Open Geospatial Consortium Inc.

Date: 2005-08-23

Reference number of this OGC® project document: **05-057r3**

Version: 0.3.0

Category: OpenGIS® Discussion Paper

Editor: Jolyon Martin

OpenGIS® Catalogue Services - Application Profile for EO Products

Warning

This document is not an OGC Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard.

Document type:	Candidate OpenGIS® Implementation Specification
Document subtype:	Application Profile
Document stage:	Discussion Paper
Document language:	English

This page left intentionally blank.

Table Of Contents

Figures

Tables

Listings

i. Preface

This candidate implementation specification document describes a minimal profile to find and order data products from catalogues of Earth Observation (EO) products.

This work is not yet at the stage of a compliant profile of OGC Catalog 2.0, it describes a first set of SOAP/HTTP protocol binding that have been inspired from on the catalogue abstract model, that will be further refined in future work. The final goal being to agree to a coherent set of interfaces for catalogue and order of EO products to support access to data from heterogeneous systems dealing with derived data products from satellite based measurements of the earth's surface and environment.

This document is based on the work of the European Space Agency (ESA), the Centre National d'Etudes Spatiales (CNES) and SPOT Image performed in the context of an interoperability experiment performed using the ESA service support environment [SSE] that resulted in an internal specification called EOLI-XML [EOLI-XML]. This EOLI-XML specification was extended to support an order interface subsequently adopted by ESA for online ordering from their catalogue.

Suggested additions, changes, and comments on this draft specification 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.

ii. Submitting organisations

The following organisations submitted the original document or its revisions to the Open GIS Consortium, Inc.

ESA – European Space Agency

The editors would like to acknowledge that this work is the result of collaboration and review of many organizations and would like to thank for the comments and contributions from CNES, ASI, DLR, EUCS, JRC, Intecs, Spacebel, Spot Image, Siemens, Kell, Seibersdorf, (also noting that this does not imply a complete endorsement from these organizations).

iii. Document contributor contact points

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

Name	Organization
Jolyon Martin	ESA
Pier Giorgio Marchetti	ESA

iv. Revision history

Date	Internal version	Editor	Primary clauses modified	Description
16 June 2005	0.1.0	Jolyon Martin	N/A	Initial Document
8 Aug 2005	0.1.0	Carl Reed	N/A except added terms and definitions	Put in proper template.
23 Aug 2005	0.2.0	Jolyon Martin	Preface and Introduction	Improvement to the overview information, setting the context of discussion paper
10 Jan 2005	0.3.0	Jolyon Martin	Title changed, annex added for mapping to CAT 2.0	Following review at OGC TC meeting Nov. 2005 it was agreed to rename the document, and to add an annex to describe the foreseen alignement towards CAT 2.0. The occasion was also take to make a few corrections to the specification based on operational experiences and provide additional information on the metadata fields.

v. Changes to the OpenGIS® Catalogue Services Specification

To be confirmed as a result of ongoing work. Any changes will be documented as official change request proposals and submitted to the OGC.

vi. Future work

In parallel to the OGC review process, this profile will be the subject of further revision in the context of the Heterogeneous Missions Accessibility study in the framework of the European Space Agency GMES preparatory activities. A number of national space agencies, which have missions contributing to GMES, will participate to this study, as well as an industrial consortium, which is being selected via an ESA open tender action.

The Agencies, which have already confirmed their participation, are:

ASI – Italian Space Agency

CNES – French Space Agency,

DLR – German Space Agency

EUSC – European Union Satellite Centre

The profile has been purposefully kept minimal to keep complexity for providers, enhancements may be considered, and will be discussed in context of the GMES preparatory work.

Areas of further work include discovery of product collections (e.g capabilities decription) should be aligned to the general discovery requirements of INSPIRE, better compliance with OGC 2.0 Catalogue Services (as outlined in Annex D), as well as order interface security considerations.

vii. Foreword

This document is a candidate profile of the OpenGIS Catalogue 2.0 Implementation Specification.

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.

Annex A, B, the Abstract Conformance Test Suite, are normative to this specification and shall be implemented when a computing environment requires catalogue services. All other annexes are informative and provide background information.

Introduction

The services proposed in this profile are intended to support the identification and subsequent ordering of EO data products from previously identified data collections. The intent of this initial profile is to describe a minimum interface that can be supported by many data providers (satellite operators, data distributors …), most of whom have existing (and relatively complex) facilities for the management of these data.

The focus of this paper is the identification and ordering of EO data products from previously identified collections (catalogues). EO data product collections are usually structured to contain data items derived from a sensor onboard a satellite or series of sensors. The key characteristics differentiating these products are date of acquisition, location and in some cases, such as the optical imagery, the possible presence of cloud obscuring the image. These are the key characteristics; there are however other metadata that are required to identify products of interest. The data collections may be very extensive, in many cases products derived from satellites data have resulted in dataset series of over 10 years, depending on the product sizing conventions this easily equates to millions of items in a collection, it is an important aspect of this profile that in the case of an interactive query session, the client is able to efficiently refine a query to identify the products of interest. However, it is equally an important feature that the profile presented here may be used in the context of computer-to-computer interaction to allow routine / complex processing tasks to be automated, for example as part of an order entry work flow.

The target audience for this document includes catalogue users, client developers, service implementers, and system testers. The candidate specification encompasses three interrelated views that reflect different viewpoints on a catalogue service. Each viewpoint[1] focuses on different areas of concern:

- *Enterprise* – describes the general capabilities of the service in light of functional and nonfunctional requirements (for catalogue users and system testers);
- *Information* – defines the kinds of information handled by the catalogue and the policies to be enforced (for catalogue users, developers, and testers);
- *Computational* – specifies the public interfaces, allowable interactions, and protocol bindings (for developers and testers).

This document presents work towards an application profile of an OGC Catalogue service. The document does not comply fully with the rules articulated in clause 11 of the *OpenGIS Catalogue Services Specification*, version 2.0 (OGC 04-021r2). This should be addressed as further work. However, the document presents the profile in the structure required for application profiles in order to facilitate the positioning of this work as a possible standard. Since the interfaces expressed are operationally used and maintained it was considered better to present the actual interfaces as they are. An informative annex D has been added in order to show the desireable evolution of the profile to better comply with OGC 04-021r2.

[1] The Reference Model of Open Distributed Processing (RM-ODP, ISO/IEC 10746) is the architectural framework adopted by the OGC and ISO/TC 211 for specifying software-intensive systems. In IEEE 1471 terminology the RM-ODP framework provides a set of *library viewpoints*.

OpenGIS® Catalogue Services – Minimum Profile for EO Products using WSDL and SOAP

1 Scope

This application profile document describes the interfaces, bindings and encodings required to search, present and order data from catalogues of Earth Observation (EO) products. The profile is presented as a minimum requirement for catalogue interoperability within the EO domain. It is anticipated that implementation of this profile can be supported in a cost-effective manner by providers of EO products.

2 Conformance

Any software product claiming conformance with this application profile shall pass all requirements described in the following abstract test suites. The definition of an abstract test suite appears in ISO 19105.

This application profile defines two classes/levels of conformance: class A, and class B. Class A defines the minimum services for catalogue search and present, class B extends the conformance towards ordering. Any software product claiming conformance with one of these classes shall pass all the requirements described in the corresponding abstract test suites below.

2.1 Abstract test suite for conformance class/level A, catalogue services

See Annex A for the ATS for conformance class A

2.2 Abstract test suite for conformance class/level B, ordering services

See Annex B for the ATS for conformance class B.

3 Normative references

The following normative documents contain provisions that, 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.

ISO 19105:2000, *Geographic information — Conformance and Testing*

OGC 05-008, *OGC Web Services Common Specification*

This OWS Common Specification contains a list of normative references that are also applicable to this Implementation Specification.

NOTE: Additional normative references to be confirmed, for a complete set of references see the Bibliography.

4 Terms and definitions

For the purposes of this document, the following terms and definitions apply:

For the purposes of this specification, the definitions specified in Clause 4 of the OWS Common Implementation Specification [OGC 05-008] shall apply. In addition, the following terms and definitions apply.

4.1
data clearinghouse
collection of institutions providing digital data, which can be searched through a single interface using a common metadata standard [ISO 19115]

4.2
data level
stratum within a set of layered levels in which data is recorded that conforms to definitions of types found at the application model level [ISO 19101]

4.3
dataset series
collection of datasets sharing the same product specification [ISO 19113, ISO 19114, ISO 19115]

4.4
identifier
a character string that may be composed of numbers and characters that is exchanged between the client and the server with respect to a specific identity of a resource

4.5
metadata dataset
metadata describing a specific dataset [ISO 19101]

4.6
metadata entity
group of metadata elements and other metadata entities describing the same aspect of data

NOTE 1 A metadata entity may contain one or more metadata entities.

NOTE 2 A metadata entity is equivalent to a class in UML terminology [ISO 19115].

4.7
metadata schema
conceptual schema describing metadata

NOTE ISO 19115 describes a standard for a metadata schema. [ISO 19101]

4.8
metadata section
subset of metadata that defines a collection of related metadata entities and elements [ISO 19115]

4.9
parameter
variable whose name and value are included in an operation **request** or **response**

4.10
profile
set of one or more base standards and - where applicable - the identification of chosen clauses, classes, subsets, options and parameters of those base standards that are necessary for accomplishing a particular function [ISO 19101, ISO 19106]

4.11
qualified name
name that is prefixed with its naming context

EXAMPLE The qualified name for the road no attribute in class Road defined in the Roadmap schema is RoadMap.Road.road_no. [ISO 19118].

4.12
schema
formal description of a model [ISO 19101, ISO 19103, ISO 19109, ISO 19118]

4.13
service interface
shared boundary between an automated system or human being and another automated system or human being [ISO 19101]

4.14
state
condition that persists for a period

NOTE The value of a particular feature attribute describes a condition of the feature [ISO 19108].

4.15
transfer protocol
common set of rules for defining interactions between distributed systems [ISO 19118]

5 Conventions

5.1 Abbreviated Terms

Some frequently used abbreviated terms:

API Application Program Interface

CSW Catalogue Service-Web

CEOS Committee on Earth Observation Satellites

DC Dublin Core

EO Earth Observation

HTTP HyperText Transport Protocol

I/F Interface

ICD Interface Control Document

ISO International Organisation for Standardisation

N/A Not Applicable

OGC Open Geospatial Consortium

SOAP Simple Object Access Protocol

TCP/IP Transmission Control Protocol/Internet Protocol

UML Unified Modeling Language

URI Uniform Resource Identifier

URL Uniform Resource Locator

URN Uniform Resource Name

UTF-8 Unicode Transformation Format-8

WSDL Web Service Definition Language

W3C World Wide Web Consortium

XML eXtensible Markup Language

XSD XML Schema Definition

6 Overview and System context

This section focuses on the purpose, scope and policies of catalogue services that comply with the given profile. It documents special requirements and describes the context of use.

6.1 Application domain

The catalogue services proposed in this profile are intended to support the identification and subsequent ordering of EO data products. The intent of the profile is to describe a minimum interface that can be supported by many data providers (e.g. - satellite operators, data distributors, ancillary content providers), many of whom have existing facilities.

Points to be discussed: large historic archive of data, push for online ordering to reduce operations costs associated to order desk, context of GMES

It is an important feature that the profile presented here may be used in the context of computer-to-computer interaction to allow routine / complex processing tasks to be automated. The need to support online ordering is also a critical feature of the Catalogue Services. The services presented may form the basis for workflows within service chaining and are presented in the light of an overall service oriented architecture. These considerations have influenced the profile to use the following W3C and OASIS recommendations.

Message-based SOAP (Simple Object Access Protocol) [SOAP] over HTTP or HTTPS for secure communication is used as protocol between the client application and the catalogue service. SOAP is firewall-friendly, and platform independent. It is thus well suited to integrate services in a heterogeneous environment.

WSDL (Web Services Description Language) 1.1 [WSDL] is used to define the SOAP interface of a service in a formal way that can be processed by the workflow editor.

XML Schema (from W3C) [XML] is used within this profile to define the structure of the operations input and output XML messages.

6.2 Essential use cases

This profile identifies four basic operations:

- The “Search” operation is used to perform a query on a remote catalogue. It returns the meta-data available in the catalogue for the EO products matching the search parameters such as area of interest and time interval. The client may present the search results in textual (list) format, and graphically on a map.
- The “Present” operation allows requesting detailed information about a search result (information on the metadata and possibly a graphical overview of the EO product).
- The “Order” operation allows a product to be requested. In many instances the EO product may not be available online, at one extreme it may have to be tasked by the satellite, or possibly it may have to be retrieved from an archive. The order may require some processing to be performed before delivery to the user, the order allows for the selection of reduced area (scene selection) and processing, delivery options that should be applied.
- The “OrderMonitor” operation allows the status of the order to be tracked.

The profile supports synchronous operations: that is the service will immediately return the result. The client application may thus display search results on the same page where the user entered the search criteria.

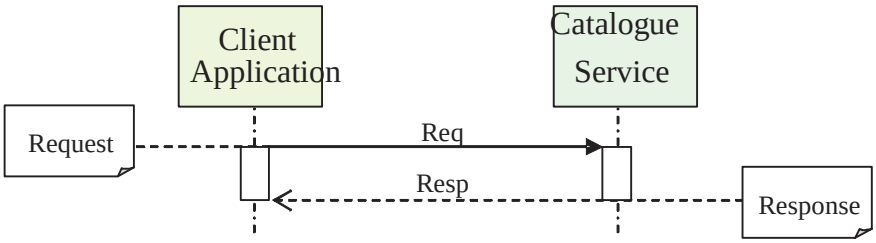

Figure 6.2: Synchronous or Request/Response Usage Scenario

The various messages that are be supported to provide these services are summarised in the table below

Input Payload Message	Output Payload Message
searchRequest	response
presentRequest	response
productOrderRequest	orderResponse
orderMonitorRequest	orderMonitorResponse

Table 6-1: Payload Message Sequence.

The system context overview presents the catalogue and order services as a whole. In practise, the catalogue services are relatively simple services, and may be offered by many organisations by implementing translations towards existing databases, these catalogue services are presented in the next section. The order services may not be supported by all data providers, and so are presented as extensions to the basic catalogue services.

7 Catalogue services

7.1 Information model for EO product catalogues

EO product catalogues provide metadata to describe the datasets derived from satellite based remote sensing which may eventually be ordered, or possibly accessed directly online. The metadata may describe physical products available in an archive, or basic acquisitions of data made by the satellite from which higher-level products may be requested, e.g. implying an additional processing of the data. The metadata may also describe acquisitions that have been planned, or potential acquisitions. The metadata framework is based on [ISO], [OpenGIS], [FGDC] standards. Extensions have been applied in order to cope these standards with this specific field of application.

The figure below illustrates the main schema for the metadata used to describe EO products within a catalogue.

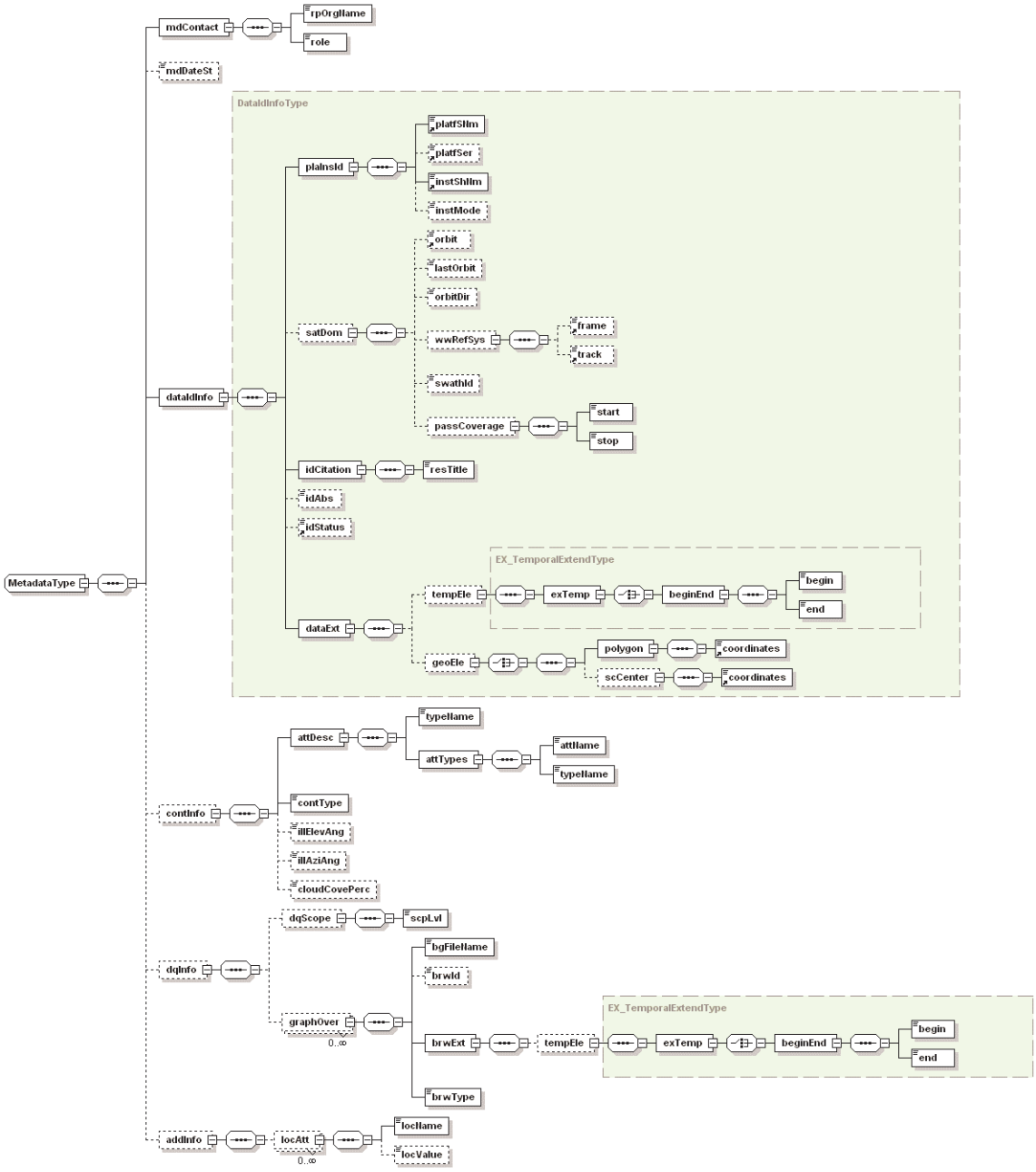

Figure 7-1: Metadata Type Diagram.

