been added to make the dimensionality information available explicitly.

```
   <element name="AA052" type="app:AA052Type" substitutionGroup="gml:_Feature"/>
   <complexType name="AA052Type">
      <complexContent>
         <extension base="gml:AbstractFeatureType">
            <sequence>
               <element name="position">
                  <complexType>
                     <choice>
                        <element ref="gml:Point"/>
                        <element ref="gml:MultiPoint"/>
                        <element ref="gml:_Surface"/>
                        <element ref="gml:MultiSurface"/>
                     </choice>
                  </complexType>
               </element>
               <element name="spatialDimension">
                  <restriction base="string">
                     <enumeration value="0"/>
                     <enumeration value="2"/>
                  </restriction>
               </element>
               <element name="txt" minOccurs="0">
                  <simpleType>
                     <restriction base="string">
                        <maxLength value="256"/>
                     </restriction>
                  </simpleType>
               </element>
            </sequence>
         </extension>
      </complexContent>
   </complexType>
```

Instance examples:

```
<AA052 gml:id="_0">
   <position>
      <gml:Surface/>
   </position>
   <spatialDimension>2</spatialDimension>
</AA052>

<AA052 gml:id="_1">
   <position>
      <gml:Point/>
   </position>
   <spatialDimension>0</spatialDimension>
   </txt>My text …</txt>
</AA052>
```

Query 1: All oil/gas fields in an area

Alternative 1a: This requires two separate queries (one per feature type) and two predicates each, one per spatial property.

```
<GetFeature version="1.0.0" service="WFS">
   <Query typeName="MSD3:AAA052">
      <ogc:Filter>
         <ogc:Or>
            <ogc:BBOX>
               <ogc:PropertyName>MSD3:surfaceGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
```

```
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
            <ogc:BBOX>
               <ogc:PropertyName>MSD3:multiSurfaceGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
         </ogc:Or>
      </ogc:Filter>
   </Query>
   <Query typeName="vmap0:PAA052">
      <ogc:Filter>
         <ogc:Or>
            <ogc:BBOX>
               <ogc:PropertyName>MSD3:pointGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
            <ogc:BBOX>
               <ogc:PropertyName>MSD3:multiPointGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
         </ogc:Or>
      </ogc:Filter>
   </Query>
</GetFeature>
```

Note that this requires that the Filter Encoding Specification is interpreted in a way that BBOX evaluates to FALSE if the spatial property is not set for the feature instance. This is not entirely clear from the specifciation! Otherwise, additional predicates with explicit tests for NULL would need to be added. This applieas to all instances of the queries for alternatives 1a and 2a.

Alternative 1b: Again, this requires two separate queries (one per feature type), but only one predicate each.

```
<GetFeature version="1.0.0" service="WFS">
   <Query typeName="MSD3:AAA052">
      <ogc:Filter>
         <ogc:BBOX>
            <ogc:PropertyName>MSD3:position</ogc:PropertyName>
            <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326[4]">
```

[4] In this example, 2D coordinates are used. For 2.5D or 3D coordinates, there has been some discussion about the correct CRS to be used. This has been clarified (thanks to Roel Nicolay, co-chair of the OGC CRS WG): Using EPSG code 4329 is wrong with any newer version of the EPSG database as code 4329 was "deprecated" in December 2002. It has been replaced by code 4327 which in turn was replaced by code 4979, which should be used.

However, the CRS with code 4979 is WGS 84 (3D) with geodetic latitude, geodetic longitude and ellipsoidal height. If the data is geodetic latitude, geodetic longitude and gravity-related height (using the EGM84 model) instead, this is currently not part of the EPSG database, but it has been proposed to be added.

