Open Geospatial Consortium

Publication Date: 2014-01-31

Approval Date: 2013-12-24

Submission Date: 2013-09-04

External identifier of this OGC® document: http://www.opengis.net/def/BP/dseo/1.0

Internal reference number of this OGC® document: 13-043

Version: 1.0

Category: OGC® Best Practice

Editor: Daniele Marchionni, Raul Cafini

OGC Download Service for Earth Observation Products Best Practice

Copyright notice

Warning

Document type:	OGC® Best Practice
Document subtype:	
Document stage:	Approved for public release
Document language:	English

Contents

Index of Figures

Index of Tables

i. Abstract

This OGC® Best Practices document specifies the interfaces, bindings, requirements, conformance classes for online download of Earth Observation products. This protocol covers several scenarios implemented by European Space Agency - ESA for providing its products to users:

- The EO Product to be downloaded is already available and can be downloaded as it is.
- The EO Product is not online available but is stored in a near online archive.
- The EO Product is advertised in a Catalogue, but it is not physically available and it has to be generated on the fly by a processing facility.
- The EO product is archived in several distributed online archives and it can be downloaded in parallel.

The basic scenarios can be simply supported by Web Browsers, the most complex ones need a dedicated client (download manager) supporting Metalink and multisource download.

This Best Practice document has been prepared basing on the work performed in the frame of ESA's Next Generation Earth Observation user services and it was initially produced during the ESA HMA (Heterogeneous Missions Accessibility) initiative [OR1] and related projects.

ii. Keywords

The following are keywords to be used by search engines and document catalogues.

ogcdoc, OGC document, best practice, dseo, eo, data access, download, hma

iii. Preface

This Best Practice document summarizes the scenarios implemented by European Space Agency – ESA for supplying its products to the users. These scenarios cover different needs of different Satellite's missions, and then they have been taken into account in separated requirements & conformance classes. In this way the developers can choose the conformance class that fits their needs without mandating the compliance with all requirements.

Actually two requirements classes have been defined, distinguishing between Download Servers dealing with already available products (**Core** class) and other servers supporting on-the-fly production (**On-Demand Download** class).

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 shall not be held responsible for identifying any or all such patent rights.

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

iv. Submitting organizations

The following organizations submitted this Document to the Open Geospatial Consortium (OGC):

- ESA – European Space Agency
- Telespazio

v. Submitters

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

Name	Affiliation
Daniele Marchionni	**Telespazio**
Raul Cafini	**Telespazio**

1. Scope

This OGC® Best Practices document specifies the interfaces, bindings, requirements, conformance classes for online download of Earth Observation products.

This document describes several ways for downloading EO Products starting from their HTTP URLs generally retrieved from an EO Product Catalogue.

Basically it distinguishes between 2 cases:

- The product is already physically on-line available for download;
- The product is NOT already physically on-line available (for various reasons e.g. the product has to be retrieved from an Offline archive, or it has to be generated from ancillary data).

Additionally, in both cases, the following alternatives are supported:

- Product file is available from multiple sources (parallel download);
- The product is composed of multiple files, each possibly available from multiple sources (multiple download);
- The product file is reachable via several HTTP redirections from the original URL.
- Any combination of previous alternatives.

The protocol uses:

- HTTP GET for all interactions;
- HTTP redirections, in case of “virtual URL”;
- Re-try mechanism according to specific HTTP returned codes, in case of on-demand / off-line products;
- Metalink files for managing the multi-file / multi source download.

2. Conformance

Two Requirements and Conformance classes have been defined:

- **Core**, regrouping all requirements for supporting download of already available EO Products including: multi file download, multi source download, and redirection.
- **On-Demand Download**, inheriting from Core class and adding the requirements for allowing download of EO product files not ready yet.

In this way "classic" download servers supporting download of already available products needs to implement just the **Core** class, while the more advanced ones, having some processing capability, might implement **On-Demand Download** class.

The following diagram shows the relationships between the defined Requirement Classes.

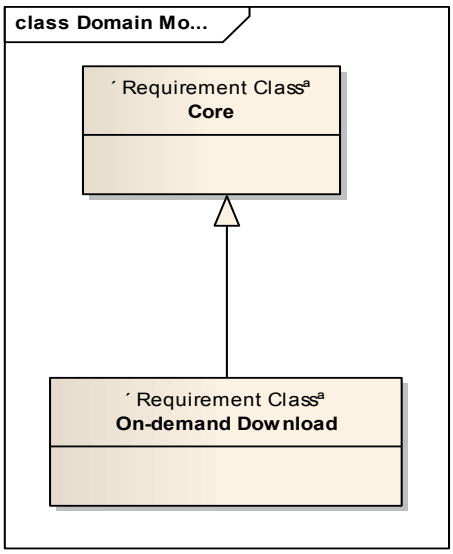


Figure 2-1: Requirement Classes.

The inheritance relationship between the different classes represents the inheritance of all requirements from the super class. E.g.: On-demand Download class defines its specific requirements and includes also the requirements defined in the Core class.

The following table reports:

- The Requirement Class name
- the URI
- The dependency with other requirements classes.

Requirement Class	Requirement Class URI	Dependency
Core	http://www.opengis.net/spec/DSEO/1.0/req/Core	
OnDemandDownload	http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload	Core

Table 2-1: Requirement classes.

The root path of all Requirements and conformance test URIs defined in this document is:

http://www.opengis.net/spec/DSEO/1.0/

Conformance with this Best Practice document shall be checked using all the relevant tests specified in Annex A (normative) of this document. The framework, concepts, and methodology for testing, and the criteria to be achieved to claim conformance are specified in the OGC Compliance Testing Policies and Procedures and the OGC Compliance Testing web site1.

In order to conform to this OGC™ interface standard, a software implementation shall choose to implement at least one of Core and optionally the other class.

All requirements-classes and conformance-classes described in this document are owned by the standard(s) identified.

1 www.opengeospatial.org/cite

3. 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.

[NR1] W3C Recommendation January 1999, Namespaces In XML, http://www.w3.org/TR/2000/REC-xml-names.

[NR2] W3C Recommendation 6 October 2000, Extensible Markup Language (XML) 1.0 (Second Edition), http://www.w3.org/TR/REC-xml

[NR3] W3C Recommendation 2 May 2001: XML Schema Part 0: Primer, http://www.w3.org/TR/2001/REC-xmlschema-0-20010502/

[NR4] W3C Recommendation 2 May 2001: XML Schema Part 1: Structures, http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/

[NR5] W3C Recommendation 2 May 2001: XML Schema Part 2: Datatypes, http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/

[NR6] OWS Common Implementation Specification, May 2005 OGC 05-008c1

[NR7] OpenGIS® Web Services Common Specification OGC 06-121r9

[NR8] The Extensible Markup Language (XML), World Wide Web Consortium, http://www.w3.org/TR/1998/REC-xml-19980210

[NR9] New OGC Document Template Draft OGC 10-176r4

[NR10] Guidance to Editors and Authors of OGC Documents that use the OGC Standards document template OGC 10-177r3

[NR11] Policy Directives for Writing and Publishing OGC Standards: TC Decisions OGC 06-135r11

[NR12] The Specification Model — A Standard for Modular specifications OGC 08-131r3

[NR13] Unified Modeling Language (UML) Version 1.3, The Object Management Group (OMG): http://www.omg.org/cgi-bin/doc?formal/00-03-01

[NR14] Metalink Internet Standard (Metalink): http://www.metalinker.org/schema/3.0/metalink.xsd

Other references:

[OR1] Heterogeneous Missions Accessibility – Design Methodology, Architecture and Use of Geospatial Standards for the Ground Segment Support of Earth Observation missions ESA TM-21
http://www.esa.int/About_Us/ESA_Publications/ESA_TM-21_Heterogeneous_Missions_Accessibility

4. Terms and Definitions

This document uses the terms defined in §5.3 of [OGC 06-121r8], which is based on the ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards. In particular, the word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this standard.

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

4.1
dataset series (dataset collection)
Collection of datasets sharing the same product specification [ISO 19113, ISO 19114, ISO 19115]. In this context, a collection metadata record in the catalogue describes a collection of EO Products, typically a dataset collection corresponds to datasets (i.e. products) generated by a single sensor in a specific mode on a particular EO satellite.

4.2
EO Product
Data product, typically stored on computer file, generated by sensors carried by Earth Observation Satellites.

4.3
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.

4.4
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]

4.5
procedure oriented architectural style
platform-independent design approach that is focused on operations, their parameters and their results, that can be defined in an abstract level specification. Concrete platform-dependent specifications can be derived from the abstract level, allowing, for example, KVP or SOAP messaging.

4.6
product order
Ordering request asking the processing and delivery of precisely identified EO Products.

4.7
resource oriented architectural style
platform-independent design approach that is focused on resources, representations and actions, that can be defined in an abstract level specification. Concrete platform-

dependent specifications can be derived from the abstract level, allowing, for example, a RESTful architecture.

4.8
URL component
text contained after the context path between two forward slashes "/", that are not placed between two matching parentheses "(...)", and before any question mark character "?"

4.9
URL component key
text contained in an URL component that comes before an opening parenthesis "("

4.10
URL component value
text contained in an URL component that is enclosed between the first opening parenthesis "(" and the last closing parenthesis ")" in the → component

Example In the http://example.org/component1/componentKey(componentValue), components are as follows:

Context Path: http://example.org/
URL Components: component1, componentKey(componentValue)
URL Component Keys: component1, componentKey
URL Component Values:componentValue

5. Conventions

This sections provides details and examples for any conventions used in the document. Examples of conventions are symbols, abbreviations, use of XML schema, or special notes regarding how to read the document.

5.1 Abbreviated terms

Acronym	Definition
API	Application Program Interface
COTS	Commercial Off The Shelf
CRS	Coordinate Reference System
CSW	Catalogue Service-Web
DSEO	Download Service for Earth Observation Products
DCE	Distributed Computing Environment
DCP	Distributed Computing Platform
EO	Earth Observation
HMA	Heterogeneous Missions Accessibility
HTTP	Hyper Text Transport Protocol
ISO	International Organization for Standardization
OGC	Open GIS Consortium
SOAP	Simple Object Access Protocol
SWG	Standard Working Group
UML	Unified Modeling Language
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
UTF-8	Unicode Transformation Format-8

Acronym	Definition
WSDL	Web Service Definition Language
W3C	World Wide Web Consortium
XML	eXtensible Markup Language

5.2 UML notation

5.2.1 Introduction

Some diagrams that appear in this document are presented using the Unified Modeling Language (UML) diagrams:

- Class Diagrams

  Class diagrams show the static structure of the model, in particular, the things that exist (such as classes and types), their internal structure, and their relationships to other things. Class diagrams do not show temporal information, although they may contain occurrences of things that have or describe temporal behaviour.

- Sequence Diagrams

  A sequence diagram shows an interaction arranged in time sequence. In particular, it shows the objects participating in the interaction by their “lifelines” and the messages that they exchange arranged in time sequence. It does not show the associations among the objects.

5.2.2 UML Class Diagrams

A class diagram is a picture providing generic descriptions of possible systems. Class diagrams and object diagrams are alternate representations of object models. Class diagrams contain classes and object diagrams contain objects, but it is possible to mix classes and objects when dealing with various kinds of metadata, so the separation is not rigid. Class diagrams contain icons representing classes, interfaces, and their relationships. In particular, class diagrams contain:

- Logical Packages

  Packages purpose is to partition the logical model of a system. They are clusters of highly related classes that are themselves cohesive, but are loosely coupled with other such clusters. You can use packages to group classes, interfaces, and other packages.