Common to many catalogue access protocols a set of presentations formats are defined to reduce the message sizes during various steps of catalogue interaction. The “Presentation” column specifies the expected use of the element in the different type of descriptors: F (full), B (brief), b (browse) and s (summary); “-“ means that the element is normally present in the response message for the specific format. Note that the schema does not prescribe tightly which elements need to be provided, it is also left to the discretion of the catalogue provider in case it is considered that certain elements are of key importance to the user for some specific collections, in this case the provider may choose to include it also in brief and summary presentations.

Tag Name	Tag Description	Presentation
mdContact	Source: [ISO], contact.	F B b s
rpOrgName	Name of the responsible organisation. Source: [ISO], organisationName.	F B b s
role	Function performed by the responsible party. Type Values: Not Empty string Permitted Value: 002 (custodian), 006 (originator), 009 (processor) Source: [ISO], role.	F B b s
mdDateSt	Metadata Date Type: date or dateTime in ISO 8601 format Date Format: CCYY-MM-DD DateTime Format: CCYY-MM-DDThh:mm[:ss]Z Source: [ISO], dateStamp Note: field "required" by ISO19115, ideally the field would mean "Date time of insertion within the catalog", but since in some catalogues metadata at are calculated on the fly based on last update of orbit information available it is also acceptable to fill with "Date time of the response"	F - - -
dataIdInfo	Source: [ISO], identificationInfo.	F B b s
plaInsId	Satellite element. Source: [FGDC], Platform_and_Instrument_identification	F B b s
platfSNm	Platform Name Type Value: string Permitted Values: see [Valids]. Source: [FGDC], Platform_Short_Name	F B b s
platfSer	Platform Identifier Type Value: string Permitted Values: see [Valids]. Source: [FGDC], Platform_Serial_Identifier	F B b s
instShNm	Sensor type. Type Value: string Permitted Values: see [Valids]. Source: [FGDC], Instrument_Short_Name	F B b s
instMode	Sensor Mode Type Value: string	F B b s
satDom	Satellite Domain element.	F B b s
orbit	Orbit Number. Type Value: integer Permitted Values: > 0 Source: 0, orbitNumber	F B b s
lastOrbit	Type Value: integer Permitted Values: > 0	F - - -
orbitDir	Orbit Direction Element. Type Value: Not Empty String Permitted Values: 0 (Ascending), 1 (Descending)	F B b s
wwRefSys	World Reference System element. Source: [FGDC], Worldwide_Reference_System	F B b s

Tag Name	Tag Description	Presentation
frame	Frame Number. Type Value: integer Permitted Values: > 0 Source: [ISO19130], frame	F B b s
track	Track Number. Type Value: integer Permitted Values: > 0 Source: [ISO19130], track	F B b s
swathId	Type Value: string Permitted Values: Not Empty String	F B b s
passCoverage	Pass Coverage element, describes the coverage of the product with respect to the orbit, specifying the start and stop time of the product relative to the ascending node crossing of the satellite. This information is typically used to subsequenty calculate possible "frames" that may be selected from the product based on a standard framing size	F B b s
start	Start pass coverage Type Value: integer	F B b s
stop	Stop pass coverage Type Value: integer	F B b s
idCitation	Source: [ISO], Identification Information	F B b s
resTitle	Product identifier element. Type Value: string Source: [ISO], Citation and responsible party information	F B b s
idAbs	Abstract element. Type Value: string Source: [ISO], Abstract. Example: ENVISAT ASAR/IM product.	F - - -
idStatus	Product Status element. Type Value: Not Empty String Permitted Values: 001 (completed), 005 (planned), 006 (required), 007 (underDevelopment), 008 (potential).	F - - -
dataExt	Extension both temporal and geographical of the product. Source: [ISO], Identification information.	F B b s
tempEle	Temporal Coverage of the product. Source: [ISO], temporalElement	F B b s
exTemp	Temporal extent information element. Source: [ISO], extent.	F B b s
beginEnd	Information containing the Product temporal coverage. Source: [ISO]	F B b s
begin	Start Date Type: dateTime in ISO 8601 format (CCYY-MM-DDThh:mm[:ss[.cc]]Z) Source: [ISO]	F B b s
end	End Date Type: dateTime in ISO 8601 format (CCYY-MM-DDThh:mm[:ss[.cc]]Z) Source: [ISO]	F B b s
geoEle	Geographic extent of the product. Source: [ISO], geographicElement	F B b s
polygon	Polygon element. Source: [ISO], polygon	F B b s

Tag Name	Tag Description	Presentation
coordinates	Coordinates element. Type: string Syntax: points are separated by blank; point's coordinates lat and long are separated by comma (,). The decimal point (.) is used to separate decimals. In order to correctly use the polygon to further allow scene selection within the acquisition the polygon shall have the following characteristics: ° first point is the first image line right corner ° second point is the first image line left corner ° N intermediate points for the left border (direction FL to LL) ° last left corner ° last right corner ° N intermediate points for the right border Example: for a polygon having 4 points: 65.00,-10.00 65.00,-8.00 63.00,-8.00 63.00,-10.00	F B b s
scCenter	Scene center element.	F B b s
coordinates	Coordinates element. Type: string Syntax: points are separated by blank; point's coordinates lat and long are separated by comma (,). The decimal point (.) is used to separate decimals. Example: 64.00,-9.00	F B b s
contInfo	Source: [ISO], contentInfo	F - - -
attDesc	Source: [ISO], attributeDescription.	F - - -
typeName	Source: [ISO19103]. Type: string Permitted Value: imageType	F - - -
attTypes	Source: [ISO19103].	F - - -
attName	Source: [ISO19103]. Type: string Permitted Value: imageQuality	F - - -
typeName	Source: [ISO19103]. Type: string Permitted Value: imageQualityType	F - - -
contType	Type: Not Empty String Permitted Values: 001(image) Source: [ISO], contentType	F - - -
illElevAng	Illumination Elevation Angle element. Type: Float Permitted Values: [-90, +90] Source: [ISO], illuminationElevationAngle	F - - -
illAziAng	Illumination Azimuth Angle element. Type: Float Permitted Values: [0, +360] Source: [ISO], illuminationAzimuthAngle.	F - - -
cloudCovePerc	Cloud Cover Percentage element. Type Value: float Permitted Values: [0, 100] Source: [ISO], cloudCoverPercentage	F - - -
dqInfo	Source: [ISO], dataQualityInfo	F - b s
dqScope	Source: [ISO], scope	F - - -
scpLvl	Level element. Type Value: string Permitted Value: DataSet Source: [ISO], level	F - - -

Tag Name	Tag Description	Presentation
graphOver	Graphic Overview element. Source: [ISO], graphicOverview	F - b s
bgFileName	HTTP URL for graphical overview, higher resolution browse should be specified in Full and Browse presentation records, lower resolution thumbnail within the Summary. Type Value: Not Empty String Source: [ISO], filename Examples: ➢ Browse HTTP URL: http://earth.esa.int:8090/XI/EN1/20000101T000000000-20000101000015100_A_B.jpg ➢ Thumbnail HTTP URL: http://earth.esa.int:8090/XI/EN1/20000101T000000000-20000101000015100_D_T.jpg	F - b s
brwId	Identifier for browse in case more than one overview is available for example thermal bands and visible bands Type Value: string Source: [ISO], graphicOverview	F - b s
brwExt	Browse Extent element.	F - b s
tempEle	Temporal Coverage of the product. Source: [ISO], temporalElement	F - b s
exTemp	Temporal extent information element. Source: [ISO], extent.	F - b s
beginEnd	Information containing the Product temporal coverage. Source: [ISO]	F - b s
begin	Start Date Type: dateTime in ISO 8601 format (CCYY-MM-DDThh:mm[:ss[.cc]]Z) Source: [ISO]	F - b s
end	End Date Type: dateTime in ISO 8601 format (CCYY-MM-DDThh:mm[:ss[.cc]]Z) Source: [ISO]	F - b s
brwType	Browse Type element. Type Value: Not Empty String	F - b s
addInfo	Additional information element.	F - - -
locAtt	Local Attribute element.	F - - -
locName	Name of Local Attribute. Type Value: string	F - - -
locValue	Value of Local Attribute. Type Value: string	F - - -

Table 7-1: Metadata Type.

7.2 External interfaces

7.2.1 Message sequencing

Three main XML elements have been identified for the Catalogue Search Interface:

- **searchRequest**: this message allows to identify products inside a single collection via a search condition and to request the number of hits or product metadata in different formats as Response.
- **presentRequest**: this message allows to retrieve (multiple) product metadata within a single collection by providing the product identification.

- **response**: this message contains the result in form of number of hits or returned product metadata

The messaging assumes a stateless interaction, it is expected that a series of search and present requests may be used to iterate over the results set from a search. It is therefore expected that the catalogue server returns results to a search in a predictable order.

The hand shaking between the SOAP/HTTP server and a SOAP/HTTP client will be the same for both Search and Present request messages. The SOAP/HTTP client sends a SOAP request message. When the response is ready, the SOAP/HTTP server builds the response message and sends the SOAP response to the client.

The message-based variant of SOAP will be used. This means that style="document" are used in the WSDL file as attribute of the <binding> element.

Table 7-2 shows the SOAP Body content messages exchanged between the EOLI-XML SOAP/HTTP server and a SOAP/HTTP client

Input Payload Message	Output Payload Message
searchRequest	response
presentRequest	response

Table 7-2: Payload Message Sequence.

7.2.2 Search request message

The body of this message is the “searchRequest XML element, derived from [OpenGIS].

Figure 7-2 gives a graphical representation:

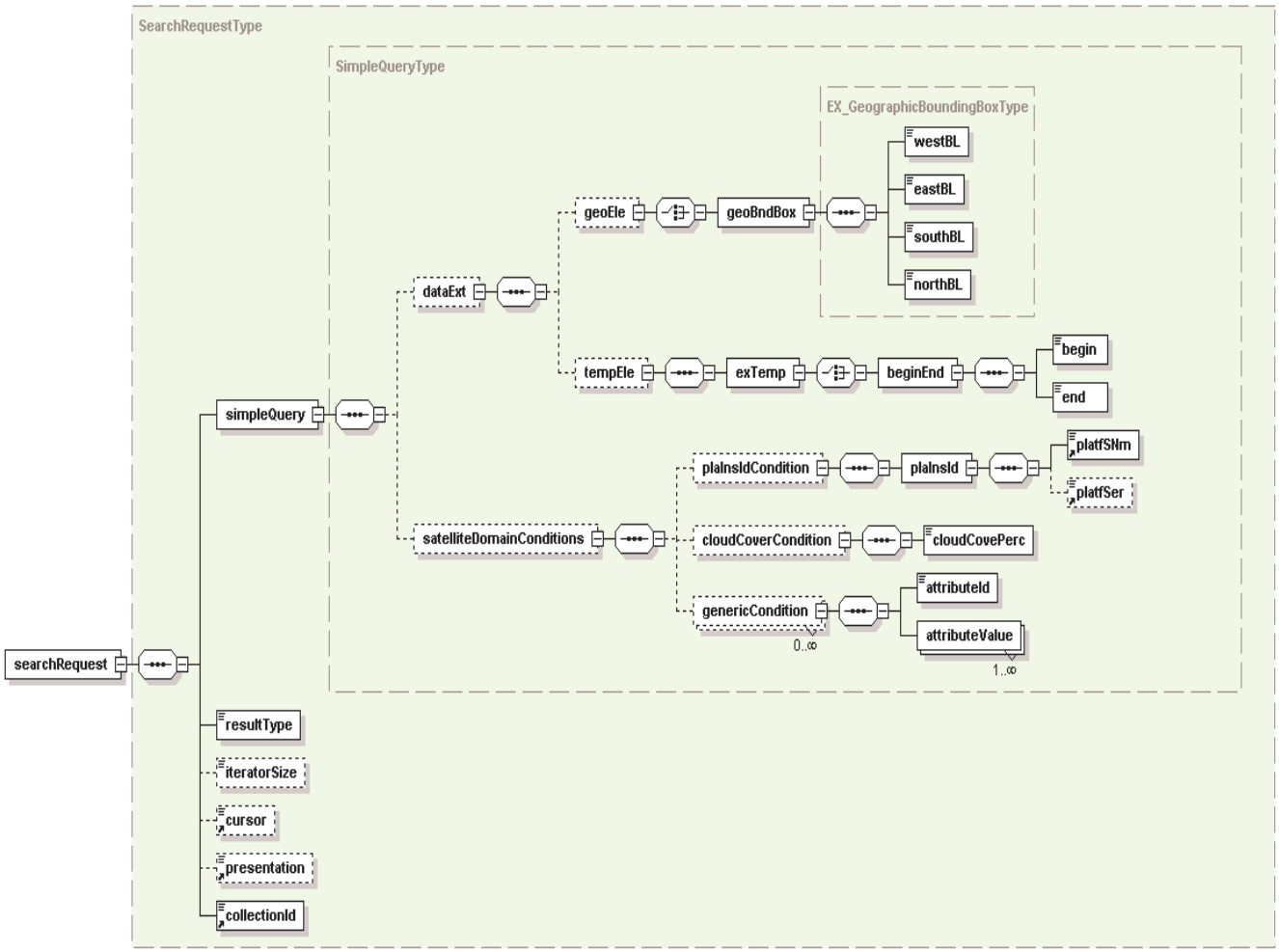

Figure 7-2: SearchRequest Diagram.

Tag Name	Tag Description
simpleQuery	SimpleQuery element. It contains the query conditions.
DataExt	Extension both temporal and geographical of the Metadata. This represents the spatial and temporal query condition. Source: **[ISO]**, Identification information.
geoEle	Geographic extent of the product. Source: **[ISO]**, geographicElement **Attribute Name**: operator **Attribute Permitted values**: OVERLAP
geoBndBox	Geographic Bounding Box element. Source: **[ISO]**, EX_GeographicBoundingBox
westBL	Western-most coordinate. Type: Float Permitted Values: [-180, +180] Source: **[ISO]**, westBoundLongitude
eastBL	Eastern-most coordinate. Type: Float Permitted Values: [-180, +180] Source: **[ISO]**, eastBoundLongitude

Tag Name	Tag Description
southBL	Southern-most coordinate. Type: Float Permitted Values: [-90, +90] Source: **[ISO]**, southBoundLongitude
northBL	Northern-most coordinate. Type: Float Permitted Values: [-90, +90] Source: **[ISO]**, northBoundLongitude
TempEle	Temporal Coverage of the product. Source: **[ISO]**, temporalElement **Attribute Name**: operator **Attribute Permitted values**: OVERLAP
exTemp	Temporal extent information element. Source: **[ISO]**, extent
beginEnd	Information containing the Product temporal coverage. Source: **[ISO]**
begin	Start Date Type: date in ISO 8601 format (CCYY-MM-DD) Source: **[ISO]**
end	End Date Type: date in ISO 8601 format (CCYY-MM-DD) Source: **[ISO]**
satelliteDomainConditions	SatelliteDomainConditions element.
plaInsIdCondition	Satellite and Instrument Identification. **Attribute Name**: operator **Attribute Permitted values**: EQUAL
plaInsId	Satellite element. Source: [FGDC], Platform_and_Instrument_identification
platfSNm	Platform Name Type Value: string Permitted Values: see [Valids]. Source: [FGDC], Platform_Short_Name
platfSer	Platform Identifier Type Value: string Permitted Values: see [Valids]. Source: [FGDC], Platform_Serial_Identifier Note: multiple values allowed in search request, in which case, a logical OR is expected to be applied
cloudCoverCondition	cloudCoverageCondition element. **Attribute Name**: operator **Attribute Permitted values**: LESS EQUAL.
cloudCovePerc	Cloud Cover Percentage element. Type Value: float Permitted Values: [0, 100] Source: **[ISO]**, cloudCoverPercentage
genericCondition	genericCondition element. **Attribute Name**: operator **Attribute Permitted values**: EQUAL, LESS EQUAL, LESS, GREATER, GREATER EQUAL, WITHIN, INTERSECTS.
attributeId	Attribute identifier. Type Value: Not empty string
attributeValue	Attribute value. Type Value: Not empty string
ResultType	Type of data to be returned in a query response message and the behaviour of the message response. Type Value: string Permitted Values: hits, results. Source: [OpenGIS], CG_QueryRequest.