In order to use gravity-related heights now, a coordinate transformation from ellipsoidal heights to gravity-related heights would need to be added and a new (non EPSG-based) reference to a Compound CRS consisting of the WGS84 2D CRS with code 4326 and a Vertical CRS code that has been defined using the transformation would need to be used.

```
               <gml:pos>1 1</gml:pos>
               <gml:pos>2 2</gml:pos>
            </gml:Envelope>
         </ogc:BBOX>
      </ogc:Filter>
   </Query>
   <Query typeName="vmap0:PAA052">
      <ogc:Filter>
         <ogc:BBOX>
            <ogc:PropertyName>MSD3:position</ogc:PropertyName>
            <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
               <gml:pos>1 1</gml:pos>
               <gml:pos>2 2</gml:pos>
            </gml:Envelope>
         </ogc:BBOX>
      </ogc:Filter>
   </Query>
</GetFeature>
```

Alternative 2a: This requires only one query, but four predicates, one per spatial property.

```
<GetFeature version="1.0.0" service="WFS">
   <Query typeName="app:AA052">
      <ogc:Filter>
         <ogc:Or>
            <ogc:BBOX>
               <ogc:PropertyName>app:surfaceGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
            <ogc:BBOX>
               <ogc:PropertyName>app:multiSurfaceGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
            <ogc:BBOX>
               <ogc:PropertyName>app:pointGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
            <ogc:BBOX>
               <ogc:PropertyName>app:multiPointGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
         </ogc:Or>
      </ogc:Filter>
   </Query>
</GetFeature>
```

Alternative 2b: This requires only one query and also only one predicate since there is just a single spatial property.

```
<GetFeature version="1.0.0" service="WFS">
   <Query typeName="app:AA052">
      <ogc:Filter>
         <ogc:BBOX>
            <ogc:PropertyName>app:location</ogc:PropertyName>
            <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
               <gml:pos>1 1</gml:pos>
```

```
               <gml:pos>2 2</gml:pos>
            </gml:Envelope>
         </ogc:BBOX>
      </ogc:Filter>
   </Query>
</GetFeature>
```

Query 2: All oil/gas fields with point representation in an area

Alternative 1a:

```
<GetFeature version="1.0.0" service="WFS">
   <Query typeName="MSD3:PAA052">
      <ogc:Filter>
         <ogc:Or>
            <ogc:BBOX>
               <ogc:PropertyName>MSD3:pointGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
            <ogc:BBOX>
               <ogc:PropertyName>MSD3:multiPointGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
         </ogc:Or>
      </ogc:Filter>
   </Query>
</GetFeature>
```

Alternative 1b:

```
<GetFeature version="1.0.0" service="WFS">
   <Query typeName="MSD3:PAA052">
      <ogc:Filter>
         <ogc:BBOX>
            <ogc:PropertyName>MSD3:position</ogc:PropertyName>
            <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
               <gml:pos>1 1</gml:pos>
               <gml:pos>2 2</gml:pos>
            </gml:Envelope>
         </ogc:BBOX>
      </ogc:Filter>
   </Query>
</GetFeature>
```

Alternative 2a:

```
<GetFeature version="1.0.0" service="WFS">
   <Query typeName="app:AA052">
      <ogc:Filter>
         <ogc:Or>
            <ogc:BBOX>
               <ogc:PropertyName>app:pointGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
            <ogc:BBOX>
               <ogc:PropertyName>app:multiPointGeometry</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
```

```
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
         </ogc:Or>
      </ogc:Filter>
   </Query>
</GetFeature>
```

Alternative 2b: Here, we are making use of the spatialDimension attribute.

```
<GetFeature version="1.0.0" service="WFS">
   <Query typeName="app:AA052">
      <ogc:Filter>
         <ogc:And>
            <ogc:BBOX>
               <ogc:PropertyName>app:location</ogc:PropertyName>
               <gml:Envelope srsName="urn:ogc:def:crs:EPSG:6.6:4326">
                  <gml:pos>1 1</gml:pos>
                  <gml:pos>2 2</gml:pos>
               </gml:Envelope>
            </ogc:BBOX>
            <ogc:PropertyIsEqualTo>
               <ogc:PropertyName>app:location/spatialDimension</ogc:PropertyName>
               <ogc:Literal>0</ogc:Literal>
            </ogc:PropertyIsEqualTo>
         </ogc:And>
      </ogc:Filter>
   </Query>
</GetFeature>
```

Mapping to the FLDB

The four alternatives listed above have different characteristics when mapped to the FLDB, where we would have two tables PAA052 and AAA052 and with one spatial slot each (e.g. an SDO_POINT for PAA052).

Alternative 1a: The two spatial properties in the choice are mapped to the same slot in the database, i.e. this requires some additional processing in the translation of the queries and the encoding of the query results in XML.