- Classes

A class captures the common structure and common behaviour of a set of objects. A class is an abstraction of real-world items. When these items exist in the real world, they are instances of the class, and referred to as objects.

- Interfaces

  An interface specifies the externally visible operations of a class and/or component, and has no implementation of its own. An interface typically specifies only a limited part of the behaviour of a class or component.

- Parameterized Classes

  A parameterized class is a template for creating any number of instantiated classes that follow its format. It declares formal parameters. You can use other classes, types, and constant expressions as parameters. You cannot use the parameterized class itself as a parameter. You must instantiate a parameterized class before you can create its objects. In its simplest form, you can use parameterized classes to build container classes.

- Instantiated Classes

  An instantiated class is a class formed from a parameterized class by supplying actual values for parameters. It is created by supplying actual values for the formal parameters of the parameterized class. This instantiation process forms a concrete class in the family of the parameterized class. The instantiated class should be put at the client end of an instantiate relationship (accessible through the Create Entry on the Tools menu) that points to the corresponding parameterized class.

- Association Relationships

  An association represents a semantic connection between two classes, or between a class and an interface. Associations are bi-directional; they are the most general relationship and also the most semantically weak.

- Aggregate Relationship

  The aggregate relationship is used for showing a whole and part relationship between two classes.

  The class at the client end of the aggregate relationship is sometimes called the aggregate class. An instance of the aggregate class is an aggregate object. The class at the supplier end of the aggregate relationship is the part whose instances are contained or owned by the aggregate object. The aggregate relationship is used for showing that the aggregate object is physically constructed from other objects or that it logically contains another object. The aggregate object has ownership of its parts.

- Generalize/Inherits Relationships

  A generalize relationship between classes shows that the subclass shares the structure or behaviour defined in one or more super-classes. Use a generalize relationship to show an "is- a" relationship between classes.

- Instantiates Relationships

  An instantiates relationship represents the act of substituting actual values for the parameters of a parameterized class or parameterized class utility to create a specialized version of the more general item. In most cases, you will also draw a uses relationship between the instantiated class and another concrete class that is used as an actual parameter.

- Dependency Relationships

  Draw a dependency relationship between two classes, or between a class and an interface, to show that the client class depends on the supplier class/interface to provide certain services, such as:

  - The client class accesses a value (constant or variable) defined in the supplier class/interface.
  - Operations of the client class invoke operations of the supplier class/interface.
  - Operations of the client class have signatures whose return class or arguments are instances of the supplier class/interface.

The next picture shows the items just explained.

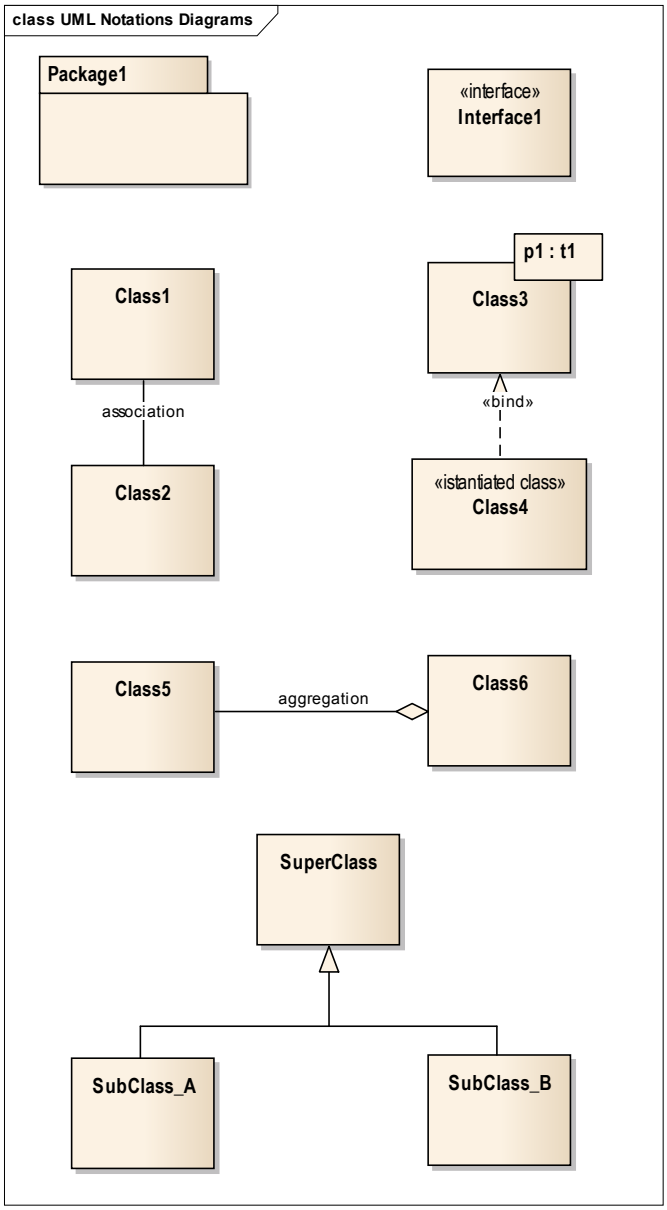


Figure 5-1: UML Class Diagram notations.

5.2.3 UML Sequence Diagrams

Sequence diagrams are a representation of an interaction between objects. A sequence diagram traces the execution of an interaction in time.

The picture below illustrates a sequence diagram.

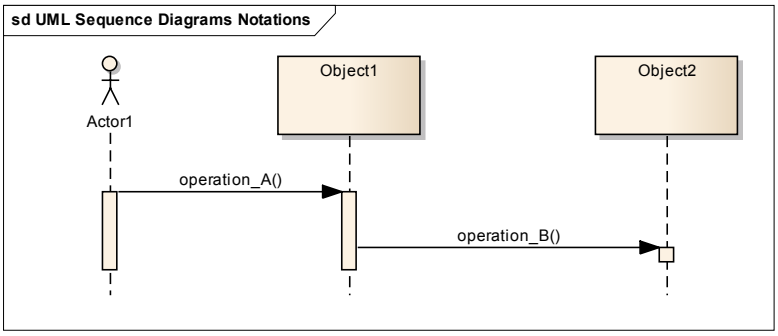


Figure 5-2: UML Sequence Diagram notations.

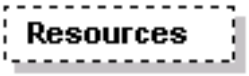
Each interaction between objects is the activation of an operation of an object, which includes input and output parameters.

5.3 XML notation

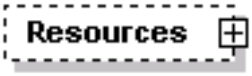
Most diagrams that appear in this specification are presented using an XML schema notation defined by the XMLSpy tool and described in this sub-clause.

Hereafter the symbols defined in the XML schema notation are described:

- Optional single element without child elements



- Optional single element with child elements



- Mandatory single element.

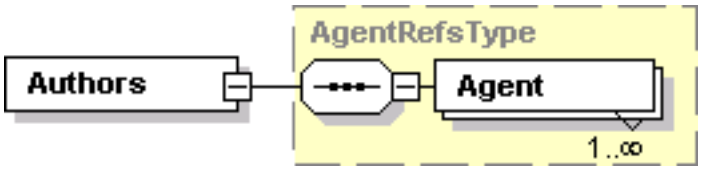
- Mandatory multiple element containing child elements. This element must occur at least once (Minimum Occurrence = 1) and may occur as often as desired (Maximum Occurrence = unbounded).

- Mandatory single element with containing simple content (e.g. text) or mixed complex content (e.g. text with xhtml markup).

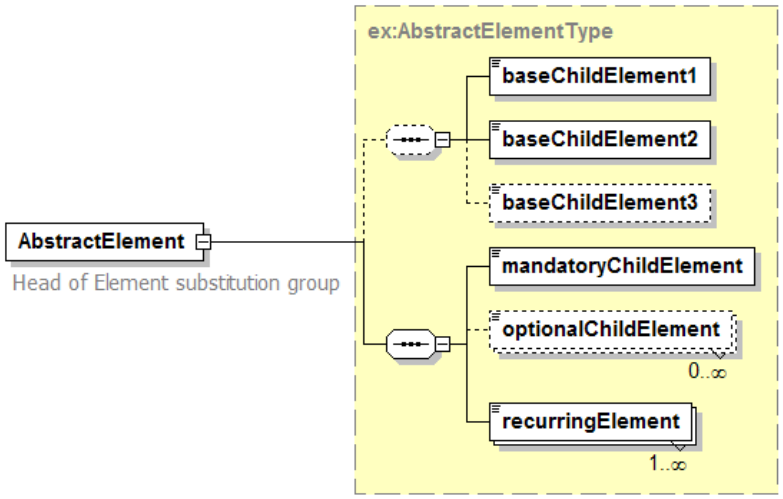
- A sequence of elements. The elements must appear exactly in the sequence in which they appear in the schema diagram.

- A choice of elements. Only a single element from those in the choice may appear at this position.

- Types. If an element refers to a complex global type, the type is shown with a border and yellow background.



- Complex Type. The following figure illustrates the use of a complex type for defining an XML element



5.4 Data dictionary tables

The XML data dictionary used to describe the parameters within this document is specified herein in a series of tables. The contents of the columns in these tables are described in Table 5-1.

Column title	Column contents
Names (left column)	Name of the XML element to describe. The name uses the XML encoding capitalization specified in Sub-clause 11.6.2 of [OGC 05-008]. The name capitalization rules used are specified in Sub-clause 11.6.2 of [OGC 05-008]. Some names in the tables may appear to contain spaces, but no names contain spaces.
Definition (second column)	Specifies the definition of this element.
Data type and value (third column)	Normally contains two items: The mandatory first item is often the data type used for this parameter, using data types appropriate in the model, in which this parameter is a named attribute of a XML entity. Alternately, the first item can identify the data structure (or class) referenced by this association, and references a separate table used to specify the contents of that class (or data structure). The optional second item in the third column of each table should indicate the source of values for this parameter, the alternative values, or other value information, unless the values are quite clear from other listed information.
Multiplicity and use (fourth column)	Normally contains two items: The item specifies the multiplicity and optionality of this parameter in this data structure, either “One (mandatory)”, “One or more (mandatory)”, “Zero or one (optional)”, or “Zero or more (optional)”. If it needs to describe the multiplicity of an XML choice element, it needs to use the word “mandatory/choice” and “optional/choice”. The choice element in fact provides an XML representation for describing a single selection from a set of element types where each selection item can be defined ‘mandatory’.

Table 5-1: Contents of data dictionary tables.

When the data type used for a parameter, in the third column of such a table, is another complex type, all the values must be specified and listed, together with the meaning of each value. When this information is extensive, these values and meanings are specified in a separated table and a cross reference to it is put in the table.

6. Download Service for Earth Observation Products (DSEO) Overview

As highlighted in the introduction sections, this document provides a simple interface for downloading EO Products available in two different ways:

- **Immediate Product access**: the requested product, specified by submitted URI, is on-line and ready for access.
- **On-Demand Download**: this type of request triggers an on-demand generation of the desired product. This option is used also when the requested product is available in off-line archive and the Download Server needs time to retrieve and restore it to on-line availability.