Tag Name	Tag Description
IteratorSize	Maximum number of the result set entries to be returned in the query response. Type Value: Integer Permitted Values: > 0 Source: [OpenGIS], CG_QueryRequest and CG_QueryResponse.
Cursor	First result set entry to be returned in the query response. Type Value: Integer Permitted Values: > 0 Source: [OpenGIS], CG_QueryRequest
presentation	Type of the descriptors returned in the query response. Type: string Permitted values: brief, summary, full, browse Source: [OpenGIS], CG_QueryRequest
CollectionId	Search space for the query requestType Value: string Permitted Values: Not empty string Syntax: Provider.Facility.Collection identifying a pre-defined TargetService (e.g. ESA.EECF.ERSE_SER for the ESA ERS SAR raw data catalogue). Source: [OpenGIS], CG_QueryRequest

Table 7-3: Search Request Tags

7.2.3 Present request message

The body of this message is the “presentRequest” XML element, derived from [OpenGIS].

Figure 7-3 gives a graphical representation:

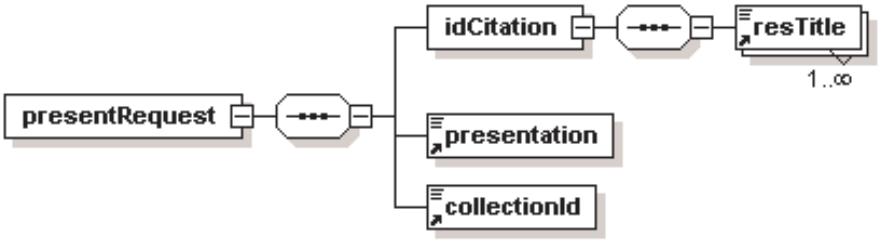

Figure 7-3: PresentRequest Diagram

Tag Name	Tag Description
idCitation	Source: **[ISO]**, Identification Information
resTitle	Product identifier element. Type Value: string Permitted Values: Not empty string Source: **[ISO]**, Citation and responsible party information Syntax: The identifier format is not mandated by the specification, but is expected to be a persistent identifier.
presentation	Type of the descriptors returned in the response. Type: string Permitted values: brief, summary, full, browse Source: [OpenGIS], CG_QueryRequest

Tag Name	Tag Description
collectionId	Search space for the query request. Type Value: string Permitted Values: Not empty string Syntax: The identifier format is not mandated by the specification. Source: [OpenGIS], CG_QueryRequest

Table 7-4: Present Request Tags

7.2.4 Response message

The body of this message is the “response” XML element, derived from [OpenGIS].

Figure 7-4 gives a graphical representation:

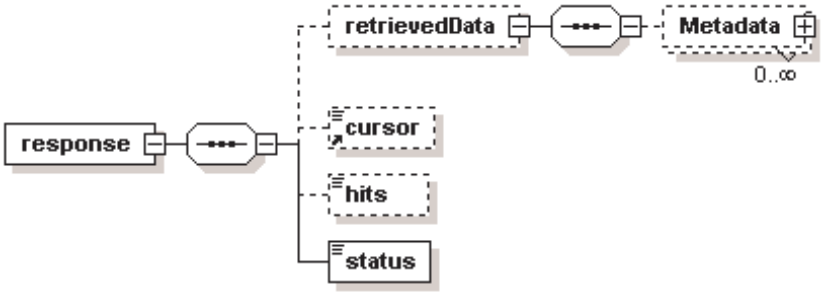

Figure 7-4: Response Diagram.

Tag Name	Tag Description
retrievedData	Subset of the results of the query request. It is organised and formatted as specified in the resultType and/or presentation parameters. Source: [OpenGIS], CG_QueryResponse **Attribute Name**: presentation **Attribute Permitted values**: brief, summary, full, browse
Metadata	Source: **[ISO]**, Metadata
Cursor	Last item in the result set that was returned in the retrieved data set. This element is not present in the present response messages. Type Value: Integer Permitted Values: >0 Source: [OpenGIS], CG_QueryResponse
Hits	Number of entries in the results set. Type Value: Integer Permitted Values: >=0 Source: [OpenGIS], CG_QueryResponse
Status	Status of the query request. Type Value: string Permitted Values: success, failure, partial Source: [OpenGIS], CG_QueryResponse Note: partial value would be used to indicate that the server may have limited the number of results returned from a query, e.g. to protect from overload

Tag Name	Tag Description
errorMessage	Error Message element. **Type:** Not empty string.

Table 7-5: Response Tags

8 Ordering Services

8.1 Information model for EO product ordering

In order to specify an order, a product from the catalogue has to be identified, and the appropriate order options specified. The following elements present the order options

Figure 8-1 gives a graphical representation of “ProductServiceOptions” Type:

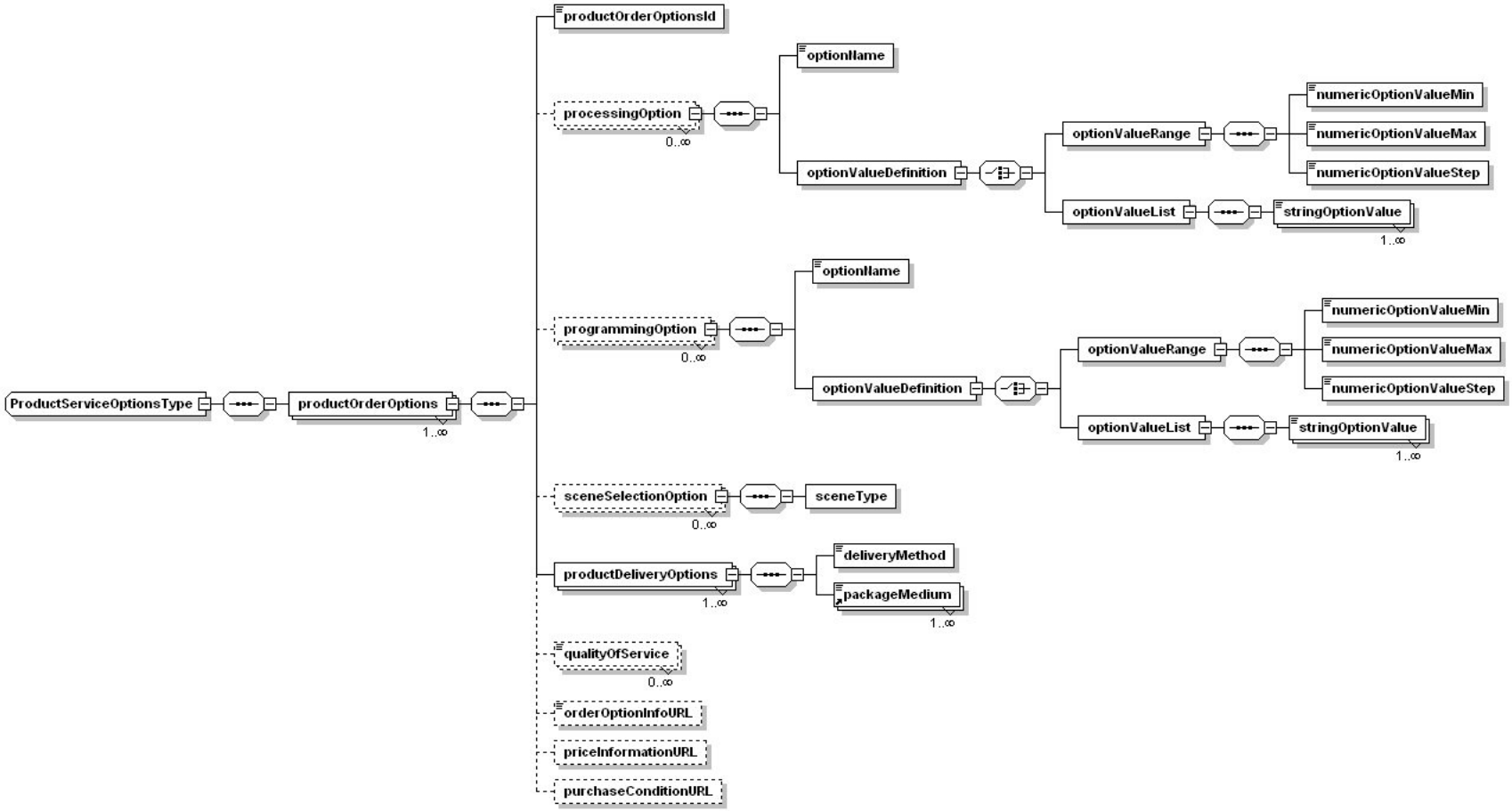

Figure 8-1: ProductServiceOptions Type Diagram.

Tag Name	Tag Description
productOrderOptions	ProductOrderOptions element. It contains the information about the different processing and/or order options available for the product order.
productOrderOptionsId	Identifier of the specific option. **Type:** Not empty string **Example:** "on-line retrieval"
processingOption	Processing option available for an order option group
optionName	Name of a processing option. **Type:** Not empty string
optionValueDefinition	Defines the allowed values that can be selected for a processing option.
optionValueRange	Describes the allowed value range of a numeric values for a processing option.
numericOptionValueMin	Minimum numeric value allowed for a processing option. **Type:** Float
numericOptionValueMax	Maximum numeric value allowed for a processing option. **Type:** Float
numericOptionValueStep	Resolution allowed to modify the numeric value allowed for a processing option. **Type:** Float
optionValueList	Lists the selectable values for a specific processing option.
stringOptionValue	String representation of a single allowed value out of a list of selectable values for a processing option. **Type:** Not empty string
programmingOption	Programming option available for an order option group
optionName	Name of a programming option. **Type:** Not empty string
optionValueDefinition	Defines the allowed values that can be selected for a programming option.
optionValueRange	Describes the allowed value range of a numeric values for a programming option.
numericOptionValueMin	Minimum numeric value allowed for a programming option. **Type:** Float
numericOptionValueMax	Maximum numeric value allowed for a programming option. **Type:** Float
numericOptionValueStep	Resolution allowed to modify the numeric value allowed for a programming option. **Type:** Float
optionValueList	Lists the selectable values for a specific programming option.
stringOptionValue	String representation of a single allowed value out of a list of selectable values for a programming option. **Type:** Not empty string
sceneSelectionOption	Identifies a scene selection option available for an order option group.
sceneType	Identifier for the scene type specific to the selected product. Detailed characteristics of the scene type are expected to be specified under link "orderOptionsInfoURL" **Type:** Not empty string (max 20 chars)
productDeliveryOptions	Delivery Options for the order item.
deliveryMethod	Delivery methods valid for the delivery medium. **Type:** String **Permitted Values**: mail, ftp-pull, ftp-push
packageMedium	Identification of a delivery Medium and a Medium Formatting Option. **Type:** Not empty string (max 40 chars) **Examples:** NTP, DAT, Exabyte, CD-ROM, DLT, D1, DVD, file.
qualityOfService	Quality of service available **Type:** Not empty string (max 20 chars) **Examples:** Standard, Rush, NRT

Tag Name	Tag Description
orderOptionInfoURL	Pointer to external information about the current order option. **Type:** Not empty string (max 255 chars)
priceInformationURL	Pointer to external information about the prices for the current order option. **Type:** Not empty string (max 255 chars)
purchaseInformationURL	Pointer to external information about the purchase conditions for the current order option. **Type:** Not empty string (max 255 chars)

Table 8-1: ProductServiceOptions Type.

8.2 External interfaces

8.2.1 Messaging sequence

The basic XML elements identified for Catalogue Search are extended and new elements are introduced for the Order Interface, the new elements introduced are:

- **productOrderRequest** this message allows to submit an order by providing one or more product identification and its collection.
- **orderMonitorRequest** this message allows to monitor the order status by providing the order identifier and to retrieve all orders for a user that have been updated since a given date.
- **orderResponse** this message contains the result in form of the order identifier.
- **orderMonitorResponse** this message contains the result in form of the order status returned.

8.2.2 Search request message extension for ordering

The searchRequest is extended in order to retrieve the order option values needed to submit an order. Changes to the "SearchRequestType"comprise the addition of the value "orderOption" to the list of possible values for the "presentation" element and the insertion of a new optional element "userInformation" (see Figure 8-2). The "userInformation" element is provided in order to retrieve those "productServiceOptions" reserved to privileged users.

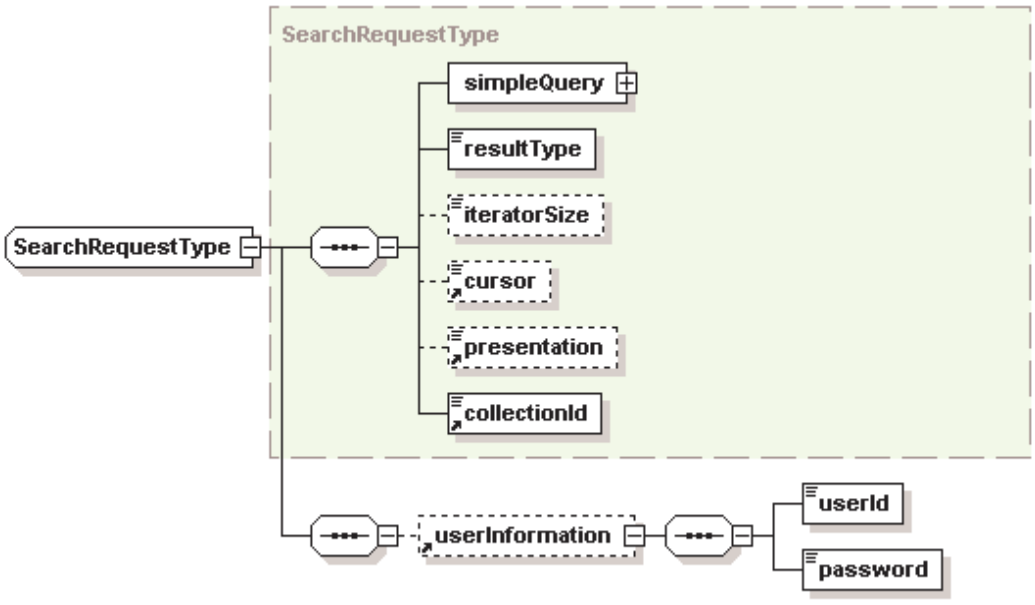

Figure 8-2: The extended SearchRequestType

8.2.3 Present request message extension for ordering

Tag Name	Tag Description
presentation	Type of the descriptors returned in the query response. **Type:** String **Permitted Values:** brief, summary, full, browse, orderOption
userInformation	It contains the personal user information as provided as input by the order issuer.
userId	User identifier. **Type:** Not empty string
password	Password to authenticate the user. **Type:** Not empty string

Table 8-2: Additional/Extended Search Request Tags

Changes to the “PresentRequestType”comprise the addition of the value “orderOption” to the list of possible values for the “presentation” element and the insertion of a new optional element “userInformation”. The “userInformation” element is provided in order to retrieve those “productServiceOptions” reserved to privileged users.

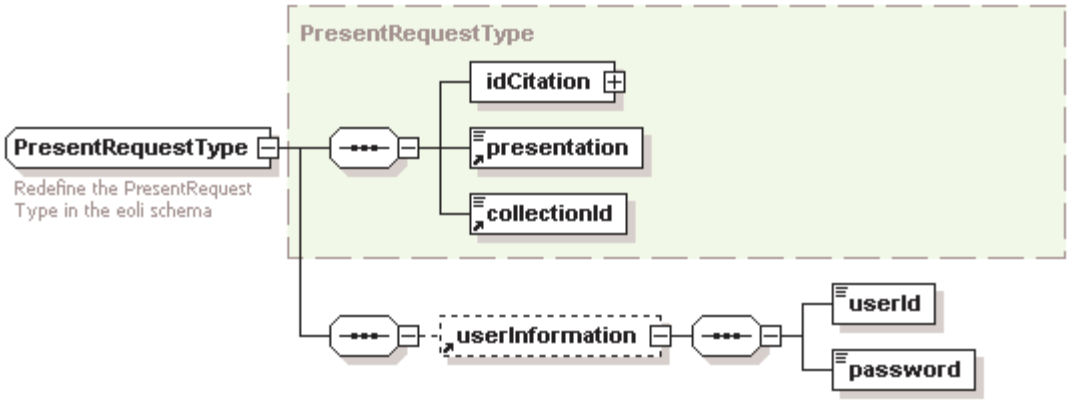

Figure 8-3: The extended PresentRequestType

Tag Name	Tag Description
Presentation	See Table 7-3
userInformation	See Table 7-3
UserId	See Table 7-3
Password	See Table 7-3

Table 8-3: Additional/Extended Present Request Tags

8.2.4 Response message extensions for ordering

The body of this message is the "response" XML element as described in 7.2.4. Only the Metadata Type has been extended in order to include the order option values needed to submit an order.