AAA052/surfaceProperty ←→ AAA052 (SDO_POLYGON)

AAA052/multiSurfaceProperty ←→ AAA052 (SDO_POLYGON)

PAA052/pointProperty ←→ PAA052 (SDO_POINT)

PAA052/multiPointProperty ←→ PAA052 (SDO_POINT)

PAA052/txt ←→ PAA052 (txt)

Alternative 1b: The mapping is one-to-one between the GML encoding and the FLDB representation.

AAA052/location ←→ AAA052 (SDO_POLYGON)

PAA052/location ←→ PAA052 (SDO_POINT)

PAA052/txt ←→ PAA052 (txt)

Alternative 2a: Like in the case 1a, some properties are mapped to the same slot in the database, i.e. this requires some additional processing in the translation of the queries and the encoding of the query results in XML.

AA052/surfaceProperty ←→ AAA052 (SDO_POLYGON)

AA052/multiSurfaceProperty ←→ AAA052 (SDO_POLYGON)

AA052/pointProperty ←→ PAA052 (SDO_POINT)

AA052/multiPointProperty ←→ PAA052 (SDO_POINT)

AA052/txt ←→ PAA052 (txt)

Alternative 2b: The mapping from the GML property is to one of two different slots in the database depending on the dimensionality:

AA052/location[spatialDimension='2'] ←→ AAA052 (SDO_POLYGON)

AA052/location[spatialDimension='0'] ←→ PAA052 (SDO_POINT)

AA052/txt[spatialDimension='0'] ←→ PAA052 (txt)

Conformance to the proposed GML Profile for Simple Feature

This discussion considers only the encoding of the spatial properties and no other aspect of the schema.

Alternative 1a: To be conformant to the Simple Feature profile, changes would be required. Instead of

```
<complexType name="PAA052Type">
   <complexContent>
      <extension base="gml:AbstractFeatureType">
         <sequence>
            <choice>
               <element name="pointGeometry" type="gml:PointPropertyType"/>
               <element name="multiPointGeometry" type="gml:MultiPointPropertyType"/>
            </choice>
            <!-- ... -->
         </sequence>
      </extension>
   </complexContent>
</complexType>
```

the spatial properties would need to be represented as

```
<complexType name="PAA052Type">
   <complexContent>
      <extension base="gml:AbstractFeatureType">
         <sequence>
            <element name="pointGeometry" type="gml:PointPropertyType" minOccurs="0"/>
            <element name="multiPointGeometry" type="gml:MultiPointPropertyType"
```

```
minOccurs="0"/>
               <!-- ... -->
            </sequence>
         </extension>
      </complexContent>
   </complexType>
```

with the normative rule that exactly one of the two spatial properties shall be set.

The Simple Feature Profile does not support the use of choice on the level of property elements.

Alternative 1b: Again, this representation is not conformant to Simple Features as it is. Instead of

```
   <complexType name="PAA052Type">
      <complexContent>
         <extension base="gml:AbstractFeatureType">
            <sequence>
               <element name="position">
                  <complexType>
                     <choice>
                        <element ref="gml:Point"/>
                        <element ref="gml:MultiPoint"/>
                     </choice>
                  </complexType>
               </element>
               <!-- ... -->
            </sequence>
         </extension>
      </complexContent>
   </complexType>
```

we would need to write

```
   <complexType name="PAA052Type">
      <complexContent>
         <extension base="gml:AbstractFeatureType">
            <sequence>
               <element name="position" type="gml:GeometryPropertyType"/>
               <!-- ... -->
            </sequence>
         </extension>
      </complexContent>
   </complexType>
```

with the normative rule that the value of location shall be either a gml:Point or a gml:MultiPoint.

Alternative 2a: Similar to 1a above, the type would need to be changed to

```
   <complexType name="AA052Type">
      <complexContent>
         <extension base="gml:AbstractFeatureType">
            <sequence>
               <element name="pointGeometry" type="gml:PointPropertyType" minOccurs="0"/>
               <element name="multiPointGeometry" type="gml:MultiPointPropertyType"
minOccurs="0"/>
               <element name="surfaceGeometry" type="gml:SurfacePropertyType"
minOccurs="0"/>
               <element name="multiSurfaceGeometry" type="gml:MultiSurfacePropertyType"
minOccurs="0"/>
               <!-- ... -->
            </sequence>
```

```
      </extension>
   </complexContent>
</complexType>
```

with the normative rule that exactly one of the four spatial properties shall be set.

