

# Open Geospatial Consortium Inc.

Date: 2007-03-12

Reference number of this OGC® IP initiative document: **OGC 06-155**

Version: 0.0.6

Category: OpenGIS® Discussion Paper

Editor: Tim Wilson

## **OWS-4 CSW ebRIM Modelling Guidelines IPR**

### **Copyright notice**

Copyright © 2007 Open Geospatial Consortium. All Rights Reserved

To obtain additional rights of use, visit <http://www.opengeospatial.org/legal>

### **Warning**

This document is not an OGC Standard. This document is an OGC Discussion Paper and is therefore not an official position of the OGC membership. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard. Further, an OGC Discussion Paper should not be referenced as required or mandatory technology in procurements.

Document type: OGC® Discussion Paper  
Document stage: Draft  
Document language: English

## Contents

<b>Preface</b> .....	<b>vi</b>
<b>i. Submitting organizations</b> .....	<b>vi</b>
<b>ii. Document terms and definitions</b> .....	<b>vi</b>
<b>iii. Document contributor contact points</b> .....	<b>vi</b>
<b>iv. Revision history</b> .....	<b>vii</b>
<b>v. Changes to the OpenGIS® Abstract Specification</b> .....	<b>vii</b>
<b>Foreword</b> .....	<b>viii</b>
<b>Introduction</b> .....	<b>ix</b>
<b>1 Scope</b> .....	<b>1</b>
<b>2 Conformance</b> .....	<b>1</b>
<b>3 Normative references</b> .....	<b>1</b>
<b>4 Terms and definitions</b> .....	<b>1</b>
<b>5 Conventions</b> .....	<b>2</b>
<b>5.1 Symbols (and abbreviated terms)</b> .....	<b>2</b>
<b>5.2 UML notation</b> .....	<b>2</b>
<b>5.3 XPath notation</b> .....	<b>3</b>
<b>6 Overview</b> .....	<b>4</b>
<b>6.1 Overview of ebRIM</b> .....	<b>4</b>
<b>6.2 Overview of CSW.ebRIM Extension Package</b> .....	<b>4</b>
<b>6.2.1 Registry Package</b> .....	<b>5</b>
<b>6.2.2 Object Types</b> .....	<b>6</b>
<b>6.2.3 Association Types</b> .....	<b>7</b>
<b>6.2.4 Slots</b> .....	<b>8</b>
<b>6.2.5 Classification Schemes</b> .....	<b>10</b>
<b>6.2.6 Stored Queries</b> .....	<b>10</b>
<b>6.3 Overview of Example Domain Model</b> .....	<b>11</b>
<b>7 Mapping a Domain Specific Model to a CSW.ebRIM Extension Package</b> ....	<b>13</b>
<b>7.1 Object Type Mapping</b> .....	<b>13</b>
<b>7.1.1 Inheritance</b> .....	<b>14</b>
<b>7.2 Attribute Mapping</b> .....	<b>15</b>
<b>7.2.1 Map to Identifier</b> .....	<b>16</b>
<b>7.2.2 Map to Name/Description</b> .....	<b>17</b>
<b>7.2.3 Map to Classification</b> .....	<b>18</b>
<b>7.2.4 Map to ExternalLink</b> .....	<b>19</b>

7.2.5	Direct Attribute Mapping .....	19
7.2.6	Mapping to Slot .....	19
7.3	Association Mapping .....	20
7.4	Stored Queries .....	21
7.5	Extension Package .....	23
8	Mapping Templates .....	29
Annex A.....		30
1	OWS-4 Sensor Web Enablement Catalog Resource Profile.....	30
1.1	Background .....	30
1.2	Abstract.....	31
1.3	Related Information.....	31
1.3.1	Use Cases for Catalog Role in Managing SWE Resources .....	31
1.3.2	Source Materials .....	31
1.4	Sensor Observation Service Management.....	31
1.4.1	Material Description.....	31
1.4.2	Related Definitions.....	32
1.4.3	Mapping of source materials into ebRIM objects.....	32
1.4.4	Query extensions / standard queries .....	35
1.4.5	Output schemas / formats .....	38
1.4.6	Resource lifecycle .....	38
1.4.7	Transaction corresponding to SOS instances harvest .....	38
1.4.8	Experiment .....	38
1.5	SensorML Management .....	39
1.5.1	Material Description.....	39
1.5.2	Mapping of source materials into ebRIM objects.....	39
1.6	Phenomenon Dictionary Management.....	43
1.6.1	Material Description.....	43
1.6.2	Mapping of source materials into ebRIM objects.....	44
Annex B OWS Domain Extension Package—Feature Catalog.....		54
1	Introduction.....	54
2	UML Model .....	54
3	Object Types.....	55
4	Association Types.....	63
5	Element Mappings .....	69
6	Security Tags .....	83
7	Mapping to OGC Core Properties .....	85
Annex C Mapping Templates .....		87
1	Types .....	87
2	Attributes .....	87
3	Associations .....	87

<b>4</b>	<b>Classifications.....</b>	<b>87</b>
<b>5</b>	<b>Stored Queries.....</b>	<b>87</b>
	<b>Bibliography .....</b>	<b>88</b>

<b>Figures</b>	<b>Page</b>
Figure 6-1: CSW.ebRIM Extension Packages .....	5
Figure 6-2: Example CRS domain model .....	11
Figure 7-1: OASIS Attribute Mapping Algorithm Flowchart .....	16

## Preface

Suggested additions, changes, and comments on this draft report are welcome and encouraged. Such suggestions may be submitted by OGC portal message, email message, or by making suggested changes in an edited copy of this document.

The changes made in this document version, relative to the previous version, are tracked by Microsoft Word, and can be viewed if desired. If you choose to submit suggested changes by editing this document, please first accept all the current changes, and then make your suggested changes with change tracking on.

### i. Submitting organizations

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

- Galdos Systems Inc.
- IONIC Software S.A.
- CubeWerx Inc.

### ii. Document terms and definitions

This document uses the specification terms defined in Subclause 5.3 of [OGC 05-008], which is based on the ISO/IEC Directives, Part 2.

### iii. Document contributor contact points

All questions regarding this document should be directed to the editor or the contributors:

Contact	Company	Email
Tim Wilson (Editor)	Galdos Systems Inc.	Twilson [at] galdosinc.com
Leif Stainsby	Galdos Systems Inc.	Lstainsby [at] galdosinc.com
Ron Lake	Galdos Systems Inc.	Rlake [at] galdosinc.com
Panagiotis (Peter) A.	CubeWerx Inc.	Pvretano[at] cubewerx.com

Vretanos		
Renato Primavera	IONIC Software S.A.	renato.primavera [at] ionicsoft.com

#### iv. Revision history

Date	Release	Editor	Primary clauses modified	Description
2006-11-13	0.0.1	T. Wilson	All	Draft outline for review.
2006-12-21	0.0.2	T. Wilson	Annexes	Outline updated.
2007-01-12	0.0.3	T. Wilson	All	Outline updated; body completed; IONIC Annex added.
2007-01-12	0.0.4	T. Wilson	All except annexes.	Minor edits to document text.
2007-03-09	0.0.5	T. Wilson	All including annexes.	CubeWerx Annex B added; outlining updated.
2007-05-07	0.0.6	Carl Reed	Various	Get ready for posting as DP.

#### v. Changes to the OpenGIS<sup>®</sup> Abstract Specification

Not applicable.

## Foreword

Attention is drawn to the possibility that some of the elements of this document 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.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the specification set forth in this document, and to provide supporting documentation.

This document assumes familiarity with OASIS (Organization for the Advancement of Structured Information Standards) *ebXML Registry Information Model* [OASIS ebRIM], *OGC Catalogue Services — ebRIM (ISO/TS 15000-3) profile of CSW* [OGC 05-025r3], and *OGC Catalogue Services for the Web* [OGC 04-021r3] specifications. Readers are also encouraged to first read the OASIS ebRIM tutorials (listed in Bibliography) which inform the mapping guidance provided in this document.

This document is structured as follows:

- The main body provides general guidance for mapping domain models to ebRIM (ebXML Registry Information Model). It is authored by Galdos (Galdos Systems Inc.) and is based on their INdicio™ CSW.ebRIM registry product implementation and [OGC 05-025r3] specification development experience.
- Annex A summarizes the specific mapping of the Sensor Web Enablement (SWE) domain to ebRIM. It is authored by IONIC (IONIC Software S.A.) and is based on their OWS4 Catalog implementation experience.
- Annex B summarizes the specific mapping of the Feature Catalog domain to ebRIM. It is authored by CubeWerx (CubeWerx Inc.) and is based on their OWS4 Catalogue implementation experience.



## Introduction

As more and more OGC members are implementing CSW.ebRIM registries, they are each faced with the recurring need to map between their domain specific information model to [OASIS ebRIM]. Currently this mapping is being done in an ad hoc manner. As such, this document provides guidance towards creating a standard methodology for mapping a geospatial domain information model to ebRIM. It also presents the results of mapping the specific SWE and Feature Catalog domain models to ebRIM for use within the OWS4 Catalog implementations.



## OWS-4 CSW ebRIM Modelling Guidelines IPR

### 1 Scope

This OpenGIS® document assists the OGC community in mapping geospatial resources to ebRIM models or packages to facilitate resource discovery in an efficient and standard way.

### 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. For undated references, the latest edition of the normative document referred to applies.

OGC 05-025r3, *Catalogue Services — ebRIM (ISO/TS 15000-3) profile of CSW*, Version 1.0.0 (October 2005).

OGC 04-021r3, *OGC™ Catalogue Services Specification*, version 2.0. (May 2005, includes Technical Corrigendum 1).

OGC 05-008, *OGC Web Services Common Specification*, Version 1.0 (May 2005).

OASIS ebRIM, *ebXML Registry Information Model Version 3.0*, OASIS Standard (May 2005), available [online]: <<http://www.oasis-open.org/committees/download.php/13591/docs.oasis-open.orgregrepv3.0specsregrep-rim-3.0-os.pdf>>.

### 4 Terms and definitions

For the purposes of this document, the definitions specified in Clause 4 of [OGC 05-008] and [OGC 05-025r3] shall apply.

## 5 Conventions

### 5.1 Symbols (and abbreviated terms)

API	Application Program Interface
BP	Basic Package
COM	Component Object Model
CORBA	Common Object Request Broker Architecture
COTS	Commercial Off The Shelf
CRS	Coordinate Reference System
CSW	Catalogue Services for the Web
CSW.ebRIM	ebRIM application profile of an OGC Catalogue service
DCE	Distributed Computing Environment
DCP	Distributed Computing Platform
DCOM	Distributed Component Object Model
ebRIM	ebXML Registry Information Model
EP	Extension Package
GML	Geography Markup Language
ISO	International Organization for Standardization
OGC	Open GIS Consortium
UML	Unified Modeling Language
WFS	Web Feature Service
XML	eXtensible Markup Language

### 5.2 UML notation

Most diagrams that appear in this specification are presented using the Unified Modeling Language (UML) static structure diagram, as described in Subclause 5.2 of [OGC 05-008].

### 5.3 XPath notation

When discussing XML instances within examples XPath notation is used to identify particular element or attributes. For more information on XPath see <http://www.w3.org/TR/xpath>.

## 6 Overview

This chapter provides an overview of the requirements for modelling an application specific domain model to a CSW.ebRIM extension package according to [OGC 05-025r3]. It also describes a simplified CRS domain model used to illustrate CSW.ebRIM modelling requirements and guidelines outlined in this document. This CRS domain model is used as the source information model while the CSW.ebRIM extension package is the target informational model for the mapping patterns defined by this document.

Note that the source model is described exclusively by UML and its concepts and terminology. Thus, the terms Object, Class, Attributes and Associations used when describing the source model are those defined by UML.

The information presented is informative and is not intended to replace the normative information defined by [OGC 05-025r3] and [OASIS ebRIM].

### 6.1 Overview of ebRIM

This document does not provide an overview of ebRIM. Instead the reader should consult the many useful OASIS tutorials that discuss mapping domain models to ebRIM (OASIS [1] and [2] listed in Bibliography) or the introduction to ebRIM provided in [OGC 05-025r3]. The reader may also refer to [OASIS ebRIM] if a deeper understanding of the ebRIM model is needed.

An important concept not always obvious to someone new to ebRIM is that a target ebRIM model is not expressed using typing or schemas. ebRIM is not a schema language like XML Schema that is used to define a domain application schema. Rather, the target model is but only XML instance(s) of the different ebRIM classes that themselves conform to the ebRIM XML schema.

### 6.2 Overview of CSW.ebRIM Extension Package

This section summarizes the CSW.ebRIM profile in terms of its requirements and implications when mapping a domain model to a CSW.ebRIM extension package. This extension package (and its member ebRIM artefacts) is the target of the mapping defined in this document. The reader should read [OGC 05-025r3] for the complete specification of a CSW.ebRIM-compliant registry as a whole.

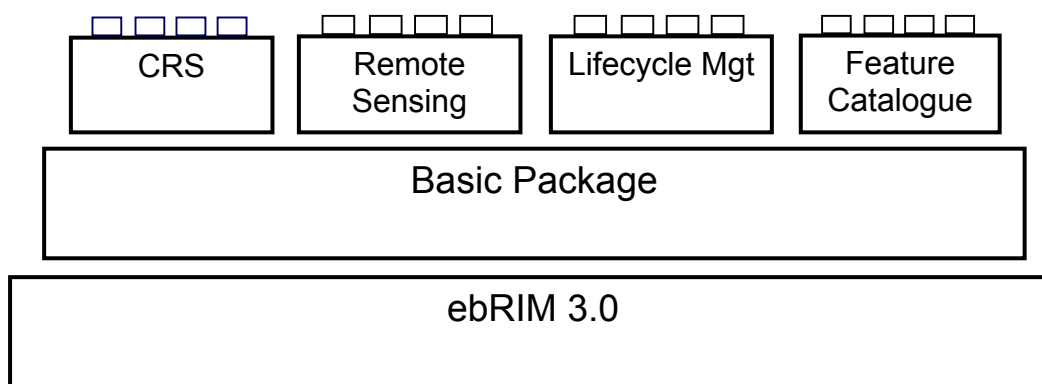
[OGC 05-025r3] defines the CSW.ebRIM Basic Package (BP) that includes the normative OGC catalogue/registry constructs that must be supported by all CSW.ebRIM implementations. These constructs are modelled as ebRIM artefacts conforming to the [OASIS ebRIM] specification.

Section 6 of [OGC 05-025r3] provides a useful summary of extension packages and their role; they are *a cohesive set of extensions that address the needs of a particular application domain or community of practice... [which] provide the principal means of*

*customizing the behaviour and content of a catalogue service.* This infers that an extension package should be tailored for a specific purpose or domain.

For the purposes of this document an extension package is inclusive of its member ebRIM artefacts. In other words mapping to an extension package means mapping to individual ebRIM artefacts and grouping them together as a registry package.

The BP serves as the primary or underlying CSW.ebRIM extension package that supports the core OGC CSW.ebRIM registry model and operations. As CSW.ebRIM registries must implement the BP it is expected that the BP will therefore be employed or extended by further geospatial (e.g. CRS) extension packages, as shown in Figure 6-1:



**Figure 6-1: CSW.ebRIM Extension Packages**

The ebRIM artefacts employed by the BP to implement the CSW.ebRIM registry requirements are summarized in the following subsections.

### 6.2.1 Registry Package

OASIS [2] provides a useful summary of an ebRIM Registry Package: *RegistryObjects may be packaged or organized in a hierarchical structure using a familiar file and folder metaphor. RegistryPackage instances serve as folders while RegistryObject instances serve as files in this metaphor. A RegistryPackage instances groups logically related RegistryObject instances together as members of that RegistryPackage.*

The BP is itself an extension package that is mapped to a rim:RegistryPackage. This object logically groups all of the BP components together for easier loading and retrieval.

[OGC 05-025r3] defines an extension package as a particular case of a rim:RegistryPackage; that is, *[extension] package members are registry objects that are subject to the following constraint: a member object may only be deleted if the package as a whole is deleted—this effectively treats an extension package as a composition.* Note however that a CSW.ebRIM registry does not enforce the semantics of composition

associations. Specifically, deleting a container object does not automatically delete contained objects. As a consequence of the above, dependencies between extension packages (e.g. “uses”) must be modelled as aggregation associations.

### 6.2.2 Object Types

The concept of typing in ebRIM is analogous to classifying its nature. As per OASIS [2], *every stored object within the registry has a defined “nature”, to say what it is and what it represents... To define this nature the registry provides some classification schemes with the idea that every contained object must have a corresponding value, the nature, within one of them... Precisely one classification scheme serves to define the nature of the registry object, the ObjectType classification scheme.*

As the BP domain has concepts not found within the core ebRIM model, it declares the following new extrinsic object types as classification nodes within the ebRIM canonical ObjectType classification scheme:

Object type ID	Description
*:ServiceProfile	Describes what the service does: its features and capabilities (from the OWL-S service ontology). Example: an OGC capabilities document
*:ServiceModel	Describes how the service works, including its essential computational characteristics and behaviours (from the OWL-S service ontology). Example: a WSDL interface description
*:ServiceGrounding	Describes how to access the service: the communications protocols and network endpoints (from the OWL-S service ontology). Example: a WSDL service description
*:Dataset	Description of a geographic data set (from ISO 19115).
*:Schema	A formal description of a conceptual model, expressed using a textual or graphical schema language (e.g., UML, XMI, XML Schema, RELAX NG, ASN.1).
*:Stylesheet	A set of rules for presenting or styling some information resource, typically expressed using a style language (e.g., XSLT, CSS).
*:Document	Documentation of any kind (e.g. specifications, manuals, reports, FAQs).
*:Annotation	Commentary intended to interpret, explain, or clarify some other resource or part thereof.
*:Image	A symbolic visual resource other than text (e.g., diagrams, photographs, drawings, maps, animations).
*:Rights	Information about the rights held in and over the resource. Typically, a Rights object embodies a rights management statement that stipulates conditions of use or distribution. Example: an ODRL or MPEG-21/REL statement



To support more powerful queries and greater interoperability between CSW.ebRIM registries, these BP object types should be subclassed (by extending the canonical ObjectType classification scheme) whenever specialized versions of them are required. For example, if a WMSCapabilitiesServiceProfile class was defined it should subclass the BP ServiceProfile object type.

The declaration of these extrinsic object types within the BP demonstrates required and/or useful patterns when mapping domain objects to ebRIM:

- A domain object type or class not already represented in ebRIM should be mapped to a new subclass of ebRIM rim:ExtrinsicObject.
- To define a subclass of rim:ExtrinsicObject you must add a new rim:ClassificationNode as a child or descendent of the canonical ExtrinsicObject rim:ClassificationNode in the canonical ObjectType classification scheme.

Further details on mapping domain classes to ebRIM are provided in Section 7.1.