The XML response message to an XML search or present request with presentation values "orderOption" will include the Metadata Type element described in 7.1 in "brief" format and the productServiceOptions element. The type of the productServiceOptions element is described in the next paragraph.

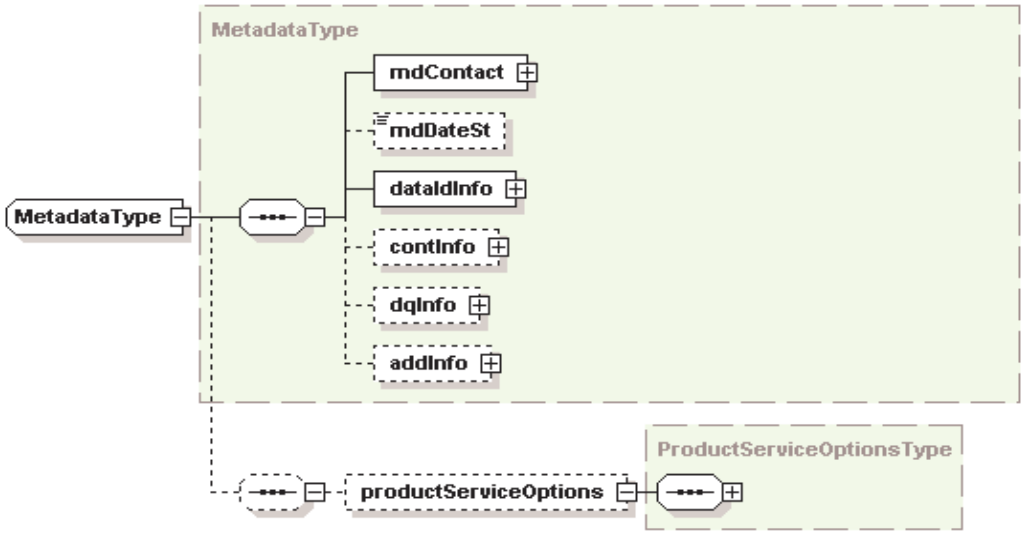

Figure 8-4: Metadata Type Diagram

8.2.5 Product order request message

The body of this message is the "productOrderRequest" XML element.

Figure 8-5 gives a graphical representation of productOrderRequest element:

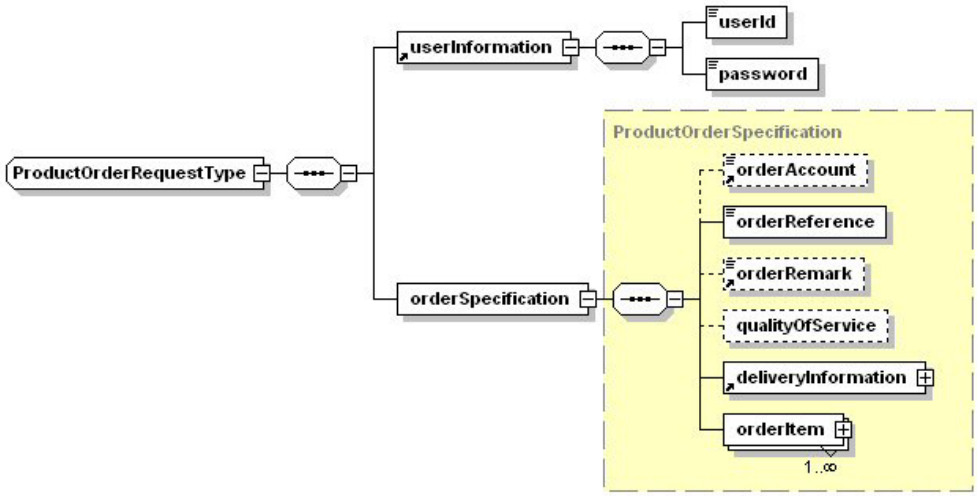

Figure 8-5: ProductOrderRequest Element Diagram.

Tag Name	Tag Description
ProductOrdeRequest	Order Submit Request element root. It contains the information to submit an order.
userInformation	See Table 7-3
userId	See Table 7-3

Tag Name	Tag Description
password	See Table 7-3
orderSpecification	Order Specification element.
orderAccount	Account under which the user is authorised to order from the specific provider. **Type:** Not empty string (max 20 chars)
orderReference	User defined name assigned to that order. **Type:** Not empty string (max 30 chars)
orderRemark	Textual remark on the order. **Type:** Not empty string (max 255 chars)
qualityOfService	See Table 7-1
deliveryInformation	Delivery Information element. Its type is described in paragraph 0
orderItem	Order Item element. Its type is described in paragraph 8.2.5.1.1

Table 8-4: ProductOrderRequest element.

8.2.5.1.1 ProductOrderItem Type

Figure 8-6 gives a graphical representation of the ProductOrderItem Type included into the ProductOrderRequest ("POR", see chapt. 8.2.5) Type.

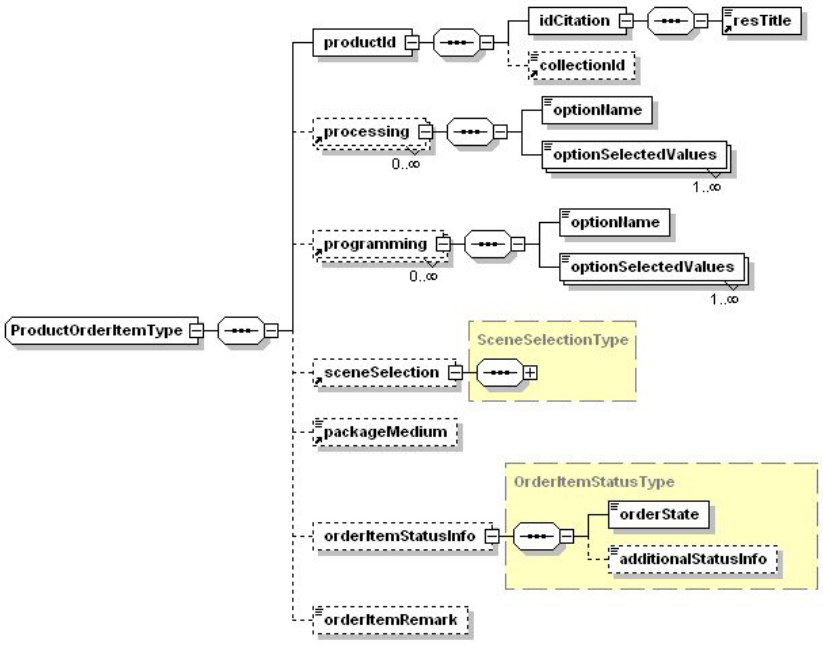

Figure 8-6: ProductOrderItem diagram.

Tag Name	Tag Description
ProductId	It identifies the target product on which the order item is based.
idCitation	Identification Information
resTitle	Product identifier element. Type Value: string Permitted Values: Not empty string Source: **[ISO]**, Citation and responsible party information This Product Identifier will be follow the convention hereafter: ➢ For available products: <2 chars SatelliteId><max 3 char SatelliteNumber>-<StartYYMMDDhhmmsscc>-<DurationDDDDhhmmsscc without leading zeros>.<max 3 chars ProductCodeId> ➢ For planned/possible future products: <2 chars SatelliteId><max 3 char SatelliteNumber>-<6digitOrbit>-<max8digitstartMillisecs>-<max8digitstopMillisecs>[-<swathId>].<max 3 chars ProductCodeId> The start and stop milliseconds are relative to the time of the last ascending node crossing of the satellite (which also defines the orbit number) Examples: pass based products: **NO15-0305111200000-100000.A** for a 10 minute NOAA pass or **ER2-03051112000000-3500000000.G13** or for an "ideal" 35 day GOME level 13 For future Envisat products: **EN1-012345-2152-25632.SEP**
collectionId	Search space for the product requested. **Type:** Not empty string **Syntax:** Provider.Facility.Collection identifying a pre-defined TargetService (e.g. ESA.EECF.ERSE_SER for the ESA ERS SAR raw data catalogue).
Processing	It specifies the processing options to be applied on the product before delivery.
optionName	Name of a processing option specific to the selected product. **Type:** Not empty string (max 40 chars)
optionSelectedValues	Values chosen for the processing option in question. **Type:** Not empty string
programming	It specifies the programming options to be applied on the product before delivery.
optionName	Name of a programming option specific to the selected product. **Type:** Not empty string (max 40 chars)
optionSelectedValues	Values chosen for the programming option in question. **Type:** Not empty string
sceneSelection	It specifies the selection of the scene from the product that is to be delivered. Its type is described in paragraph 0.
packageMedium	Identification of a delivery Medium and a Medium Formatting Option. **Type:** Not empty string (max 40 chars) **Examples:** NTP, DAT, Exabyte, CD-ROM, DLT, D1, DVD, file.
orderItemRemark	Textual remark on the order item put by the XML issuer. **Type:** Not empty string

Table 8-5: ProductOrderItem Type.

8.2.5.1.2 DeliveryInformation Type

Figure 8-7 gives a graphical representation of DeliveryMethod Type:

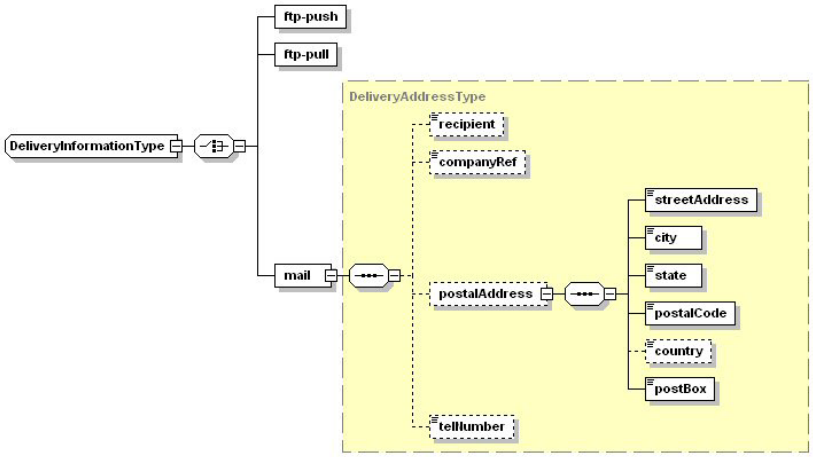

Figure 8-7: DeliveryMethod Type Diagram.

Tag Name	Tag Description
ftp-push	FTP URL: address of a user-owned FTP server to which a product can be posted containing also directory, username, password information. **Type:** Not empty string (max 255 chars) **Syntax:** **ftp://**'ftpUserName':'ftpPassword'@'ftpAddress'/'ftpDirectory' **Example: ftp://**muis_intecs:intecs@ftp.intecs.it/MUIS
ftp-pull	FTP URL: address of a provider-owned FTP server from which user can fetch products containing also directory, username, password information. The value is set by the provider, therefore the element has to be set to <blank> in the OrderSubmitRequest **Type:** string (max 255 chars) **Syntax:** **ftp://**'ftpUserName':'ftpPassword'@'ftpAddress'/'ftpDirectory' **Example: ftp://**userOder:userpwd@ftp.esa.int/XI/EN1
mail	Mail element.
recipient	Identification of the receiving entity. **Type:** Not empty string (max 40 chars)
companyRef	Identification of the receiving person.
postalAddress	Postal Address of the user.
streetAddress	Street Adress element. **Type:** String
city	City element. **Type:** String
state	State element. **Type:** String
postalCode	Postal Code element. **Type:** String

Tag Name	Tag Description
country	Country element. **Type:** String
postalBox	Postal Box element. **Type:** String
telNumber	Telephone number of the receiving person. **Type:** Not empty string (max 18 chars)

Table 8-6: DeliveryMethod Type.

8.2.5.1.3 SceneSelection Type

Figure 8-8 gives a graphical representation of the SceneSelection Type:

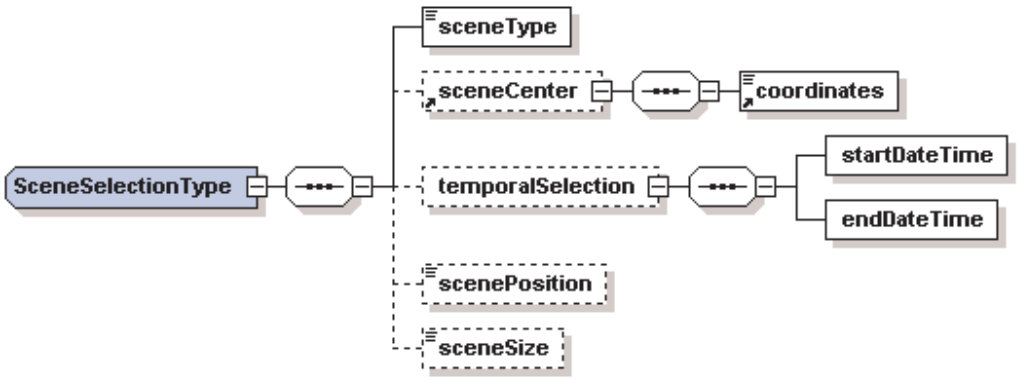

Figure 8-8: SceneSelectionOptions Type diagram

Tag Name	Tag Description
sceneType	Identifier for the scene type specific to the selected product. **Type:** Not empty string (max 20 chars)
sceneCenter	Scene center element.
coordinates	Coordinates element. **Type:** String **Syntax:** points are separated by blank; point's coordinates lat and long are separated by comma (,). The decimal point (.) is used to separate decimals. **Example:** 64.00,-9.00
temporalSelection	Temporal selection element
startDateTime	Start time for the temporal selection in the following format: CCYY-MM-DDThh:mm:ss.ccZ **Type:** DateTime
endDateTime	Stop time for the temporal selection in the following format CCYY-MM-DDThh:mm:ss.ccZ **Type:** DateTime

Tag Name	Tag Description
scenePosition	Provider specific system to define the position of the scene. Examples: “frame 1234” **Type:** Not empty string
sceneSize	Provider specific system to define the size of the scene. Examples: “frames: 3”, “40x50 km”, “26 seconds” **Type:** Not empty string

Table 8-7: SceneSelectionOptions Type.

8.2.6 Order response message

The body of this message is the “orderResponse” XML element.

Figure 8-9 gives a graphical representation of OrderSubmitResponse Type:

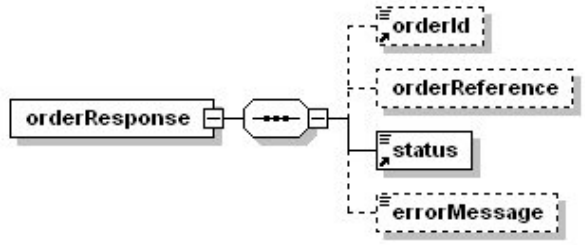

Figure 8-9: OrderResponse element diagram.

Tag Name	Tag Description
orderResponse	OrderResponse element.
orderId	Order identification number unique for this Provider. Mandatory if status=”success”. **Type:** Not empty string (max 16 chars)
orderReference	See Table 8-4
Status	Status of the order submit request. **Type:** String **Permitted Values:** success, failure, partial
errorMessage	Error Message element. **Type:** Not empty string.

Table 8-8: OrderResponse Element.

8.2.7 Order monitor request message

The body of this message is the “orderMonitorRequest” XML element.

Figure 8-10 gives a graphical representation of OrderMonitorRequest element:

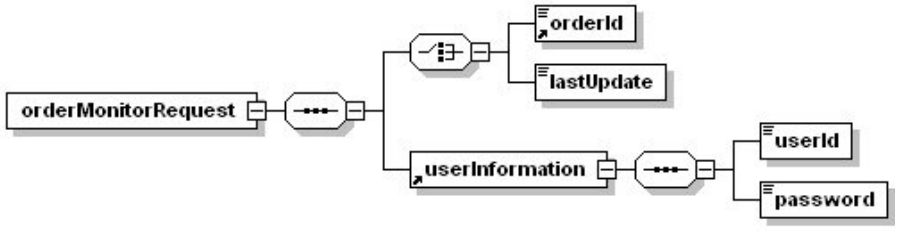

Figure 8-10: OrderMonitorRequest element diagram.

Tag Name	Tag Description
orderMonitorRequest	OrderMonitorRequest element.
orderId	See Table 8-8
lastUpdate	Last update of the order. Type: Type: date in ISO 8601 format (CCYY-MM-DD)
userInformation	See Table 7-3
userId	See Table 7-3
password	See Table 7-3

Table 8-9: OrderMonitorRequest element.

8.2.8 Order monitor response message

The body of this message is the "orderMonitorResponse" XML element for submitted orders.