6.1 Protocol Operations

This specification is composed of the following operations:

- **GetCapabilities**, which allows a client to receive service metadata (Capabilities document) that describes the abilities of the specific server (see §8.1).
- **GetProduct**, which allows a client to request a specified Product providing its unique URI (see §8.2). The server returns different responses in case the requested product is on-line (directly available / available on another dissemination server / Metalink download) or off-line / on-demand generated (retry later to check product availability).

6.2 Essential Use-cases

6.2.1 On-Line available product Scenario

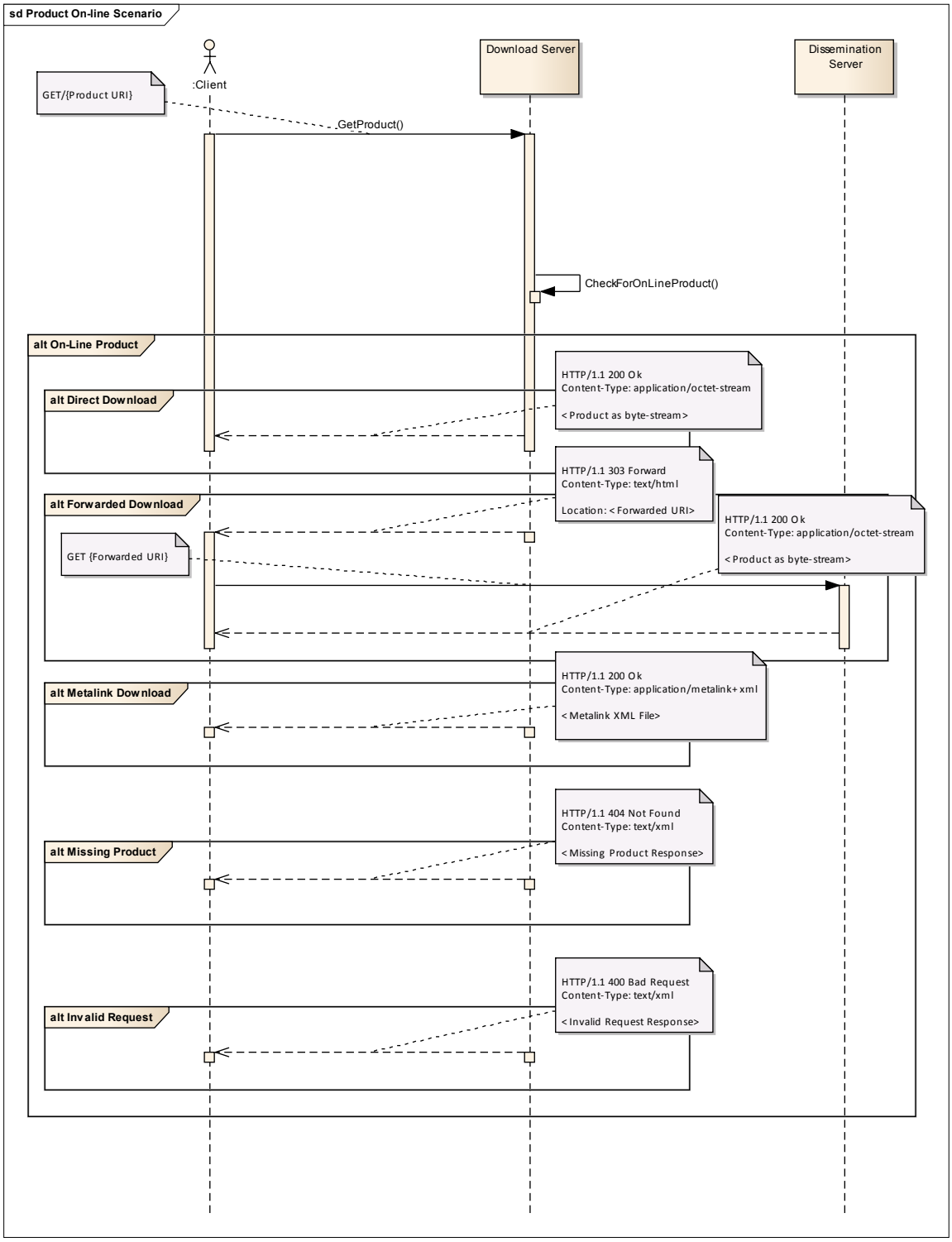


Figure 6-1: Product On-line scenario.

The scenario involves the following entities:

- **Client**: which is the one asking for EO Products;
- **Download Server**: it is the server implementing this specification;
- **Dissemination Server**: it is an external server different from the previous one;

Scenario description:

- The Client asks to the Download Server for a specific product (using GetProduct with a specific product URI).
- The Download Server checks for the on-line availability of the product.
    - If the product is on-line available, there are the following sub-cases:
        - The product is returned as byte stream.
        - The product is online, but at different site: the Download Server returns a forwarded download message where the alternative location of the requested product is specified. The client will download the product form the re-directed server.
        - The product is online, but composed of multiple files: the Download Server sends back a Metalink XML file. The Client will parse the Metalink to obtain all the items that are part of the requested product. This sub-case is detailed in §6.2.1.1.
    - If the product is missing the Download Server returns a Missing Product exception message.
    - If the request is invalid (the submitted request is not well formed, etc.) the Download Server returns a Bad Request exception message.

6.2.1.1 Product Download via Metalink Scenario

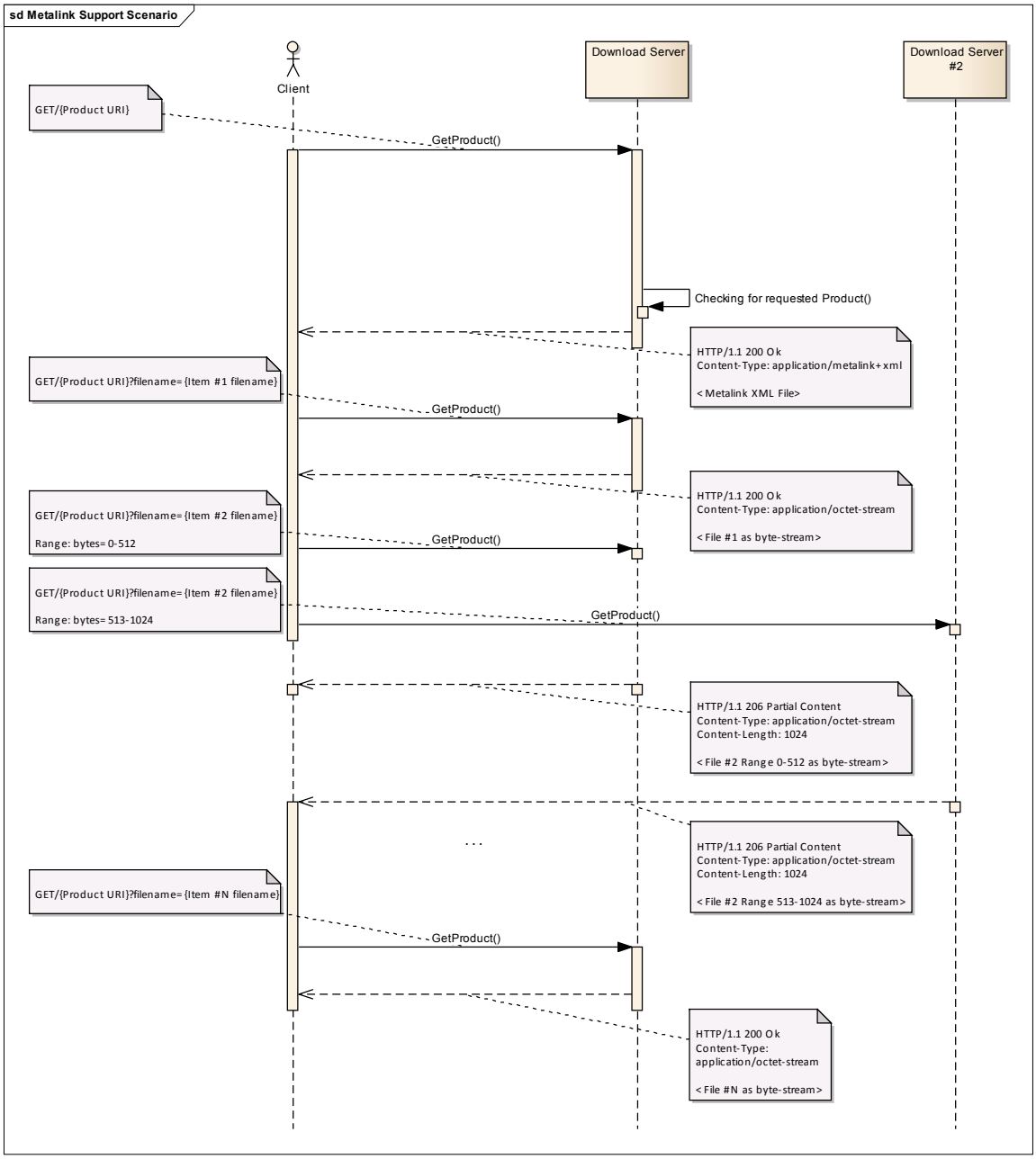


Figure 6-2: Product Download via Metalink Scenario.

The scenario involves the following entities:

- **Client**: which is the one asking for EO Products;
- **Download Server, Download Server#2**: servers implementing this specification.

Scenario description:

- The client asks to the Download Server to retrieve a product with specified URI (GetProduct).
- The Download Server internally checks if the specified product is available in on-line storage. The product is on-line available and it is composed of multiple files so the Download Server returns a Metalink XML file.

  A Metalink is an XML file where the various items composing the EO Product are listed and the possible sources are specified as well. Major details can be found on section §7.1.

- On Client side, the Metalink response message triggers the parsing of the file. For each item composing the product, one or more sub-URIs is defined.
- Each sub-URI is retrieved from the Client by an HTTP GET request:
  - In case the requested item has one source, then it is directly downloaded.
  - In case the requested item has two or more sources (e.g. item #2 is currently available both from main Download Server and from Download Server #2) then, depending on Client capability, a parallel download can be performed. The client sends a GetProduct request to each server asking for a byte range of the product. Each server sends back the partial content.

    When the client has received all trunks, then merge them rebuilding the whole file.