### 6.2.3 Association Types

The BP declares the following association types within the ebRIM canonical AssociationType classification scheme:

Association type ID	Description
*:OperatesOn	Associates a Service offer with a description of the data that the service operates on as input or output (from ISO 19119).
*:Presents	Associates a Service offer with a description of what the service does (from the OWL-S 1.1 service ontology).
*:Supports	Associates a Service offer with a description of how an agent can access the service using some communication protocol (from the OWL-S 1.1 service ontology).
*:DescribedBy	Associates a Service offer with a description of its computational characteristics and semantic content of requests (from the OWL-S 1.1 service ontology).
*:Annotates	Associates an annotation resource with the registry object that it explains or evaluates.
*:RepositoryItemFor	Associates a source ExternalLink (that refers to an item in an external repository) with a target ExtrinsicObject.
*:GraphicOverview	Associates a source Dataset with an Image that illustrates or summarizes the data (e.g., a browsing aid).

To support more powerful queries and greater interoperability between CSW.ebRIM registries, these BP association types should be subclassed (by extending the canonical AssociationType classification scheme) whenever specialized versions of them are required. For example, if an OperatesOnCoverage association was defined it should subclass the BP OperatesOn association type.

The declaration of these association types within the BP demonstrates required and/or useful patterns when mapping domain objects to ebRIM:

- A domain association type not already represented in ebRIM should be mapped to a new association type.
- To define a new association type you must add a new rim:ClassificationNode as a child or descendent of the canonical AssociationType classification scheme.

Further details on mapping domain associations to ebRIM are provided in Section 7.2.

#### 6.2.4 Slots

The BP defines the following CSW DCMI metadata terms as slots within a ‘Slots’ classification scheme:

Slot name	Definition
<a href="http://purl.org/dc/elements/1.1/contributor">http://purl.org/dc/elements/1.1/contributor</a>	An entity responsible for making contributions to the content of the resource.
<a href="http://purl.org/dc/elements/1.1/coverage">http://purl.org/dc/elements/1.1/coverage</a>	The extent or scope of the content of the resource.
<a href="http://purl.org/dc/elements/1.1/creator">http://purl.org/dc/elements/1.1/creator</a>	An entity primarily responsible for making the content of the resource.
<a href="http://purl.org/dc/elements/1.1/date">http://purl.org/dc/elements/1.1/date</a>	A date associated with an event in the life cycle of the resource.
<a href="http://purl.org/dc/elements/1.1/description">http://purl.org/dc/elements/1.1/description</a>	An account of the content of the resource.
<a href="http://purl.org/dc/elements/1.1/format">http://purl.org/dc/elements/1.1/format</a>	The physical or digital manifestation of the resource.
<a href="http://purl.org/dc/elements/1.1/identifier">http://purl.org/dc/elements/1.1/identifier</a>	An unambiguous reference to the resource within a given context.
<a href="http://purl.org/dc/elements/1.1/language">http://purl.org/dc/elements/1.1/language</a>	A language of the intellectual content of the resource.
<a href="http://purl.org/dc/elements/1.1/publisher">http://purl.org/dc/elements/1.1/publisher</a>	An entity responsible for making the resource available.
<a href="http://purl.org/dc/elements/1.1/relation">http://purl.org/dc/elements/1.1/relation</a>	A reference to a related resource.
<a href="http://purl.org/dc/elements/1.1/rights">http://purl.org/dc/elements/1.1/rights</a>	Information about rights held in and over the resource.
<a href="http://purl.org/dc/elements/1.1/source">http://purl.org/dc/elements/1.1/source</a>	A reference to a resource from which the present resource is derived.
<a href="http://purl.org/dc/elements/1.1/subject">http://purl.org/dc/elements/1.1/subject</a>	The topic of the content of the resource.
<a href="http://purl.org/dc/elements/1.1/title">http://purl.org/dc/elements/1.1/title</a>	A name given to the resource.
<a href="http://purl.org/dc/elements/1.1/type">http://purl.org/dc/elements/1.1/type</a>	The nature or genre of the content of the resource.
<a href="http://purl.org/dc/terms/abstract">http://purl.org/dc/terms/abstract</a>	A summary of the content of the resource (refinement of description).

http://purl.org/dc/terms/modified	Date on which the resource was changed (refinement of date).
http://purl.org/dc/terms/spatial	Spatial characteristics of the intellectual content of the resource (refinement of coverage).
http://purl.org/dc/terms/temporal	Temporal characteristics of the intellectual content of the resource (refinement of coverage).
http://purl.org/dc/terms/valid	Date (often a range) of validity of a resource (refinement of date).

[OGC 05-025r3] asserts the following recommendations or rules with respect to the use of these slots:

- When a BP slot is added to a registry object, its name must match the name of the corresponding classification node.
- They may all be applied to any registry object.
- For simple values the slotType property should reference either an XML Schema datatype or a classification scheme if the values are selected from a controlled vocabulary; for complex values slotType must reference a suitable type definition.

The BP also provides a mapping between a csw:record and ebRIM:

CSW property name	ebRIM information item(s)
dc:identifier <sup>a</sup>	rim:RegistryObject/@id rim:RegistryObject/rim:ExternalIdentifier/@value
dc:type	rim:RegistryObject/@objectType
dc:title	rim:RegistryObject/rim:Name
dc:description	rim:RegistryObject/rim:Description
dc:subject <sup>b</sup>	rim:RegistryObject/rim:Slot/rim:ValueList/rim:Value
dc:format	rim:ExtrinsicObject/@mimeType
ows:BoundingBox <sup>c</sup>	rim:RegistryObject/rim:Slot
<p><sup>a</sup> The first identifier (in document order) is taken as the principal identifier; remaining identifiers are considered to be external identifiers.</p> <p><sup>b</sup> This accommodates keywords from uncontrolled vocabularies; multiple values give rise to multiple properties. The corresponding slot name is “http://purl.org/dc/elements/1.1/subject”. Note that formal classifications from ebRIM are not mapped.</p> <p><sup>c</sup> The corresponding slot name is “http://www.opengis.net/gml/Envelope”.</p>	

This demonstrates the following useful pattern when mapping domain attributes to ebRIM:

- A domain attribute should be mapped to an existing and suitable ebRIM information item if one exists. For example, the CSW property dc:description is mapped to rim:RegistryObject/rim:Description; dc:format to rim:ExtrinsicObject/@mimeType.

Further details on mapping domain attributes to ebRIM slots are provided in Section 7.2.6.

### 6.2.5 Classification Schemes

In addition to extending the ebRIM canonical ObjectType and AssociationType classification schemes, the BP adds the following new schemes:

- Services taxonomy (source: ISO 19119, Subclause 8.3)
- Slots (source: DCMI metadata terms)
- Country codes (source: ISO 3166-1)
- Geographical regions (source: UN Statistics Division)
- Feature codes (source: DIGEST v2.1, Part 4)

to allow classification of geospatial domain objects by such standard taxonomies.

### 6.2.6 Stored Queries

Stored queries are essentially parameterized, named queries that are included within an extension package to facilitate the discovery of the domain specific objects. The BP includes the following stored queries:

- findServices
- listExtensionPackages
- showStoredQueries
- getVersionHistory

The BP asserts the following rules or recommendations when defining stored queries:

- The supported method bindings must be described in the stored query definition (rim:AdhocQuery/rim:Description).
- The response must have either csw:GetRecordsResponse or rim:RegistryObjectList (if invoked using the GET method) as the document element.

- The id attribute specifies the stored query to execute; any parameters are passed in as child Slot elements.
- The value of the rim:QueryExpression/@queryLanguage attribute must refer to a node in the canonical query language scheme (which may be extended as needed to accommodate additional query languages).
- Formal query parameters are included as slots in the rim:AdhocQuery instance; parameter bindings in the query specification are indicated using *\$parameter-name* expressions.
- Within a definition, the slotType attribute should specify the appropriate value space for actual parameters.
- The rim:ValueList element must be empty, unless a default value is declared.

Further details on ebRIM stored queries are provided in Section 7.4.

### 6.3 Overview of Example Domain Model

A simplified CRS domain model is employed within this document to illustrate the application of guidelines presented in this document. It is described using UML in Figure 6-2. For simplicity only those attributes and role names which will be discussed are shown.

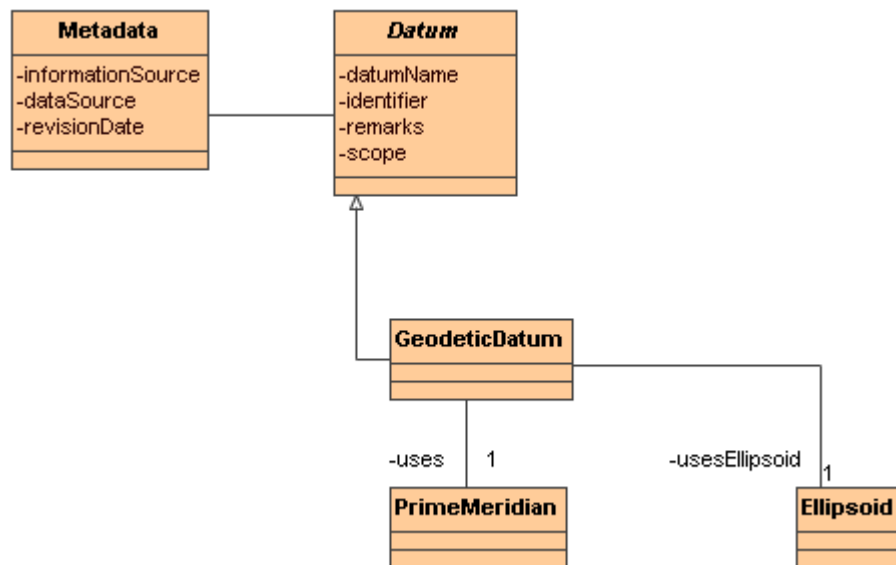


Figure 6-2: Example CRS domain model

With regard to the attributes:

- informationSource has a string value describing the source of the original information for the datum.
- dataSource has a value from a controlled list of data publishers.
- revisionDate has a date value indicating the latest revision date for the datum.
- datumName has a string value supplying the common name.
- identifier has a URN value uniquely identifying the datum.
- remarks has a string value describing the datum.
- scope has a string value describing the suitable application of the datum.

As the design of any target eBRIM model is also a function of the uses that it must serve, for the purposes of this document it must support the following scenario:

- GML instances of GeodeticDatum, Ellipsoid and PrimeMeridian shall be stored as separate, opaque and individual XML files within the local repository. Note that the Datum class is abstract.
- Discovery use cases include searching for all Datum instances by any of its attributes *except* scope.
- Queries should be able to return any associated objects of an identified datum.

Finally, the following WGS84 GeodeticDatum instance (expressed as a GML object) is used to show concrete mappings between the example domain CRS model and target eBRIM model.

```
<?xml version="1.0" encoding="UTF-8"?>
<GeodeticDatum gml:id="epsg6326"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:epsg="urn:x-ogp:spec:schema-xsd:EPSG:0.1:metadata"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns="http://www.opengis.net/gml">
  <metaDataProperty>
    <epsg:MetaData>
      <epsg:informationSource>NIMA TR8350.2 June 2004
revision.</epsg:informationSource>
      <epsg:dataSource>EPSG</epsg:dataSource>
      <epsg:revisionDate>2005-06-09</epsg:revisionDate>
    </epsg:MetaData>
  </metaDataProperty>
  <identifier codeSpace="EPSG">urn:x-
ogp:def:datum:EPSG:6326</identifier>
  <datumName>World Geodetic System 1984</datumName>
  <remarks>The WGS 84 datum is widely used.</remarks>
```

```

<scope>Satellite navigation.</scope>
<uses xlink:href="urn:x-ogp:def:meridian:EPSG:8901"/>
<usesEllipsoid xlink:href="urn:x-ogp:def:ellipsoid:EPSG:7030"/>
</GeodeticDatum>

```

## 7 Mapping a Domain Specific Model to a CSW.ebRIM Extension Package

### 7.1 Object Type Mapping

There are several questions to initially consider when mapping domain objects to ebRIM:

1. Which domain objects will be stored explicitly within the repository? All object types for them require a mapping to an ebRIM registry object as there must be a unique, associated registry object for each and every unique repository item.
2. Are any domain objects a composition of child objects? The object types for such child objects do not necessarily require mapping to ebRIM if separate instances of them will not be stored explicitly within the repository.
3. Do any of the domain object types represent or serve as taxonomic values for other domain objects? If so they may be represented best as/within a rim:ClassificationScheme as opposed to a rim:ExtrinsicObject.

In the case of the example CRS model:

1. Instances of GeodeticDatum, Ellipsoid and PrimeMeridian will be stored as explicit repository resources (GML instances as files). They must therefore be represented in the target ebRIM model as a rim:ExtrinsicObject.
2. Instances of Metadata will not be stored as explicit repository items. Mapping this object type to the target ebRIM model is therefore not required nor useful. Note that the Metadata attributes may still be exposed as attributes of the parent GeodeticDatum object (see Section 7.2).

The following OASIS [1] mapping algorithm should be used to map those domain objects that will be represented as classes within ebRIM:

1. **Direct Class Mapping To Rim:** First determine if there is a class in ebRIM that closely matches the class in the source model. For example, if a domain Organization object maps closely to rim:Organization, then its use is preferred.
2. **Mapping To ExtrinsicObject Sub-Class:** If no class in ebRIM is a good match then define a new sub-class of ExtrinsicObject class in ebRIM and map the source class to the new subclass.

As stated in OASIS [1] and per [OASIS ebRIM], to define a sub-class of ExtrinsicObject you must extend the canonical ObjectType ClassificationScheme and add a new

*ClassificationNode as a child or descendent of the canonical ClassificationNode for ExtrinsicObject in the ObjectType ClassificationScheme.*

As per [OASIS ebRIM] the id value must be unique within the CSW.ebRIM registry and of type URN.

### 7.1.1 Inheritance

When specializing the ObjectType classification scheme it is recommended that type hierarchies be maintained wherever possible. This is done, as per OASIS [1], *by defining a corresponding inheritance relationship among the ClassificationNodes defined when extending the ObjectType scheme.*

For example, the GeodeticDatum class inherits from the abstract Datum class. Both concepts and the hierarchy between them are created in the ObjectType classification scheme:

```
<rim:ClassificationNode code="Datum"
id="urn:x-ogp:csw-ebRIM:def:epsg:ObjectType:Datum"
parent='urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExtrinsicObject'>
<rim:Name>
  <rim:LocalizedString xml:lang="en" value="Datum"/>
</rim:Name>
</rim:ClassificationNode>

<rim:ClassificationNode code="GeodeticDatum"
  id="urn:x-ogp:csw-ebRIM:def:epsg:ObjectType:GeodeticDatum"
  parent='urn:x-ogp:csw-ebRIM:def:epsg:ObjectType:Datum' >
  <rim:Name>
    <rim:LocalizedString xml:lang="en" value="Geodetic Datum"/>
  </rim:Name>
</rim:ClassificationNode>
```

Note in the case of GeodeticDatum the parent attribute value is the id value for the Datum classification node, thereby establishing the hierarchy between these concepts. In addition, the Datum parent attribute value is the id value for the ExtrinsicObject classification node, since there is no existing concept of Datum within the canonical ObjectType classification scheme or its BP extensions.

The value of maintaining the Datum type hierarchy increases as the CRS model is expanded to include additional Datum subtypes, e.g. EngineeringDatum. It would then be possible to query and retrieve all concrete datum types by issuing a single query using the Datum parent objectType.

Once these classification nodes are submitted to a CSW.ebRIM registry, a rim:ExtrinsicObject within it could then be classified as a GeodeticDatum object. For example:



```

<rim:ExtrinsicObject mimeType="application/xml" isOpaque="false"
  id="urn:anId"
  objectType="urn:x-ogc:def:ObjectType:GML:GeodeticDatum">
  ...
</rim:ExtrinsicObject>

```

could describe the example WGS84 GeodeticDatum instance existing as a repository item.

## 7.2 Attribute Mapping

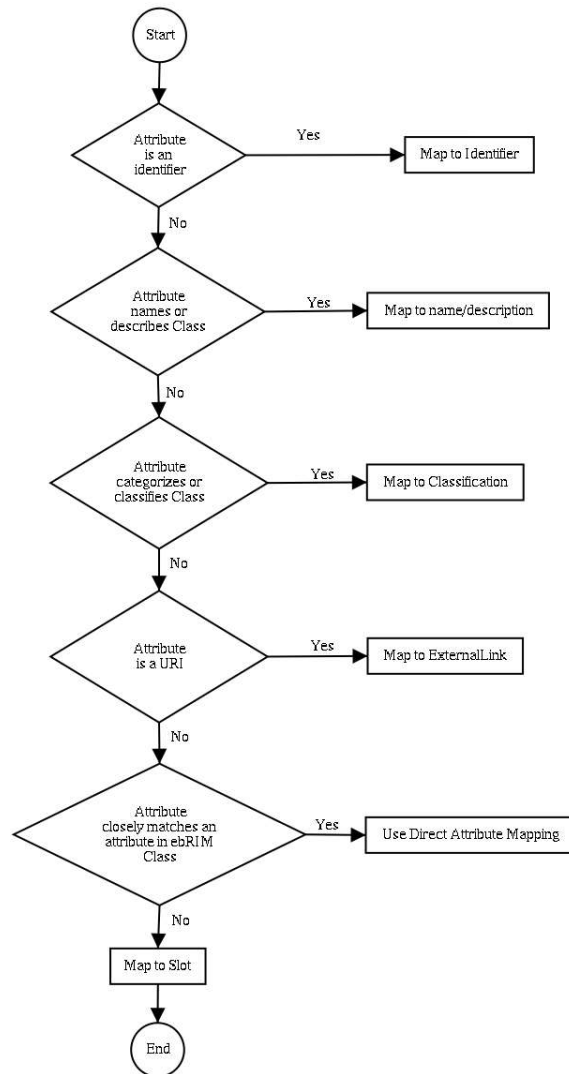
As with domain objects, there are several questions to initially consider when mapping domain attributes to ebRIM:

1. Which attributes are required by discovery use cases? These should generally be exposed in the ebRIM model to facilitate discovery in an efficient manner.
2. What is the minimum set of domain attributes that require exposure within ebRIM for discovery or other purposes? For example, ID, name, description, and geometry. As additional attributes can be easily added to ebRIM objects, it is recommended that only the minimum set be added or exposed initially, with any others added over time based on changing query requirements.

In the case of the example CRS model:

1. The attributes required by the discovery use cases are described in Overview of Example Domain Model.
2. As the domain scope attribute is not required for discovery use cases, it will not be exposed in the initial target ebRIM model. If over time its use for discovery was deemed helpful registry objects could be updated to expose this attribute.

Use of the following OASIS [1] mapping algorithm is recommended for mapping those domain attributes that will be mapped to ebRIM:



**Figure 7-1: OASIS Attribute Mapping Algorithm Flowchart**

OASIS [1] can be consulted for detailed guidance on applying this algorithm. The following subsections demonstrate its use by applying it to the source CRS domain model. Patterns or rules asserted by the BP are also demonstrated.