Figure 8-11 gives a graphical representation of OrderMonitorResponse element:

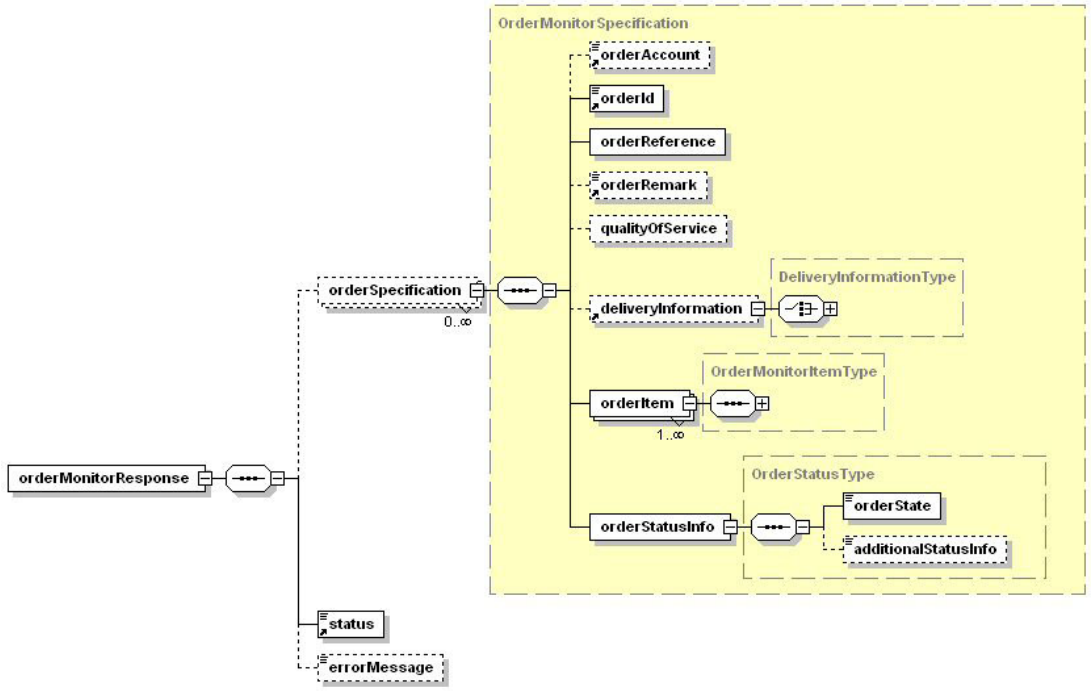

Figure 8-11: OrderMonitorResponse element.

Tag Name	Tag Description
orderMonitorResponse	OrderMonitorResponse element for submitted order.
orderSpecification	Order Specification element. Mandatory if status=success”
orderAccount	See Table 8-4
orderId	See Table 8-8
orderReference	See Table 8-4
orderRemark	See Table 8-4
qualityOfService	See Table 7-1
deliveryInformation	Delivery Information element. Its type is described in paragraph 0
orderItem	See Table 8-4
orderStatusInfo	Contains the Order status information.
orderState	See Table 8-5
additionalStatusInfo	See Table 8-5
status	Status of the order monitor response. **Type:** String **Permitted Values:** success, failure
ErrorMessage	See Table 8-8

Table 8-10: OrderMonitorResponse element.

8.2.8.1.1 OrderMonitorItemType

Figure 8-12 gives a graphical representation of the OrderMonitorItemType:

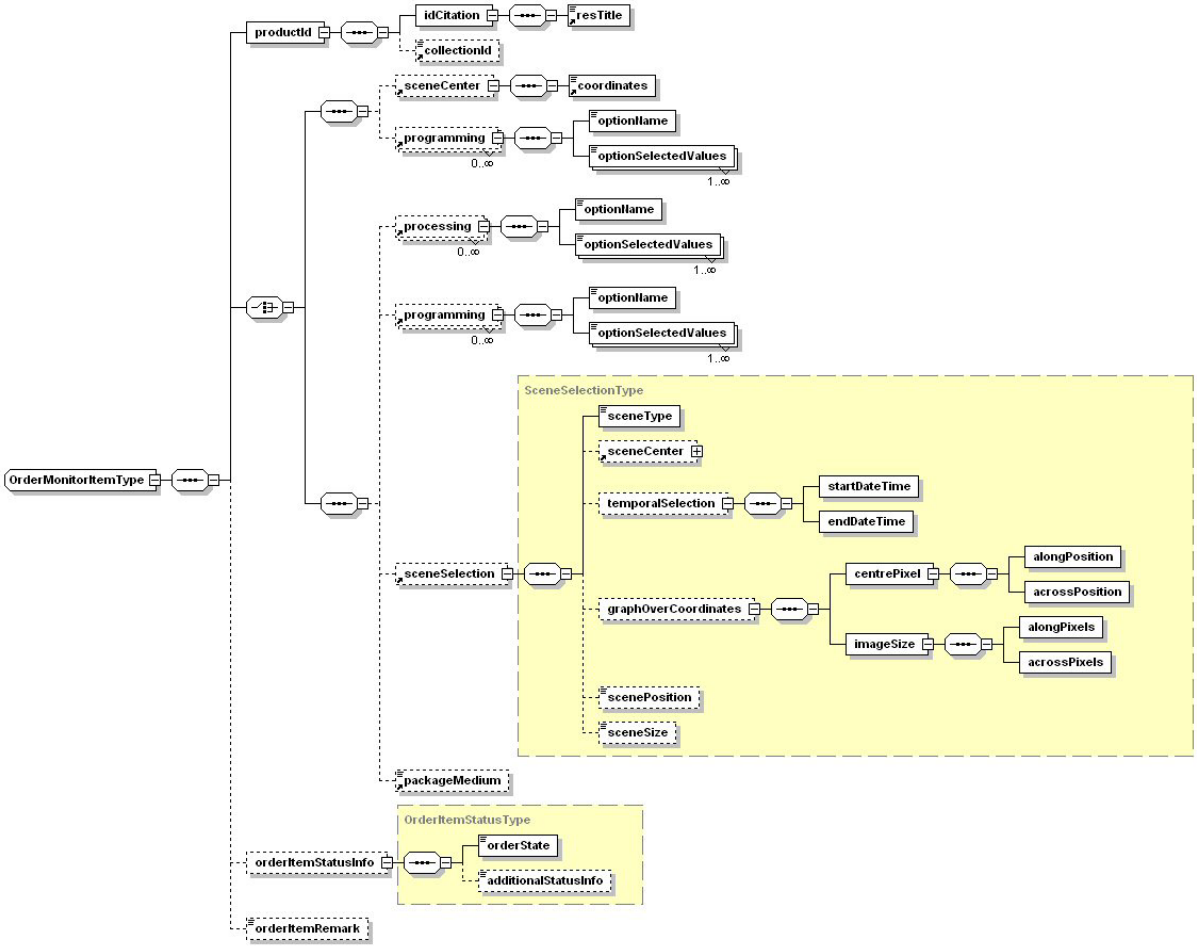

Figure 8-12: OrderMonitorItem diagram.

Tag Name	Tag Description
productId	See Table 8-5
idCitation	See Table 8-5
resTitle	See Table 8-5
collectionId	See Table 8-5
sceneCenter	See Table 8-7
coordinates	See Table 8-7
programming	See Table 8-5
optionName	See Table 8-5
optionSelectedValues	See Table 8-5

Tag Name	Tag Description
processing	See Table 8-5
optionName	See Table 8-5
optionSelectedValues	See Table 8-5
programming	See Table 8-5
optionName	See Table 8-5
optionSelectedValues	See Table 8-5
sceneSelection	It specifies the selection of the scene from the product that is to be delivered. Its type is described in paragraph 0.
packageMedium	See Table 8-5
orderItemStatusInfo	See Table 8-5
orderState	See Table 8-5
additionalStatusInfo	See Table 8-5
orderItemRemark	See Table 8-5

Table 8-11: OrderMonitorItem Type.

9 Standards Extensions

The application profile at hand demands some extensions to the referenced ISO specifications. These extensions and recommendations are described in the following paragraphs.

9.1 Extensions to ISO19115:2003

A few extensions have been defined in the domain to provide metadata that are commonly referenced in EO products, e.g. orbit number, it may be necessary to align/propose these metadata to 19115 part 2.

9.2 Comparison to OGC CS 2.0 base specification

A preliminary analysis shows that the profile shares many common features to the OGC catalogue interface specification CSW 2.0. An alignment towards CSW 2.0 is presented in Annex E. and may provide a suitable framework to include the discovery of product collections. However, the requirements for ordering of data should be addressed at the same time.

9.2.1 Mappings to the common XML Record format

To be provided.

10 Implementation guidance

The following section gives developers help when setting up a catalogue service instance that complies with the defined application profile. Any information provided here is non-normative.

10.1.1 Technical issues

-

10.1.2 Semantic issues

-

10.1.3 Examples

The following sections show example of the requests / responses

10.1.3.1 SearchRequest message

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/">
   <SOAP:Body>
      <searchRequest xmlns="http://earth.esa.int/XML/eoli">
         <simpleQuery>
            <dataExt>
               <geoEle operator="OVERLAP">
                  <geoBndBox>
                     <westBL>180.0</westBL>
                     <eastBL>180.0</eastBL>
                     <southBL>90.0</southBL>
                     <northBL>90.0</northBL>
                  </geoBndBox>
               </geoEle>
               <tempEle operator="OVERLAP">
                  <exTemp>
                     <beginEnd>
                        <begin>2000-08-13</begin>
                        <end>2000-08-13</end>
                     </beginEnd>
                  </exTemp>
               </tempEle>
            </dataExt>
            <satelliteDomainConditions>
               <plaInsIdCondition operator="EQUAL">
                  <plaInsId>
                     <platfSNm>ERS</platfSNm>
                     <platfSer>2</platfSer>
                  </plaInsId>
               </plaInsIdCondition>
               <genericCondition operator="EQUAL">
                  <attributeId>track</attributeId>
                  <attributeValue>1</attributeValue>
               </genericCondition>
               <genericCondition operator="EQUAL">
                  <attributeId>frame</attributeId>
                  <attributeValue >27</attributeValue>
               </genericCondition>
               <genericCondition operator="WITHIN">
                  <attributeId>orbit</attributeId>
                  <attributeValue >1000</attributeValue>
                  <attributeValue>1300</attributeValue>
               </genericCondition >
            </satelliteDomainConditions>
         </simpleQuery>
         <resultType>results</resultType>
         <iteratorSize>1000</iteratorSize>
         <cursor>1</cursor>
         <presentation>brief</presentation>
         <collectionId>ESA.EECF.ERSE_SER</collectionId>
      </searchRequest>
   </SOAP:Body>
</SOAP:Envelope>
```

Listing 1 – searchRequest

10.1.3.2 PresentRequest message

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/">
   <SOAP:Body>
```



```
      <presentRequest xmlns="http://earth.esa.int/XML/eoli">
         <idCitation>
            <resTitle>ER2-00081308530342-12858.SE</resTitle>
            <resTitle>ER2-00081310283992-61308.SE</resTitle>
         </idCitation>
         <presentation>brief</presentation>
         <collectionId>ESA.EECF.ERSE_SER</collectionId>
      </presentRequest>
   </SOAP:Body>
</SOAP:Envelope>
```

Listing 2 – presentRequest

10.1.3.3 Response message (full format)

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/">
   <SOAP:Body>
      <response xmlns="http://earth.esa.int/XML/eoli">
         <retrievedData presentation="full">
            <Metadata>
               <mdContact>
                  <rpOrgName>ESA</rpOrgName>
                  <role>002</role>
               </mdContact>
               <mdDateSt>2000-01-01T10:00:00Z</mdDateSt>
               <dataIdInfo>
                  <plaInsId>
                     <platfSNm>ENVISAT</platfSNm>
                     <instShNm>ASAR/IM</instShNm>
                     <instMode>high</instMode>
                  </plaInsId>
                  <satDom>
                     <orbit>1234</orbit>
                     <lastOrbit>1</lastOrbit>
                     <orbitDir>1</orbitDir>
                     <wwRefSys>
                        <frame>27</frame>
                        <track>1</track>
                     </wwRefSys>
                     <swathId>1</swathId>
                     <passCoverage>
                        <start>2152</start>
                        <stop>25632</stop>
                     </ passCoverage >
                  </satDom>
                  <idCitation>
                     <resTitle>EN1-012345-2152-25632.XI</resTitle>
                  </idCitation>
                  <idAbs>ENVISAT ASAR/IM product</idAbs>
                  <idStatus>001</idStatus>
                  <dataExt>
                     <tempEle>
                        <exTemp>
                           <beginEnd>
                              <begin>2000-01-01T00:00:00.00Z</begin>
                              <end>2000-01-01T00:00:15.10Z</end>
                           </beginEnd>
                        </exTemp>
                     </tempEle>
                     <geoEle>
                        <polygon>
                           <coordinates>65.00,-10.00 65.00,-8.00 63.00,-8.00 63.00,-
10.00</coordinates>
                        </polygon>
                        <scCenter>
                           <coordinates>64.00,-9.00</coordinates>
                        </scCenter>
                     </geoEle>
```