6.2.2 On-demand download Scenario

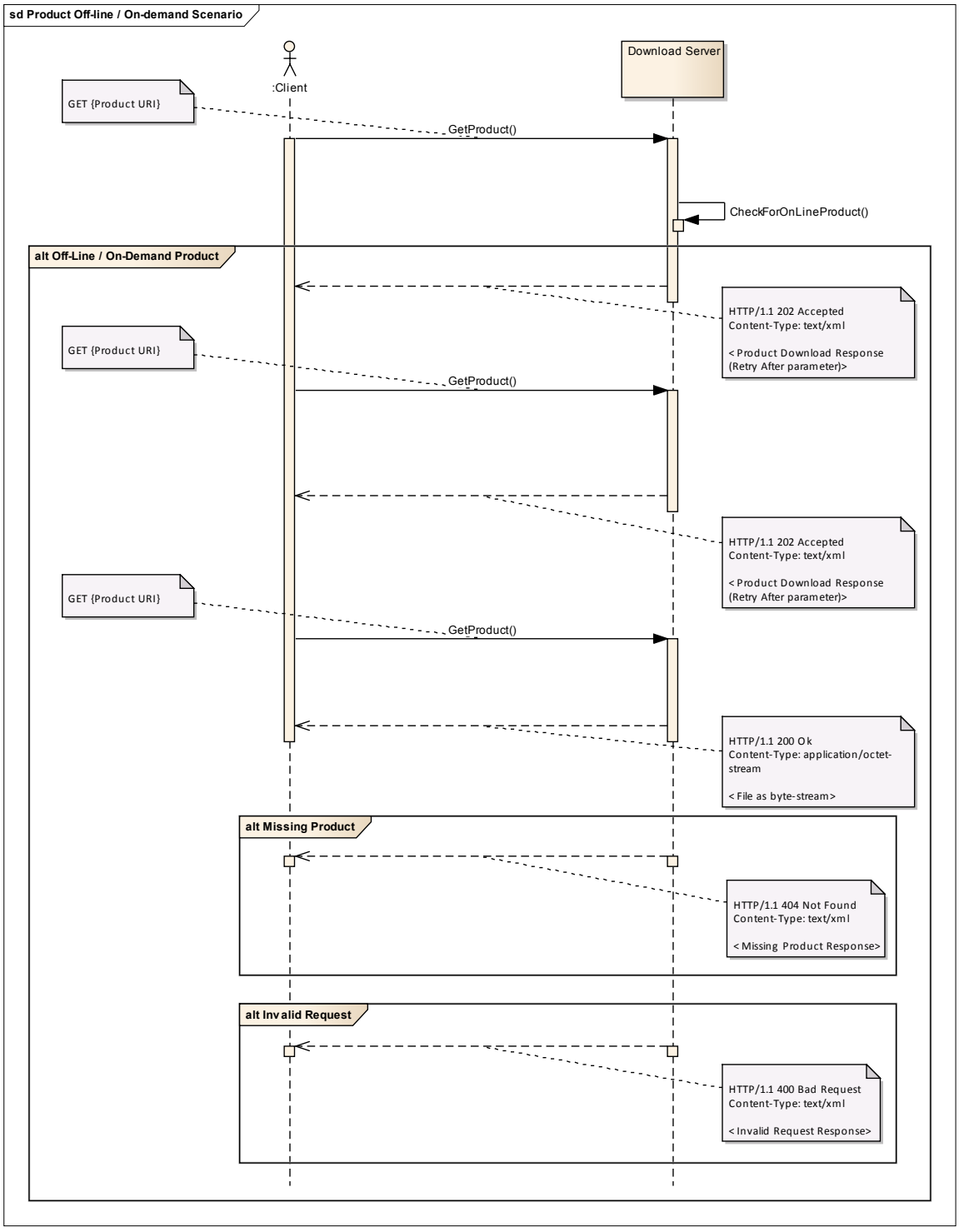


Figure 6-3: Product Off-Line / On-Demand Scenario.

The scenario involves the following entities:

- **Client**: which is the one asking for EO Products;
- **Download Server**: it is the server implementing this specification;

Scenario description:

- The client asks a product with specified URI (**GetProduct**) to the Download Server.
- The Download Server internally checks whether the specified product is online available or not.
- The Download Server returns a Product Download Response XML message informing the client that the request has been accepted, but the product is not ready yet. Also a “retry after” time interval is included as well.
- Later on the Client tries to get the product by calling GetProduct again with the same URL:
    - If the product is not yet available, the Download Server will reply with a new Product Download Response XML message with an updated “retry after” time interval.
    - If the product is ready, the Download Server returns it to the Client.
    - If the product is missing the Download Server returns a Missing Product exception message.
    - If the request is invalid (the submitted request is not well formed, etc.) the Download Server returns a Bad Request exception message.

7. Shared Aspects

7.1 Metalink

7.1.1 Introduction

An EO Product is generally an image of a certain earth surface area. In some cases a product can also include metadata such as: image description, telemetry-related data, sensor information and so on.
All items of a product are usually managed as single files (e.g. ENVISAT products or the ones in the Earth Explorer format stored as single TAR files) and in this way they can be downloaded starting directly from a dedicated Product URI. Special considerations have to be taken into account in the case the product is composed of several files (e.g. in the case of SAFE products) or the product can be downloaded in parallel from different sources. In these cases Metalink files are an appropriate solution.
A Metalink is an xml file that refers to other files by their HTTP URLs. It is a description of a collection of files, each file being potentially another Metalink file referencing another sub-collection of files.
Several clients have already a built-in support for Metalink, but in general to enable download via Metalink a plug-in must be installed.

7.1.2 Multiple-file Product

As per HTTP protocol, from a single URL a client can get a single byte stream, which means that from a single URL only a single file can be retrieved.
In case the download is composed of multiple files, the following options arise:

- The different files are packed together in a single archive file.
- The download is performed via processing a Metalink file referencing the URLs of all items composing the complete download.

The first solution is not suitable when the product files are not already archived in a single file, but must be packed at run time. In this case the Metalink file download solution recommended.

7.1.3 Parallel Download

Another scenario where Metalink files are very useful is when the product files are replicated on different storage locations. In this case the various replica of the same product can be referred from the Metalink and then a client can activate the download in parallel from the various sources improving the download bandwidth and also the reliability of the download process.

7.1.4 Metalink XSD Schema

In this section we briefly describe the Metalink XSD schema focusing on the main aspects of this representation which we use in this document, leaving all the details to its reference [NR14].

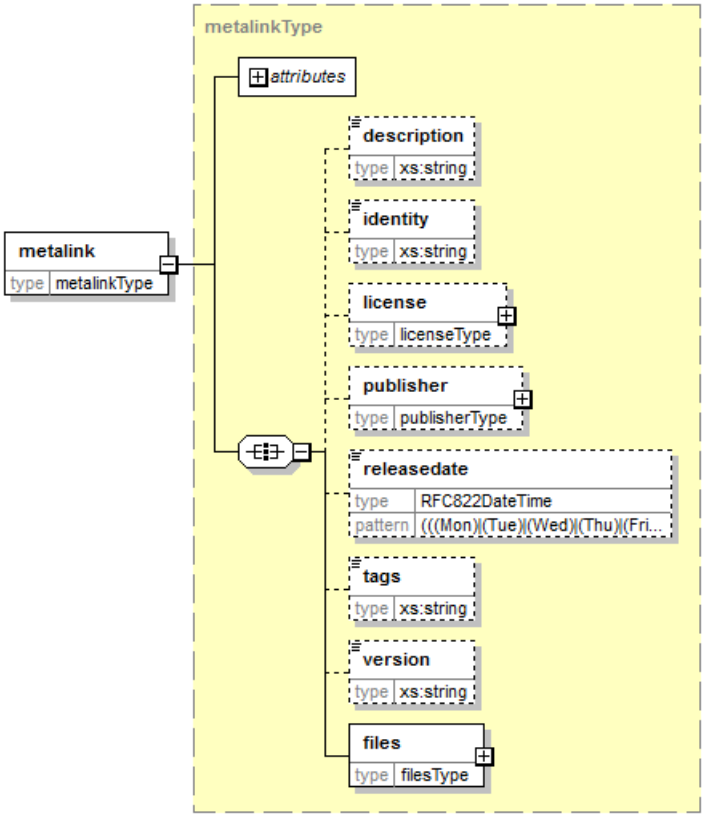


Figure 7-1: Metalink XML Schema diagram.

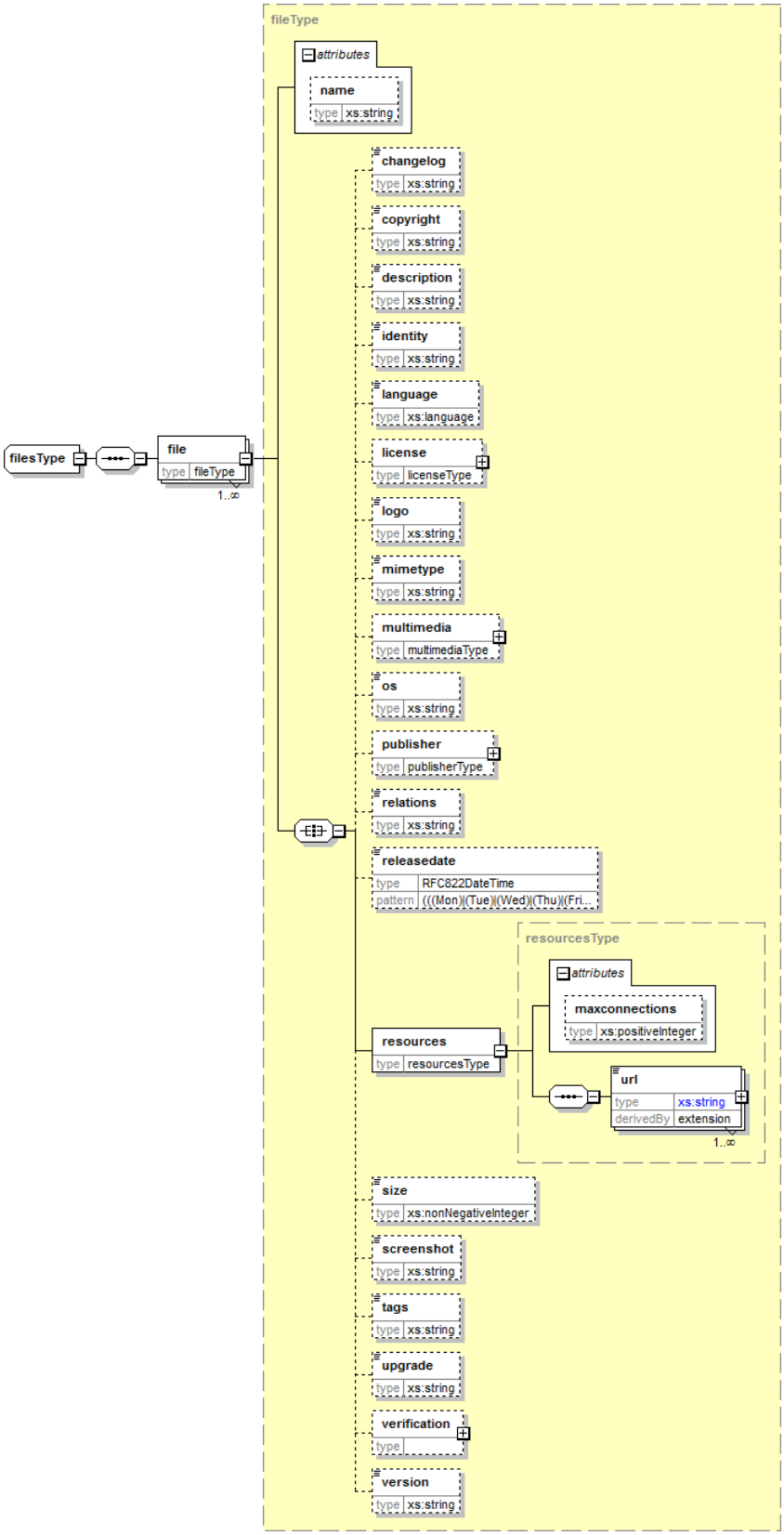


Figure 7-2: Metalink - filesType XML schema diagram.

As shown in Figure 7-1, the Metalink data structure is composed of two major xml elements:

- Files: in this section the Metalink regroups all items composing a product with one file entry for each item. Each file contains one or more resource object type (described below) and a series of elements and attributes better described in [NR14].
- Resources: in this sub-section the Metalink structure describes all the resources that have a local copy of the item-related file. Each resource is described by a URL to the requested file. See [NR14] for details.