### 7.2.1 Map to Identifier

While the GeodeticDatum/identifier property uniquely identifies the datum and is of type URN, its value is not of UUID format recommended by OASIS [1] as the preferred identifier scheme. In addition, there are often implementation reasons for not using the source object identifier as the rim:ExtrinsicObject identifier. As such, this property is not mapped to rim:ExtrinsicObject/@id.

In addition, while mapping it to rim:ExtrinsicObject/@lid has merit, most known CSW.ebRIM implementations have not yet implemented versioning for which lid is used. As such, it is likely prudent to reserve use of lid during early CSW.ebRIM adoption.

Therefore it is mapped to rim:ExternalIdentifier within the EPSG namespace:

```
<rim:ExtrinsicObject mimeType="application/xml" isOpaque="false"
  id="urn:anId"
  objectType="urn:x-ogc:def:ObjectType:GML:GeodeticDatum">
  <rim:ExternalIdentifier
    objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ExternalIdentifier"
    identificationScheme="urn:x-ogp:spec:EPSG:0.1:dataset"
    id="ext-id-001"
    value="urn:x-ogp:def:axis:EPSG:6326"
    registryObject="eoid-ext-id-001"/>
</rim:ExtrinsicObject>
```

### 7.2.2 Map to Name/Description

GeodeticDatum/datumName provides a common name for the datum and is of type string, therefore is best mapped to rim:Name.

GeodeticDatum/remarks provides a description for the datum, and therefore is best mapped to rim:Description.

For example:

```
<rim:ExtrinsicObject mimeType="application/xml" isOpaque="false"
id="urn:anId" objectType="urn:x-ogp:csw-
ebRIM:def:epsg:ObjectType:GeodeticDatum">
  <rim:ExternalIdentifier
    objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ExternalIdentifier"
    identificationScheme="urn:x-ogp:spec:EPSG:0.1:dataset"
    id="ext-id-001"
    value="urn:x-ogp:def:axis:EPSG:6326"
    registryObject="eoid-ext-id-001"/>
  <rim:Name>
    <rim:LocalizedString xml:lang="en"
value="World Geodetic System 1984"/>
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en"
value="The WGS 84 datum is widely used."/>
  </rim:Description>
</rim:ExtrinsicObject>
```

### 7.2.3 Map to Classification

GeodeticDatum/metaDataProperty/epsg:MetaData/epsg:dataSource has values from a controlled list of data suppliers. There are several options for mapping this property to eBRIM:

- establish a rim:ClassificationScheme and map dataSource to the rim:ClassificationNode created within said scheme representing its value.
- map the data supplier as rim:Organization object, and dataSource as a rim:Association to said object.
- map it to the existing BP <http://purl.org/dc/elements/1.1/publisher> Slot.

For demonstration purposes the first option is chosen, providing the following example:

**Step 1:** Define the Data Source ClassificationScheme which includes a ClassificationNode for each source of CRS data:

```
<rim:ClassificationScheme id="urn:x-
ogp:def:ebrim:scheme:epsg:data-source"
  xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
  objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ClassificationScheme"
  isInternal="true"
  nodeType="UniqueCode">
  <rim:Name>
    <rim:LocalizedString value="Data Sources for CRS entities in
the EPSG dataset"/>
  </rim:Name>
  <rim:ClassificationNode id="urn:x-
ogp:def:ebrim:scheme:epsg:data-source:OGP"
    parent="urn:x-ogp:def:ebrim:scheme:epsg:data-source"
    objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ClassificationNode"
    code="OGP">
    <rim:Name>
      <rim:LocalizedString value="Oil & Gas Producers
(OGP)"/>
    </rim:Name>
  </rim:ClassificationNode>
  <rim:ClassificationNode id="urn:x-
ogp:def:ebrim:scheme:epsg:data-source:OilCo1"
    parent="urn:x-ogp:def:ebrim:scheme:epsg:data-source"
    objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ClassificationNode"
    code="OilCo1">
    <rim:Name>
      <rim:LocalizedString value="Major Oil Company #1"/>
    </rim:Name>
```

```

</rim:ClassificationNode>
<rim:ClassificationNode id="urn:x-
ogp:def:ebri:m:scheme:epsg:data-source:OilCo2"
  parent="urn:x-ogp:def:ebri:m:scheme:epsg:data-source"
  objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ClassificationNode"
  code="OilCo2">
  <rim:Name>
    <rim:LocalizedString value="Major Oil Company #2"/>
  </rim:Name>
</rim:ClassificationNode>
</rim:ClassificationScheme>

```

**Step 2:** When adding a GeodeticDatum instance to the registry, create a Classification instance which refers to a ClassificationNode in the Data Source scheme defined in Step 1. For example:

```

<rim:Classification id="classification-001"
  classificationScheme="urn:x-ogp:def:ebri:m:scheme:epsg:data-
source"
  classificationNode="urn:x-ogp:def:ebri:m:scheme:epsg:data-
source:OGP"
  classifiedObject="{datum-instance-identifier}"/>

```

#### 7.2.4 Map to ExternalLink

As there are no properties with URL values representing a link to an external resource, no properties are mapped to rim:ExternalLink.

#### 7.2.5 Direct Attribute Mapping

Other than the earlier mappings to rim:Name and rim:Description, there are no suitable mappings between the source attributes to ebRIM attributes.

#### 7.2.6 Mapping to Slot

In most cases specific domain attributes that require exposure within the target ebRIM model (for discovery purposes) will be mapped to Slots. They should be mapped to existing BP Slots as possible. In the case of the source CRS model, attributes are mapped to Slots as follows:

- GeodeticDatum/metaDataProperty/epsg:MetaData/epsg:informationSource maps to the BP <http://purl.org/dc/elements/1.1/source> Slot.
- GeodeticDatum/metaDataProperty/epsg:MetaData/epsg:revisionDate maps to the BP <http://purl.org/dc/terms/modified> Slot.

For example:

```

<rim:ExtrinsicObject mimeType="application/xml" isOpaque="false"
id="urn:anId" objectType="urn:x-ogp:csw-
ebrim:def:epsg:ObjectType:GeodeticDatum">
  <rim:ExternalIdentifier
    objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier"
    identificationScheme="urn:x-ogp:spec:EPSG:0.1:dataset"
    id="ext-id-001"
    value="urn:x-ogp:def:axis:EPSG:6326"
    registryObject="eoid-ext-id-001"/>
  <rim:Name>
    <rim:LocalizedString xml:lang="en"
value="World Geodetic System 1984"/>
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en"
value="The WGS 84 datum is widely used."/>
  </rim:Description>
  <rim:Slot name="http://purl.org/dc/elements/1.1/source">
    <rim:ValueList>
      <rim:Value>NIMA TR8350.2 June 2004 revision.</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="http://purl.org/dc/terms/modified">
    <rim:ValueList>
      <rim:Value>2005-06-09</rim:Value>
    </rim:ValueList>
  </rim:Slot>
</rim:ExtrinsicObject>

```

### 7.3 Association Mapping

OASIS [1] can be consulted for detailed guidance on mapping to associations. The following subsections demonstrate its use by applying it to the source CRS domain model. Patterns or rules asserted by the BP are also demonstrated.

As discussed in Section 6.2.3:

- A domain association not already represented in ebRIM should be mapped to a new subclass of ebRIM AssociationType.
- To define a new association type you must extend the canonical AssociationType ClassificationScheme and add a new ClassificationNode as a child or descendent of the AssociationType ClassificationScheme.
- To support more powerful queries and greater interoperability between CSW.ebRIM registries, the BP association types should be subclassed (by extending the canonical AssociationType classification scheme) whenever specialized versions of them are required.

For example:

- The Geodetic/uses association can be suitably mapped to the existing ebRIM ‘uses’ AssociationType.
- A new association type should be created for the Geodetic/usesEllipsoid association that extends the existing ebRIM ‘uses’ ClassificationNode in the canonical AssociationType classification scheme. Expressed as ebRIM this would appear as:

```
<rim:ClassificationNode
  xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
  id="urn:x-opg:def:csw-ebrim:AssociationType:UsesEllipsoid"
  parent="urn:oasis:names:tc:ebxml-regrep:AssociationType:Uses"
  code="UsesEllipsoid">
  <rim:Name>
    <rim:LocalizedString charset="UTF-8" value="UsesEllipsoid"/>
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString charset="UTF-8" value="Associates a
Datum with an Ellipsoid"/>
  </rim:Description>
</rim:ClassificationNode>
```

An example of a Geodetic/uses association for WGS84 GeodeticDatum object is shown below:

```
<!-- GeodeticDatum[6326] ==> Uses ==> Ellipsoid[7030] -->
<rim:Association
  xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
  id="urn:uuid:14976e7a-253f-4221-956b-627617ae2d19"
  objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Association"
  associationType="urn:oasis:names:tc:ebxml-
regrep:AssociationType:Uses"
  sourceObject="urn:x-opg:def:datum:EPSG:6326"
  targetObject="urn:x-opg:def:ellipsoid:EPSG:7030"/>
```

## 7.4 Stored Queries

As discussed in Section 6.2.6, stored queries are essentially parameterized, named queries that are included within an extension package to facilitate the discovery of the domain specific objects. They should be included whenever possible.

A stored query maps to a rim:AdhocQuery object. The reader should consult [OGC 05-025r3] and [OASIS ebRIM] for guidance on creating, using and implementing stored queries.

For the example CRS domain, the following stored query expressed as a CSW.ebRIM query statement would retrieve Datums by name:

```
<rim:AdhocQuery xmlns:rim="urn:oasis:names:tc:ebxml-
regrep:xsd:rim:3.0"
  id="urn:x-ogc:specification:csw-ebRim:Query:findDatumByName"
  objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:AdhocQuery">
  <rim:Slot name="searchText" slotType="xsd:anyURI">
    <rim:ValueList/>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString charset="UTF-8" xml:lang="en-US"
value="findDatumByName"/>
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString charset="UTF-8" xml:lang="en-US"
      value="Returns a list of Datum RegistryObjects, of any sub-
type (i.e. GeodeticDatum, EngineeringDatum etc.). The REQUIRED
parameter 'searchText' is the text used to match records."
    />
  </rim:Description>
  <rim:QueryExpression
    queryLanguage="urn:x-ogc:specification:csw-
ebRim:QueryLanguage:CSWQuery"
    xmlns:csw="http://www.opengis.net/cat/csw"
    xmlns:ogc="http://www.opengis.net/ogc">
    <csw:Query typeNames="ExtrinsicObject
ClassificationNode=cn1,cn2">
      <csw:ElementSetName
typeNames="ExtrinsicObject">full</csw:ElementSetName>
      <csw:Constraint version="1.1.0">
        <ogc:Filter>
          <ogc:And>
            <ogc:PropertyIsEqualTo>
<ogc:PropertyName>/ExtrinsicObject/@objectType</ogc:PropertyName>
              <ogc:PropertyName>/$cn1/@id</ogc:PropertyName>
            </ogc:PropertyIsEqualTo>
            <ogc:PropertyIsEqualTo>
              <ogc:PropertyName>/$cn1/@parent</ogc:PropertyName>
              <ogc:PropertyName>/$cn2/@id</ogc:PropertyName>
            </ogc:PropertyIsEqualTo>
            <ogc:PropertyIsEqualTo>
              <ogc:PropertyName>/$cn2/@id</ogc:PropertyName>
              <ogc:Literal>urn:x-
ogc:def:ObjectType:GML:Datum</ogc:Literal>
            </ogc:PropertyIsEqualTo>
          </ogc:Or>
          <ogc:PropertyIsLike escapeChar='/' wildcard='*'
singleChar='?'>

```



```

<ogc:PropertyName>/ExtrinsicObject/Name/LocalizedString/@value</o
gc:PropertyName>
    <ogc:Literal>*${searchText}*</ogc:Literal>
  </ogc:PropertyIsLike>
  <ogc:PropertyIsLike escapeChar='/' wildCard='*'
singleChar='?'>

<ogc:PropertyName>/ExtrinsicObject/Description/LocalizedString/@v
alue</ogc:PropertyName>
    <ogc:Literal>*${searchText}*</ogc:Literal>
  </ogc:PropertyIsLike>
  </ogc:Or>
  </ogc:And>
  </ogc:Filter>
  </csw:Constraint>
  </csw:Query>
  </rim:QueryExpression>
</rim:AdhocQuery>

```

## 7.5 Extension Package

As noted previously in Section 6.2, geospatial domain models should be modelled as extension packages that are mapped to a `rim:RegistryPackage`. This provides a useful means to insert, retrieve and otherwise manage all registry objects associated with the extension package.

To further simplify management of an extension package it is recommended that where a source extension package has a dependency on another target extension package the dependency should be explicitly identified. This can be easily modelled by instantiating a `rim:Association` of `rim:Association.AssociationType = "uses"` between the source and target extension packages. This dependency association must be an aggregation association. For example:

```

<!--
  Illustration of ExtensionPackage dependency modelled as an
  Association:
  Example ExtensionPackage ==> Uses ==> BasicPackage
-->
<rim:Association
  xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
  id="urn:uuid:0076222a-417d-4603-956b-325817bb2e18"
  objectType="urn:oasis:names:tc:ebxml-
regrep:Object:RegistryObject:Association"
  associationType="urn:oasis:names:tc:ebxml-
regrep:AssociationType:Uses"
  sourceObject="urn:x-ogc:def:ext-pkg:example-01"
  targetObject="urn:x-ogc:specification:csw-
ebrim:package:Basic"/>

```

Note that the extension package does not include any ebRIM registry objects related to actual domain objects stored as repository instances. Rather, it should include only those ebRIM artifacts necessary to support the inclusion of such instances. For the example domain model, this includes:

- Classification nodes that extend the canonical ObjectType and AssociationType classification schemes for the Datum and GeodeticDatum objects and the usesEllipsoid association.
- Classification scheme for the dataSource attribute.
- Association for the extension package dependency on the BP.
- Stored queries.

The result of mapping the example domain model to ebRIM is shown below.

```
<!--
Example Datum Extension Package

Note 1:  the "Uses" association type is already defined in the
CSW-ebRIM Basic Extension Package as:
urn:oasis:names:tc:ebxml-regrep:AssociationType:Uses
therefore, there is no need to define any association types
this extension package.

Note 2:  this example extension package is provided here wrapped
in
a CSW Transaction Insert statement in order that it may be
directly submitted to a registry.
-->
<Transaction
  xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
  xmlns="http://www.opengis.net/cat/csw"
  service="CSW"
  version="2.0.1"
  verboseResponse="true">
  <Insert>
    <rim:RegistryPackage id="urn:x-ogc:def:ext-pkg:example-01"
      objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:RegistryPackage">
      <rim:Name>
        <rim:LocalizedString xml:lang="en"
          value="Example Datum Extension Package"/>
      </rim:Name>
      <rim:VersionInfo versionName="1.0" comment="Last revised
2007-Jan-10"/>
      <rim:RegistryObjectList>
```

```

        <!--
            ObjectType ClassificationNodes (define new entity
types)
        -->
        <rim:ClassificationNode code="Datum"
            id="urn:x-ogc:def:ObjectType:GML:Datum"
            parent="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExtrinsicObject">
            <rim:Name>
                <rim:LocalizedString xml:lang="en"
value="Datum"/>
            </rim:Name>
            <rim:ClassificationNode code="GeodeticDatum"
                id="urn:x-
ogc:def:ObjectType:GML:GeodeticDatum"
                parent="urn:x-ogc:def:ObjectType:GML:Datum">
                <rim:Name>
                    <rim:LocalizedString xml:lang="en"
value="Geodetic Datum"/>
                </rim:Name>
            </rim:ClassificationNode>
            <!-- other Datum sub-types could be included here
        -->
        </rim:ClassificationNode>

        <rim:ClassificationNode code="Ellipsoid"
            id="urn:x-ogc:def:ObjectType:GML:Ellipsoid"
            parent="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExtrinsicObject">
            <rim:Name>
                <rim:LocalizedString xml:lang="en"
value="Ellipsoid"/>
            </rim:Name>
        </rim:ClassificationNode>

        <rim:ClassificationNode code="PrimeMeridian"
            id="urn:x-ogc:def:ObjectType:GML:PrimeMeridian"
            parent="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExtrinsicObject">
            <rim:Name>
                <rim:LocalizedString xml:lang="en"
value="Prime Meridian"/>
            </rim:Name>
        </rim:ClassificationNode>

        <!--
            AssociationType ClassificationNodes (define new
association types)
        -->
        <rim:ClassificationNode

```

```

        xmlns:rim="urn:oasis:names:tc:ebxml-
regrep:xsd:rim:3.0"
        id="urn:x-opg:def:csw-
ebrim:AssociationType:UsesEllipsoid"
        parent="urn:oasis:names:tc:ebxml-
regrep:AssociationType:Uses"
        code="UsesEllipsoid">
        <rim:Name>
            <rim:LocalizedString charset="UTF-8"
value="UsesEllipsoid"/>
        </rim:Name>
        <rim:Description>
            <rim:LocalizedString charset="UTF-8"
value="Associates a Datum with an Ellipsoid"/>
        </rim:Description>
    </rim:ClassificationNode>

    <!--
        Classification scheme for data sources
    -->
    <rim:ClassificationScheme id="urn:x-
ogp:def:ebrim:scheme:epsg:data-source"
        xmlns:rim="urn:oasis:names:tc:ebxml-
regrep:xsd:rim:3.0"
        objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ClassificationScheme"
        isInternal="true"
        nodeType="UniqueCode">
        <rim:Name>
            <rim:LocalizedString value="Data Sources for
CRS entities in the EPSG dataset"/>
        </rim:Name>
        <rim:ClassificationNode id="urn:x-
ogp:def:ebrim:scheme:epsg:data-source:OGP"
            parent="urn:x-opg:def:ebrim:scheme:epsg:data-
source"
                objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ClassificationNode"
                code="OGP">
                <rim:Name>
                    <rim:LocalizedString value="Oil & Gas
Producers (OGP)"/>
                </rim:Name>
            </rim:ClassificationNode>
        <rim:ClassificationNode id="urn:x-
ogp:def:ebrim:scheme:epsg:data-source:OilCo1"
            parent="urn:x-opg:def:ebrim:scheme:epsg:data-
source"
                objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ClassificationNode"