```
                    </dataExt>
                </dataIdInfo>
                <contInfo>
                    <attDesc>
                        <typeName>imageType</typeName>
                        <attTypes>
                            <attName>imageQuality</attName>
                            <typeName>imageQualityType</typeName>
                        </attTypes>
                    </attDesc>
                    <contType>001</contType>
                    <illElevAng>90</illElevAng>
                    <illAziAng>50</illAziAng>
                    <cloudCovePerc>1</cloudCovePerc>
                </contInfo>
                <dqInfo>
                    <dqScope>
                        <scpLvl>DataSet</scpLvl>
                    </dqScope>
                </dqInfo>
                <addInfo>
                    <locAtt>
                        <locName>Polarisation</locName>
                        <locValue>VV</locValue>
                    </locAtt>
                </addInfo>
            </Metadata>
        </retrievedData>
        <cursor>1</cursor>
        <hits>1</hits>
        <status>success</status>
    </response>
  </SOAP:Body>
</SOAP:Envelope>
```

Listing 3 – response

11 Security considerations

This document does not demand any specific security considerations regarding a compliant catalogue service.

However, for the order interfaces, it has to be acknowledged that use of user/password credentials passed over HTTP are not a secure solution, in practice ordering of EO data products within the context of ESA's science community is regulated further by quota allocation and so this is not a major concern, a longer term solution is expected to be based on WS security standards.

Annex A
(normative)

Abstract test suite for catalogue services

A.1

A.1.1 Basic Client

A.1.1.3 searchRequest
a) Test purpose: Verify that a client satisfies all requirements for a searchRequest.
b) Test method: Generate an adequate sample of searchRequest from the client and verify that each is a valid request.
c) Reference:
d) Test type: Basic

A.1.1.4 presentRequest
a) Test purpose: Verify that a client satisfies all requirements for a presentRequest.
b) Test method: Generate an adequate sample of presentRequest from the client and verify that each is a valid request.
c) Reference:
d) Test type: Basic

A.1.2 Basic Server

A.1.2.1 Version negotiation <<NOT YET ADDRESSED IN THIS PROFILE>>
a) Test Purpose: Verify that the server interface satisfies the requirements for version negotiation.
b) Test Method: Submit requests containing version number both lower than and higher than the version supported by the server. Verify that the server responses is in accord with the rules for version negotiation.
c) Reference: CSW 2.0 base specification
d) Test Type: Basic

A.1.2.2 Request parameter rules <<NOT YET ADDRESSED IN THIS PROFILE>>
a) Test Purpose: Verify that the server interface satisfies the requirements for request parameter rules.
b) Test Method: Generate a sample of requests from a client. Include both invalid requests and valid request that vary within the limits allowed by the rules. Verify that the server provides an appropriate response in each case.
c) Reference: CSW 2.0 base specification
d) Test Type: Basic

A.1.2.3 GetCapabilities response <<NOT YET ADDRESSED IN THIS PROFILE>>
a) Test Purpose: Verify that a basic CSW server interface satisfies all requirements of the GetCapabilities operation.
b) Test Method: Make several GetCapabilities requests using a variety of input parameters. Verify that an appropriate response is returned in each case.
c) Reference:
d) Test Type: Basic

A.1.2.3 searchRequest response
a) Test Purpose: Verify that the server satisfies all requirements of the searchRequest operation.
b) Test Method: Make several searchRequest requests using a variety of input parameters. Verify that an appropriate response is returned in each case.
c) Reference:
d) Test Type: Basic

A.1.2.3 presentRequest response
a) Test Purpose: Verify that the server satisfies all requirements of the presentRequest operation.
b) Test Method: Make several GetRecordsById requests using a variety of input parameters. Verify that an appropriate response is returned in each case.
c) Reference:
d) Test Type: Basic

Conformance to the catalogue search and present interfaces may be tested via the Service Support Environment in the ESA's Eoportal [SSE]

Annex B
(normative)

Abstract test suite for ordering services (to be completed)

B.1

Annex B
(informative)

Design rationale (to be completed)

Annex C
(normative)

XSD & WSDL Specification

Catalogue XSD

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://earth.esa.int/XML/eoli"
targetNamespace="http://earth.esa.int/XML/eoli" elementFormDefault="qualified">
    <!-- ================== Element Section ==================== -->
    <!-- ======================================================== -->
    <xsd:element name="searchRequest" type="SearchRequestType"/>
    <xsd:element name="beginEnd">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="begin" type="xsd:dateTime"/>
                <xsd:element name="end" type="xsd:dateTime"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="collectionId" type="NonNullStringType"/>
    <xsd:element name="cursor" type="xsd:positiveInteger" nillable="false"/>
    <xsd:element name="presentation" type="PresentationType" nillable="false"/>
    <xsd:element name="status" type="StatusType" nillable="false"/>
    <xsd:element name="frame" type="xsd:positiveInteger" nillable="false"/>
    <xsd:element name="track" type="xsd:positiveInteger" nillable="false"/>
    <xsd:element name="orbit" type="xsd:positiveInteger" nillable="false"/>
    <xsd:element name="presentRequest" type="PresentRequestType"/>
    <xsd:element name="platfSNm" type="NonNullStringType"/>
    <xsd:element name="platfSer" type="CharacterString"/>
    <xsd:element name="instShNm" type="NonNullStringType"/>
    <xsd:element name="resTitle" type="NonNullStringType"/>
    <xsd:element name="response" type="ResponseType"/>
    <xsd:element name="coordinates" type="CoordinatesType" nillable="false"/>
    <xsd:element name="idStatus" type="IdStatusEnumType"/>
    <!-- ================== Simple Type Section ================== -->
    <!-- ======================================================== -->
    <xsd:simpleType name="DateTimeTypes">
        <xsd:union memberTypes="xsd:date xsd:dateTime"/>
    </xsd:simpleType>
    <xsd:simpleType name="NonNullStringType">
        <xsd:restriction base="CharacterString">
            <xsd:minLength value="1"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="CharacterString">
        <xsd:restriction base="xsd:string">
            <xsd:minLength value="0" fixed="false"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="LongitudeType">
        <xsd:restriction base="xsd:float">
            <xsd:minInclusive value="-180.0"/>
            <xsd:maxInclusive value="180.0"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="LatitudeType">
        <xsd:restriction base="xsd:float">
            <xsd:minInclusive value="-90.0"/>
            <xsd:maxInclusive value="90.0"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="cloudCoveragePercType">
        <xsd:restriction base="xsd:float">
            <xsd:minInclusive value="0.0"/>
            <xsd:maxInclusive value="100.0"/>
```

```
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="PresentationType">
        <xsd:restriction base="CharacterString">
            <xsd:enumeration value="brief"/>
            <xsd:enumeration value="summary"/>
            <xsd:enumeration value="full"/>
            <xsd:enumeration value="browse"/>
            <xsd:enumeration value="orderOption"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="StatusType">
        <xsd:restriction base="CharacterString">
            <xsd:enumeration value="success"/>
            <xsd:enumeration value="failure"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="RoleEnumType">
        <xsd:restriction base="NonNullStringType">
            <xsd:enumeration value="002"/>
            <xsd:enumeration value="006"/>
            <xsd:enumeration value="009"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="IdStatusEnumType">
        <xsd:restriction base="NonNullStringType">
            <xsd:enumeration value="001"/>
            <xsd:enumeration value="005"/>
            <xsd:enumeration value="006"/>
            <xsd:enumeration value="007"/>
            <xsd:enumeration value="008"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="DegreeType">
        <xsd:restriction base="xsd:float">
            <xsd:minInclusive value="0.0"/>
            <xsd:maxInclusive value="360.0"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="ContentTypeEnum">
        <xsd:restriction base="NonNullStringType">
            <xsd:enumeration value="001"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="OrbitDirectionEnumType">
        <xsd:restriction base="CharacterString">
            <xsd:enumeration value="0"/>
            <xsd:enumeration value="1"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="PolarisationEnumType">
        <xsd:restriction base="NonNullStringType">
            <xsd:enumeration value="HH"/>
            <xsd:enumeration value="VV"/>
            <xsd:enumeration value="HV"/>
            <xsd:enumeration value="VH"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="FullSceneEnumType">
        <xsd:restriction base="NonNullStringType">
            <xsd:enumeration value="F"/>
            <xsd:enumeration value="P"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="CoordinatesType">
        <xsd:restriction base="NonNullStringType"/>
    </xsd:simpleType>
    <!-- ================== Complex Type Section ================== -->
    <!-- ========================================================== -->
    <xsd:complexType name="SearchRequestType">
        <xsd:sequence>
```



```
            <xsd:element name="simpleQuery" type="SimpleQueryType"/>
            <xsd:element name="resultType" nillable="false">
                <xsd:simpleType>
                    <xsd:restriction base="NonNullStringType">
                        <xsd:enumeration value="results"/>
                        <xsd:enumeration value="hits"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:element>
            <xsd:element name="iteratorSize" type="xsd:positiveInteger" nillable="false" minOccurs="0"/>
            <xsd:element ref="cursor" minOccurs="0"/>
            <xsd:element ref="presentation" minOccurs="0"/>
            <xsd:element ref="collectionId"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="SimpleQueryType">
        <xsd:sequence>
            <xsd:element name="dataExt" minOccurs="0">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:restriction base="DataExtType">
                            <xsd:sequence>
                                <xsd:element name="geoEle" minOccurs="0">
                                    <xsd:complexType>
                                        <xsd:choice>
                                            <xsd:element name="geoBndBox">
                                              <xsd:complexType>
                                                <xsd:complexContent>
                                                  <xsd:extension base="EX_GeographicBoundingBoxType"/>
                                                </xsd:complexContent>
                                              </xsd:complexType>
                                            </xsd:element>
                                        </xsd:choice>
                                        <xsd:attribute name="operator">
                                            <xsd:simpleType>
                                              <xsd:restriction base="CharacterString">
                                                <xsd:enumeration value="OVERLAP"/>
                                              </xsd:restriction>
                                            </xsd:simpleType>
                                        </xsd:attribute>
                                    </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="tempEle" minOccurs="0">
                                    <xsd:complexType>
                                        <xsd:sequence>
                                            <xsd:element name="exTemp">
                                              <xsd:complexType>
                                                <xsd:choice>
                                                  <xsd:element name="beginEnd">
                                                    <xsd:complexType>
                                                      <xsd:sequence>
                                                        <xsd:element name="begin" type="xsd:date" nillable="false"/>
                                                        <xsd:element name="end" type="xsd:date" nillable="false"/>
                                                      </xsd:sequence>
                                                    </xsd:complexType>
                                                  </xsd:element>
                                                </xsd:choice>
                                              </xsd:complexType>
                                            </xsd:element>
                                        </xsd:sequence>
                                        <xsd:attribute name="operator">
                                            <xsd:simpleType>
                                              <xsd:restriction base="CharacterString">
                                                <xsd:enumeration value="OVERLAP"/>
                                              </xsd:restriction>
                                            </xsd:simpleType>
                                        </xsd:attribute>
                                    </xsd:complexType>
                                </xsd:element>
                            </xsd:sequence>
                        </xsd:restriction>
                    </xsd:complexContent>
                </xsd:complexType>
```

```
            </xsd:element>
            <xsd:element name="satelliteDomainConditions" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="plaInsIdCondition" minOccurs="0">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="plaInsId">
                                        <xsd:complexType>
                                            <xsd:sequence>
                                              <xsd:element ref="platfSNm"/>
                                              <xsd:element ref="platfSer" minOccurs="0"/>
                                            </xsd:sequence>
                                        </xsd:complexType>
                                    </xsd:element>
                                </xsd:sequence>
                                <xsd:attribute name="operator">
                                    <xsd:simpleType>
                                        <xsd:restriction base="CharacterString">
                                            <xsd:enumeration value="EQUAL"/>
                                        </xsd:restriction>
                                    </xsd:simpleType>
                                </xsd:attribute>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="cloudCoverCondition" minOccurs="0">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="cloudCovePerc" type="cloudCoveragePercType" nillable="false"/>
                                </xsd:sequence>
                                <xsd:attribute name="operator">
                                    <xsd:simpleType>
                                        <xsd:restriction base="CharacterString">
                                            <xsd:enumeration value="LESS EQUAL"/>
                                        </xsd:restriction>
                                    </xsd:simpleType>
                                </xsd:attribute>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="genericCondition" minOccurs="0" maxOccurs="unbounded">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="attributeId" type="NonNullStringType" nillable="false"/>
                                    <xsd:element name="attributeValue" type="NonNullStringType" nillable="false"
maxOccurs="unbounded"/>
                                </xsd:sequence>
                                <xsd:attribute name="operator">
                                    <xsd:simpleType>
                                        <xsd:restriction base="CharacterString">
                                            <xsd:enumeration value="EQUAL"/>
                                            <xsd:enumeration value="LESS EQUAL"/>
                                            <xsd:enumeration value="LESS"/>
                                            <xsd:enumeration value="GREATER"/>
                                            <xsd:enumeration value="GREATER EQUAL"/>
                                            <xsd:enumeration value="WITHIN"/>
                                            <xsd:enumeration value="INTERSECTS"/>
                                        </xsd:restriction>
                                    </xsd:simpleType>
                                </xsd:attribute>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="DataExtType">
        <xsd:sequence>
            <xsd:element name="geoEle" minOccurs="0">
                <xsd:complexType>
                    <xsd:choice>
```



```
                        <xsd:sequence>
                            <xsd:element name="polygon">
                                <xsd:complexType>
                                    <xsd:sequence>
                                        <xsd:element ref="coordinates"/>
                                    </xsd:sequence>
                                </xsd:complexType>
                            </xsd:element>
                            <xsd:element name="scCenter" minOccurs="0">
                                <xsd:complexType>
                                    <xsd:sequence>
                                        <xsd:element ref="coordinates"/>
                                    </xsd:sequence>
                                </xsd:complexType>
                            </xsd:element>
                        </xsd:sequence>
                        <xsd:element name="geoBndBox">
                            <xsd:complexType>
                                <xsd:complexContent>
                                    <xsd:extension base="EX_GeographicBoundingBoxType"/>
                                </xsd:complexContent>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:choice>
                    <xsd:attribute name="operator">
                        <xsd:simpleType>
                            <xsd:restriction base="CharacterString">
                                <xsd:enumeration value="OVERLAP"/>
                            </xsd:restriction>
                        </xsd:simpleType>
                    </xsd:attribute>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="tempEle" type="EX_TemporalExtendType" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="EX_GeographicBoundingBoxType">
        <xsd:sequence>
            <xsd:element name="westBL" type="LongitudeType" nillable="false"/>
            <xsd:element name="eastBL" type="LongitudeType" nillable="false"/>
            <xsd:element name="southBL" type="LatitudeType" nillable="false"/>
            <xsd:element name="northBL" type="LatitudeType" nillable="false"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="EX_TemporalExtendType">
        <xsd:sequence>
            <xsd:element name="exTemp">
                <xsd:complexType>
                    <xsd:choice>
                        <xsd:element name="beginEnd">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="begin" type="xsd:dateTime" nillable="false"/>
                                    <xsd:element name="end" type="xsd:dateTime" nillable="false"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:choice>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="PresentRequestType">
        <xsd:sequence>
            <xsd:element name="idCitation">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element ref="resTitle" maxOccurs="unbounded"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element ref="presentation"/>
            <xsd:element ref="collectionId"/>
```

```
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="MetadataType">
        <xsd:sequence>
            <xsd:element name="mdContact">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="rpOrgName" type="CharacterString"/>
                        <xsd:element name="role" type="RoleEnumType"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="mdDateSt" type="DateTimeTypes" nillable="false" minOccurs="0"/>
            <xsd:element name="dataIdInfo" type="DataIdInfoType"/>
            <xsd:element name="contInfo" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="attDesc">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="typeName" fixed="imageType" nillable="false">
                                        <xsd:simpleType>
                                            <xsd:restriction base="CharacterString">
                                                <xsd:enumeration value="imageType"/>
                                            </xsd:restriction>
                                        </xsd:simpleType>
                                    </xsd:element>
                                    <xsd:element name="attTypes">
                                        <xsd:complexType>
                                            <xsd:sequence>
                                                <xsd:element name="attName" fixed="imageQuality" nillable="false">
                                                    <xsd:simpleType>
                                                        <xsd:restriction base="CharacterString">
                                                            <xsd:enumeration value="imageQuality"/>
                                                        </xsd:restriction>
                                                    </xsd:simpleType>
                                                </xsd:element>
                                                <xsd:element name="typeName" fixed="imageQualityType" nillable="false">
                                                    <xsd:simpleType>
                                                        <xsd:restriction base="CharacterString">
                                                            <xsd:enumeration value="imageQualityType"/>
                                                        </xsd:restriction>
                                                    </xsd:simpleType>
                                                </xsd:element>
                                            </xsd:sequence>
                                        </xsd:complexType>
                                    </xsd:element>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="contType" type="ContentTypeEnum" nillable="false"/>
                        <xsd:element name="illElevAng" type="LatitudeType" nillable="false" minOccurs="0"/>
                        <xsd:element name="illAziAng" type="DegreeType" nillable="false" minOccurs="0"/>
                        <xsd:element name="cloudCovePerc" type="cloudCoveragePercType" nillable="false" minOccurs="0"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="dqInfo" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="dqScope" minOccurs="0">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="scpLvl" fixed="DataSet">
                                        <xsd:simpleType>
                                            <xsd:restriction base="NonNullStringType">
                                                <xsd:enumeration value="DataSet"/>
                                            </xsd:restriction>
                                        </xsd:simpleType>
                                    </xsd:element>
                                </xsd:sequence>
```



```
                        </xsd:complexType>
                    </xsd:element>
                    <xsd:element name="graphOver" minOccurs="0" maxOccurs="unbounded">
                        <xsd:complexType>
                            <xsd:sequence>
                                <xsd:element name="bgFileName" type="NonNullStringType"/>
                                <xsd:element name="brwExt">
                                    <xsd:complexType>
                                        <xsd:sequence>
                                            <xsd:element name="tempEle" type="EX_TemporalExtendType"
minOccurs="0"/>
                                        </xsd:sequence>
                                    </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="brwType" type="NonNullStringType"/>
                            </xsd:sequence>
                        </xsd:complexType>
                    </xsd:element>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="addInfo" minOccurs="0">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="locAtt" minOccurs="0" maxOccurs="unbounded">
                        <xsd:complexType>
                            <xsd:sequence>
                                <xsd:element name="locName" type="NonNullStringType"/>
                                <xsd:element name="locValue" type="NonNullStringType" minOccurs="0"/>
                            </xsd:sequence>
                        </xsd:complexType>
                    </xsd:element>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DataIdInfoType">
    <xsd:sequence>
        <xsd:element name="plaInsId">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element ref="platfSNm"/>
                    <xsd:element ref="platfSer" minOccurs="0"/>
                    <xsd:element ref="instShNm"/>
                    <xsd:element name="instMode" type="CharacterString" minOccurs="0"/>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="satDom" minOccurs="0">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element ref="orbit" minOccurs="0"/>
                    <xsd:element name="lastOrbit" type="xsd:positiveInteger" nillable="false" minOccurs="0"/>
                    <xsd:element name="orbitDir" type="OrbitDirectionEnumType" nillable="false" minOccurs="0"/>
                    <xsd:element name="wwRefSys" minOccurs="0">
                        <xsd:complexType>
                            <xsd:sequence>
                                <xsd:element ref="frame" minOccurs="0"/>
                                <xsd:element ref="track" minOccurs="0"/>
                            </xsd:sequence>
                        </xsd:complexType>
                    </xsd:element>
                    <xsd:element name="swathId" type="NonNullStringType" minOccurs="0"/>
                    <xsd:element name="passCoverage" minOccurs="0">
                        <xsd:complexType>
                            <xsd:sequence>
                                <xsd:element name="start" type="xsd:integer"/>
                                <xsd:element name="stop" type="xsd:integer"/>
                            </xsd:sequence>
                        </xsd:complexType>
                    </xsd:element>
                </xsd:sequence>
```

```
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="idCitation">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="resTitle" type="NonNullStringType"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="idAbs" type="CharacterString" minOccurs="0"/>
            <xsd:element ref="idStatus" minOccurs="0"/>
            <xsd:element name="dataExt">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:restriction base="DataExtType">
                            <xsd:sequence>
                                <xsd:element name="tempEle" type="EX_TemporalExtendType" minOccurs="0"/>
                                <xsd:element name="geoEle" minOccurs="0">
                                    <xsd:complexType>
                                        <xsd:choice>
                                            <xsd:sequence>
                                              <xsd:element name="polygon">
                                                <xsd:complexType>
                                                  <xsd:sequence>
                                                    <xsd:element ref="coordinates"/>
                                                  </xsd:sequence>
                                                </xsd:complexType>
                                              </xsd:element>
                                              <xsd:element name="scCenter" minOccurs="0">
                                                <xsd:complexType>
                                                  <xsd:sequence>
                                                    <xsd:element ref="coordinates"/>
                                                  </xsd:sequence>
                                                </xsd:complexType>
                                              </xsd:element>
                                            </xsd:sequence>
                                        </xsd:choice>
                                    </xsd:complexType>
                                </xsd:element>
                            </xsd:sequence>
                        </xsd:restriction>
                    </xsd:complexContent>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="ResponseType">
        <xsd:sequence>
            <xsd:element name="retrievedData" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="Metadata" type="MetadataType" minOccurs="0" maxOccurs="unbounded"/>
                    </xsd:sequence>
                    <xsd:attribute name="presentation" type="PresentationType"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:element ref="cursor" minOccurs="0"/>
            <xsd:element name="hits" type="xsd:nonNegativeInteger" nillable="false" minOccurs="0"/>
            <xsd:element ref="status"/>
            <xsd:element name="errorMessage" type="NonNullStringType" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:schema>
```