7.1.5 Metalink example

In this section we provide a Metalink response example for an EO product composed of two items with multiple sources.

```
<?xml version="1.0" encoding="UTF-8"?>
<metalink xmlns=http://www.metalinker.org/
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:schemaLocation="http://www.metalinker.org/schema/3.0/metalink.xsd">
<files>
  <file name="product.SAFE/manifest.mf">
    <copyright>European Space Agency</copyright>
    <resources>
      <url type="http">http://DSEOServer/product.SAFE?file=manifest</url>
      <url type="http">http://DSEOServer_bis/product.SAFE?file=manifest</url>
    </resources>
  </file>
  <file name="product.SAFE/image.jp2">
    <copyright>European Space Agency</copyright>
    <resources>
      <url type="http">http://DSEOServer/product.SAFE?file=image</url>
      <url type="http">http://DSEOServer_bis/product.SAFE?file=image</url>
    </resources>
  </file>
</files>
</metalink>
```

As shown in the example above we have one EO product called “product.SAFE” which is composed of two items (two file XML elements): “manifest.mf”, a manifest file and “image.jp2” a JPEG2000 image file. Each item is currently available from two different locations (a resource XML element for each file with two url XML element) that makes the file available from two different Download Servers (in this case DSEOServer and DSEOServer_bis).

8. DSEO Operations

DSEO service operations supports Key-Value Pairs (KVP) encoding and then their parameters are set as list of key value pairs over HTTP GET requests.
Each pair is defined using the name of the parameter followed by an equals sign, '=', followed by the value given to the parameter, for example "productURI=http://www.example.org/product1".
In HTTP GET messages, the KVP lists are part of the URL.

8.1 GetCapabilities

The GetCapabilities operation allows clients to retrieve service metadata from a server. The response to a GetCapabilities request shall be an XML document containing service metadata about the server, including specific information about the DSEO server.

This section specifies the request parameters and the XML document that a DSEO server must return to describe its capabilities.

8.1.1 GetCapabilities operation request

The GetCapabilities operation request shall be as specified in Sub-clauses 7.2 and 7.3 of [NR7]. The value of the “service” parameter shall be “DSEO”. The allowed set of service metadata (or Capabilities) XML document section names and meanings shall be as specified in Tables 3 and 7 of [NR7].

The “Multiplicity and use” column in Table 1 of [NR7] specifies the optionality of each listed parameter in the GetCapabilities operation request. The following table specifies the implementation of those parameters by DSEO clients and servers.

Names	Definition	Data types and values	Multiplicity and use
Service	Service type identifier	Character String type, not empty. SHALL be “DSEO"	One (mandatory)
Request	Operation name	Character String type, not empty. SHALL be "GetCapabilities"	One (mandatory)
Accept Version	Prioritized sequence of one or more standard versions accepted by client, with preferred versions listed first	Sequence of Character String type, each not empty Value is list of x.y.z “version” values. SHALL contain "1.0.0"	Zero or one (optional) When omitted, return latest supported version
Sections	Unordered list of zero or more names of requested sections in complete service metadata document	Sequence of Character String type, each not empty Value is list of section names Allowed section names are in Table 18	Zero or one (optional) When omitted or not supported by server, return complete service metadata document

Names	Definition	Data types and values	Multiplicity and use
Update Sequence	Service metadata document version, value is "increased" whenever any change is made in complete service metadata document	Character String type, not empty Values are selected by each server, and are always opaque to clients	Zero or one (optional) When omitted or not supported by server, return latest service metadata document
Accept Formats	Prioritized sequence of zero or more response formats desired by client, with preferred formats listed first.	Sequence of Character String type, each not empty. Value is list of format identifiers Identifiers are MIME types of formats useful for service metadata documents.	Zero or one (optional) When omitted or not supported by server, return service metadata document using MIME type "application/xml"

Table 8-1: Parameters in GetCapabilities operation request.

8.1.1.1 GetCapabilities request example

The following is an example of GetCapabilities request.

```
http://www.dseo.example.org/<context path>?service=DSEO&request=GetCapabilities&version=1.0.0
```

8.1.2 GetCapabilities operation response

The GetCapabilities operation response is based on the ServiceMetadata document. It is the entry point resource that represents the resources available on the service and communication requirements for the service.
The following figure provides a graphical representation of the Capabilities XML document:

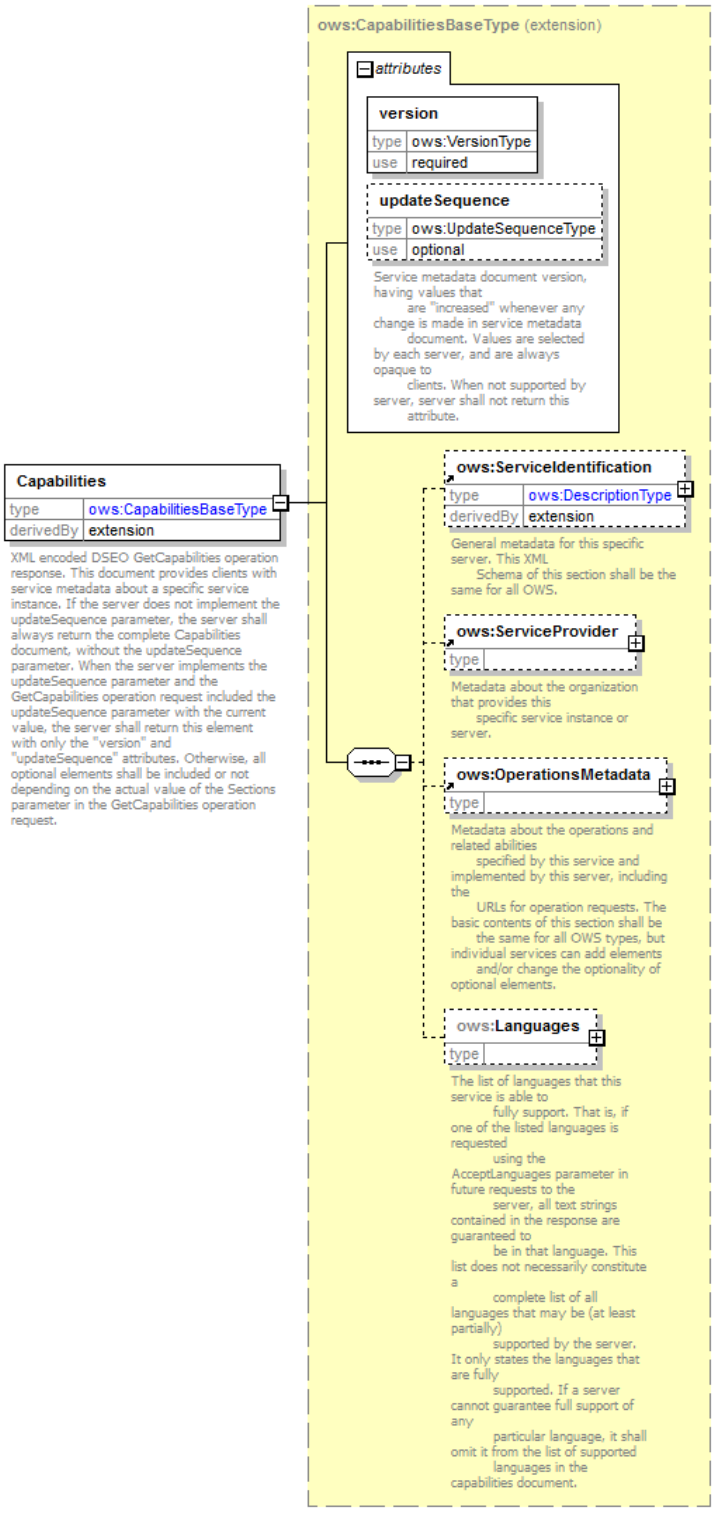


Figure 8-1: GetCapabilities response schema.

Names	Definition	Data type and value	Multiplicity and use
@version	Standard version for operation. It is specified in Table 6 in Subclause 7.4.2 of [NR7].	It shall be set to 1.0.0	1
@updateSequence	Specified in Table 6 in Subclause 7.4.2 of [NR7].	Can be left empty	0..1
ServiceIdentification	Metadata about this specific server. The contents and organization of this section should be the same for all OWSs, as specified in §7.4.4 of [NR7], plus the tailoring specified below.		
ServiceProvider	Metadata about the organization operating this server. The contents and organization of this section should be the same for all OWSs, as specified in §7.4.5 of [NR7].		
OperationsMetadata	Metadata about the operations specified by this service and implemented by this server, including the URLs for operation requests. The basic contents and organization of this section shall be the same for all OWSs, as specified in §7.4.6 of [NR7].	Can be left empty Comment: DSEO specification is operation oriented, then this section must be filled with provided operations: GetCapabilites GetProduct.	0..1
Languages	Languages supported by this server. The contents and organization of this section shall be the same for all OWSs, as specified in §7.4.9 of [NR7]		

Table 8-2: Parameters in GetCapabilities operation response.

8.1.2.1 OperationsMetadata section

The OperationsMetadata section is the same as for all OGC Web Services, as specified in section §7.4.6 and owsOperationsMetadata.xsd of OWS Common [OGC 06-121r3]. The parameters are specified in table Table 8-3:

Name	Value	Meaning of parameter value
Operation.name	GetCapabilities	The GetCapabilities operation is implemented by this server.
	GetProduct	The GetProduct operation is implemented by this server.

Table 8-3: OperationsMetadata section.

The “Name” column uses dot-separator notation to identify parts of a parent item. The “Value” column references an operation parameter, in this case an operation name, and the meaning of including that value is listed in the right column.
The Operation data type allows specifying distributed computing platform (DCP) parameters and the encoding of this DCP as a Constraint within the DCP parameter.

8.1.2.2 GetCapabilities response example

```
<?xml version="1.0" encoding="UTF-8"?>
<Capabilities version="1.0.0" xmlns:ows="http://www.opengis.net/ows/2.0" xmlns="http://www.opengis.net/dseo/1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.opengis.net/dseo/1.0
./dseo.xsd">
        <ows:ServiceProvider>
                <ows:ProviderName>ESA</ows:ProviderName>
                <ows:ProviderSite/>
```

```
                <ows:ServiceContact>
                        <ows:IndividualName>John Smith</ows:IndividualName>
                        <ows:PositionName>EO Operator</ows:PositionName>
                        <ows:ContactInfo>
                                <ows:Phone/>
                        </ows:ContactInfo>
                        <ows:Role codeSpace="http://www.xmlspy.com">String</ows:Role>
                </ows:ServiceContact>
        </ows:ServiceProvider>
        <ows:OperationsMetadata>
                <ows:Operation name="GetCapabilities">
                        <ows:DCP>
                                <ows:HTTP>
                                        <ows:Get/>
                                </ows:HTTP>
                        </ows:DCP>
                </ows:Operation>
                <ows:Operation name="GetProduct">
                        <ows:DCP>
                                <ows:HTTP>
                                        <ows:Get/>
                                </ows:HTTP>
                        </ows:DCP>
                </ows:Operation>
        </ows:OperationsMetadata>
</Capabilities>
```

8.1.3 GetCapabilities response exceptions

When a DSEO Server encounters an error while performing a GetCapabilities operation, it SHALL return an exception-report message as specified in §8 of OWS Common [OGC 06-121r3]. Then it returns an HTTP response including:

- HTTP Status Code: 4XX for errors on the client side; 5XX for errors on server side.
- HTTP Entity Body: ows:ExceptionReport element set as specified in §8 of [NR7].

The following table reports the possible error conditions with the defined HTTP responses.