Alternative 2b: Similar to 1b above, the type would need to be changed to

```
<complexType name="AA052Type">
   <complexContent>
      <extension base="gml:AbstractFeatureType">
         <sequence>
            <element name="position" type="gml:GeometryPropertyType"/>
            <element name="spatialDimension">
               <restriction base="string">
                  <enumeration value="0"/>
                  <enumeration value="2"/>
               </restriction>
            </element>
            <!-- ... -->
         </sequence>
      </extension>
   </complexContent>
</complexType>
```

with the normative rule that the value of location shall be either a gml:Point, gml:MultiPoint, gml:Surface or gml:MultiSurface.

In principle, the geometry properties have to be changed to use the generic "gml:GeometryPropertyType", because of the support for multiple geometry types. This is because there is not just a single specific geometry type per feature type but more than one. A PGA065 feature may have a gml:Point or a gml:MultiPoint as the value of "position". Using a choice of gml:Point and gml:MultiPoint in a property type is not allowed by GML-SF. Alternatively, gml:PointPropertyType could be used with a maxOccurs="unbounded" as an alternative way to encode the geometric aggregation. However, MultiPoint was used as this geometry object is intended for collections of points (also by GML-SF). As a result, gml:GeometryPropertyType has to be used and the "real" constriants can only be provided in additional normative text.

SLD examples

This discussion provides SLD 1.0.0 examples for each representation.

Alternative 1a requires two Rules for each feature type.

```
<sld:UserStyle>
   <sld:Name>AAA052</sld:Name>
   <sld:Title>AAA052 style</sld:Title>
   <sld:FeatureTypeStyle xmlns:MSD3="http://www.example.com/MSD3">
      <sld:FeatureTypeName>MSD3:AAA052</sld:FeatureTypeName>
      <sld:Rule>
         <sld:PolygonSymbolizer>
            <sld:Geometry>
               <ogc:PropertyName>MSD3:surfaceGeometry</ogc:PropertyName>
            </sld:Geometry>
            <sld:Fill>
               <sld:CssParameter name="fill">
                  <ogc:Literal>rgb(175,57,132)</ogc:Literal>
               </sld:CssParameter>
```

```
               </sld:Fill>
               <sld:Stroke/>
            </sld:PolygonSymbolizer>
         </sld:Rule>
         <sld:Rule>
            <sld:PolygonSymbolizer>
               <sld:Geometry>
                  <ogc:PropertyName>MSD3:multiSurfaceGeometry</ogc:PropertyName>
               </sld:Geometry>
               <sld:Fill>
                  <sld:CssParameter name="fill">
                     <ogc:Literal>rgb(175,57,132)</ogc:Literal>
                  </sld:CssParameter>
               </sld:Fill>
               <sld:Stroke/>
            </sld:PolygonSymbolizer>
         </sld:Rule>
      </sld:FeatureTypeStyle>
   </sld:UserStyle>
   <sld:UserStyle>
      <sld:Name>PAA052</sld:Name>
      <sld:Title>PAA052 style</sld:Title>
      <sld:FeatureTypeStyle xmlns:MSD3="http://www.example.com/MSD3">
         <sld:FeatureTypeName>MSD3:PAA052</sld:FeatureTypeName>
         <sld:Rule>
            <sld:PointSymbolizer>
               <sld:Geometry>
                  <ogc:PropertyName>MSD3:pointGeometry</ogc:PropertyName>
               </sld:Geometry>
               <sld:Graphic>
                  <sld:ExternalGraphic>
                     <sld:OnlineResource xlink:href="http://example.com/symbol/#BAR"
xlink:title="BAR"
                        xlink:type="simple" xmlns:xlink="http://www.w3.org/1999/xlink"/>
                     <sld:Format>image/svg+xml</sld:Format>
                  </sld:ExternalGraphic>
                  <sld:Size>12</sld:Size>
               </sld:Graphic>
            </sld:PointSymbolizer>
         </sld:Rule>
         <sld:Rule>
            <sld:PointSymbolizer>
               <sld:Geometry>
                  <ogc:PropertyName>MSD3:multiPointGeometry</ogc:PropertyName>
               </sld:Geometry>
               <sld:Graphic>
                  <sld:ExternalGraphic>
                     <sld:OnlineResource xlink:href="http://example.com/symbol/#FOO"
xlink:title="FOO"
                         xlink:type="simple" xmlns:xlink="http://www.w3.org/1999/xlink"/>
                     <sld:Format>image/svg+xml</sld:Format>
                  </sld:ExternalGraphic>
                  <sld:Size>12</sld:Size>
               </sld:Graphic>
            </sld:PointSymbolizer>
         </sld:Rule>
      </sld:FeatureTypeStyle>
   </sld:UserStyle>
```