Catalogue WSDL

```
?xml version="1.0" encoding="UTF-8"?>
```



```
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://earth.esa.int/XML/dsm" xmlns:eoli="http://earth.esa.int/XML/eoli"
targetNamespace="http://earth.esa.int/XML/dsm">
    <import namespace="http://earth.esa.int/XML/eoli" location="eoli.xsd"/>
    <message name="searchRequestInput">
        <part name="searchRequestParameter" element="eoli:searchRequest"/>
    </message>
    <message name="presentRequestInput">
        <part name="searchRequestParameter" element="eoli:presentRequest"/>
    </message>
    <message name="responseOutput">
        <part name="responseParameter" element="eoli:response"/>
    </input message="tns:searchRequestInput"/>
            <output message="tns:responseOutput"/>
        </operation>
        <operation name="processPresentRequest">
            <input message="tns:presentRequestInput"/>
            <output message="tns:responseOutput"/>
        </operation>
    </portType>
    <binding name="dsmBinding" type="tns:dsmPortType">
        <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
        <operation name="processSearchRequest">
            <soap:operation soapAction="http://earth.esa.int/SOAP/searchRequest"/>
            <input>
                <soap:body use="literal"/>
            </input>
            <output>
                <soap:body use="literal"/>
            </output>
        </operation>
        <operation name="processPresentRequest">
            <soap:operation soapAction="http://earth.esa.int/SOAP/presentRequest"/>
            <input>
                <soap:body use="literal"/>
            </input>
            <output>
                <soap:body use="literal"/>
            </output>
        </operation>
    </binding>
    <service name="dsmService">
        <port name="dsmSoap" binding="tns:dsmBinding">
            <soap:address location="http://earth.esa.int/"/>
        </port>
    </service>
</definitions>
```

Order XSD

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.4 U (http://www.xmlspy.com) by Simone Gianfranceschi (ESA - European Space Agency) -->
<!--                    :_Order.xsd
Type     :C Schema
         :     October 2005
    :Schema
         :
         :Patrizia
         :.3 -->
<xsd:schema xmlns="http://earth.esa.int/XML/eoli" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://earth.esa.int/XML/eoli" elementFormDefault="qualified" attributeFormDefault="unqualified">
    <xsd:redefine schemaLocation="./eoli.xsd">
        <xsd:complexType name="MetadataType">
            <xsd:complexContent>
                <xsd:extension base="MetadataType">
                    <xsd:sequence minOccurs="0">
                        <xsd:element name="productServiceOptions" type="ProductServiceOptionsType" minOccurs="0"/>
                    </xsd:sequence>
                </xsd:extension>
            </xsd:complexContent>
        </xsd:complexType>
        <xsd:complexType name="ResponseType">
```

```
            <xsd:complexContent>
                <xsd:restriction base="ResponseType">
                    <xsd:sequence>
                        <xsd:element name="retrievedData" minOccurs="0">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="Metadata" type="MetadataType" minOccurs="0"
maxOccurs="unbounded"/>
                                </xsd:sequence>
                                <xsd:attribute name="presentation" type="PresentationType"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element ref="cursor" minOccurs="0"/>
                        <xsd:element name="hits" type="xsd:nonNegativeInteger" nillable="false" minOccurs="0"/>
                        <xsd:element ref="status"/>
                        <xsd:element name="errorMessage" type="NonNullStringType" minOccurs="0"/>
                    </xsd:sequence>
                </xsd:restriction>
            </xsd:complexContent>
        </xsd:complexType>
        <xsd:complexType name="SearchRequestType">
            <xsd:complexContent>
                <xsd:extension base="SearchRequestType">
                    <xsd:sequence>
                        <xsd:element ref="userInformation" minOccurs="0"/>
                    </xsd:sequence>
                </xsd:extension>
            </xsd:complexContent>
        </xsd:complexType>
        <xsd:complexType name="PresentRequestType">
            <xsd:complexContent>
                <xsd:extension base="PresentRequestType">
                    <xsd:sequence>
                        <xsd:element ref="userInformation" minOccurs="0"/>
                    </xsd:sequence>
                </xsd:extension>
            </xsd:complexContent>
        </xsd:complexType>
    </xsd:redefine>
    <!-- ========== Functions root element ==========-->
    <!-- ============================================================ -->
    <xsd:element name="productOrderRequest" type="ProductOrderRequestType"/>
    <xsd:element name="orderMonitorRequest">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:choice>
                    <xsd:element ref="orderId"/>
                    <xsd:element name="lastUpdate" type="DateTimeTypes"/>
                </xsd:choice>
                <xsd:element ref="userInformation"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="orderMonitorResponse">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="orderSpecification" type="OrderMonitorSpecification" minOccurs="0"
maxOccurs="unbounded"/>
                <xsd:element name="status" type="ExtendedStatusType"/>
                <xsd:element name="errorMessage" type="NonNullStringType" minOccurs="0"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <!-- ========== Elements RE-Definition Section ========== -->
    <!-- ============================================================ -->
    <!-- ========== Elements Definition Section ========== -->
    <!-- ============================================================ -->
    <xsd:element name="sceneCenter">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="coordinates"/>
```



```
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="programming">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="optionName">
                    <xsd:simpleType>
                        <xsd:restriction base="NonNullStringType">
                            <xsd:maxLength value="40"/>
                        </xsd:restriction>
                    </xsd:simpleType>
                </xsd:element>
                <xsd:element name="optionSelectedValues" type="NonNullStringType" maxOccurs="unbounded"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="userInformation">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="userId" type="NonNullStringType"/>
                <xsd:element name="password" type="NonNullStringType"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="orderAccount" nillable="false">
        <xsd:simpleType>
            <xsd:restriction base="NonNullStringType">
                <xsd:maxLength value="20"/>
            </xsd:restriction>
        </xsd:simpleType>
    </xsd:element>
    <xsd:element name="alongGridUnitType">
        <xsd:simpleType>
            <xsd:restriction base="xsd:string">
                <xsd:enumeration value="A"/>
                <xsd:enumeration value="S"/>
            </xsd:restriction>
        </xsd:simpleType>
    </xsd:element>
    <xsd:element name="alongGrid" type="xsd:integer"/>
    <xsd:element name="acrossGrid" type="xsd:integer"/>
    <xsd:element name="orderId">
        <xsd:simpleType>
            <xsd:restriction base="NonNullStringType">
                <xsd:maxLength value="16"/>
            </xsd:restriction>
        </xsd:simpleType>
    </xsd:element>
    <!--    <xsd:element name="deliveryMedium">        <xsd:simpleType>            <xsd:restriction
base="NonNullStringType">                <xsd:maxLength value="40"/>            </xsd:restriction>        </xsd:simpleType>
    </xsd:element>    -->
    <xsd:element name="packageMedium">
        <xsd:simpleType>
            <xsd:restriction base="NonNullStringType">
                <xsd:maxLength value="40"/>
            </xsd:restriction>
        </xsd:simpleType>
    </xsd:element>
    <xsd:element name="deliveryInformation" type="DeliveryInformationType"/>
    <xsd:element name="price" type="PriceType"/>
    <xsd:element name="listPrice" type="PriceType"/>
    <xsd:element name="expectedPrice" type="PriceType"/>
    <xsd:element name="productId">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="idCitation">
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:element ref="resTitle"/>
                        </xsd:sequence>
                    </xsd:complexType>
                </xsd:element>
```

```
                <xsd:element ref="collectionId"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="orderItemRemark" type="NonNullStringType"/>
    <xsd:element name="processing">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="optionName">
                    <xsd:simpleType>
                        <xsd:restriction base="NonNullStringType">
                            <xsd:maxLength value="40"/>
                        </xsd:restriction>
                    </xsd:simpleType>
                </xsd:element>
                <xsd:element name="optionSelectedValues" type="NonNullStringType" maxOccurs="unbounded"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="sceneSelection" type="SceneSelectionType"/>
    <xsd:element name="orderRemark">
        <xsd:simpleType>
            <xsd:restriction base="xsd:string">
                <xsd:maxLength value="255"/>
            </xsd:restriction>
        </xsd:simpleType>
    </xsd:element>
    <xsd:element name="packageRemark" type="NonNullStringType"/>
    <xsd:element name="packageLabel" type="NonNullStringType"/>
    <xsd:element name="packageCopies" type="xsd:integer"/>
    <xsd:element name="deliveryUnitRemark" type="NonNullStringType"/>
    <xsd:element name="latestAcceptedDeliveryDate" type="xsd:date"/>
    <xsd:element name="priority" type="xsd:integer"/>
    <!-- ====================  Order Specification Definition ==================== -->
    <!-- ======================================================================== -->
    <xsd:complexType name="ProductOrderSpecification">
        <xsd:sequence>
            <xsd:element ref="orderAccount" minOccurs="0"/>
            <xsd:element name="orderReference">
                <xsd:simpleType>
                    <xsd:restriction base="NonNullStringType">
                        <xsd:maxLength value="30"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:element>
            <xsd:element ref="orderRemark" minOccurs="0"/>
            <xsd:element ref="deliveryInformation"/>
            <xsd:element name="orderItem" type="ProductOrderItemType" maxOccurs="unbounded"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="OrderDataAccessSpecification">
        <xsd:sequence>
            <xsd:element name="orderReference">
                <xsd:simpleType>
                    <xsd:restriction base="NonNullStringType">
                        <xsd:maxLength value="30"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:element>
            <xsd:element name="orderItem" type="ProductOrderItemType"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="OrderMonitorSpecification">
        <xsd:sequence>
            <xsd:element ref="orderAccount" minOccurs="0"/>
            <xsd:element ref="orderId"/>
            <xsd:element name="orderReference"/>
            <xsd:element ref="orderRemark" minOccurs="0"/>
            <xsd:element ref="deliveryInformation" minOccurs="0"/>
            <xsd:element name="orderItem" type="OrderMonitorItemType" maxOccurs="unbounded"/>
            <xsd:element name="orderStatusInfo" type="OrderStatusType"/>
```



```
        </xsd:sequence>
    </xsd:complexType>
    <!--========================  Order Item Definition ========================== -->
    <!--======================================================================== -->
    <xsd:complexType name="ProductOrderItemType">
        <xsd:sequence>
            <xsd:element name="productId">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="idCitation">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element ref="resTitle"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element ref="collectionId" minOccurs="0"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element ref="processing" minOccurs="0" maxOccurs="unbounded"/>
            <xsd:element ref="programming" minOccurs="0" maxOccurs="unbounded"/>
            <xsd:element ref="sceneSelection" minOccurs="0"/>
            <xsd:element ref="packageMedium" minOccurs="0"/>
            <xsd:element name="qualityOfService" minOccurs="0"/>
            <xsd:element name="orderItemStatusInfo" type="OrderItemStatusType" minOccurs="0"/>
            <xsd:element name="orderItemRemark" type="xsd:string" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
    <!--====================== ComplexType  Definition Section ====================== -->
    <!--======================================================================== -->
    <xsd:complexType name="ProductOrderRequestType">
        <xsd:sequence>
            <xsd:element ref="userInformation"/>
            <xsd:element name="orderSpecification" type="ProductOrderSpecification"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="ProductServiceOptionsType">
        <xsd:sequence>
            <xsd:element name="productOrderOptions" maxOccurs="unbounded">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="productOrderOptionsId" type="CharacterString"/>
                        <xsd:element name="processingOption" minOccurs="0" maxOccurs="unbounded">
                            <xsd:complexType>
                                <xsd:complexContent>
                                    <xsd:extension base="ProductProcessingOptionType">
                                        <xsd:sequence>
                                            <xsd:element name="optionName" type="NonNullStringType"/>
                                            <xsd:element name="optionValueDefinition">
                                                <xsd:complexType>
                                                    <xsd:choice>
                                                        <xsd:element name="optionValueRange">
                                                            <xsd:complexType>
                                                                <xsd:sequence>
                                                                    <xsd:element name="numericOptionValueMin"
type="xsd:float"/>
                                                                    <xsd:element name="numericOptionValueMax"
type="xsd:float"/>
                                                                    <xsd:element name="numericOptionValueStep"
type="xsd:float"/>
                                                                </xsd:sequence>
                                                            </xsd:complexType>
                                                        </xsd:element>
                                                        <xsd:element name="optionValueList">
                                                            <xsd:complexType>
                                                                <xsd:sequence>
                                                                    <xsd:element name="stringOptionValue"
type="NonNullStringType" maxOccurs="unbounded"/>
                                                                </xsd:sequence>
                                                            </xsd:complexType>
                                                        </xsd:element>
                                                    </xsd:choice>
```

```
                                        </xsd:complexType>
                                    </xsd:element>
                                </xsd:sequence>
                            </xsd:extension>
                        </xsd:complexContent>
                    </xsd:complexType>
                </xsd:element>
                <xsd:element name="programmingOption" minOccurs="0" maxOccurs="unbounded">
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:element name="optionName" type="NonNullStringType"/>
                            <xsd:element name="optionValueDefinition">
                                <xsd:complexType>
                                    <xsd:choice>
                                        <xsd:element name="optionValueRange">
                                            <xsd:complexType>
                                                <xsd:sequence>
                                                    <xsd:element name="numericOptionValueMin"
type="xsd:float"/>
                                                    <xsd:element name="numericOptionValueMax"
type="xsd:float"/>
                                                    <xsd:element name="numericOptionValueStep"
type="xsd:float"/>
                                                </xsd:sequence>
                                            </xsd:complexType>
                                        </xsd:element>
                                        <xsd:element name="optionValueList">
                                            <xsd:complexType>
                                                <xsd:sequence>
                                                    <xsd:element name="stringOptionValue"
type="NonNullStringType" maxOccurs="unbounded"/>
                                                </xsd:sequence>
                                            </xsd:complexType>
                                        </xsd:element>
                                    </xsd:choice>
                                </xsd:complexType>
                            </xsd:element>
                        </xsd:sequence>
                    </xsd:complexType>
                </xsd:element>
                <xsd:element name="sceneSelectionOption" minOccurs="0" maxOccurs="unbounded">
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:element name="sceneType"/>
                        </xsd:sequence>
                    </xsd:complexType>
                </xsd:element>
                <xsd:element name="productDeliveryOptions" maxOccurs="unbounded">
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:element name="deliveryMethod">
                                <xsd:simpleType>
                                    <xsd:restriction base="xsd:string">
                                        <xsd:enumeration value="ftp-push"/>
                                        <xsd:enumeration value="ftp-pull"/>
                                        <xsd:enumeration value="mail"/>
                                    </xsd:restriction>
                                </xsd:simpleType>
                            </xsd:element>
                            <xsd:element ref="packageMedium" maxOccurs="unbounded"/>
                        </xsd:sequence>
                    </xsd:complexType>
                </xsd:element>
                <xsd:element name="qualityOfService" type="CharacterString" minOccurs="0"
maxOccurs="unbounded"/>
                <xsd:element name="orderOptionInfoURL" type="CharacterString" minOccurs="0"/>
                <xsd:element name="priceInformationURL" minOccurs="0"/>
                <xsd:element name="purchaseConditionURL" minOccurs="0"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
```



```
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="HorizontalSceneSelectionDefinitionType">
        <xsd:choice>
            <xsd:element name="rectangleSceneSelectionOption">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element ref="acrossGrid"/>
                        <xsd:element ref="alongGridUnitType"/>
                        <xsd:element ref="alongGrid"/>
                        <xsd:element name="acrossSizeLimitation">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="acrossSizeMin" type="xsd:integer"/>
                                    <xsd:element name="acrossSizeMax" type="xsd:integer"/>
                                    <xsd:element name="acrossSizeStep" type="xsd:integer"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="acrossPositionLimitation">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="acrossStart" type="xsd:integer"/>
                                    <xsd:element name="acrossStop" type="xsd:integer"/>
                                    <xsd:element name="acrossStep" type="xsd:integer"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="alongSizeLimitation">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="alongSizeMin" type="xsd:integer"/>
                                    <xsd:element name="alongSizeMax" type="xsd:integer"/>
                                    <xsd:element name="alongSizeStep" type="xsd:integer"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="alongPositionLimitation">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="alongStart" type="xsd:integer"/>
                                    <xsd:element name="alongStop" type="xsd:integer"/>
                                    <xsd:element name="alongStep" type="xsd:integer"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:choice>
    </xsd:complexType>
    <xsd:complexType name="ProductProcessingOptionType"/>
    <xsd:complexType name="OrderStatusType">
        <xsd:sequence>
            <xsd:element name="orderState">
                <xsd:simpleType>
                    <xsd:restriction base="xsd:string">
                        <xsd:enumeration value="BeingEstimated"/>
                        <xsd:enumeration value="Estimated"/>
                        <xsd:enumeration value="BeingQuoted"/>
                        <xsd:enumeration value="NotValid"/>
                        <xsd:enumeration value="BeingProcessed"/>
                        <xsd:enumeration value="Cancelled"/>
                        <xsd:enumeration value="BeingCancelled"/>
                        <xsd:enumeration value="Deleted"/>
                        <xsd:enumeration value="BeingDeleted"/>
                        <xsd:enumeration value="Completed"/>
                        <xsd:enumeration value="BeingOnHold"/>
                        <xsd:enumeration value="Terminated"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:element>
            <xsd:element name="additionalStatusInfo" type="NonNullStringType" minOccurs="0"/>
```