```

```

        code="OilCo1">
        <rim:Name>
            <rim:LocalizedString value="Major Oil
Company #1"/>
        </rim:Name>
    </rim:ClassificationNode>
    <rim:ClassificationNode id="urn:x-
ogp:def:ebrim:scheme:epsg:data-source:OilCo2"
parent="urn:x-ogp:def:ebrim:scheme:epsg:data-
source"
objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ClassificationNode"
code="OilCo2">
    <rim:Name>
        <rim:LocalizedString value="Major Oil
Company #2"/>
    </rim:Name>
</rim:ClassificationNode>
</rim:ClassificationScheme>

<!--
    ExtensionPackage dependency on BasicPackage
-->
<rim:Association
xmlns:rim="urn:oasis:names:tc:ebxml-
regrep:xsd:rim:3.0"
id="urn:uuid:0076222a-417d-4603-956b-325817bb2e18"
objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Association"
associationType="urn:oasis:names:tc:ebxml-
regrep:AssociationType:Uses"
sourceObject="urn:x-ogc:def:ext-pkg:example-01"
targetObject="urn:x-ogc:specification:csw-
ebrim:package:Basic"/>

<!--
    Stored queries for discovery purposes
-->
<rim:AdhocQuery xmlns:rim="urn:oasis:names:tc:ebxml-
regrep:xsd:rim:3.0"
id="urn:x-ogc:specification:csw-
ebrim:Query:findDatumByName"
objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:AdhocQuery">
    <rim:Slot name="searchText" slotType="xsd:anyURI">
        <rim:ValueList/>
    </rim:Slot>
    <rim:Name>
        <rim:LocalizedString charset="UTF-8"
xml:lang="en-US" value="findDatumByName"/>
    </rim:Name>

```

```

        <rim:Description>
            <rim:LocalizedString charset="UTF-8"
xml:lang="en-US"
                value="Returns a list of Datum
RegistryObjects, of any sub-type (i.e. GeodeticDatum, EngineeringDatum
etc.). The REQUIRED parameter 'searchText' is the text used to match
records."
            />
        </rim:Description>
        <rim:QueryExpression
            queryLanguage="urn:x-ogc:specification:csw-
ebrim:QueryLanguage:CSWQuery"
            xmlns:csw="http://www.opengis.net/cat/csw"
            xmlns:ogc="http://www.opengis.net/ogc">
            <csw:Query typeNames="ExtrinsicObject
ClassificationNode=cn1,cn2">
                <csw:ElementSetName
typeNames="ExtrinsicObject">full</csw:ElementSetName>
                <csw:Constraint version="1.1.0">
                    <ogc:Filter>
                        <ogc:And>
                            <ogc:PropertyIsEqualTo>
                                <ogc:PropertyName>/ExtrinsicObject/@ObjectType</ogc:PropertyName>
                                <ogc:PropertyIsEqualTo>
                                    <ogc:PropertyName>/${cn1}/@id</ogc:PropertyName>
                                    </ogc:PropertyIsEqualTo>
                                    <ogc:PropertyIsEqualTo>
                                        <ogc:PropertyName>/${cn1}/@parent</ogc:PropertyName>
                                        <ogc:PropertyIsEqualTo>
                                            <ogc:PropertyName>/${cn2}/@id</ogc:PropertyName>
                                            </ogc:PropertyIsEqualTo>
                                            <ogc:PropertyIsEqualTo>
                                                <ogc:PropertyName>/${cn2}/@id</ogc:PropertyName>
                                                <ogc:Literal>urn:x-
ogc:def:ObjectType:GML:Datum</ogc:Literal>
                                                </ogc:PropertyIsEqualTo>
                                                <ogc:Or>
                                                    <ogc:PropertyIsLike
escapeChar='/' wildCard='*' singleChar='?'>
                                <ogc:PropertyName>/ExtrinsicObject/Name/LocalizedString/@value</ogc:Pr
opertyName>
                                <ogc:Literal>*${searchText}*</ogc:Literal>
                                    </ogc:PropertyIsLike>
                                    <ogc:PropertyIsLike
escapeChar='/' wildCard='*' singleChar='?'>

```

```

<ogc:PropertyName>/ExtrinsicObject/Description/LocalizedString/@value<
/ogc:PropertyName>

<ogc:Literal>*${searchText}*</ogc:Literal>
                                </ogc:PropertyIsLike>
                                </ogc:Or>
                                </ogc:And>
                                </ogc:Filter>
                                </csw:Constraint>
                                </csw:Query>
                                </rim:QueryExpression>
                                </rim:AdhocQuery>

                                </rim:RegistryObjectList>
                                </rim:RegistryPackage>
                                </Insert>
</Transaction>

```

## 8 Mapping Templates

OASIS [2] describes mapping templates (tables) that can be used by those new to ebRIM to help map source models to ebRIM. Examples of these tables are included in Annex C.

## Annex A

### 1 OWS-4 Sensor Web Enablement Catalog Resource Profile

#### 1.1 Background

OGC's Interoperability Program is a global, hands-on and collaborative prototyping program designed to rapidly develop, test and deliver proven candidate specifications into OGC's Specification Program, where they are formalized for public release. In OGC's Interoperability Initiatives, an international team of technology providers' work together to solve specific geo-processing interoperability problems posed by the initiative's sponsoring organizations. OGC Interoperability Initiatives include test beds, pilot projects, interoperability experiments, and interoperability support services all designed to encourage rapid development, testing, validation and adoption of open, consensus based standards specifications.

In the Fall of 2005, the OGC issued a call for sponsors for an OGC OWS-4 Interoperability initiative testbed activity to advance OGC's open framework for interoperability in the geospatial industry. Three meetings were conducted with potential OWS-4 sponsors to review the OGC technical baseline, to discuss OWS 3 results, and to identify OWS 4 requirements. Sponsors have expressed keen interest in advancing standards for sensor webs, geospatial digital rights management, geospatial semantics and knowledge management. After analyzing the sponsors input, the OGC Interoperability Team recommended to the sponsors that the content of the OWS-4 initiative be organized around the following 7 threads:

3. Sensor Web Enablement (SWE)
4. Geo Processing Workflow (GPW)
5. Geo Decision Support (GeoDSS)
6. Geo-Digital Rights Management (GeoDRM)
7. CAD / GIS / BIM
8. OGC Location Services (OpenLS)
9. Compliance Testing (CITE)

The first one, the Sensor Web subtask, will continue to mature the existing set of SWE work items to enable the federation of sensors, platforms and management infrastructure into a single sensor enterprise. This enterprise will enable the discovery and tasking of sensors as well as the delivery of sensor measurements regardless of sensor type and controlling organization. The ultimate vision is of a sensor market place where users can



identify, evaluate, select and request a sensor collection regardless of sensor type, platform or owner.

## 1.2 Abstract

This document will describe the integration of a CSW eBRIM Catalogue in the OWS-4 SWE infrastructure. Since a lot of resources must be managed in the SWE context - like Services (i.e. SOS, SPS, SAS and WNS), metadata files (SensorML, TransducerML) or resources (GML dictionaries defining units of measures or phenomena) – a CSW can be used to index all these informations and provide search and retrieval facilities to clients.

## 1.3 Related Information

### 1.3.1 Use Cases for Catalog Role in Managing SWE Resources

- [Use cases](#)
- [Hot Spot Use cases](#)
- [Registry Role](#)

### 1.3.2 Source Materials

#### 1.3.2.1 General

- [SweImplementations](#)

#### 1.3.2.2 Sensor Observation Service (SOS)

- [SweIonicExperiment](#)

#### 1.3.2.3 Sensor Planning Service (SPS)

#### 1.3.2.4 SensorML

- [Instances](#)
- [Instances \(SVN\)](#)

#### 1.3.2.5 TransducerML

- [Instances](#)

#### 1.3.2.6 Unit of Measures Dictionary

- [Instance \(SVN\)](#)

#### 1.3.2.7 Phenomenon Dictionary

- [Instance \(SVN\)](#)

## 1.4 Sensor Observation Service Management

### 1.4.1 Material Description

Sensor Observation Service (SOS) is part of the OGC SWE group of specifications which provides an interface (through a OGC compliant webservice) for managing deployed sensors and retrieving sensor data.

#### 1.4.2 Related Definitions

Concepts related to SOS are:

**Observation:** an event with a result which has a value describing some phenomenon,

**Observation Offering:** a logical grouping of observations offered by a service that are related in some way,

**Observed Value:** a value describing a natural phenomenon, which may use one of a variety of scales including nominal, ordinal ratio and interval.

**Sensor:** an entity capable of observing a phenomenon and returning an observed value. A sensor can be an instrument or a living organism (e.g. a person).

#### Feature Of Interest

#### 1.4.3 Mapping of source materials into ebRIM objects

Every SOS is represented in the CSW ebRIM catalogues by a structure of ebRIM objects. This structure is intended to highlight the most representative data of the webservice and to optimize most popular searches.

So the ebRIM structure must optimize this kind of typical searches from clients, or any combination of them:

Give me all SOS (or Observation Offerings) provided by this organization,

Give me all SOS (or Observation Offerings) providing observations of this phenomenon,

Give me all SOS (or Observation Offerings) providing observations within this bounding box,

Give me all SOS (or Observation Offerings) providing observations taken in this temporal interval,

Give me all SOS (or Observation Offerings) related to this keyword,

Give me all SOS (or Observation Offerings) provided by this organization and providing observations of this phenomenon, taken in this time interval within this bounding box...

The harvesting (the way a resource is pulled and indexed in the catalogue) of the SOS is based on the capabilities document. Like for others OGC webservices, the ServiceIdentification, ServiceProvider and OperationsMetadata parts are mapped to a structure including these ebRIM objects : Service, ServiceBinding, Associations, Organization, User and ExtrinsicObjects

The following table defines the mapping between the ServiceIdentification section of the Capabilities and ebRIM objects

SOS Capabilities Element	EbRIM Element	Type
/Capabilities/ServiceIdentification/Title	Service/Name/LocalizedString/@value	String
/Capabilities/ServiceIdentification/Abstract	Service/Description/LocalizedString/@value	String
/Capabilities/ServiceIdentification/Keywords/Keyword	Service/Slot[@name='Keywords']/ValueList/Value[]	String
/Capabilities/ServiceIdentificationServiceType	Service/Slot[@name='Identifier']/ValueList/Value[1]	String
/Capabilities/ServiceIdentification/Fees	Service/Slot[@name='Fees']/ValueList/Value[1]	String
/Capabilities/ServiceIdentification/AccessConstraints	Service/Slot[@name='AccessConstraints']/ValueList/Value[1]	String
Date of the harvesting in the Catalog	Service/Slot[@name='PublicationDate']/ValueList/Value[1]	ISO8601 Date

Actually, the Service object is the main object, the entry point to the SOS in the Catalogue. It is classified to the SOS ClassificationNode of the OGC Services taxonomy.

The following table defines the mapping between the ServiceProvider section of the Capabilities and ebRIM objects

SOS Capabilities Element	EbRIM Element	Type
/Capabilities/ServiceProvider/ProviderSite/@href	Service/Slot[@name='OnlineResource']/ValueList/Value[1]	URL
/Capabilities/ServiceProvider/ProviderName	Organization/Name/LocalizedString/@value (linked to the source Service object by a 'OffersService' Association)	String
/Capabilities/ServiceProvider/ServiceContact/IndividualName	User/PersonName (the primaryContact of the organization)	String
/Capabilities/ServiceProvider/ServiceContact/PositionName	User/Slot[@name='Position']/ValueList/Value[1]	String
/Capabilities/ServiceProvider/ServiceContact/ContactInfo/Phone/Voice	User/TelephoneNumber	String
/Capabilities/ServiceProvider/ServiceContact/ContactInfo/Address/DeliveryPoint	User/Address/@street	String
/Capabilities/ServiceProvider/ServiceContact/ContactInfo/Address/City	User/Address/@city	String
/Capabilities/ServiceProvider/ServiceContact/ContactInfo/Address/AdministrativeArea	User/Address/@stateOrProvince	String
/Capabilities/ServiceProvider/ServiceContact/ContactInfo/Address/PostalCode	User/Address/@postalCode	String
/Capabilities/ServiceProvider/ServiceContact/ContactInfo/Address/Country	User/Address/@country	String
/Capabilities/ServiceProvider/ServiceContact/ContactInfo/Address/ElectronicMailAddress	User/EmailAddress/@address	String

The following table defines the mapping between the OperationsMetadata section of the Capabilities and ebRIM objects

SOS Capabilities Element	EbRIM Element	Type
/Capabilities/OperationsMetadata/Operation[@name='GetCapabilities']/DCP/HTTP/GET/@href	ServiceBinding/@accessURI (linked to the Service object)	URL
/Capabilities/OperationsMetadata/Operation/@name	ExtrinsicObject[@objectType='Service_Operation']/Name/LocalizedString/@value (linked to the source Service object by an 'Implements' Association)	String
/Capabilities/OperationsMetadata/Operation/DCP/HTTP/GET/@href	ExtrinsicObject[@objectType='Service_Operation']/Slot[@name='URLGet']/ValueList/Value[1]	URL

/Capabilities/OperationsMetadata/Operation/DCP/HTTP/POST/@href	ExtrinsicObject[@objectType='Service_Operation']/Slot[@name='URLPost']/ValueList/Value[1]	URL
--	---	-----

The Contents section of the capabilities is obviously also mapped to an ebRIM structure. The Observation Offering list is transformed to a set of ExtrinsicObjects representing each Observation Offering and linked to the Service object by an Association.

These ExtrinsicObjects, representing each Observation Offering, keep a set of information like the name, procedures, observed properties, result formats and result models. They also keep spatial and temporal information. Each FeatureOfInterest is also modeled by an ExtrinsicObject linked to the Observation Offering ExtrinsicObject by an Association.

The following table defines the mapping between the Contents section of the Capabilities and ebRIM objects

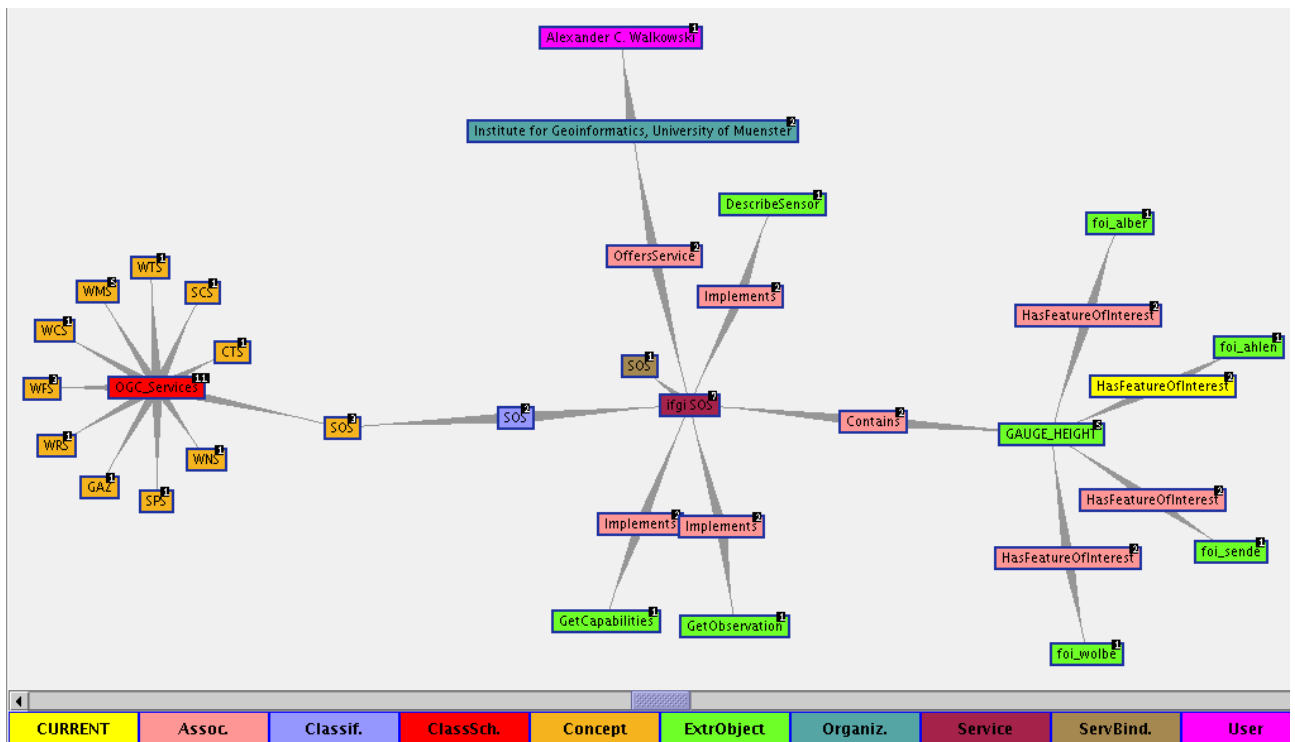
SOS Capabilities Element	EbRIM Element	Type
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/@id	ExtrinsicObject[@objectType='ObservationOffering']/Name/LocalizedString/@value (linked to the source Service object by a 'Contains' Association)	String
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/name	ExtrinsicObject[@objectType='ObservationOffering']/Description/LocalizedString/@value	String
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/boundedBy	ExtrinsicObject[@objectType='ObservationOffering']/Slot[@name='FootPrint']/ValueList/Value[1] (Transformed in EPSG:4326)	Geometry
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/eventTime/TimePeriod/beginPosition	ExtrinsicObject[@objectType='ObservationOffering']/Slot[@name='BeginTime']/ValueList/Value[1]	ISO8601 Date
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/eventTime/TimePeriod/end Position	ExtrinsicObject[@objectType='ObservationOffering']/Slot[@name='EndTime']/ValueList/Value[1]	ISO8601 Date
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/procedure/@href	ExtrinsicObject[@objectType='ObservationOffering']/Slot[@name='Procedures']/ValueList/Value[]	String
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/observedProperty/@href	ExtrinsicObject[@objectType='ObservationOffering']/Slot[@name='ObservedProperties']/ValueList/Value[]	String
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/resultFormat	ExtrinsicObject[@objectType='ObservationOffering']/Slot[@name='ResultFormats']/ValueList/Value[]	String
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/resultModel	ExtrinsicObject[@objectType='ObservationOffering']/Slot[@name='ResultModels']/ValueList/Value[]	String
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/featureOfInterest/FeatureCollection/featureMember/GeoReferenceableFeature/@id	ExtrinsicObject[@objectType='FeatureOfInterest']/Name/LocalizedString/@value (linked to the source ObservationOffering ExtrinsicObject object by a 'HasFeatureOfInterest' Association)	String
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/featureOfInterest/FeatureCollection/featureMember/GeoReferenceableFeature/name	ExtrinsicObject[@objectType='FeatureOfInterest']/Description/LocalizedString/@value	String
/Capabilities/Contents/ObservationOfferingList/ObservationOffering/featureOfInterest/FeatureCollection/featureMember/GeoReferenceableFeature/location	ExtrinsicObject[@objectType='FeatureOfInterest']/Slot[@name='FootPrint']/ValueList/Value[1] (Transformed in EPSG:4326)	Geometry

The following diagram represents an SOS instance mapped to an ebRIM structure.

On the left of the diagram, we can see the OGC services taxonomy, composed by a ClassificationScheme (red box) with its children ClassificationNode (orange boxes).