Error Description	HTTP Error Code	OGC Exception Report		
		"exceptionCode"	**"locator"**	**"ExceptionText"**
Bad Request E.g. an invalid parameter in the request.	400	"BadRequest"	Name of parameter with invalid value	"Invalid value for Parameter"
Internal Server Error An error occurred inside the server while processing the request.	500	"NoApplicableCode"		"Internal Server Error"

Table 8-4: GetCapabilities error conditions.

8.2 GetProduct

The GetProduct operation allows DSEO clients to request a particular EO product identified by an URI.

8.2.1 GetProduct operation request

The following table reports the parameters for sending a GetProduct request:

Names	Definition	Data types and values	Multiplicity and use
Service	Service type identifier	Character String type, not empty SHALL be “DSEO"	One (mandatory)
Request	Operation name	Character String type, not empty SHALL be "GetProduct"	One (mandatory)
Version	Standard version for operation	Character String type, not empty SHALL contain "1.0.0"	One (mandatory)
ProductURI	URI of the product to be downloaded.	URI type, not empty.	One (mandatory)

Table 8-5: Parameters in GetProduct operation request.

8.2.1.1 GetProduct operation request example

http://<hostname>:<port>/<contextpath>?service=DSEO&request=GetProduct&version=1.0.0&ProductURI={RequestedProductURI}

8.2.2 GetProduct operation response

As explained in section §6.2, different responses are possible depending on the availability of the product (on-line / non on line) and on the server capability (Metalink download, forwarded download).
The following table reports the possible responses, all based on standard HTTP protocol:

Response Name	HTTP Status	Response type	MIME Type	Response Format
Direct Download	200 – Ok	Binary	application/octet-stream	Specific to the product file format.
Metalink Download	200 – Ok	XML	application/metalink+xml	Metalink.xsd
Partial Content	206 – Partial Content	Binary	application/octet-stream	
Accepted Download	202 – Accepted	XML	text/xml	dseo.xsd
Forwarded Download	303 – See Other	ASCII	text/html	See http specifications

Table 8-6: GetProduct operation responses.

8.2.2.1 Direct Download response

The EO Product is on-line, either because it is an as-it-is product or because has been moved on-line after a transfer from off-line or on-demand processing has been

completed. It is sent back to the client as a byte-stream directly from the invoked DSEO Server.

8.2.2.2 Metalink Download response

The requested EO Product is composed of several files and, for different reasons (e.g. for performances or bandwidth restrictions), cannot be downloaded as a single file. A Metalink file is generated by the Download Server which provides the product file tree structure and the relevant sub-URI's. The use of parallel download is suggested to retrieve the entire product quickly. Metalink files can also be used by the Download Server to indicate that the product is available from different locations. See §7.1 for further details on Metalink structure and examples.

8.2.2.3 Accepted Download response

The request is valid but the product is still not available for the download. This happens when the product is either off-line or needs on-demand processing.

The following figure provides a graphical representation of the DSEO Accepted Download Response element.

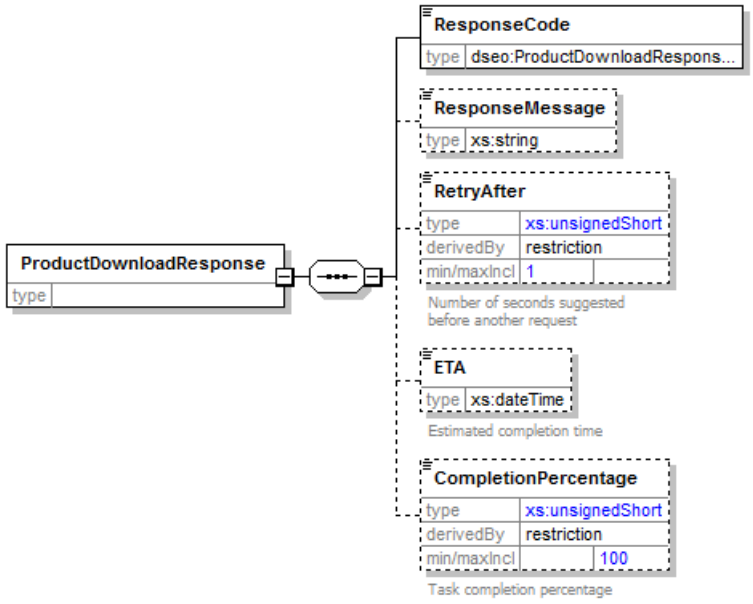


Figure 8-2: GetProduct – Accepted Download response schema.

Names	Definition	Data type and value	Multiplicity and use
ResponseCode	The response code	Type: ProductDownloadResponseType. Allowed values: - ACCEPTED - IN_PROGRESS - SUSPENDED - ERROR	1

Names	Definition	Data type and value	Multiplicity and use
ResponseMessage	Response Message	Type: xs:string.	0..1
RetryAfter	Number of seconds suggested before retry the request.	Type: restricted xs:unsignedShort Allowed values: □ minInclusive value=”1”	0..1
ETA	Estimated completion time	Type: xs:dateTime	0..1
CompletionPercentage	Task completion percentage	Type: restricted xs:unsignedShort Allowed values: □ maxInclusive value=”100”	0..1

Table 8-7: Parameters in GetProduct – Accepted Download operation response.

8.2.2.4 Forwarded Download response

It is the same as in case of Direct Download, but this time the hosting server is not the DSEO Server but another server to which the client is redirected to, following another valid URL.

8.2.2.5 Direct Download response example

```
HTTP/1.1 200 Ok |
Content-Type: application/octet-stream;

010100010....
```

8.2.2.6 Metalink Download response example

```
HTTP/1.1 200 OK |
Content-Type: application/metalink+xml;

<?xml version="1.0" encoding="UTF-8"?>
<metalink xmlns=="http://www.metalinker.org/"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:schemaLocation="http://www.metalinker.org/schema/3.0/metalink.xsd">
  <files>
      <file name="product.SAFE/manifest.mf">
            <copyright>European Space Agency</copyright>
            <resources>
                  <url type="http">http://server/product.SAFE?file=manifest</url>
            </resources>
      </file>
      <file name="product.SAFE/image.j2p">
            <copyright>European Space Agency</copyright>
            <resources>
                  <url type="http">http://server_bis/product.SAFE?file=image"</url>
            </resources>
      </file>
  </files>
</metalink>
```

8.2.2.7 Accepted Download response example

```
HTTP/1.1 202 Accepted |
Content-Type: text/xml;

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

```
<dseo:ProductDownloadResponse
xmlns:dseo=http://www.opengis.net/spec/DSEO/1.0/schema/ProductDownloadResponse
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/spec/DSEO/1.0/schema/ProductDownloadResponse">
        <dseo:ResponseCode>IN_PROGRESS</dseo:ResponseCode>
        <dseo:RetryAfter>30</dseo:RetryAfter>
</dseo:ProductDownloadResponse>
```

8.2.2.8 Forwarded Download response example

```
HTTP/1.1 303 See Other |
Content-Type: text/html;