Alternative 1b requires only one Rule element for each feature type.

```
   <sld:UserStyle>
      <sld:Name>AAA052</sld:Name>
      <sld:Title>AAA052 style</sld:Title>
      <sld:FeatureTypeStyle xmlns:MSD3="http://www.example.com/MSD3">
         <sld:FeatureTypeName>MSD3:AAA052</sld:FeatureTypeName>
         <sld:Rule>
            <sld:PolygonSymbolizer>
               <sld:Geometry>
                  <ogc:PropertyName>MSD3:position</ogc:PropertyName>
```

```
               </sld:Geometry>
               <sld:Fill>
                  <sld:CssParameter name="fill">
                     <ogc:Literal>rgb(175,57,132)</ogc:Literal>
                  </sld:CssParameter>
               </sld:Fill>
               <sld:Stroke/>
            </sld:PolygonSymbolizer>
         </sld:Rule>
      </sld:FeatureTypeStyle>
   </sld:UserStyle>
   <sld:UserStyle>
      <sld:Name>PAA052</sld:Name>
      <sld:Title>PAA052 style</sld:Title>
      <sld:FeatureTypeStyle xmlns:MSD3="http://www.example.com/MSD3">
         <sld:FeatureTypeName>MSD3:PAA052</sld:FeatureTypeName>
         <sld:Rule>
            <sld:PointSymbolizer>
               <sld:Geometry>
                  <ogc:PropertyName>MSD3:position</ogc:PropertyName>
               </sld:Geometry>
               <sld:Graphic>
                  <sld:ExternalGraphic>
                     <sld:OnlineResource xlink:href="http://example.com/symbol/#BAR"
xlink:title="BAR"
                        xlink:type="simple" xmlns:xlink="http://www.w3.org/1999/xlink"/>
                     <sld:Format>image/svg+xml</sld:Format>
                  </sld:ExternalGraphic>
                  <sld:Size>12</sld:Size>
               </sld:Graphic>
            </sld:PointSymbolizer>
         </sld:Rule>
      </sld:FeatureTypeStyle>
   </sld:UserStyle>
```