```
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="OrderItemStatusType">
        <xsd:sequence>
            <xsd:element name="orderState" type="xsd:string"/>
            <xsd:element name="additionalStatusInfo" type="NonNullStringType" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="PackagePriceType">
        <xsd:sequence>
            <xsd:element ref="price" minOccurs="0"/>
            <xsd:element ref="listPrice" minOccurs="0"/>
            <xsd:element ref="expectedPrice" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="PriceType">
        <xsd:sequence>
            <xsd:element name="refCurrency">
                <xsd:simpleType>
                    <xsd:restriction base="NonNullStringType">
                        <xsd:maxLength value="3"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:element>
            <xsd:element name="refAmount">
                <xsd:simpleType>
                    <xsd:restriction base="xsd:unsignedLong">
                        <xsd:maxInclusive value="4294967296"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:element>
            <xsd:element name="userCurrency" minOccurs="0">
                <xsd:simpleType>
                    <xsd:restriction base="NonNullStringType">
                        <xsd:maxLength value="3"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:element>
            <xsd:element name="userAmount" minOccurs="0">
                <xsd:simpleType>
                    <xsd:restriction base="xsd:unsignedLong">
                        <xsd:maxInclusive value="4294967296"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="ProductDeliveryOptionsType">
        <xsd:sequence>
            <xsd:element name="productByteSize" type="xsd:integer" minOccurs="0"/>
            <xsd:element name="productFormat" type="NonNullStringType" minOccurs="0"/>
            <xsd:element name="productCompression" type="NonNullStringType" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="SceneSelectionType">
        <xsd:sequence>
            <xsd:element name="sceneType">
                <xsd:simpleType>
                    <xsd:restriction base="NonNullStringType">
                        <xsd:maxLength value="20"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:element>
            <xsd:element ref="sceneCenter" minOccurs="0"/>
            <xsd:element name="temporalSelection" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="startDateTime"/>
                        <xsd:element name="endDateTime"/>
                    </xsd:sequence>
                </xsd:complexType>
```



```
            </xsd:element>
            <xsd:element name="scenePosition" type="CharacterString" minOccurs="0"/>
            <xsd:element name="sceneSize" type="CharacterString" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="DeliveryUnitType">
        <xsd:sequence/>
    </xsd:complexType>
    <xsd:complexType name="DeliveryInformationType">
        <xsd:choice>
            <xsd:element name="ftp-push"/>
            <xsd:element name="ftp-pull"/>
            <xsd:element name="mail" type="DeliveryAddressType"/>
        </xsd:choice>
    </xsd:complexType>
    <xsd:complexType name="DeliveryAddressType">
        <xsd:sequence>
            <xsd:element name="recipient" minOccurs="0">
                <xsd:simpleType>
                    <xsd:restriction base="NonNullStringType">
                        <xsd:maxLength value="40"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:element>
            <xsd:element name="companyRef" type="CharacterString" minOccurs="0"/>
            <xsd:element name="postalAddress" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="streetAddress" type="xsd:string"/>
                        <xsd:element name="city" type="xsd:string"/>
                        <xsd:element name="state" type="xsd:string"/>
                        <xsd:element name="postalCode" type="xsd:string"/>
                        <xsd:element name="country" type="xsd:string"/>
                        <xsd:element name="postBox" type="xsd:string"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="telNumber" minOccurs="0">
                <xsd:simpleType>
                    <xsd:restriction base="NonNullStringType">
                        <xsd:maxLength value="18"/>
                    </xsd:restriction>
                </xsd:simpleType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="OrderMonitorItemType">
        <xsd:sequence>
            <xsd:element name="productId">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="idCitation">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element ref="resTitle"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element ref="collectionId"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:choice>
                <xsd:sequence>
                    <xsd:element ref="sceneCenter" minOccurs="0"/>
                    <xsd:element ref="programming" minOccurs="0" maxOccurs="unbounded"/>
                </xsd:sequence>
                <xsd:sequence>
                    <xsd:element ref="processing" minOccurs="0" maxOccurs="unbounded"/>
                    <xsd:element ref="programming" minOccurs="0" maxOccurs="unbounded"/>
                    <xsd:element ref="sceneSelection" minOccurs="0"/>
                    <xsd:element ref="packageMedium" minOccurs="0"/>
                    <xsd:element name="qualityOfService" minOccurs="0"/>
```

```
                </xsd:sequence>
            </xsd:choice>
            <xsd:element name="orderItemStatusInfo" type="OrderItemStatusType" minOccurs="0"/>
            <xsd:element name="orderItemRemark" type="xsd:string" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
    <!--============================ Simple Types-->
    <xsd:simpleType name="ExtendedStatusType">
        <xsd:restriction base="CharacterString">
            <xsd:enumeration value="success"/>
            <xsd:enumeration value="failure"/>
            <xsd:enumeration value="incomplete"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:schema>
```

Order WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://earth.esa.int/XML/dsm" xmlns:eoli="http://earth.esa.int/XML/eoli"
targetNamespace="http://earth.esa.int/XML/dsm">
    <!-- Modified import schema in order to define this WSDL as  WS-I Basic Profile compliant -->
    <types>
        <schema attributeFormDefault="qualified" elementFormDefault="qualified"
targetNamespace="http://schemas.xmlsoap.org/wsdl/" xmlns="http://www.w3.org/2001/XMLSchema">
            <import namespace="http://earth.esa.int/XML/eoli" schemaLocation="Eoli_Order.xsd"/>
        </schema>
    </types>
    <message name="searchRequestInput">
        <part name="searchRequestParameter" element="eoli:searchRequest"/>
    </message>
    <message name="presentRequestInput">
        <part name="searchRequestParameter" element="eoli:presentRequest"/>
    </message>
    <message name="productOrderRequestInput">
        <part name="productOrderRequestParameter" element="eoli:productOrderRequest"/>
    </message>
    <message name="orderMonitorRequestInput">
        <part name="orderMonitoRequestParameter" element="eoli:orderMonitorRequest"/>
    </message>
    <message name="responseOutput">
        <part name="responseParameter" element="eoli:response"/>
    </message>
    <message name="orderResponseOutput">
        <part name="orderResponseParameter" element="eoli:orderResponse"/>
    </message>
    <message name="orderMonitorResponseOutput">
        <part name="orderMonitorResponseParameter" element="eoli:orderMonitorResponse"/>
    </message>
    <portType name="dsmPortType">
        <operation name="processSearchRequest">
            <input message="tns:searchRequestInput"/>
            <output message="tns:responseOutput"/>
        </operation>
        <operation name="processPresentRequest">
            <input message="tns:presentRequestInput"/>
            <output message="tns:responseOutput"/>
        </operation>
        <operation name="processProductOrderRequest">
            <input message="tns:productOrderRequestInput"/>
            <output message="tns:orderResponseOutput"/>
        </operation>
        <operation name="processOrderMonitorRequest">
            <input message="tns:orderMonitorRequestInput"/>
            <output message="tns:orderMonitorResponseOutput"/>
        </operation>
    </portType>
    <binding name="dsmBinding" type="tns:dsmPortType">
        <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
        <operation name="processSearchRequest">
```



```
            <soap:operation soapAction="http://earth.esa.int/SOAP/searchRequest"/>
            <input>
                <soap:body use="literal"/>
            </input>
            <output>
                <soap:body use="literal"/>
            </output>
        </operation>
        <operation name="processPresentRequest">
            <soap:operation soapAction="http://earth.esa.int/SOAP/presentRequest"/>
            <input>
                <soap:body use="literal"/>
            </input>
            <output>
                <soap:body use="literal"/>
            </output>
        </operation>
        <operation name="processProductOrderRequest">
            <soap:operation soapAction="http://earth.esa.int/SOAP/productOrderRequest"/>
            <input>
                <soap:body use="literal"/>
            </input>
            <output>
                <soap:body use="literal"/>
            </output>
        </operation>
        <operation name="processOrderMonitorRequest">
            <soap:operation soapAction="http://earth.esa.int/SOAP/orderMonitorRequest"/>
            <input>
                <soap:body use="literal"/>
            </input>
            <output>
                <soap:body use="literal"/>
            </output>
        </operation>
    </binding>
    <service name="dsmService">
        <port name="dsmSoap" binding="tns:dsmBinding">
            <soap:address location="http://earth.esa.int/"/>
        </port>
    </service>
</definitions>
```

Annex D
(informative)

Proposed evolution towards OGC CAT 2.0 and CSW Compliance

This annex describes the desireable and anticipated evolution of the profile to better comply with OGC 04-021r2 and specifically an alignment with CSW.

It should be kept in mind that a key driver for this profile is to allow it to be implemented as a standard interface to existing catalogues of EO products in a cost effective manner, and thus we wish to require quite a minimal set of interfaces. These interfaces may subsequently be extended to ensure compatability with other systems (for example when required INSPIRE directives), it should be anticipated that translators/adaptors may be developed to provide further interoperability by offering alternative bindings, however, the profile is kept to a minimum so that the overhead needn't be implemented by all catalogue providers.

Catalogue Schema and Searchable Properties

The EO profile is targeted towards the identification of EO products from large collections of data that are typically contain data of a certain product type generated from a same sensor on board a satellite or series of satellites. The key attributes that distinguish the products within a collection are those identified within this profile.

It is proposed to align further the geographical metadata within this profile with GML 3.1.1

The core catalogue schema of CAT 2.0 is not really appropriate for the individual product metadata, but more towards collection level metadata (e.g. textual information that is common to all data within the collection), this is also the case for the core searchable properties. Inclusion of additional metadata and queriables may be included in the profile in the future, but would be expected to be aligned towards the requirements from INSPIRE.

Interfaces

The basic search operations of the EO Profile have a good mapping to CSW. The anticipated evolution is summarised below:

EOProfile	CSW	Comment
EOProfile:SearchRequest	CSW:GetRecords	The operations shall be renamed to be inline with the CSW implementation
EOProfile:PresentRequest	CSW:GetRecordById	The operations shall be renamed to be inline with the CSW implementation

-	OGC_Service:GetCapabilities	Service metadata discovery should be aligned with INSPIRE in a future iteration of the EO profile
-	CSW:DescribeRecord	Since we currently intend to only to specify SOAP bindings, and so schema described are published and referenced via the WSDL there seems little added value to provide this interface within the EO profile.

Request Encoding

The current required binding to be mandated by this profile is SOAP, it is not anticipated to require a KVP encoding.

Query Format

The query format within the EO profile currently defined by the simpleRequest. The advantages of this format is that it provides a well defined set of parameters that need to be supported by each catalogue provider, and search performances can be optimised towards these parameters. The simpleRequest can also be described using OGC Filter Syntax, and so it would be anticipated to migrate towards this syntax. However, it appears not to be possible to define using OGC Filter Syntax restrictions on what has to be supported by a catalogue provider, or to limit the possibilities to avoid "unreasonable" queries that would not be feasible to respond. These issues have been also previously reported to OGC WFS working group in change request 05-022.

GetRecords Request

Considering the mandatory or request parameters and those already present in the EO profile the following table shows the anticipated evolution

EOProfile	CSW	Comment
simpleQuery	constraint, constraintLanguageVersion	Use of OGC Filter Syntax is anticipated
resultType	resultType	Within EO profile we do not foresee much use in the *validate* value available in CSW
iteratorSize cursor	maxRecords startPosition	These parameters shall be renamed to align to CSW
presentation	elementSetName	These parameters shall be renamed to align to CSW, the EO profile adds a *browse* value

collectionId	typeNames	CSW foresees search against service, dataset, dataset collection and application. The EO profile is targeted towards searches within already identified collections, at this level no further alignment is foreseen

GetRecords Response

EOProfile	CSW	Comment
retreivedData	abstractRecords	No further alignment foreseen at this time
cursor hits	nextRecord numberOfRecordsMatched numberOfRecordsReturned	These parameters shall be renamed to align to CSW
status errorMessage	status	

GetRecordById Request

EOProfile	CSW	Comment
idCitation collectionId	id	It may be possible to combine collectionId and idCitation within the EO profile to result in a URI of the type prescribed by CSW although we would seem to loose alignment with ISO (TBC)
presentation	elementSetName	These parameters shall be renamed to align to CSW, the EO profile adds a *browse* value

GetRecords Response

EOProfile	CSW	Comment
retreivedData	abstractRecords	No further alignment foreseen at this time
hits		To be renamed numberOfRecordsReturned,

		which does not seem to be specified by CSW as part of response, but is considered useful in the EO profile
status errorMessage		status does not seem to be specified by CSW as part of response, but is considered useful in the EO profile

Bibliography

[SSE] ESA Service Support Environment
http://services.eoportal.org

[EOLI-XML] http://earth.esa.int/rtd/Documents/EOLI-XML-ICD.pdf

[CIP2.4] Catalogue Interoperability Protocol (CIP) Specification – Release B, CEOS/WGISS/PTT/CIP-B, June 1998, Issue 2.4
http://wgiss.ceos.org/ics/documentation.html

[SOAP] Simple Object Access Protocol (SOAP) 1.1, W3C Note 08 May 2000, http://www.w3.org/TR/2000/NOTE-SOAP-20000508/

[WSDL] Web Services Description Language (WSDL) 1.1, W3C Note 15 March 2001, http://www.w3.org/TR/wsdl

[WSDL-Tutorial] Web Services Description Language (WSDL) 1.1 Tutorial, http://www.w3schools.com/wsdl/

[XML] W3C Recommendation 6 October 2000, Extensible Markup Language (XML) 1.0 (Second Edition), http://www.w3.org/TR/REC-xml.

W3C Recommendation January 1999, Namespaces In XML, http://www.w3.org/TR/2000/REC-xml-names.

W3C Recommendation 2 May 2001: XML Schema Part 0: Primer, http://www.w3.org/TR/2001/REC-xmlschema-0-20010502/

W3C Recommendation 2 May 2001: XML Schema Part 1: Structures, http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/

W3C Recommendation 2 May 2001: XML Schema Part 2: Datatypes, http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/

[UML] Unified Modeling Language (UML) Version 1.3, The Object Management Group (OMG): http://www.omg.org/cgi-bin/doc?formal/00-03-01

[Valids] Catalogue Interoperable Catalogue System Valids, CEOS/WGISS/PTT/Valids

[FGDC] Content Standard for Digital Geospatial Metadata: Extensions for Remote Sensing Metadata, FGDC-STD-012-2002.

[GML] Geography Markup Language (GML) 2.0, OpenGIS Implementation Specification, 20 February 2001, OGC Document Number 01-029, http://www.opengis.net/gml/01-029/GML2.html and http://opengis.net/schema.htm.

[HTTP] Hypertext Transfer Protocol -- HTTP/1.1, RFC 2616, U.C. Irvine, DEC W3C/MIT, DEC, W3C/MIT, W3C/MIT, January 1997, http://www.normos.org/ietf/rfc/rfc2616.txt

[ISO] ISO/TC 211, Geographic information – Metadata, ISO/FDIS 19115, 23/01/2003.

[ISO19130] ISO/TC 211 N. 1167, WD 19130, Geographic information – Sensor and data model for imagery and gridded data, 211n1167.pdf, 25/09/2002.

[ISO19103] Geographic information – Conceptual schema language.

[OpenGIS] OpenGIS–Catalog Interface Implementation Specification (Version 1.0), Document 99-051s.

[Profile] ISO/IEC TR 10000-1:1998. *Information Technology – Framework and taxonomy of International Standardised Profiles – Part 1: General principles and documentation framework*. Technical Report, JTC 1. Fourth edition. Available [online]: <http://www.iso.ch/iso/en/ittf/PubliclyAvailableStandards/c030726_ISO_IEC_TR_10000-1_1998(E).zip>.

[ODP] ISO/IEC 10746-2:1996. *Information Technology – Open Distributed Processing – Reference Model: Foundations*. Common text with ITU-T Recommendation X.902. Available [online]: <http://www.iso.ch/iso/en/ittf/PubliclyAvailableStandards/s018836_ISO_IEC_107462_1996(E).zip>.