At the center of the diagram, we can see the Service object (purple box) linked to the SOS ClassificationNode with a Classification (blue box) and the ServiceBinding (brown box) attached to this Service. Service has also a set of Associations (light pink boxes) of 3 different AssociationType. The first AssociationType, OffersService, is the Association to the Organization object (dark cyan box) itself attached to the User object (flashy pink box). The second AssociationType, Implements, is the Associations to the ExtrinsicObject (green boxes) representing the Operation of the Service. The last AssociationType, Contains, is the Association to the ExtrinsicObject (green box) representing the Observations of the ObservationOffering.

At the right of the diagram, we can see the ebRIM structure representing the Observation. So we have the ExtrinsicObject (green box) representing the Observations of the ObservationOffering, linked through Associations of type HasFeatureOfInterest (light pink boxes) to ExtrinsicObject (green boxes) representing FeatureOfInterest of the Observation.



#### 1.4.4 Query extensions / standard queries

The following query retrieves all registered SOS:

```
<?xml version="1.0" encoding="UTF-8"?>
<csw:GetRecords
  xmlns:csw="http://www.opengis.net/cat/csw"
  xmlns:ogc="http://www.opengis.net/ogc"
  version="2.0.0">
```

```

outputSchema="EBRIM">
<csw:Query typeNames="Service Concept Classification">
  <csw:ElementName>/Service</csw:ElementName>
  <csw:Constraint version="1.0.0">
    <ogc:Filter>
      <ogc:And>
        <ogc:PropertyIsEqualTo>

<ogc:PropertyName>/Concept/Name/LocalizedString/@value</ogc:Prope
rtyName>
          <ogc:Literal>SOS</ogc:Literal>
        </ogc:PropertyIsEqualTo>
        <ogc:PropertyIsEqualTo>
          <ogc:PropertyName>/Concept/@id</ogc:PropertyName>

<ogc:PropertyName>/Classification/@classificationNode</ogc:Proper
tyName>
          </ogc:PropertyIsEqualTo>
        <ogc:PropertyIsEqualTo>
          <ogc:PropertyName>/Service/@id</ogc:PropertyName>

<ogc:PropertyName>/Classification/@classifiedObject</ogc:Property
Name>
          </ogc:PropertyIsEqualTo>
        </ogc:And>
      </ogc:Filter>
    </csw:Constraint>
  </csw:Query>
</csw:GetRecords>

```

The following query retrieves all Observation Offerings intersecting a given bounding box:

```

<?xml version="1.0" encoding="UTF-8"?>
<csw:GetRecords
  xmlns:csw="http://www.opengis.net/cat/csw"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:gml="http://www.opengis.net/gml"
  version="2.0.0"
  outputSchema="EBRIM">
  <csw:Query typeNames="ExtrinsicObject">
    <csw:ElementName>/ExtrinsicObject</csw:ElementName>
    <csw:Constraint version="1.0.0">
      <ogc:Filter>
        <ogc:And>
          <ogc:PropertyIsEqualTo>

<ogc:PropertyName>/ExtrinsicObject/@objectType</ogc:PropertyName>
          <ogc:Literal>ObservationOffering</ogc:Literal>
        </ogc:PropertyIsEqualTo>
        <ogc:Intersects>

```

```

<ogc:PropertyName>/ExtrinsicObject/Slot[@name="FootPrint"]/ValueList/Value[1]
  </ogc:PropertyName>
  <gml:Box srsName="EPSG:4326">
    <gml:coordinates>5.210,50.890
5.965,55.125</gml:coordinates>
  </gml:Box>
  </ogc:Intersects>
  </ogc:And>
  </ogc:Filter>
  </csw:Constraint>
</csw:Query>
</csw:GetRecords>

```

The following query retrieves all Observation Offerings containing data for a given date:

```

<?xml version="1.0" encoding="UTF-8"?>
<csw:GetRecords
  xmlns:csw="http://www.opengis.net/cat/csw"
  xmlns:ogc="http://www.opengis.net/ogc"
  version="2.0.0"
  outputSchema="EBRIM">
  <csw:Query typeNames="ExtrinsicObject">
    <csw:ElementName>/ExtrinsicObject</csw:ElementName>
    <csw:Constraint version="1.0.0">
      <ogc:Filter>
        <ogc:And>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/ExtrinsicObject/@objectType</ogc:PropertyName>
            <ogc:Literal>ObservationOffering</ogc:Literal>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsLessThanOrEqualTo>
            <ogc:PropertyName>/ExtrinsicObject/Slot[@name="BeginTime"]/ValueList/Value[1]
            </ogc:PropertyName>
            <ogc:Literal>2006-01-01T00:00:00Z</ogc:Literal>
          </ogc:PropertyIsLessThanOrEqualTo>
          <ogc:PropertyIsGreaterThanOrEqualTo>
            <ogc:PropertyName>/ExtrinsicObject/Slot[@name="EndTime"]/ValueList/Value[1]
            </ogc:PropertyName>
            <ogc:Literal>2006-01-01T00:00:00Z</ogc:Literal>
          </ogc:PropertyIsGreaterThanOrEqualTo>
        </ogc:And>
      </ogc:Filter>
    </csw:Constraint>
  </csw:Query>
</csw:GetRecords>

```

**1.4.5 Output schemas / formats**

**1.4.6 Resource lifecycle**

**1.4.7 Transaction corresponding to SOS instances harvest**

transaction1.xml: SOS at <http://eol.geobliki.com/sos>

transaction2.xml: SOS at <http://mars.uni-muenster.de:8080/52nSOSv2/sos>

**1.4.8 Experiment**

An Catalogue able to harvest SOS has been provided by Ionic for the OWS-4 Demo :  
<http://dev.ionicsoft.com:8082/ows4catalog/wrs/WRS?request=getCapabilities>

A JSP with some queries samples is also available :

<http://dev.ionicsoft.com:8082/ows4catalog/elements/sos.jsp>

The following diagram shows how these 3 SOS instances are mapped into the eBRIM model and linked to the OGC Services Taxonomy.

<http://ren.3eti.net:8080/ogc2/GetCapabilities.ogc>

<http://vast.uah.edu:8080/ows-dev/scoopTest>

<http://imasoft.spotimage.com/52nSOSv2/sos>





This metadata includes identifiers, classifiers, constraints (time, legal, and security), capabilities, characteristics, contacts, and references, in addition to inputs, outputs, parameters, and system location.

Essential elements of SensorML documents are:

**System:** Composite model of a group or array of components, which can include detectors, actuators, or sub-systems. A System relates a Process Chain to the real world and therefore provides additional definitions regarding relative positions of its components and communication interfaces.

**Sensor:** Specific type of System representing a complete Sensor. This could be for example a complete airborne scanner which includes several Detectors (one for each band).

**ProcessModel:** Atomic processing block usually used within a more complex Process Chain. It is associated to a Process Method which defines the process interface as well as how to execute the model. It also precisely defines its own inputs, outputs and parameters.

**ProcessChain:** Composite processing block consisting of interconnected sub-processes, which can in turn be Process Models or Process Chains. A process chain also includes possible data sources as well as connections that explicitly link input and output signals of sub-processes together. It also precisely defines its own inputs, outputs and parameters.

**Detector:** Atomic part of a composite Measurement System defining sampling and response characteristic of a simple detection device. A detector has only one input and one output, both being scalar quantities. More complex Sensors such as a frame camera which are composed of multiple detectors can be described as a detector group or array using a System or Sensor.

**Actuator:** An actuator, the reverse of a sensor, is a device that converts an electrical signal to an action.

**PropertyList:** List of properties

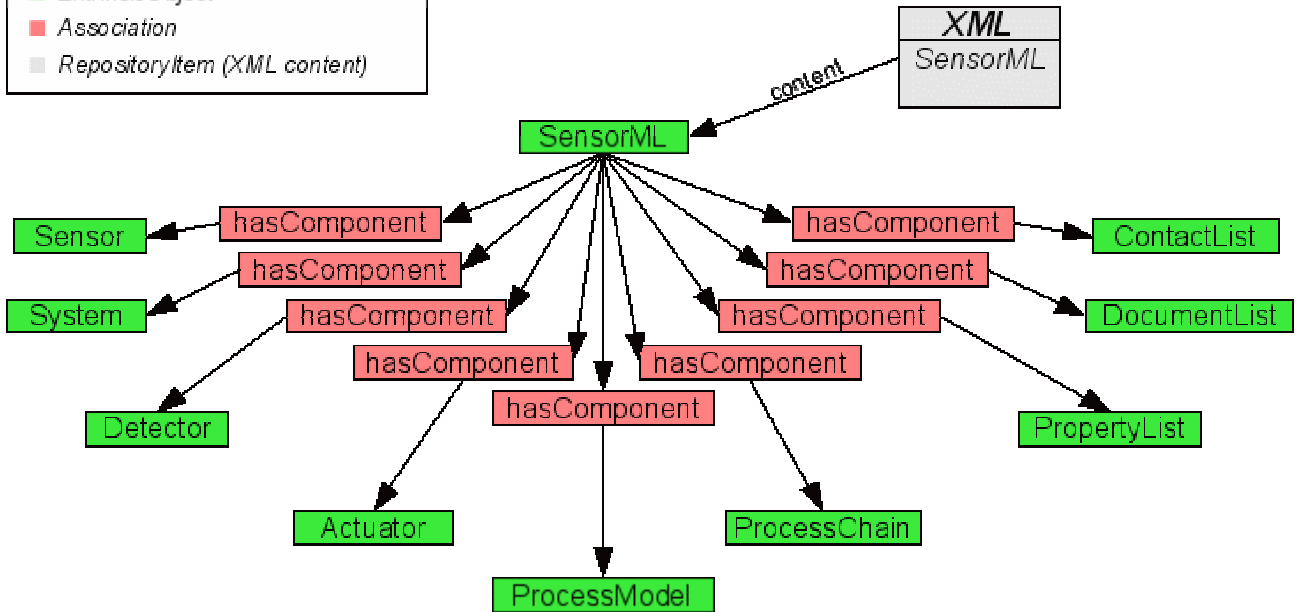
**DocumentList:** List of documents

**ContactList:** List of contacts

Each SensorML harvested in Catalog is represented by an ExtrinsicObject, having as content (related RepositoryItem) the XML file itself, and linked through Associations to additional ExtrinsicObjects representing essential elements, as described in the following schema:

ebRIM Structure representing a SensorML instance

- ExtrinsicObject
- Association
- RepositoryItem (XML content)



Following tables define the mapping between SensorML instances and the corresponding ebRIM structure.

**1.5.2.1 The SensorML root element:**

SensorML Element	EbRIM Element	Type
/SensorML	ExtrinsicObject[@objectType='SensorML']	String

**1.5.2.2 The System element:**

SensorML Element	EbRIM Element	Type
/SensorML/System	ExtrinsicObject[@objectType='System'] linked to the SensorML ExtrinsicObject by an 'hasComponent' Association	String
/SensorML/System/@id	ExtrinsicObject[@objectType='System']/Slot[@name='Identifier']/ValueList/Value[1]	String
/SensorML/System/identification/IdentifierList/identifier[@name='uid']/Term	ExtrinsicObject[@objectType='System']/@uuid	String
/SensorML/System/identification/IdentifierList/identifier[@name='short_name']/Term	ExtrinsicObject[@objectType='System']/Name/LocalizedString/@value	String
/SensorML/System/identification/IdentifierList/identifier[@name='long_name']/Term	ExtrinsicObject[@objectType='System']/Description/LocalizedString/@value	String
/SensorML/System/capabilities/PropertyList/property/@name	ExtrinsicObject[@objectType='System']/Slot[@name='Capabilities']/ValueList/Value[]	String
/SensorML/System/characteristics/PropertyList/property/@name	ExtrinsicObject[@objectType='System']/Slot[@name='Characteristics']/ValueList/Value[]	String
/SensorML/System/contact/ResponsibleParty/organizationName	Organization/Name/LocalizedString/@value linked to the System ExtrinsicObject by an 'providedBy' Association	String
/SensorML/System/contact/ResponsibleParty/individualName	User/PersonName/@fullName , registered as User of the providing Organization	String
/SensorML/System/contact/ResponsibleParty/contact	User/TelephoneNumbers	/

Info/phone		
/SensorML/System/contact/ResponsibleParty/contact Info/address	User/PostalAddresses	/

Let's talk about his XML part of the System definition:

```
<classification>
  <ClassifierList>
    <classifier name="sensor_type">
      <Term qualifier="urn:ogc:classifier:sensorType">Positioning
System</sml:Term>
    </classifier>
    <classifier name="application">
      <Term qualifier="urn:ogc:classifier:application">Navigation
Measurements</sml:Term>
    </classifier>
  </ClassifierList>
</classification>
```

Actually, every term's qualifier is represented in the catalogue as a taxonomy. So in this example, 2 Classifications are created. The first one links the System ExtrinsicObject to the 'Positioning System' ClassificationNode, child of the 'urn:ogc:classifier:sensorType' ClassificationScheme. The second one links the same System ExtrinsicObject to the 'Navigation Measurements' ClassificationNode, child of the 'urn:ogc:classifier:application' ClassificationScheme.

Such ClassificationScheme or ClassificationNode are created during the harvesting process if they don't exist yet.

**1.5.2.3 The Sensor element:**

SensorML Element	EbRIM Element	Type
/SensorML/Sensor	ExtrinsicObject[@objectType='Sensor'] linked to the SensorML ExtrinsicObject by an 'hasComponent' Association	String
/SensorML/Sensor/@id	ExtrinsicObject[@objectType='Sensor']/Slot[@name='Identifier']/ValueList/Value[1]	String
/SensorML/Sensor/identification/IdentifierList/identifier[@name='uid']/Term	ExtrinsicObject[@objectType='Sensor']/@uuid	String
/SensorML/Sensor/identification/IdentifierList/identifier[@name='short_name']/Term	ExtrinsicObject[@objectType='Sensor']/Name/LocalizedString/@value	String
/SensorML/Sensor/identification/IdentifierList/identifier[@name='long_name']/Term	ExtrinsicObject[@objectType='Sensor']/Description/LocalizedString/@value	String
/SensorML/Sensor/inputs/InputList/input/@name	ExtrinsicObject[@objectType='Sensor']/Slot[@name='Inputs']/ValueList/Value[]	String
/SensorML/Sensor/outputs/OutputList/output/@name	ExtrinsicObject[@objectType='Sensor']/Slot[@name='Outputs']/ValueList/Value[]	String
/SensorML/Sensor/processes/ProcessList/process/@name	ExtrinsicObject[@objectType='Sensor']/Slot[@name='Processes']/ValueList/Value[]	String

Like for the System element, classification part of the XML definition is transformed into Classifications to corresponding ClassificationNode of the right taxonomy.

**1.5.2.4 The ProcessModel element:**

SensorML Element	EbRIM Element	Type
/SensorML/ProcessModel	ExtrinsicObject[@objectType='ProcessModel'] linked to the SensorML ExtrinsicObject by an 'hasComponent' Association	String
/SensorML/ProcessModel/@id	ExtrinsicObject[@objectType='ProcessModel']/Slot[@name='Identifier']/ValueList/Value[1]	String

**1.5.2.5 The ProcessChain element:**

SensorML Element	EbRIM Element	Type
/SensorML/ProcessChain	ExtrinsicObject[@objectType='ProcessChain'] linked to the SensorML ExtrinsicObject by an 'hasComponent' Association	String
/SensorML/ProcessChain/@id	ExtrinsicObject[@objectType='ProcessChain']/Slot[@name='Identifier']/ValueList/Value[1]	String

**1.5.2.6 The Detector element:**

SensorML Element	EbRIM Element	Type
/SensorML/Detector	ExtrinsicObject[@objectType='Detector'] linked to the SensorML ExtrinsicObject by an 'hasComponent' Association	String
/SensorML/Detector/@id	ExtrinsicObject[@objectType='Detector']/Slot[@name='Identifier']/ValueList/Value[1]	String

**1.5.2.7 The Actuator element:**

SensorML Element	EbRIM Element	Type
/SensorML/Actuator	ExtrinsicObject[@objectType='Actuator'] linked to the SensorML ExtrinsicObject by an 'hasComponent' Association	String
/SensorML/Actuator/@id	ExtrinsicObject[@objectType='Actuator']/Slot[@name='Identifier']/ValueList/Value[1]	String

**1.5.2.8 The PropertyList element:**

SensorML Element	EbRIM Element	Type
/SensorML/PropertyList	ExtrinsicObject[@objectType='PropertyList'] linked to the SensorML ExtrinsicObject by an 'hasComponent' Association	String

**1.5.2.9 The DocumentList element:**

SensorML Element	EbRIM Element	Type
/SensorML/DocumentList	ExtrinsicObject[@objectType='DocumentList'] linked to the SensorML ExtrinsicObject by an 'hasComponent' Association	String

**1.5.2.10 The ContactList element:**

SensorML Element	EbRIM Element	Type
/SensorML/ContactList	ExtrinsicObject[@objectType='ContactList'] linked to the SensorML ExtrinsicObject by an 'hasComponent' Association	String

**1.6 Phenomenon Dictionary Management****1.6.1 Material Description**

SWE Community needs GML dictionary defining phenomenons. These phenomenons are used:

1. When defining the output of a process (system) in SensorML
2. When defining the GetObservation on the returning capabilities document (see values of parameter)
3. In the getObservation response (See observedProperty)

4. When describing the output in SensorML document (swe:component)

**1.6.2 Mapping of source materials into ebRIM objects**

Catalogs handle GML dictionaries and organize phenomena in a hierarchical taxonomy. The taxonomy is useful to search and retrieve for example SOS or Observations linked to a phenomenon or any of its descendants: e.g. a user asks for all Observations served by an SOS and related to the “AirTemperature” or any of its descendants (more specific) like “AverageAirTemperature15Minute” and “AverageAirTemperature30Minute”.

The entire dictionary is represented by a ClassificationScheme object. Each dictionary entry (of type Phenomenon, ConstrainedPhenomenon, CompositePhenomenon or PhenomenonSeries) is represented by a ClassificationNode, child of the ClassificationScheme.