Alternative 2a requires four Rules for the (single) feature type.

```
   <sld:UserStyle>
      <sld:Name>AA052</sld:Name>
      <sld:Title>AA052 style</sld:Title>
      <sld:FeatureTypeStyle xmlns:MSD3="http://www.example.com/MSD3">
         <sld:FeatureTypeName>MSD3:AA052</sld:FeatureTypeName>
         <sld:Rule>
            <sld:PointSymbolizer>
               <sld:Geometry>
                  <ogc:PropertyName>MSD3:pointGeometry</ogc:PropertyName>
               </sld:Geometry>
               <sld:Graphic>
                  <sld:ExternalGraphic>
                     <sld:OnlineResource xlink:href="http://example.com/symbol/#BAR"
xlink:title="BAR"
                        xlink:type="simple" xmlns:xlink="http://www.w3.org/1999/xlink"/>
                     <sld:Format>image/svg+xml</sld:Format>
                  </sld:ExternalGraphic>
                  <sld:Size>12</sld:Size>
               </sld:Graphic>
            </sld:PointSymbolizer>
         </sld:Rule>
         <sld:Rule>
            <sld:PointSymbolizer>
               <sld:Geometry>
                  <ogc:PropertyName>MSD3:multiPointGeometry</ogc:PropertyName>
               </sld:Geometry>
               <sld:Graphic>
                  <sld:ExternalGraphic>
                     <sld:OnlineResource xlink:href="http://example.com/symbol/#FOO"
xlink:title="FOO"
                         xlink:type="simple" xmlns:xlink="http://www.w3.org/1999/xlink"/>
                     <sld:Format>image/svg+xml</sld:Format>
                  </sld:ExternalGraphic>
```

```
                  <sld:Size>12</sld:Size>
               </sld:Graphic>
            </sld:PointSymbolizer>
         </sld:Rule>
         <sld:Rule>
            <sld:PolygonSymbolizer>
               <sld:Geometry>
                  <ogc:PropertyName>MSD3:surfaceGeometry</ogc:PropertyName>
               </sld:Geometry>
               <sld:Fill>
                  <sld:CssParameter name="fill">
                     <ogc:Literal>rgb(175,57,132)</ogc:Literal>
                  </sld:CssParameter>
               </sld:Fill>
               <sld:Stroke/>
            </sld:PolygonSymbolizer>
         </sld:Rule>
         <sld:Rule>
            <sld:PolygonSymbolizer>
               <sld:Geometry>
                  <ogc:PropertyName>MSD3:multiSurfaceGeometry</ogc:PropertyName>
               </sld:Geometry>
               <sld:Fill>
                  <sld:CssParameter name="fill">
                     <ogc:Literal>rgb(175,57,132)</ogc:Literal>
                  </sld:CssParameter>
               </sld:Fill>
               <sld:Stroke/>
            </sld:PolygonSymbolizer>
         </sld:Rule>
      </sld:FeatureTypeStyle>
   </sld:UserStyle>
```

Alternative 2b requires two Rules for the feature type, each using an ogc:Filter to select the appropriate features based on the value of spatialDimension.

```
   <sld:UserStyle>
      <sld:Name>AA052</sld:Name>
      <sld:Title>AA052 style</sld:Title>
      <sld:FeatureTypeStyle xmlns:MSD3="http://www.example.com/MSD3">
         <sld:FeatureTypeName>MSD3:AA052</sld:FeatureTypeName>
         <sld:Rule>
            <ogc:Filter>
              <ogc:PropertyIsEqualTo>
                <ogc:PropertyName>MSD3:spatialDimension</ogc:PropertyName>
                <ogc:Literal>0</ogc:Literal>
              </ogc:PropertyIsEqualTo>
            </ogc:Filter>
            <sld:PointSymbolizer>
               <sld:Geometry>
                  <ogc:PropertyName>MSD3:position</ogc:PropertyName>
               </sld:Geometry>
               <sld:Graphic>
                  <sld:ExternalGraphic>
                     <sld:OnlineResource xlink:href="http://example.com/symbol/#FOO"
xlink:title="FOO"
                         xlink:type="simple" xmlns:xlink="http://www.w3.org/1999/xlink"/>
                     <sld:Format>image/svg+xml</sld:Format>
                  </sld:ExternalGraphic>
                  <sld:Size>12</sld:Size>
               </sld:Graphic>
            </sld:PointSymbolizer>
         </sld:Rule>
         <sld:Rule>
            <ogc:Filter>
              <ogc:PropertyIsEqualTo>
                <ogc:PropertyName>MSD3:spatialDimension</ogc:PropertyName>
                <ogc:Literal>2</ogc:Literal>
              </ogc:PropertyIsEqualTo>
```

```
            </ogc:Filter>
            <sld:PolygonSymbolizer>
               <sld:Geometry>
                  <ogc:PropertyName>MSD3:position</ogc:PropertyName>
               </sld:Geometry>
               <sld:Fill>
                  <sld:CssParameter name="fill">
                     <ogc:Literal>rgb(175,57,132)</ogc:Literal>
                  </sld:CssParameter>
               </sld:Fill>
               <sld:Stroke/>
            </sld:PolygonSymbolizer>
         </sld:Rule>
      </sld:FeatureTypeStyle>
   </sld:UserStyle>
```

Implementation issues

The following other issues and alternatives came up as part of the GML Investigations track of OWS-3 GeoDSS when the MSD3 schema was implemented using an Oracle database, because the most general geometry property type gml:GeometryPropertyType is being used for the 'msd3:position' property and this generality created problems in the implementation.