Location: http://disseminationserveraddress/<ProductURI>
```

8.2.3 GetProduct response exceptions

When a DSEO Server encounters an error while performing the GetProduct operation, it SHALL return an exception report message as specified in §7.4.1 of OWS Common [OGC 06-121r3] and then it returns an HTTP response including:

- HTTP Status Code: 4XX for errors on the client side; 5XX for errors on server side.
- HTTP Entity Body: ows:ExceptionReport element set as specified in §8 of [NR7].

The following table reports the possible error conditions with the defined HTTP responses.

Error Description	HTTP Error Code	OGC Exception Report		
		"exceptionCode"	**"locator"**	**"ExceptionText"**
Bad Request E.g. an invalid parameter in the request.	400	"BadRequest"	Name of parameter with invalid value	"Invalid value for Parameter"
Missing Product An error occurred inside the server while retrieving the product.	404	"MissingProduct"		"Requested product is missing"
Internal Server Error An error occurred inside the server while processing the request.	500	"NoApplicableCode"		"Internal Server Error"
Authentication Fail	401	AuthenticationFailed	"identity_token"	"Invalid or missing identity information"
Authorization Fail	403	AuthorizationFailed	Possible values: "ProductURI"	Text describing the item not authorized.

Table 8-8: GetProduct error conditions.

9. DSEO "Core" Requirement Class

This section reports all the requirements a DSEO Server has to comply with for claiming the conformance with respect to the Core class.

This section has been structured by operations: for each DSEO operation a dedicated sub-section has been prepared reporting all related requirements:

- GetCapabilities requirements have been reported in §9.1.
- GetProduct (On-Line) requirements have been reported in §9.2.

9.1 GetCapabilities

Requirements Class	
http://www.opengis.net/spec/DSEO/1.0/req/Core	
Target type	Download Server
Dependency	**None**
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities** *The DSEO Server shall implement the GetCapabilities operation.*
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Request** *The GetCapabilities request shall consist of a HTTP GET with the following KVP parameters:* - *Service=DSEO* - *Operation=GetCapabilities* - *Version=1.0.0*
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Response** *The GetCapabilities response shall consist of an XML instance document as validated by the entity Capabilities in the dseo.xsd XML Schema.*
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Metadata** *The GetCapabilities response shall contain all information about the operations supported by the DSEO Server.*
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Encoding** *A compliant DSEO Server shall support KVP encoding.* *The returned Capabilities document shall have:* - *Capabilities/Contents/ContentsType supportedEncoding=KVP.*

Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Exceptions** *When a DSEO Server encounters an error while performing a GetCapabilities operation it shall return:* - *HTTP Status Code: 4XX for errors on the client side; 5XX for errors on server side as specified in* Table 8-4. - *HTTP Entity Body: an ows:ExceptionReport according to the clause 8 of [NR7].*

9.2 GetProduct (On-Line)

Requirements Class	
http://www.opengis.net/spec/DSEO/1.0/req/Core	
Target type	Download Server
Dependency	**None**
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct** *The DSEO Server shall implement the GetProduct operation.*
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request** *The GetProduct operation must be invoked using the following mandatory parameters:* - *service=DSEO* - *request=GetProduct* - *version=1.0.0* - *ProductURI={the requested product URI}*
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response** *The DSEO Server shall reply to a correct GetProduct request for on-line product one of the following response types:* - *DirectDownload* - *PartialContent* - *MetalinkDownload* - *ForwardedDownload*
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/DirectDownload** *In case of DirectDownload response, a DSEO server shall return the required file as binary stream.*

Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/PartialContent** *A DSEO server shall support HTTP GET request with RANGE attribute in order to return only the specified trunk of data.*
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/MetalinkDownload** *In case of product composed of multiple files a DSEO server shall return a Metalink file reporting the URI of all related files.*
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/ForwardedDownload** *In case of Forwarded Download the DSEO server shall return an HTTP re-direction to the actual file location.*
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Exceptions** *When a DSEO Server encounters an error while performing a GetProduct request it shall return:* - *HTTP Status Code: 4XX for errors on the client side; 5XX for errors on server side as specified in* Table 8-8. - *HTTP Entity Body: an ows:ExceptionReport according to the clause 8 of [NR7].*

10. DSEO "On-Demand Download" Requirement Class

This section reports all the requirements a DSEO Server has to comply with for claiming the conformance with respect to the On-Demand Download class.

This section has been structured by operations: for each DSEO operation a dedicated sub-section has been prepared reporting all related requirements.

10.1 GetProduct (On-Demand)

Requirements Class	
http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload	
Target type	Download Server
Dependency	**http://www.opengis.net/spec/DSEO/1.0/req/Core**
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response** *In case of non online available product a DSEO server shall return the following response:* - *AcceptedDownload*
Requirement	**http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload** *In case of AcceptedDownload a DSEO server shall return a ProductDownloadResponse element:* - *validated by dseo.xsd schema* - *including ResponseCode* - *including RetryAfter*

11. Annex A (Normative): Conformance Class Abstract Test Suite

This section describes the Abstract Test Suite (ATS) of the

OGC Download Service for Earth Observation Products.

An ATS provides a basis for developing an executable test suite (ETS) to verify that an Implementation Under Test (IUT) conforms to all relevant functional specifications.

A number of Conformance classes have been defined for verifying the various Requirement classes defined in this specification:

- **Core** conformance class, regrouping all tests for verifying Core class requirements;
- **On-demand Download**, regrouping all tests for verifying On-demand Download class requirements.

A conformance class is implemented via a set of Conformance Tests, each testing one or more requirements of the corresponding Requirements Class.

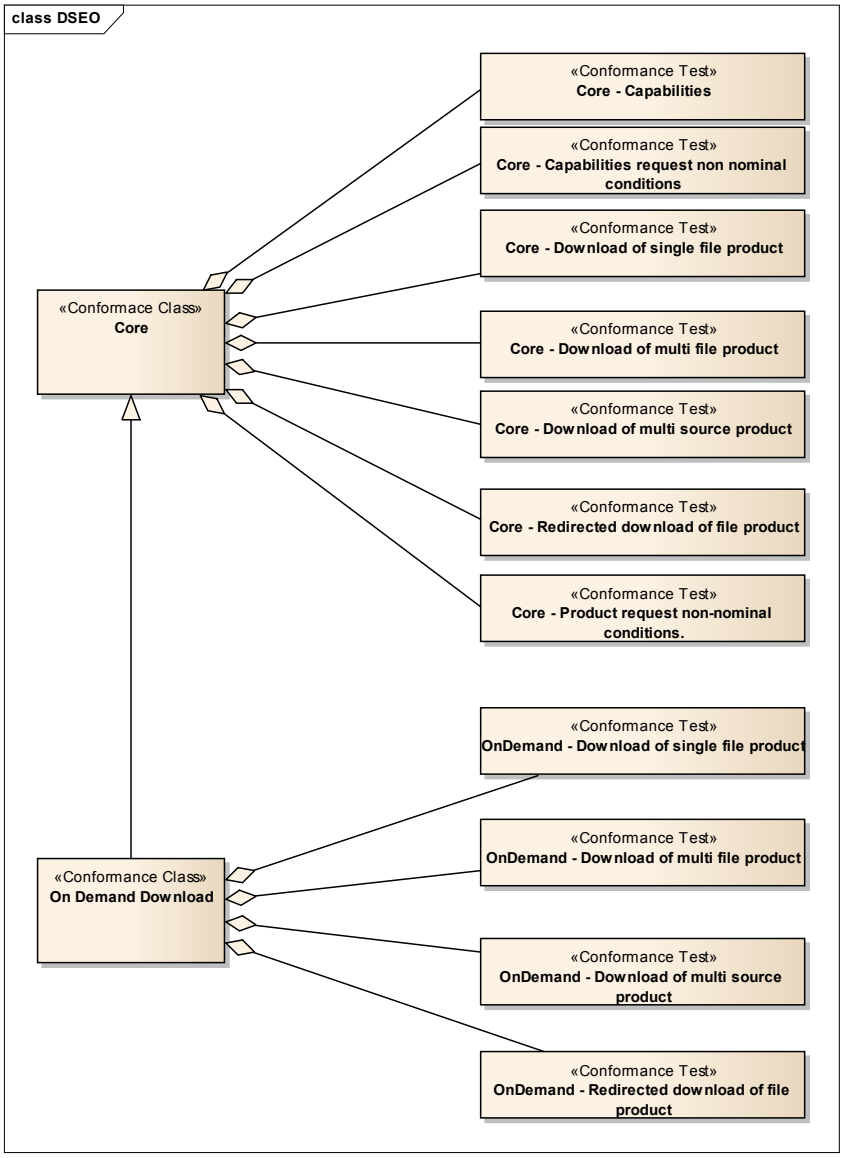


Figure 11-1: Requirement Classes & Abstract Test Suite.

11.1 Conformance Class: Core (http://www.opengis.net/spec/DSEO/1.0/conf/Core)

This Conformance Class is in charge of verifying the compliance of the DSEO server under test with respect to the Core Requirement Class, which includes the basic functions that every DSEO Server shall implement.

11.1.1 Core – Capabilities

a) **Test id**: http://www.opengis.net/spec/DSEO/1.0/conf/Core/GetCapabilities

b) **Test Purpose**: To verify that the DSEO Server under test correctly supports the DSEO GetCapabilities operation.

c) **Test Method**: Verify that the DSEO Server under test accepts a valid DSEO GetCapabilities request with service name "DSEO" with KVP encoding.

Verify that the body of the response message:

- Complies with the Capabilities element of dseo.xsd schema.
- The ows:OperationsMetadata element is filled-in with the list of supported operations.
- The supported encoding (Capabilities/Contents/ContentsType supportedEncoding) is KVPEncoding.

Pass if the assertion is satisfied; fail otherwise.

d) **References**:

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Request
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Response
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Metadata
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Encoding

e) **Test type**: Capability

11.1.2 Core – Capabilities request non-nominal conditions

a) **Test id**: http://www.opengis.net/spec/DSEO/1.0/conf/Core/GetCapabilities-non-nominal

b) **Test Purpose**: Verification that the DSEO Server under test throws a correct error message when incorrect GetCapabilities request is received.

c) **Test Method**: Send an incorrect DSEO GetCapabilities request via KVP HTTP GET without service name. Verify that the response to the request is a HTTP Error including an ows:ExceptionReport according to the clause 8 of [NR7].

d) **References**:

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Request
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Exceptions

e) **Test type**: Capability

11.1.3 Core – Download of single file product

a) **Test id**: http://www.opengis.net/spec/DSEO/1.0/conf/Core/GetProduct-single-file

b) **Test Purpose**: to verify that a DSEO Server complies with all requirements for a GetProduct operation for a single file product download.

c) **Test Method**:

- The Client sends an HTTP GET request including:
  - service=DSEO
  - request=GetProduct
  - version=1.0.0
  - ProductURI={on-line single file product URI}
- Verify that the DSEO Server replies with:
  - HTTP/1.1 response with
    - Status = 200 Ok
    - Content-type: application/octet-stream
    - The HTTP payload contains the binary data representation of the requested product.

d) **References**:

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/DirectDownload

e) **Test Type**: Capability

11.1.4 Core – Download of multi file product

a) **Test id**: http://www.opengis.net/spec/DSEO/1.0/conf/Core/GetProduct-multi-file

b) **Test Purpose**: to verify that a DSEO Server complies with all requirements of a GetProduct operation for a multi file product download.

c) **Test Method**:

- The Client sends an HTTP GET request including:
  - service=DSEO
  - request=GetProduct
  - version=1.0.0
  - ProductURI={on-line multi file product URI}
- Verify that the DSEO Server replies with:
  - HTTP/1.1 response
    - Status = 200 Ok
    - Content-type: application/metalink+xml
    - The HTTP payload contains a Metalink file valid with respect to Metalink.xsd schema definition.
- Verify that the Client parses the Metalink file and sends an HTTP GET message for each item present in Metalink file.
- The DSEO server replies to all requests with:
  - HTTP/1.1 response
    - Status = 200 Ok
    - Content-type: application/octet-stream
    - The HTTP payload contains the binary code representation of the item #N of requested product.

d) **Reference**:

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/PartialContent

e) **Test Type**: Capability

11.1.5 Core – Download of multi source product

a) **Test id**: http://www.opengis.net/spec/DSEO/1.0/conf/Core/GetProduct-multi-source

b) **Test Purpose**: to verify that a DSEO Server complies with all requirements of GetProduct operation for a multi source product download.

c) **Test Method**:

- The Client sends an HTTP GET request including:
  - service=DSEO
  - request=GetProduct
  - version=1.0.0
  - ProductURI={on-line multi source product URI}
- Verify that the DSEO Server replies with:
  - An HTTP/1.1 response
    - Status = 200 Ok
    - Content-type: application/metalink+xml
    - The HTTP payload contains a Metalink file valid against Metalink.xsd schema definition.
- Verify that the Client parses the Metalink file and sends an HTTP GET message for each source defined in the Metalink file including appropriate range for every trunk.
- The DSEO server replies with N:
  - HTTP/1.1 response
    - Status = 206 Partial Content

- Content-type: application/octet-stream
- The HTTP payload contains the binary code representation of the trunk #N of requested file.

d) **Reference**:

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/MetalinkDownload

e) **Test Type**: Capability

11.1.6 Core – Redirected download of single file product

a) **Test id**: http://www.opengis.net/spec/DSEO/1.0/conf/Core/GetProduct-redirect

b) **Test Purpose**: to verify that a DSEO Server complies with all requirements for a GetProduct operation for a redirected download of single file product.

c) **Test Method**:

- The Client sends an HTTP GET request including:
  - service=DSEO
  - request=GetProduct
  - version=1.0.0
  - ProductURI={product located on another server URI}
- Verify that the DSEO Server replies with:
  - HTTP/1.1 response
    - Status = 303 See Other and the alternative location of product on alternative server.
- Verify that the Client sends an HTTP GET message to the suggested alternative server.

- Verify that the Server replies with:
  - HTTP/1.1 response
    - Status = 200 Ok
    - Content-type: application/octet-stream.
    - The HTTP payload contains the binary data representation of the requested file.

d) **Reference**:

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/ForwardedDownload

e) **Test Type**: Capability

11.1.7 Core – Product request non-nominal conditions

a) **Test id**: http://www.opengis.net/spec/DSEO/1.0/conf/Core/GetProduct-non-nominal

b) **Test Purpose**: to verify that the DSEO Server under test throws a correct error message when incorrect GetProduct request is received.

c) **Test Method**: To send an incorrect DSEO GetProduct request with invalid product ID. Verify that the response is a HTTP Error including an ows:ExceptionReport according to the clause 8 of [NR7].

d) **References**:

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Exceptions

e) **Test Type**: Capability

11.2 Conformance Class: On-demand Download (http://www.opengis.net/spec/DSEO/1.0/conf/OnDemandDownload)

11.2.1 OnDemand – Download of single file on-demand product

a) **Test id**: http://www.opengis.net/spec/DSEO/1.0/conf/OnDemandDownload/GetProduct-single-file

b) **Test Purpose**: to verify that the DSEO Server complies with all requirements of GetProduct operation for the download of single file on-demand product.

c) **Test Method**:

- The Client sends an HTTP GET request including:
  - service=DSEO
  - request=GetProduct
  - version=1.0.0
  - ProductURI={on-demand single file product URI}
- Verify that the DSEO Server replies with:
  - An HTTP/1.1 response with
    - Status = 202 Accepted
    - Content-type: text/xml
    - The HTTP payload contains the Product Download Response XML message with retry-after parameter.
- Repeat the test until the DSEO server replies with the required product:
  - An HTTP/1.1 response with
    - Status = 200 Ok
    - Content-type: application/octet-stream
    - The HTTP payload contains the binary code representation of the requested product.

d) **References**:

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/DirectDownload
- http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload

e) **Test Type**: Capability

11.2.2 OnDemand – Download of multi file product

a) **Test id**: http://www.opengis.net/spec/DSEO/1.0/conf/OnDemandDownload/GetProduct-multi-file

b) **Test Purpose**: to verify that the DSEO Server complies with all requirements of GetProduct operation for the download of multi file on-demand product.

c) **Test Method**:

- The Client sends an HTTP GET request including:
  - service=DSEO
  - request=GetProduct
  - version=1.0.0
  - ProductURI={on-demand multi file product URI}
- Verify that the DSEO Server replies with:
  - An HTTP/1.1 response with
    - Status: 202 Accepted
    - Content-type: text/xml
    - The HTTP payload contains the Product Download Response XML message with retry-after parameter.

- The Client repeat the test waiting for the time interval specified in retry-after value until the DSEO server replies with the Metalink file:
  - HTTP/1.1 response with
    - Status = 200 Ok