The following table defines the mapping between the Dictionary and ebRIM objects:

Dictionary Element	EbRIM Element	Type
/Dictionary/identifier	ClassificationScheme/@uuid	URN
/Dictionary/name	ClassificationScheme/Name/LocalizedString/@value	String
/Dictionary/description	ClassificationScheme/Description/LocalizedString/@value	String

The following table defines the mapping between the **Phenomenon** Dictionary entry and ebRIM objects:

Dictionary Element	EbRIM Element	Type
/Dictionary/dictionaryEntry/Phenomenon	ClassificationNode child of the ClassificationScheme	/
"Phenomenon" (fixed Value)	ClassificationNode/Slot[@name="PhenomenonType"]/ValueList/Value[1]	String
/Dictionary/dictionaryEntry/Phenomenon/identifier	ClassificationNode/@uuid	URN
/Dictionary/dictionaryEntry/Phenomenon/name	ClassificationNode/Name/LocalizedString/@value	String
/Dictionary/dictionaryEntry/Phenomenon/description	ClassificationNode/Description/LocalizedString/@value	String

Next schema shows how simple phenomena are represented into the taxonomy.

```

<gml:Dictionary>
  <gml:description>A dictionary of phenomena, compiled through
  OWS-1, OWS-1.2 OWS-3. SJDC 2005-10-03</gml:description>
  <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
  010)">urn:x-ogc:def:phenomenon:OGC:ALL</gml:identifier>
  <gml:name>OWS Phenomena</gml:name>

  <gml:dictionaryEntry>
    <swe:Phenomenon gml:id="AtmosphericPressure">

```

```

    <gml:description
xlink:href="http://sweet.jpl.nasa.gov/ontology/property.owl#AtmosphericPressure">fluid pressure exerted due to the gravitational effect on the column of atmosphere above the position of interest</gml:description>
    <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-010)">urn:x-ogc:def:phenomenon:OGC:AtmosphericPressure</gml:identifier>
    <gml:name>Atmospheric Pressure</gml:name>
  </swe:Phenomenon>
</gml:dictionaryEntry>

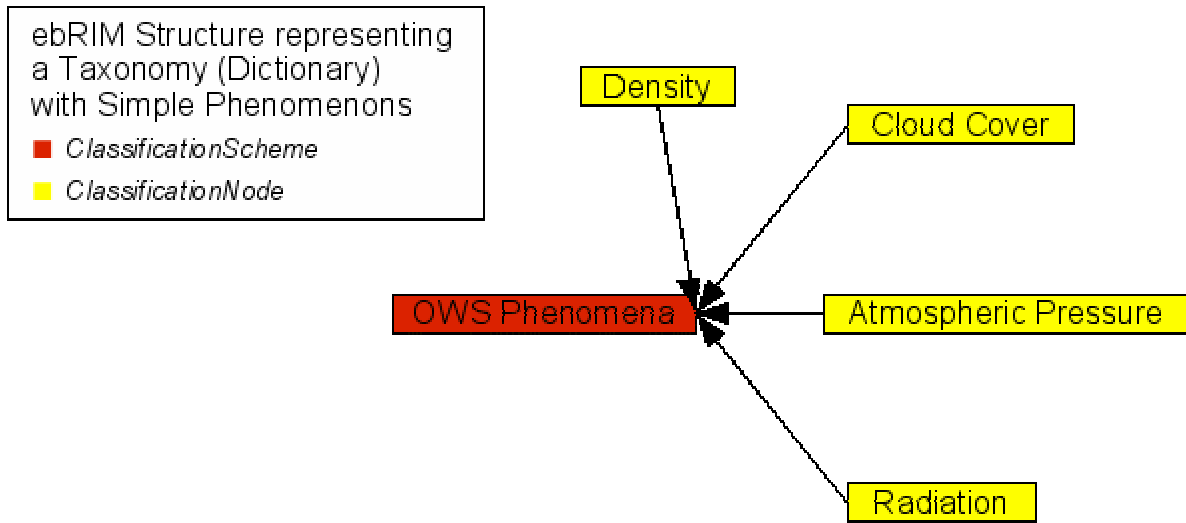
<gml:dictionaryEntry>
  <swe:Phenomenon gml:id="CloudCover">
    <gml:description>fraction of sky occupied by visible cloud</gml:description>
    <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-010)">urn:x-ogc:def:phenomenon:OGC:CloudCover</gml:identifier>
    <gml:name>Cloud Cover</gml:name>
  </swe:Phenomenon>
</gml:dictionaryEntry>

<gml:dictionaryEntry>
  <swe:Phenomenon gml:id="Density">
    <gml:description/>
    <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-010)">urn:x-ogc:def:phenomenon:OGC:Density</gml:identifier>
    <gml:name>Density</gml:name>
  </swe:Phenomenon>
</gml:dictionaryEntry>

<gml:dictionaryEntry>
  <swe:Phenomenon gml:id="Radiation">
    <gml:description
xlink:href="http://sweet.jpl.nasa.gov/ontology/property.owl#RadiantEnergy"/>
    <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-010)">urn:x-ogc:def:phenomenon:OGC:Radiation</gml:identifier>
    <gml:name>Radiation</gml:name>
  </swe:Phenomenon>
</gml:dictionaryEntry>

  [...]
</gml:Dictionary>

```



The following table defines the mapping between the **ConstrainedPhenomenon** Dictionary entry and ebRIM objects:

Dictionary Element	EbRIM Element	Type
/Dictionary/dictionaryEntry/ConstrainedPhenomenon	ClassificationNode child of the ClassificationScheme	/
"ConstrainedPhenomenon" (fixed Value)	ClassificationNode/Slot[@name="PhenomenonType"]/ValueList/Value[1]	String
/Dictionary/dictionaryEntry/ConstrainedPhenomenon/identifier	ClassificationNode/@uuid	URN
/Dictionary/dictionaryEntry/ConstrainedPhenomenon/name	ClassificationNode/Name/LocalizedString/@value	String
/Dictionary/dictionaryEntry/ConstrainedPhenomenon/description	ClassificationNode/Description/LocalizedString/@value	String
/Dictionary/dictionaryEntry/ConstrainedPhenomenon/base/@href	Association (with associationType equals to 'derivedFrom') to the ClassificationNode of the same taxonomy having this uuid	String
/Dictionary/dictionaryEntry/ConstrainedPhenomenon/singleConstraint/TypedValue/property	ClassificationNode/Slot[@name="ConstraintProperty"]/ValueList/Value[1]	String

Next schema shows how constrained phenomenons are represented into the taxonomy.

```

<gml:Dictionary>
  <gml:description>A dictionary of phenomena, compiled through
  OWS-1, OWS-1.2 OWS-3. SJDC 2005-10-03</gml:description>
  <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
  010) ">urn:x-ogc:def:phenomenon:OGC:ALL</gml:identifier>
  <gml:name>OWS Phenomena</gml:name>

  <gml:dictionaryEntry>
    <swe:Phenomenon gml:id="Radiation">
      <gml:description
      xlink:href="http://sweet.jpl.nasa.gov/ontology/property.owl#RadiantEnergy"/>
      <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
      010) ">urn:x-ogc:def:phenomenon:OGC:Radiation</gml:identifier>
    </swe:Phenomenon>
  </gml:dictionaryEntry>
</gml:Dictionary>

```



```

    <gml:name>Radiation</gml:name>
  </swe:Phenomenon>
</gml:dictionaryEntry>

<gml:dictionaryEntry>
  <swe:ConstrainedPhenomenon gml:id="Radiance">
    <gml:description
xlink:href="http://sweet.jpl.nasa.gov/ontology/property.owl#Radiance"/>
    <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-010)">urn:x-ogc:def:phenomenon:OGC:Radiance</gml:identifier>
    <gml:name>Radiance</gml:name>
    <swe:base xlink:href="#Radiation"/>
    <swe:singleConstraint>
      <swe:TypedValue>
        <swe:property codeSpace="urn:x-ogc:def:nil:OGC:unknown">mode</swe:property>
        <swe:value xsi:type="gml:CodeType">passive</swe:value>
      </swe:TypedValue>
    </swe:singleConstraint>
  </swe:ConstrainedPhenomenon>
</gml:dictionaryEntry>

<gml:dictionaryEntry>
  <swe:ConstrainedPhenomenon gml:id="_19V">
    <gml:description>19 GHz Radiation Vertical Polarisation</gml:description>
    <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-010)">urn:x-ogc:def:phenomenon:OGC:19V</gml:identifier>
    <gml:name>19V</gml:name>
    <swe:base xlink:href="#Radiation"/>
    <swe:singleConstraint>
      <swe:TypedValue>
        <swe:property
codeSpace="./">PeakWavelength</swe:property>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#GHz">19.35</swe:value>
      </swe:TypedValue>
    </swe:singleConstraint>
    <swe:singleConstraint>
      <swe:TypedValue>
        <swe:property
codeSpace="./">PolarisationDirection</swe:property>
        <swe:value xsi:type="gml:CodeType"
codeSpace="http://www.opengis.net/sensorGlossary">V</swe:value>
      </swe:TypedValue>
    </swe:singleConstraint>
  </swe:ConstrainedPhenomenon>
</gml:dictionaryEntry>

<gml:dictionaryEntry>

```

```

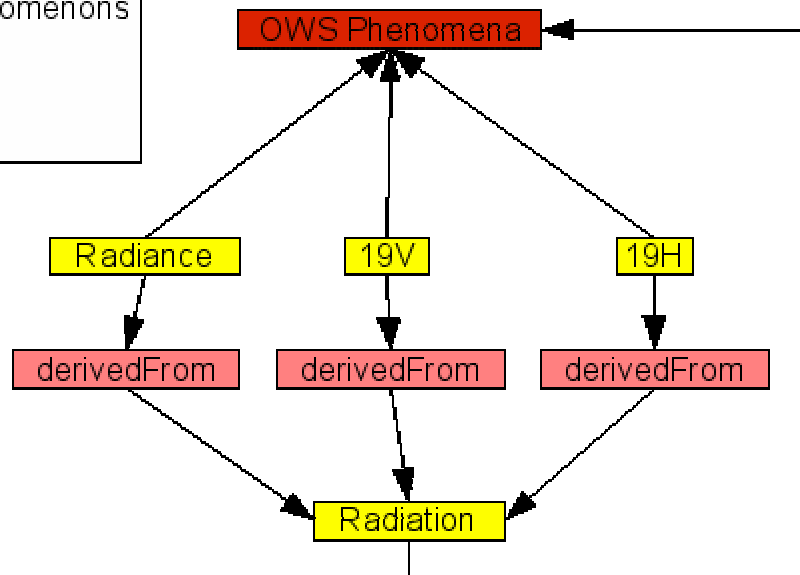
    <swe:ConstrainedPhenomenon gml:id="_19H">
      <gml:description>19 GHz Radiation Horizontal
Polarisation</gml:description>
      <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
010)">urn:x-ogc:def:phenomenon:OGC:19H</gml:identifier>
      <gml:name>19H</gml:name>
      <swe:base xlink:href="#Radiation"/>
      <swe:singleConstraint>
        <swe:TypedValue>
          <swe:property
codeSpace="."/>peakWavelength</swe:property>
          <swe:value xsi:type="gml:MeasureType"
uom="#GHz">19.35</swe:value>
        </swe:TypedValue>
      </swe:singleConstraint>
      <swe:singleConstraint>
        <swe:TypedValue>
          <swe:property
codeSpace="."/>PolarisationDirection</swe:property>
          <swe:value xsi:type="gml:CodeType"
codeSpace="http://www.opengis.net/sensorGlossary">H</swe:value>
        </swe:TypedValue>
      </swe:singleConstraint>
    </swe:ConstrainedPhenomenon>
  </gml:dictionaryEntry>

  [...]
</gml:Dictionary>

```

ebRIM Structure representing a Taxonomy (Dictionary) with Constrained Phenomenons

- Association
- ClassificationScheme
- ClassificationNode



The following table defines the mapping between the **CompositePhenomenon** Dictionary entry and ebRIM objects:

Dictionary Element	EbRIM Element	Type
/Dictionary/dictionaryEntry/Composite Phenomenon	ClassificationNode child of the ClassificationScheme	/
"CompositePhenomenon" (fixed Value)	ClassificationNode/Slot[@name="PhenomenonType"]/ValueList/Value[1]	String
/Dictionary/dictionaryEntry/Composite Phenomenon/identifier	ClassificationNode/@uuid	URN
/Dictionary/dictionaryEntry/Composite Phenomenon/name	ClassificationNode/Name/LocalizedString/@value	String
/Dictionary/dictionaryEntry/Composite Phenomenon/description	ClassificationNode/Description/LocalizedString/@value	String
/Dictionary/dictionaryEntry/Composite Phenomenon/base/@href	Association (with associationType equals to 'derivedFrom') to the ClassificationNode of the same taxonomy having this uuid	String
/Dictionary/dictionaryEntry/Composite Phenomenon/component/@href	Association (with associationType equals to 'composedWith') to the ClassificationNode of the same taxonomy having this uuid	String

Next schema shows how composite phenomenons are represented into the taxonomy.

```

<gml:Dictionary>
  <gml:description>A dictionary of phenomena, compiled through
  OWS-1, OWS-1.2 OWS-3. SJDC 2005-10-03</gml:description>
  <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
  010)">urn:x-ogc:def:phenomenon:OGC:ALL</gml:identifier>
  <gml:name>OWS Phenomena</gml:name>

  <gml:dictionaryEntry>
    <swe:Phenomenon gml:id="Direction">
      <gml:description
  xlink:href="http://sweet.jpl.nasa.gov/ontology/space.owl#Directio
  n">Orientation of a vector relative to a reference
  frame.</gml:description>
      <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
  010)">urn:x-ogc:def:phenomenon:OGC:Direction</gml:identifier>
      <gml:name>Direction</gml:name>
    </swe:Phenomenon>
  </gml:dictionaryEntry>

  <gml:dictionaryEntry>
    <swe:Phenomenon gml:id="Speed">
      <gml:description
  xlink:href="http://sweet.jpl.nasa.gov/ontology/property.owl#Speed
  ">Scalar rate of movement</gml:description>
      <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
  010)">urn:x-ogc:def:phenomenon:OGC:Speed</gml:identifier>
      <gml:name>Speed</gml:name>
    </swe:Phenomenon>
  </gml:dictionaryEntry>

  <gml:dictionaryEntry>
    <swe:CompositePhenomenon gml:id="Velocity" dimension="2">

```

```

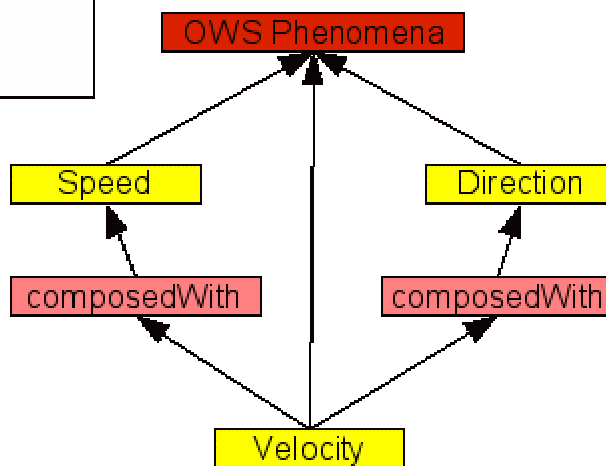
    <gml:description
xlink:href="http://sweet.jpl.nasa.gov/ontology/property.owl#Velocity">Vector rate of movement relative to a reference
frame</gml:description>
    <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
010)">urn:x-ogc:def:phenomenon:OGC:Velocity</gml:identifier>
    <gml:name>Velocity</gml:name>
    <swe:component xlink:href="#Speed"/>
    <swe:component xlink:href="#Direction"/>
  </swe:CompositePhenomenon>
</gml:dictionaryEntry>

[...]
</gml:Dictionary>

```

ebRIM Structure representing a Taxonomy (Dictionary) with Composed Phenomenons

- Association
- ClassificationScheme
- ClassificationNode



The following table defines the mapping between the **PhenomenonSeries** Dictionary entry and ebRIM objects:

Dictionary Element	EbRIM Element	Type
/Dictionary/dictionaryEntry/PhenomenonSeries	ClassificationNode child of the ClassificationScheme	/
"PhenomenonSeries" (fixed Value)	ClassificationNode/Slot[@name="PhenomenonType"]/ValueList/Value[1]	String
/Dictionary/dictionaryEntry/PhenomenonSeries/identifier	ClassificationNode/@uuid	URN
/Dictionary/dictionaryEntry/PhenomenonSeries/name	ClassificationNode/Name/LocalizedString/@value	String
/Dictionary/dictionaryEntry/PhenomenonSeries/description	ClassificationNode/Description/LocalizedString/@value	String
/Dictionary/dictionaryEntry/PhenomenonSeries/base/@href	Association (with associationType equals to 'derivedFrom') to the ClassificationNode of the same taxonomy having this uuid	String

Next schema shows how phenomenon SERIES are represented into the taxonomy.

```

<gml:Dictionary>
  <gml:description>A dictionary of phenomena, compiled through
  OWS-1, OWS-1.2 OWS-3. SJDC 2005-10-03</gml:description>
  <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
  010)">urn:x-ogc:def:phenomenon:OGC:ALL</gml:identifier>
  <gml:name>OWS Phenomena</gml:name>

  <gml:dictionaryEntry>
    <swe:Phenomenon gml:id="Radiation">
      <gml:description
  xlink:href="http://sweet.jpl.nasa.gov/ontology/property.owl#Radia
  ntEnergy"/>
      <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
  010)">urn:x-ogc:def:phenomenon:OGC:Radiation</gml:identifier>
      <gml:name>Radiation</gml:name>
    </swe:Phenomenon>
  </gml:dictionaryEntry>

  <gml:dictionaryEntry>
    <swe:ConstrainedPhenomenon gml:id="Radiance">
      <gml:description
  xlink:href="http://sweet.jpl.nasa.gov/ontology/property.owl#Radia
  nce"/>
      <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
  010)">urn:x-ogc:def:phenomenon:OGC:Radiance</gml:identifier>
      <gml:name>Radiance</gml:name>
      <swe:base xlink:href="#Radiation"/>
      <swe:singleConstraint>
        <swe:TypedValue>
          <swe:property codeSpace="urn:x-
  ogc:def:nil:OGC:unknown">mode</swe:property>
          <swe:value xsi:type="gml:CodeType">passive</swe:value>
        </swe:TypedValue>
      </swe:singleConstraint>
    </swe:ConstrainedPhenomenon>
  </gml:dictionaryEntry>

  <gml:dictionaryEntry>
    <swe:PhenomenonSeries gml:id="uSpectrum" dimension="17">
      <gml:description>Simple spectrum with uniform spacing of
  bands</gml:description>
      <gml:identifier codeSpace="urn:x-ogc:tc:arch:doc-rp(05-
  010)">urn:x-ogc:def:phenomenon:OGC:uSpectrum</gml:identifier>
      <gml:name>UniformSpectrum</gml:name>
      <swe:base xlink:href="#Radiance"/>
      <swe:constraintList>
        <swe:TypedValueList>
          <swe:property codeSpace="."/>Wavelength</swe:property>
          <swe:value xsi:type="gml:MeasureType"
  uom="./units.xml#um">0.300</swe:value>
        </swe:TypedValueList>
      </swe:constraintList>
    </swe:PhenomenonSeries>
  </gml:dictionaryEntry>

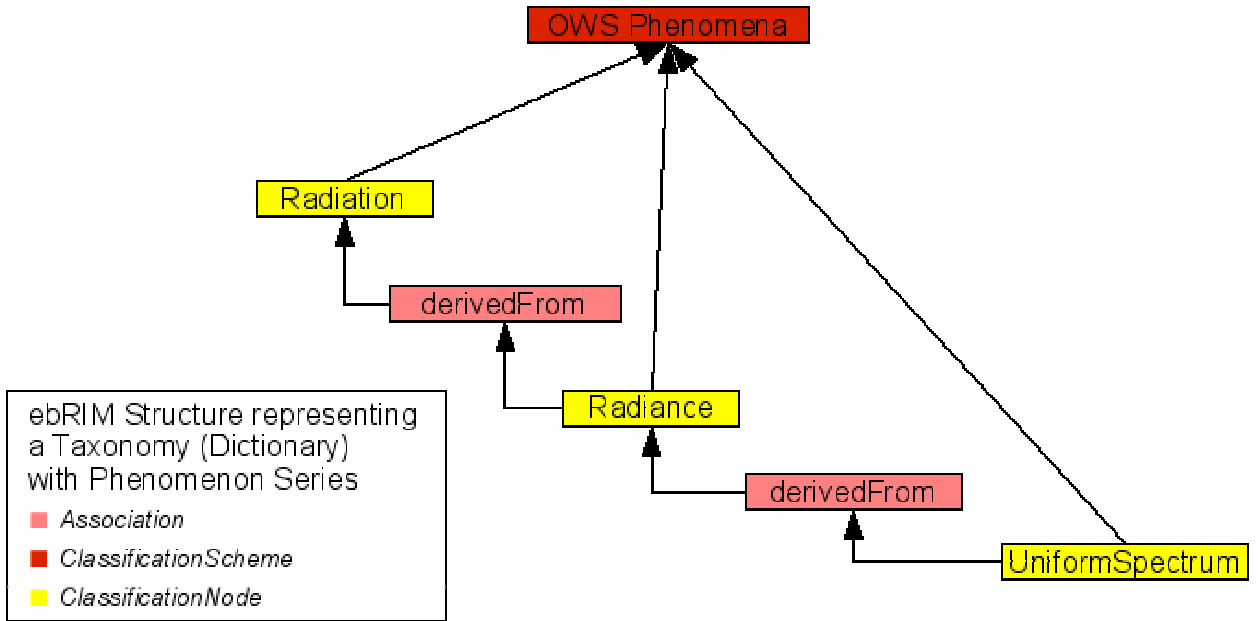
```

```

        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.304</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.308</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.312</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.316</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.320</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.324</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.328</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.332</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.336</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.340</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.344</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.348</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.352</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.356</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.360</swe:value>
        <swe:value xsi:type="gml:MeasureType"
uom="./units.xml#um">0.364</swe:value>
    </swe:TypedValueList>
    </swe:constraintList>
    </swe:PhenomenonSeries>
</gml:dictionaryEntry>

[... ]
</gml:Dictionary>