The issue was that the mapping tool employed by one participant that reads the GML application schema and creates the relational schema in Oracle cannot interpret what the 'msd3:position' property maps to (SDO Point, SDO Line, SDO Polygon) based on its type.

Therefore, replacing gml:GeometryPropertyType with a choice of more specific property types (gml:PointPropertyType and gml:MultiPointPropertyType) was proposed. In particular the following alternatives were brought up and discussed.

Alterative 3a

```
<complexType name="PAA010Type">
 <complexContent>
  <extension base="gml:AbstractFeatureType">
   <sequence>
    <choice>
     <element name="position" type="gml:PointPropertyType"/>
     <element name="multiPosition" type="gml:MultiPointPropertyType"/>
    </choice>
    ...
   </sequence>
  </extension>
 </complexContent>
</complexType>
```

It was also noted that replacing the more general property type with a choice between two different spatial types for the same property also causes problems loading the data.

Other issues with this alternative were:

- It does not conform to GML-SF.

- Using two spatial properties where conceptually there is only one (it is the spatial position of the feature, whether this is a single point or a set of points), is not really good modelling - in particular since the object-property-pattern explicitly allows to determine the type of the property value easily.

- This style was used in OWS-2 and the feedback has been that this creates implementation issues and 3b should be preferred compared to 1a.

Therefore this alternative was rejected.

Alternative 3b

```
<complexType name="PAA010Type">
```

```
 <complexContent>
  <extension base="gml:AbstractFeatureType">
   <sequence>
     <element name="position">
       <complexType>
         <choice>
            <element ref="gml:Point"/>
            <element ref="gml:MultiPoint"/>
         </choice>
       </complexType>
     </element>
     ...
   </sequence>
  </extension>
 </complexContent>
</complexType>
```

Since this alternative had the following issues, it was rejected, too.

- It does not conform with the GML-SF profile.
- There are currently no rules in GML Annex E that could produces such property types.
- The Schema Assembly Tool is not able to handle such property types.

Alternative 3c

```
<complexType name="PAA010Type">
 <complexContent>
  <extension base="gml:AbstractFeatureType">
   <sequence>
    <element name="position" type="gml:PointPropertyType" minOccurs="0"/>
    <element name="multiPosition" type="gml:MultiPointPropertyType" minOccurs="0"/>
    ...
   </sequence>
  </extension>
 </complexContent>
</complexType>
```

While this alternative would conform to the proposed GML-SF profile, the arguments raised in Alternative 3a against such "double properties" are still valid.

There was also a discussion on the use of the MultiXYZPropertyType types and if there might be other ways to map the spatial properties without using them. One option would be to use a single spatial property that allowed multiple occurances:

Alternative 3d

```
<complexType name="PAA010Type">
 <complexContent>
  <extension base="gml:AbstractFeatureType">
   <sequence>
    <element name="position" type="gml:PointPropertyType" minOccurs="0"
maxOccurs="unbounded"/>
    ...
   </sequence>
  </extension>
 </complexContent>
</complexType>
```

This alternative would be a possibility and actually match the conceptual model as the aggregates are not part of the DGIWG profile of ISO 19107 and it is assumed that such properties with a multiplicity greater than one or Set<T> parametrized types would be used. However, if this argument is followed, it would raise the question how useful the geometry aggregates are in ISO 19107, Simple Feature for SQL, GML, etc. at all.

Alternative 3e

```
<complexType name="PAA010Type">
 <complexContent>
  <extension base="gml:AbstractFeatureType">
   <sequence>
    <element name="position" type="gml:MultiPointPropertyType"/>
    ...
   </sequence>
  </extension>
 </complexContent>
</complexType>
```

This alternative finally uses the geometry aggregate only and would be an alternative approach. It would require that a geometry primitive would be encoded as a geometry aggregate with a single member.

Bibliography

ISO 19115:2003, *Geographic Information - Metadata*

ISO/DTS 19139, *Geographic Information - Metadata XML Schema Implementation*

Dublin Core

OGC 04-100, NGA Application Schema Development (OWS-2)