    - Content-type: application/metalink+xml
    - The HTTP payload contains a Metalink file valid against Metalink.xsd schema definition.
- The Client parses the Metalink file and sends an HTTP GET message for each item present in Metalink file.
- The DSEO server replies to all requests with:
  - HTTP/1.1 response
    - Status = 200 Ok
    - Content-type: application/octet-stream
    - The HTTP payload contains the binary code representation of the item #N of requested product.

d) **Reference**:

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/PartialContent
- http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload

e) **Test Type**: Capability

11.2.3 OnDemand – Download of multi source product

a) **Test id**: http://www.opengis.net/spec/DSEO/1.0/conf/OnDemandDownload/GetProduct-multi-source

b) **Test Purpose**: verify that the DSEO Server complies with all requirements of GetProduct operation for the download of multi source on-demand product.

c) **Test Method**:

- The Client sends an HTTP GET request including:
  - service=DSEO
  - request=GetProduct
  - version=1.0.0
  - ProductURI={an on-demand multi source product URI}
- Verify that the DSEO Server replies with:
  - HTTP/1.1 response
    - Status = 202 Accepted
    - Content-type: text/xml
    - The HTTP payload contains the Product Download Response XML message with retry-after parameter.
- The Client waits for the time interval equal to specified retry-after value and repeat the request until the DSEO Server replies with:
  - HTTP/1.1 response
    - Status = 200 Ok
    - Content-type: application/metalink+xml
    - The HTTP payload contains a Metalink file valid against Metalink.xsd schema definition.
- The Client parses the Metalink file and sends an HTTP GET message for each source in the Metalink file with appropriate range for every trunk.
- The DSEO Server replies with:
  - HTTP/1.1 response
    - Status = 206 Partial Content

- Content-type: application/octet-stream
- The HTTP payload contains the binary code representation of the trunk #N of requested file.

d) **Reference**:

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/MetalinkDownload
- http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload

e) **Test Type**: Capability

11.2.4 OnDemand – Redirected download of single file product

a) **Test id**: http://www.opengis.net/spec/DSEO/1.0/conf/OnDemandDownload/GetProduct-redirect

b) **Test Purpose**: to verify that a DSEO Server satisfies all requirements of GetProduct operation for redirected download of single file on-demand product.

c) **Test Method**:

- The Client sends an HTTP GET request including:
  - service=DSEO
  - request=GetProduct
  - version=1.0.0
  - ProductURI={on-demand product located on another server URI}
- Verify that the DSEO Server replies with:
  - An HTTP/1.1 response

- Status = 202 Accepted
- Content-type: text/xml
- The HTTP payload contains the Product Download Response XML message with retry-after parameter.

- The Client waits the time interval equal to specified retry-after value and then repeats the request.
- The DSEO Server replies with:
  - An HTTP/1.1 response with Status = 303 See Other and the alternative location of product on alternative server.
- The Client sends an HTTP GET message to the suggested alternative server.
- The alternative server replies with:
  - HTTP/1.1 response
    - Status = 200 Ok
    - Content-type: application/octet-stream.
    - The HTTP payload contains the binary code representation of the requested file.

d) **Reference**:

- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/ForwardedDownload
- http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response
- http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload

e) **Test Type**: Capability

11.3 Traceability Matrix

11.3.1 Requirements vs. Conformance Tests Traceability Matrix

Requirement URL	Conformance Test
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities	11.1.1 Core – Capabilities
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities	11.1.2 Core – Capabilities request non-nominal conditions
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Request	11.1.1 Core – Capabilities
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Request	11.1.2 Core – Capabilities request non-nominal conditions
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Response	11.1.1 Core – Capabilities
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Metadata	11.1.1 Core – Capabilities
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Encoding	11.1.1 Core – Capabilities
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Exceptions	11.1.2 Core – Capabilities request non-nominal conditions
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct	11.1.3 Core – Download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct	11.1.4 Core – Download of multi file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct	11.1.5 Core – Download of multi source product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct	11.1.6 Core – Redirected download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct	11.1.7 Core – Product request non-nominal conditions
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct	11.2.1 OnDemand – Download of single file on-demand product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct	11.2.2 OnDemand – Download of multi file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct	11.2.3 OnDemand – Download of multi source product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct	11.2.4 OnDemand – Redirected download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request	11.1.3 Core – Download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request	11.1.4 Core – Download of multi file product

Requirement URL	Conformance Test
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request	11.1.5 Core – Download of multi source product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request	11.1.6 Core – Redirected download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request	11.1.7 Core – Product request non-nominal conditions
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request	11.2.1 OnDemand – Download of single file on-demand product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request	11.2.2 OnDemand – Download of multi file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request	11.2.3 OnDemand – Download of multi source product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request	11.2.4 OnDemand – Redirected download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response	11.1.3 Core – Download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response	11.1.4 Core – Download of multi file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response	11.1.5 Core – Download of multi source product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response	11.1.6 Core – Redirected download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response	11.2.1 OnDemand – Download of single file on-demand product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response	11.2.2 OnDemand – Download of multi file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response	11.2.3 OnDemand – Download of multi source product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response	11.2.4 OnDemand – Redirected download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/DirectDownload	11.1.3 Core – Download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/DirectDownload	11.2.1 OnDemand – Download of single file on-demand product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/PartialContent	11.1.4 Core – Download of multi file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/PartialContent	11.2.2 OnDemand – Download of multi file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/MetalinkDownload	11.1.5 Core – Download of multi source product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/MetalinkDownload	11.2.3 OnDemand – Download of multi source product

Requirement URL	Conformance Test
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/ForwardedDownload	11.1.6 Core – Redirected download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/ForwardedDownload	11.2.4 OnDemand – Redirected download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Exceptions	11.1.7 Core – Product request non-nominal conditions
http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response	11.2.1 OnDemand – Download of single file on-demand product
http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response	11.2.2 OnDemand – Download of multi file product
http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response	11.2.3 OnDemand – Download of multi source product
http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response	11.2.4 OnDemand – Redirected download of single file product
http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload	11.2.1 OnDemand – Download of single file on-demand product
http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload	11.2.2 OnDemand – Download of multi file product
http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload	11.2.3 OnDemand – Download of multi source product
http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload	11.2.4 OnDemand – Redirected download of single file product

Table 11-1: Requirements vs. Conformance Tests Traceability Matrix.

11.3.2 Conformance Tests vs. Requirements Traceability Matrix

Conformance Test	Requirement URL
11.1.1 Core – Capabilities	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities
11.1.1 Core – Capabilities	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Request
11.1.1 Core – Capabilities	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Response
11.1.1 Core – Capabilities	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Metadata
11.1.1 Core – Capabilities	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Encoding
11.1.2 Core – Capabilities request non-nominal conditions	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities

Conformance Test	Requirement URL
11.1.2 Core – Capabilities request non-nominal conditions	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Request
11.1.2 Core – Capabilities request non-nominal conditions	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetCapabilities/Exceptions
11.1.3 Core – Download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
11.1.3 Core – Download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
11.1.3 Core – Download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
11.1.3 Core – Download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/DirectDownload
11.1.4 Core – Download of multi file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
11.1.4 Core – Download of multi file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
11.1.4 Core – Download of multi file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
11.1.4 Core – Download of multi file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/PartialContent
11.1.5 Core – Download of multi source product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
11.1.5 Core – Download of multi source product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
11.1.5 Core – Download of multi source product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
11.1.5 Core – Download of multi source product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/MetalinkDownload
11.1.6 Core – Redirected download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
11.1.6 Core – Redirected download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
11.1.6 Core – Redirected download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response

Conformance Test	Requirement URL
11.1.6 Core – Redirected download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/ForwardedDownload
11.1.7 Core – Product request non-nominal conditions	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
11.1.7 Core – Product request non-nominal conditions	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
11.1.7 Core – Product request non-nominal conditions	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Exceptions
11.2.1 OnDemand – Download of single file on-demand product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
11.2.1 OnDemand – Download of single file on-demand product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
11.2.1 OnDemand – Download of single file on-demand product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
11.2.1 OnDemand – Download of single file on-demand product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/DirectDownload
11.2.1 OnDemand – Download of single file on-demand product	http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response
11.2.1 OnDemand – Download of single file on-demand product	http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload
11.2.2 OnDemand – Download of multi file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
11.2.2 OnDemand – Download of multi file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
11.2.2 OnDemand – Download of multi file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
11.2.2 OnDemand – Download of multi file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/PartialContent
11.2.2 OnDemand – Download of multi file product	http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response
11.2.2 OnDemand – Download of multi file product	http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload
11.2.3 OnDemand – Download of multi source product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct

Conformance Test	Requirement URL
11.2.3 OnDemand – Download of multi source product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
11.2.3 OnDemand – Download of multi source product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
11.2.3 OnDemand – Download of multi source product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/MetalinkDownload
11.2.3 OnDemand – Download of multi source product	http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response
11.2.3 OnDemand – Download of multi source product	http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload
11.2.4 OnDemand – Redirected download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct
11.2.4 OnDemand – Redirected download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Request
11.2.4 OnDemand – Redirected download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response
11.2.4 OnDemand – Redirected download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/Core/GetProduct/Response/ForwardedDownload
11.2.4 OnDemand – Redirected download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response
11.2.4 OnDemand – Redirected download of single file product	http://www.opengis.net/spec/DSEO/1.0/req/OnDemandDownload/GetProduct/Response/AcceptedDownload

Table 11-2: Conformance Tests vs. Requirements Traceability Matrix.

12. Annex B (Normative): XML Schema Documents

In addition to this document, this BP includes several normative XML Schema Documents. These XML Schema Documents are bundled in a zip file with the present document. After OGC acceptance of a Version 1.0.0 of this BP, these XML Schema Documents will also be posted online at the URL http://schemas.opengeospatial.net/oseop/1.0. In the event of a discrepancy between the bundled and online versions of the XML Schema Documents, the online files shall be considered authoritative.

The DSEO protocol specified in this document uses the XML Schema Document included in the zip file with this document. This XML Schema Document combines the XML schema fragments listed in various sub-clauses of this document, eliminating duplications. This XML Schema Document is named:

- dseo.xsd

This XML Schema Document uses and builds on the following XML Schema Documents:

- ows/2.0/owsAll.xsd
- http://www.metalinker.org/schema/3.0/metalink.xsd

13. Annex C: Revision history

Date	Release	Author	Paragraph modified	Description
2013-05-14	**0.9.1**	**R. Cafini**	**All**	**First issue**

14. Annex D: Bibliography