```



## Annex B OWS Domain Extension Package—Feature Catalog

### 1 Introduction

The purpose of this ANNEX is to describe how to register artefacts, described using the Department of Defence Metadata Standard (DDMS V1.3), into the ebRIM (V3.0). This annex specifically describes how DDMS metadata was mapped into the information model of a CSW-ebRIM compliant catalogue for the OWS4 testbed.

In the testbed an automated tool, called UGAS, was used to generate a GML application schemas from a UML models (encoded in XML as an XMI file). Embedded within the generated GML application schemas, DDMS were used to describe resources (or artefacts) declared therein (e.g. <http://demo.cubewerx.com/FeatureCatalogue/GSIP.xsd>). The CubeWerx catalogue was modified to harvest this embedded DDMS metadata and register the artefacts declared in the schema into the catalogue's information model (ebRIM V3.0).

### 2 UML Model

The following class diagram shows the artefacts, and their inter-relationships, that are registered as a result of the catalogue processing DDMS metadata embedded in a GML application schema.

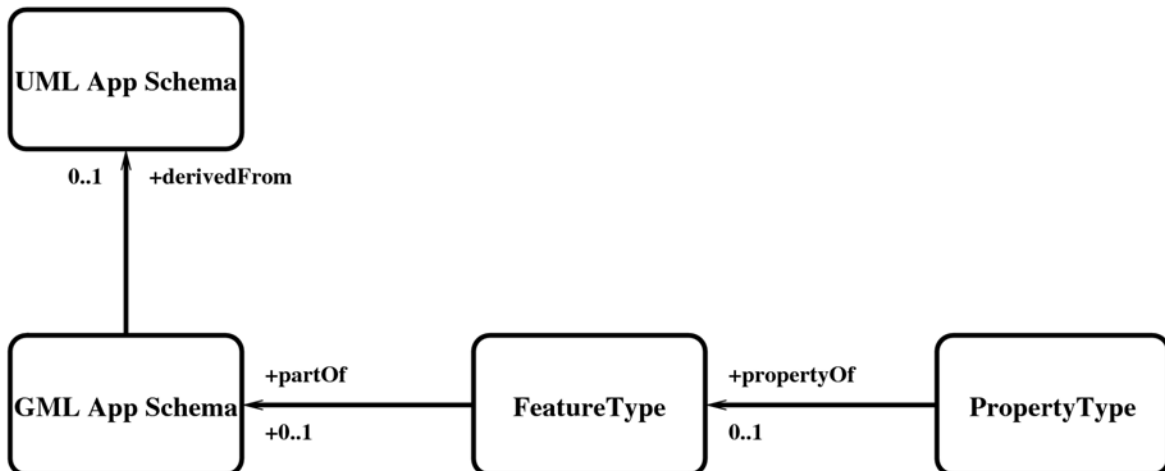


Figure B.1 – Feature Catalogue UML



### 3 Object Types

As shown in subclause B.2, four basic object types were processed in the OWS4 testbed. Within the embedded DDMS metadata, these object types were designated using the codes: UMLApplicationSchema, Schema, FeatureType and PropertyType. For the OWS4 testbed, the "schema" types were further organized into the hierarchy shown in figure A.1 in order to discriminate an XML-Schema from a GML Application Schema from a UML Application Schema.

For the purpose of designation slot types, several DDMS-specific object types were also defined. These types include a boolean, string, double and a couple of extended date types. These types were organized into a classification sub-tree with the root node designated DDMS\_Slot\_Types, as shown in figure B.2.

Figure B.2 shows the portion of the ebRIM type classification scheme relevant to the OWS4 project. Namespace-like prefixes are used in the diagram to discriminate the origin of each node. The prefix 'ebrim' is used to designate nodes that already exist in the classification scheme. The prefix 'csw-ebrim' is used to designate nodes added by the csw-ebRIM extension package. The prefix 'ows4-ddms' is used to designate nodes added during the OWS4 testbed to support the harvesting of DDMS metadata.

```

ebrim:ObjectType
|
+ ebrim:RegistryObject
|
+ ebrim:ExtrinsicObject
||
| +- csw-ebrim:Schema
|| |
|| +- ows4-ddms:XML-Schema
|| | |
|| | +- ows4-ddms:GML-Application-Schema
|| | |
|| +- ows4-ddms:UML-Application-Schema
||

```

```

| +- ows4-ddms:FeatureType
||
| +- ows4-ddms:PropertyType
|
+- ows4-ddms:DDMS_Slot_Types
|
| +- ows4-ddms:String
|
| +- ows4-ddms:Boolean
|
| +- ows4-ddms:Double
|
| +- ows4-ddms:CombinedDateType
|
| +- ows4-ddms:ExtendedCombinedDateType

```

**Figure B.2 – ObjectType classification scheme**

The following XML fragments define the additional nodes that need to be added to the ObjectType classification scheme:

```

<rim:ClassificationNode code="XML-Schema"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:csw-ebrim:ObjectType:Schema:XML-Schema"
  id="urn:x-ogc:specification:csw-ebrim:ObjectType:Schema:XML-Schema"
  parent="urn:x-ogc:specification:csw-ebrim:ObjectType:Schema">
  <rim:Name>

```

```

    <rim:LocalizedString value="XML-Schema" />
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en"
      value="An XML schema document."/>
  </rim:Description>
</rim:ClassificationNode>

<rim:ClassificationNode code="GML-Application-Schema"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:csw-ebrim:ObjectType:Schema:XML-Schema:GML-
  Application-Schema"
  id="urn:x-ogc:specification:csw-ebrim:ObjectType:Schema:XML-Schema:GML-
  Application-Schema"
  parent="urn:oasis:names:tc:ebxml-regrep:ObjectType:Schema:XML-Schema">
  <rim:Name>
    <rim:LocalizedString value="GML-Application-Schema" />
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en"
      value="A GML application schema document."/>
  </rim:Description>
</rim:ClassificationNode>

<rim:ClassificationNode code="UML-Application-Schema"

```

```

    objectType="urn:oasis:names:tc:ebxml-
    regrep:ObjectType:RegistryObject:ClassificationNode"

```

```

    lid="urn:x-ogc:specification:csw-ebrim:ObjectType:Schema:UML-Application-
    Schema"

```

```

    id="urn:x-ogc:specification:csw-ebrim:ObjectType:Schema:UML-Application-
    Schema"

```

```

    parent="urn:x-ogc:specification:csw-ebrim:ObjectType:Schema">

```

```

    <rim:Name>

```

```

        <rim:LocalizedString value="UML-Application-Schema" />

```

```

    </rim:Name>

```

```

    <rim:Description>

```

```

        <rim:LocalizedString xml:lang="en"

```

```

            value="A UML model encoded as an XMI document."/>

```

```

    </rim:Description>

```

```

</rim:ClassificationNode>

```

```

<rim:ClassificationNode code="FeatureType"

```

```

    objectType="urn:oasis:names:tc:ebxml-
    regrep:ObjectType:RegistryObject:ClassificationNode"

```

```

    lid="urn:x-ogc:specification:csw-ebrim:ObjectType:FeatureType"

```

```

    id="urn:x-ogc:specification:csw-ebrim:ObjectType:FeatureType"

```

```

    parent="urn:oasis:names:tc:ebxml-
    regrep:ObjectType:RegistryObject:ExtrinsicObject">

```

```

    <rim:Name>

```

```

        <rim:LocalizedString value="FeatureType" />

```

```

    </rim:Name>

```

```

    <rim:Description>

```

```

    <rim:LocalizedString xml:lang="en" value=""/>
  </rim:Description>
</rim:ClassificationNode>

<rim:ClassificationNode code="PropertyType"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:csw-ebrim:ObjectType:PropertyType"
  id="urn:x-ogc:specification:csw-ebrim:ObjectType:PropertyType"
  parent="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ExtrinsicObject">
  <rim:Name>
    <rim:LocalizedString value="PropertyType" />
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en" value=""/>
  </rim:Description>
</rim:ClassificationNode>

<rim:ClassificationNode code="DDMS_Slot_Types"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types"
  id="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types"
  parent="urn:oasis:names:tc:ebxml-regrep:ObjectType">
  <rim:Name>

```

```

    <rim:LocalizedString value="DDMS_Slot_Types" />
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en" value="Types defined by the DDMS"/>
  </rim:Description>
</rim:ClassificationNode>

<rim:ClassificationNode code="String"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types:String"
  id="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types:String"
  parent="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types">
  <rim:Name>
    <rim:LocalizedString value="String" />
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en"
      value="A character string type."/>
  </rim:Description>
</rim:ClassificationNode>

<rim:ClassificationNode code="Boolean"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types:Boolean">

```

```

id="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types:Boolean"
parent="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types">
<rim:Name>
  <rim:LocalizedString value="Boolean" />
</rim:Name>
<rim:Description>
  <rim:LocalizedString xml:lang="en"
    value="A value of TRUE or FALSE"/>
</rim:Description>
</rim:ClassificationNode>

<rim:ClassificationNode code="Double"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types:Double"
  id="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types:Double"
  parent="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types">
<rim:Name>
  <rim:LocalizedString value="Double" />
</rim:Name>
<rim:Description>
  <rim:LocalizedString xml:lang="en"
    value="An IEEE double value."/>
</rim:Description>
</rim:ClassificationNode>

```

```

<rim:ClassificationNode code="CombinedDateType"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:csw-
  ebrim:ObjectType:DDMS_Slot_Types:CombinedDateType"
  id="urn:x-ogc:specification:csw-
  ebrim:ObjectType:DDMS_Slot_Types:CombinedDateType"
  parent="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS_Slot_Types">
  <rim:Name>
    <rim:LocalizedString value="CombinedDateType" />
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en"
      value="A value whose type corresponds to one of the
        XML types dateTime, date, gYearMonth, gYear"/>
  </rim:Description>
</rim:ClassificationNode>

```

```

<rim:ClassificationNode code="ExtendedCombinedDateType"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:csw-
  ebrim:ObjectType:DDMS_Slot_Types:ExtendedCombinedDateType"
  id="urn:x-ogc:specification:csw-
  ebrim:ObjectType:DDMS_Slot_Types:ExtendedCombinedDateType"
  parent="urn:x-ogc:specification:csw-
  ebrim:ObjectType:DDMS_Slot_Types:CombinedDateType">
  <rim:Name>

```



```

    <rim:LocalizedString value="ExtendedCombinedDateType"/>
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en"
    value="A value whose type corresponds to one of the
      XML types dateTime, date, gYearMonth, gYear OR one of
      the strings 'Not Applicable' or 'Unknown'"/>
  </rim:Description>
</rim:ClassificationNode>

```

#### 4 Association Types

In the embedded DDMS metadata, the **<relation>** element is used to declare relationships or associations between the resource containing the element and other resources.

The following association types have been used within the OWS4 testbed:

1. **derivedFrom**  
an association between a GML Application Schema and the UML Application Schema from which it was generated (also used to indicate that one resource is derived from another resource)
2. **dependsOn**  
an association between schema documents indicating that one schema depend on another schema
3. **partOf**  
an association between a feature type and the GML Application Schema within which it is declared
4. **resource**  
an association between a resource and itself typically used to specify a URL reference to the declaration of the resource within a GML Application Schema
5. **propertyOf**  
an association indicating that the resource is a property of a specified feature type

## 6. **inView**

an association indicating that a feature is part of the specified View (this is mapped as a classification in ebRIM)

Examples:

The following XML fragments are examples of the relation element:

```
<relation xlink:href="GSIP_320/GSIP.xsd">resource</relation>
```

```
<relation xlink:href="urn:x-nsgfc:AS:id:gsip:v1.8_draft">derivedFrom</relation>
```

```
<relation xlink:href="urn:x-nsgfc:AS:id:gsip:v1.8_draft">partOf</relation>
```

```
<relation xlink:href="urn:x-nsgfc:GMLAS:id:app:unknown">dependsOn</relation>
```

```
<relation xlink:href="urn:x-nsgfc:VIEW:id:elevation">inView</relation>
```

```
<relation xlink:href="urn:x-nsgfc:VIEW:id:inlandWater">inView</relation>
```

```
<relation xmlns:xs="http://www.w3.org/2001/XMLSchema"
xlink:href="/GSIP.xsd#element(//xs:element[@name='ExtractionMine'])">resource</rel
ation>
```

With respect to the class diagram in subclause A.2, a GML Application Schema is **derivedFrom** a UML Application Schema. Each **FeatureType** is **partOf** a GML Application Schema and each **PropertyType** is a **propertyOf** a **FeatureType**.

Figure B.3 shows a portion of the AssociationType classification scheme and the additional nodes added to that scheme to support the harvesting of DDMS metadata. The prefix 'ebrim' is used to indicate nodes that are already part of the canonical ebXML AssociationType classification scheme. The prefix 'csw-ebrim' is used to indicate nodes added to the canonical AssociationType classification scheme by the csw-ebRIM

extension package. Finally, the prefix 'ows4-ddms' indicates nodes added to the canonical AssociationType classification scheme during the OWS4 testbed to support harvesting DDMS metadata.

```

ebrim:AssociationType
|
+- csw-ebrimAssociationType
|
+- ows4-ddms:derivedFrom
|
+- ows4-ddms:dependsOn
|
+- ows4-ddms:partOf
|
+- ows4-ddms:resource
|
+- ows4-ddms:propertyOf
|
+- ows4-ddms:inView

```

### Figure B.3 – AssociationType classification scheme

The following XML fragments define the additional association type nodes that need to be added to a catalogue to support harvesting of DDMS metadata:

```

<rim:ClassificationNode code="derivedFrom"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:ows4-ddms:AssociationType:derivedFrom"
  id="urn:x-ogc:specification:ows4-ddms:AssociationType:derivedFrom"

```

```

parent="urn:oasis:names:tc:ebxml-regrep:classificationScheme:AssociationType">
  <rim:Name>
    <rim:LocalizedString value="derivedFrom" />
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en" value="an association between a GML
Application Schema and the UML Application Schema from which it was generated"/>
  </rim:Description>
</rim:ClassificationNode>

<rim:ClassificationNode code="dependsOn"
  objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:ows4-ddms:AssociationType:dependsOn"
  id="urn:x-ogc:specification:ows4-ddms:AssociationType:dependsOn"
  parent="urn:oasis:names:tc:ebxml-regrep:classificationScheme:AssociationType">
  <rim:Name>
    <rim:LocalizedString value="dependsOn" />
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en" value="an association between schema
documents indicating that one schema depend on another schema"/>
  </rim:Description>
</rim:ClassificationNode>

<rim:ClassificationNode code="partOf"

```

```

    objectType="urn:oasis:names:tc:ebxml-
    regrep:ObjectType:RegistryObject:ClassificationNode"

    lid="urn:x-ogc:specification:ows4-ddms:AssociationType:partOf"
    id="urn:x-ogc:specification:ows4-ddms:AssociationType:partOf"
    parent="urn:oasis:names:tc:ebxml-regrep:classificationScheme:AssociationType">
    <rim:Name>
        <rim:LocalizedString value="partOf" />
    </rim:Name>
    <rim:Description>
        <rim:LocalizedString xml:lang="en" value="an association between a feature type
        and the GML Application Schema within which it is declared"/>
    </rim:Description>
</rim:ClassificationNode>

<rim:ClassificationNode code="resource"
    objectType="urn:oasis:names:tc:ebxml-
    regrep:ObjectType:RegistryObject:ClassificationNode"
    lid="urn:x-ogc:specification:ows4-ddms:AssociationType:resource"
    id="urn:x-ogc:specification:ows4-ddms:AssociationType:resource"
    parent="urn:oasis:names:tc:ebxml-regrep:classificationScheme:AssociationType">
    <rim:Name>
        <rim:LocalizedString value="resource" />
    </rim:Name>
    <rim:Description>
        <rim:LocalizedString xml:lang="en" value="an association between a resource and
        itself typically used to specify a URL reference to the declaration of the resource within a
        GML Application Schema"/>

```

```

</rim:Description>

</rim:ClassificationNode>

<rim:ClassificationNode code="propertyOf"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:ows4-ddms:AssociationType:propertyOf"
  id="urn:x-ogc:specification:ows4-ddms:AssociationType:propertyOf"
  parent="urn:oasis:names:tc:ebxml-regrep:classificationScheme:AssociationType">
  <rim:Name>
    <rim:LocalizedString value="propertyOf" />
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en" value="an association indicating that the
    resource is a property of a specified feature type"/>
  </rim:Description>
</rim:ClassificationNode>

<rim:ClassificationNode code="inView"
  objectType="urn:oasis:names:tc:ebxml-
  regrep:ObjectType:RegistryObject:ClassificationNode"
  lid="urn:x-ogc:specification:ows4-ddms:AssociationType:inView"
  id="urn:x-ogc:specification:ows4-ddms:AssociationType:inView"
  parent="urn:oasis:names:tc:ebxml-regrep:classificationScheme:AssociationType">
  <rim:Name>
    <rim:LocalizedString value="inView" />

```

</rim:Name>

<rim:Description>

<rim:LocalizedString xml:lang="en" value="an association indicating that a feature is part of a specific view or theme"/>

</rim:Description>

</rim:ClassificationNode>

## 5 Element Mappings

This subclause describes how each source element from a DDMS metadata document is mapped to elements in the ebRIM in order to create a feature catalogue.

-----

### Source Path:

Resource/identifier/@qualifier

Resource/identifier/@value

### Target Path:

ExtrinsicObject/@id

- OR -

ExtrinsicObject/ExternalIdentifier/@id

### Notes:

If the value of the Resource/identifier/@qualifier value is set to 'urn:uuid' the value of the attribute Resource/identifier/@value is mapped to the attribute ExtrinsicObject/@id.

Otherwise the catalogue generates a value for the [ExtrinsicObject/@id](#) attribute and creates an ExternalIdentifier object where the attribute value of the Resource/identifier/@qualifier attribute is mapped to the ExternalIdentifier/@identificationScheme attribute and the value of the Resource/identifier/@value attribute is mapped to the attribute ExternalIdentifier/@value.

The value of the system generated id is then copied to the ExternalIdentifier/@registryObject attribute.

---

**Source Path:**

Resource/title

**Target Path:**

ExtrinsicObject/Name/LocalizedString/[@xml:lang="..."]

ExtrinsicObject/Name/LocalizedString/[@charset="..."]

ExtrinsicObject/Name/LocalizedString/[@value="title"]

**Notes:**

The **lang** attribute may be set of the Resource/language element is set. Similarly, the **charset** attribute can be set from the declared character set of the XML document (e.g. UTF-8).

---

**Source Path:**

Resource/subtitle

**Target Path:**

ExtrinsicObject/Slot/[@name="subtitle" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:String"]/ValueList/Value

**Notes:**

The subtitle is mapped to a slot. It's type is set using the id of the string object type defined in subclause A.3 (i.e. urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:String).

---



**Source Path:**

Resource/description

**Target Path:**

ExtrinsicObject/Description/LocalizedString/[@xml:lang="TOKEN1"]

ExtrinsicObject/Description/LocalizedString/[@charset="TOKEN2"]

ExtrinsicObject/Description/LocalizedString/[@value="description"]

**Notes:**

The **lang** attribute may be set of the Resource/language element is set. Similarly, the **charset** attribute can be set from the declared character set of the XML document (e.g. UTF-8).

**Source Path:**

Resource/language/@qualifier

Resource/language/@value

**Target Path:**

.../xml:lang="value"

**Notes:**

If the language element is set, its value is used to populate the localized strings in the eBRIM. For OWS4, the only supported qualifier was "ISO 639-1".

**Source Path:**

Resource/dates@posted

Resource/dates@created

Resource/dates@validTil

Resource/dates@infoCutOff

**Target Path:**

ExtrinsicObject/Slot/[@name="posted" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:CombinedDateType"]/ValueList/Value

ExtrinsicObject/Slot/[@name="created" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:CombinedDateType"]/ValueList/Value

ExtrinsicObject/Slot/[@name="validTil" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:CombinedDateType"]/ValueList/Value

ExtrinsicObject/Slot/[@name="infoCutOff" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:CombinedDateType"]/ValueList/Value

**Notes:**

The various date source elements are mapped, in eBRIM, to slots of the same name. The slotType attribute is set using the id of the CombinedDateType object type defined in subclause A.3 (i.e. urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:CombinedDateType).

---

**Source Path:**

Resource/rights/@privacyAct

**Target Path:**

ExtrinsicObject/Slot/[@name="rights" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:Boolean"]/ValueList/Value

---

**Source Path:**

Resource/rights/@copyright

**Target Path:**

ExtrinsicObject/Slot/[@name="copyright" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:Boolean"]/ValueList/Value

---

**Source Path:**

Resource/rights/@intellectualProperty

**Target Path:**

ExtrinsicObject/Slot/[@name="intellectualProperty" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:Boolean"]/ValueList/Value

---

**Source Path:**

Resource/source/@schemaHref

Resource/source/@schemaQualifier

**Target Path:**

ExternalLink@externalURI

ExternalLink/Slot/[@name="schemaQualifier" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:URI"]/ValueList/Value

**Notes:**

An ExternalLink object is created and the value of the attribute Resource/source/@schemaHref is mapped to ExternalLink@externalURI.

An Association object is then created to relate the ExternalLink object to the ExtrinsicObject created for the record. The association mapping is:

Association/sourceObject

-> ExtrinsicObject/@id

Association/targetObject

-> ExternalLink/@id

Association/associationType

-> urn:x-ogc:specification:csw-ebrim:AssociationType:derivedFrom

---

**Source Path:**

Resource/source/@qualifier

Resource/source/@value

**Target Path:**

Association/@sourceObject

Association/@targetObject

Association/@associationType

**Notes:**

An eBRIM association is used to reference the resource from which the current resource was derived.

The value of the Association/@sourceObject attribute is set to value of the id of the current resource (see: Source Path: Resource/identifier)

If the value of the Resource/source/@qualifier attribute is set to "urn:uuid" then the Resource/source/@value value is mapped to [Association/@targetObject](#). Otherwise, an ExternalLink object must be created and the value of the Association/@targetObject is set to the value of the [ExternalLink/@id](#) attribute.

The value of the Association/@associationType is set to:

urn:x-ogc:specification:ows4-ddms:AssociationType:derivedFrom

---

**Source Path:**

Resource/type@qualifier

Resource/type@value

**Target Path:**

ExtrinsicObject/@objectType

**Notes:**

For OWS4, one of:

urn:x-ogc:specification:csw-ebrim:ObjectType:Schema:XML-Schema:GML-Application-Schema

urn:x-ogc:specification:csw-ebrim:ObjectType:Schema:UML-Application-Schema

urn:x-ogc:specification:csw-ebrim:ObjectType:FeatureType

urn:x-ogc:specification:csw-ebrim:ObjectType:PropertyType

---

**Source Path:**

Resource/creator/Organization/name

Resource/creator/Organization/phone

Resource/creator/Organization/email

Resource/publisher/Organization/phone

Resource/publisher/Organization/name

Resource/publisher/Organization/email

Resource/contributor/Organization/name

Resource/contributor/Organization/phone

Resource/contributor/Organization/email

Resource/pointOfContact/Organization/name

Resource/pointOfContact/Organization/phone

Resource/pointOfContact/Organization/email

**Target Path:**

Organization/Name/LocalizedString/[@xml:lang="..."]

Organization/Name/LocalizedString/[@charset="..."]

Organization/Name/LocalizedString/[@value="name"]

Organization/Slot/[@name="OrganizationType" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:String"]/ValueList/Value

Organization/TelephoneNumber/@number

Organization/EmailAddress/@address

**Notes:**

The xml:lang attribute may be set of the Resource/language element is set. Similarly, the charset attribute can be set from the declared character set of the XML document (e.g. UTF-8).

A Slot name "OrganizationType" is used to discriminate the different types or organizations found in a DDMS document. The allowed values are:

- creator
- publisher
- contributor
- pointOfContact

The corresponding telephone number is mapped to the attribute Organization/TelephoneNumber/@number.

The Organization object is associated to the ExtrinsicObject created for the artefact via the Organization/@parent attribute. That is, that the value of the Organization/@parent attribute equals the value of the ExtrinsicObject/@id attribute.

---

**Source Path:**

Resource/format/Media/mimeType

**Target Path:**

ExtrinsicObject/@mimeType

---

**Source Path:**

Resource/format/Media/extent@qualifier

Resource/format/Media/extent@value

**Target Path:**

ExtrinsicObject/Slot/[@name="extent" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:String"]/ValueList/Value

---

**Source Path:**

Resource/format/Media/medium

**Target Path:**

ExtrinsicObject/Slot/[@name="medium" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:String"]/ValueList/Value

---

**Source Path:**

Resource/subjectCoverage/Subject/category/@code

Resource/subjectCoverage/Subject/category/@qualifier

Resource/subjectCoverage/Subject/category/@label

**Target Path:**

Classification/@classificationScheme

Classification/@classificationNode

Classification/@classifiedObject

**Notes:**

There is no direct mapping between the Resource/subjectCoverage values and the Classification object. Basically, the catalogue has to recognize the DDMS subjectCoverage category and map it to an appropriate classification node in a classification scheme that is loaded into the catalogue.



In the OWS4 testbed the following classification schemes were used to categorize resources:

- [http://www.isotc211.org/2005/gmd/MD\\_TopicCategoryCode](http://www.isotc211.org/2005/gmd/MD_TopicCategoryCode)
- <http://www.dgiwg.org/dfdd/category>
- <http://www.nga.mil/>
- <http://www.nga.mil/gsip/classification/DataContentStandards>
- <http://www.nga.mil/gsip/classification/Views>

The ISO Topic Categories were encoded as an eBRIM classification scheme (see MD\_Topic\_Categories.xml).

---

**Source Path:**

Resource/subjectCoverage/Subject/keyword/@value

**Target Path:**

ExtrinsicObject/Slot/[@name="keyword" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:String"]/ValueList/Value[N]

---

**Source Path:**

Resource/virtualCoverage/@address

Resource/virtualCoverage/@protocol

**Target Path:**

UNMAPPED (not used in OWS4 testbed)

---

**Source Path:**

Resource/temporalCoverage/TimePeriod/name

Resource/temporalCoverage/TimePeriod/start

Resource/temporalCoverage/TimePeriod/end

**TargetPath:**

ExtrinsicObject/Slot/[@name="TimePeriod\_name" and @slotType="urn:x-ogc:specification:csw-ebriim:ObjectType:DDMS\_Slot\_Types:String"]/ValueList/Value

ExtrinsicObject/Slot/[@name="TimePeriod\_start" and @slotType="urn:x-ogc:specification:csw-ebriim:ObjectType:DDMS\_Slot\_Types:ExtendedCombinedDateType"]/ValueList/Value

ExtrinsicObject/Slot/[@name="TimePeriod\_end" and @slotType="urn:x-ogc:specification:csw-ebriim:ObjectType:DDMS\_Slot\_Types:ExtendedCombinedDateType"]/ValueList/Value

-----

**Source Path:**

Resource/geospatialCoverage/GeospatialExtent/geographicIdentifier/name

Resource/geospatialCoverage/GeospatialExtent/geographicIdentifier/region

**Target Path:**

UNMAPPED (not used in OWS4 testbed)

-----

**Source Path:**

Resource/geospatialCoverage/GeospatialExtent/geographicIdentifier/countryCode@qualifier

Resource/geospatialCoverage/GeospatialExtent/geographicIdentifier/countryCode@value

**Target Path:**

UNMAPPED (not used in OWS4 testbed)

---

**Source Path:**

Resource/geospatialCoverage/GeospatialExtent/geographicIdentifier/facilityIdentifier/@beNumber

Resource/geospatialCoverage/GeospatialExtent/geographicIdentifier/facilityIdentifier/@osuffix

**Target Path:**

ExtrinsicObject/Slot/[@name="FacilityIdentifier\_beNumber" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:String"]/ValueList/Value

ExtrinsicObject/Slot/[@name="FacilityIdentifier\_osuffix" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:String"]/ValueList/Value

---

**Source Path:**

Resource/geospatialCoverage/GeospatialExtent/boundingBox/WestBL

Resource/geospatialCoverage/GeospatialExtent/boundingBox/EastBL

Resource/geospatialCoverage/GeospatialExtent/boundingBox/SouthBL

Resource/geospatialCoverage/GeospatialExtent/boundingBox/NorthBL

**Target Path:**

ExtrinsicObject/Slot/[@name="boundingBox"]/ValueList/Value

**Notes:**

The values are converted to a gml:Envelope and stored in a Slot named "boundingBox".

---

**Source Path:**

Resource/geospatialCoverage/GeospatialExtent/verticalExtent/@unitOfMeasure  
Resource/geospatialCoverage/GeospatialExtent/verticalExtent/@datum  
Resource/geospatialCoverage/GeospatialExtent/verticalExtent/MinVerticalExtent  
Resource/geospatialCoverage/GeospatialExtent/verticalExtent/MaxVerticalExtent

**Target Path:**

ExtrinsicObject/Slot/[@name="VerticalExtent\_UOM" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:String"]/ValueList/Value

ExtrinsicObject/Slot/[@name="VerticalExtent\_Datum" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:String"]/ValueList/Value

ExtrinsicObject/Slot/[@name="VerticalExtent\_Min" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:Double"]/ValueList/Value

ExtrinsicObject/Slot/[@name="VerticalExtent\_Max" and @slotType="urn:x-ogc:specification:csw-ebrim:ObjectType:DDMS\_Slot\_Types:Double"]/ValueList/Value

---

**Source Path:**

Resource/geospatialCoverage/GeospatialExtent/boundingGeometry

**Target Path:**

ExtrinsicObject/Slot/[@name="boundingGeometry"]/ValueList/Value

**Note:**

The bounding geometry is either a gml:Point or gml:Polygon according to the DDMS schema.

-----

**Source Path:**

Resource/geospatialCoverage/GeospatialExtent/postalAddress/street  
 Resource/geospatialCoverage/GeospatialExtent/postalAddress/city  
 Resource/geospatialCoverage/GeospatialExtent/postalAddress/state  
 Resource/geospatialCoverage/GeospatialExtent/postalAddress/postalCode  
 Resource/geospatialCoverage/GeospatialExtent/postalAddress/countryCode@qualifier  
 Resource/geospatialCoverage/GeospatialExtent/postalAddress/countryCode@value

**Target Path:**

Organization/Address/@streetNumber  
 Organization/Address/@street  
 Organization/Address/@city  
 Organization/Address/@stateOrProvince  
 Organization/Address/@postalCode  
 Organization/Address/@country

## 6 Security Tags

Two approaches were tried to map security information into eBRIM.

The first approach was to use eBRIM Slots to store the security metadata. The problem with this approach was that some of the DDMS metadata elements are mapped to Slots and it not possible to dynamically extend a Slot by adding a Slot. Methods were developed to circumvent this limitation but they relied on non-standard conventions and in the end were determined to be awkward to query.

The second approach tried was to extend the ebRIM model by adding a **Security** element. The following XML-Schema fragment defines the necessary extension:

```
<element name="Security" type="DDMS_Security_Type"/>
<complexType name="DDMS_Security_Type">
  <attribute name="parent" type="tns:referenceURI" use="required"/>
  <attribute name="about" type="string" use="optional"/>
  <attribute name="name" type="string" use="optional"/>
  <attributeGroup ref="ICISM:SecurityAttributesGroup"/>
</complexType>
```

The **parent** attribute references the id of the resource to which the security information is being attached. The **about** attribute references the name of the ebRIM element to which the security information is attached. The **name** attribute is used to further qualify the value of the **about** attribute (e.g. as would be required if the security information was attached to a Slot).

Example:

Consider the title element of a resource with the identifier "urn:x-nsgfc:GMLAS:id:gsip:v1.8\_draft:AerialFarm":

```
<title ICISM:classification="U" ICISM:ownerProducer="U.S. National Geospatial-Intelligence Agency">Aerial Farm</title>
```

In ebRIM, the DDMS title element would be mapped to a Slot:

```
<ExtrinsicObject id="urn:x-nsgfc:GMLAS:id:gsip:v1.8_draft:AerialFarm">
  <Slot name="title"
    slotType="urn:x_ogc:specification:csw_ebrim:ObjectType:DDMS_Slot_Types:String">
    <ValueList>
      <Value>Aerial Farm</Value>
    </ValueList>
```

```

</Slot>
</ExtrinsicObject>

```

and the security metadata, using the newly added Security element, would be mapped as:

```

<Security
  parent="urn:x-nsgfc:GMLAS:id:gsip:v1.8_draft:AerialFarm"
  about="Slot"
  name="title"
  ICISM:classification="U"
  ICISM:ownerProducer="U.S. National Geospatial-Intelligence Agency"/>

```

## 7 Mapping to OGC Core Properties

The following table defines the mapping of DDMS metadata elements to the OGC Core Properties. Since both DDMS and the OGC Core Properties are based on Dublin Core metadata elements, the mapping is straightforward.

DDMS Metadata Element	OGC Core Element
ddms:identifier	dc:identifier
ddms:title	dc:title
ddms:subtitle	dct:alternative
ddms:description	dc:description
ddms:language	dc:language
ddms:dates/@created	dct:dateSubmitted
ddms:dates/@posted	dct:issued

<b>DDMS Metadata Element</b>	<b>OGC Core Element</b>
ddms:dates/@validTil	dct:valid
ddms:rights	dc:rights
ddms:source	dc:source
ddms:type	dc:type
ddms:creator	dc:creator
ddms:publisher	dc:publisher
ddms:contributor	dc:contributor
ddms:format	dc:format
ddms:subjectCoverage	dc:coverage
ddms:temporalCoverage	dct:temporal
ddms:geospatialCoverage	dct:spatial
sm:relation	dc:relation



## Annex C Mapping Templates

See OASIS [1] for discussion of using the following source to ebRIM mapping templates.

### 1 Types

Source Concept	ebRIM Object Type Name	ebRIM Parent Type Name	ebRIM code	ebRIM ObjectType ID	Comment
----------------	------------------------	------------------------	------------	---------------------	---------

### 2 Attributes

Source Attribute	ebRIM Attribute	Cardinality	Comment
------------------	-----------------	-------------	---------

### 3 Associations

AssociationType	ID	Description
-----------------	----	-------------

Association Source Object Type	Association Target Object Type	ebRIM Association Type	ebRIM Association Name	Comment
--------------------------------	--------------------------------	------------------------	------------------------	---------

### 4 Classifications

Name	ID	Reference	Comment
------	----	-----------	---------

### 5 Stored Queries

Search by (Parameters)	AdHocQuery Expression	Description
------------------------	-----------------------	-------------

## Bibliography

- [1] OASIS regrep-tutorial-05, *ebXML Registry – A Tutorial*, Draft Technical Note, Version 0.5, April 2005. Available [online]:  
<<http://ebxmlrr.sourceforge.net/presentations/regrep-tutorial-05.pdf>>.
  
- [2] OASIS ebRR-3.0-DeploymentProfileTemplate-wd-024, *Deployment Profile Template*, Draft OASIS Profile, Version 0.2.4, February 2006. Available [online]:  
<<http://www.oasis-open.org/committees/download.php/16859/ebRR-3.0-DeploymentProfileTemplate-wd-024.pdf>>.