Ordering Services for Earth Observation Products

Copyright notice
Copyright © 2007 Open Geospatial Consortium, Inc. All Rights Reserved. To obtain additional rights of use, visit http://www.opengeospatial.org/legal/.

Warning
This document defines an OGC Best Practices position on a particular technology or approach related to an OGC standard. This document is not an OGC Standard and may not be referred to as an OGC Standard. It is subject to change without notice. However, this document is an official position of the OGC membership on this particular technology topic.
Contents

1 SCOPE .................................................................................................................................................. 20

2 CONFORMANCE .................................................................................................................................... 21

3 REFERENCES ......................................................................................................................................... 22

3.1 NORMATIVE REFERENCES ............................................................................................................ 22

3.2 OTHER REFERENCES ..................................................................................................................... 22

4 TERMS AND DEFINITIONS .................................................................................................................. 24

5 SYMBOLS AND ABBREVIATIONS ........................................................................................................ 28

5.1 SYMBOLS (AND ABBREVIATED TERMS) ....................................................................................... 28

5.2 UML NOTATION .............................................................................................................................. 29

5.2.1 UML Class Diagrams ..................................................................................................................... 29

5.2.2 UML Sequence Diagrams ............................................................................................................. 30

5.3 XML NOTATION .............................................................................................................................. 31

5.4 DOCUMENT TERMS AND DEFINITIONS ...................................................................................... 32

6 SYSTEM CONTEXT ............................................................................................................................... 33

6.1 APPLICATION DOMAIN ................................................................................................................. 33

6.2 ESSENTIAL USE-CASES ................................................................................................................ 34

6.2.1 Ordering from catalogue of EO Products ....................................................................................... 34

6.2.2 Order of Future Products derived from tasking requests.......................................................... 36

6.2.3 Subscribe to EO Products ............................................................................................................ 37

7 INFORMATION MODELS .................................................................................................................... 40

7.1 INFORMATION MODEL FOR EO PRODUCT ORDERING ................................................................ 40

7.1.1 XML schema approach ................................................................................................................... 41

7.1.2 Order Options ............................................................................................................................. 41

7.1.2.1 CommonOrderOptionsType ....................................................................................................... 41

7.1.2.2 ParameterDescriptorType ......................................................................................................... 44

7.1.2.3 sps:Parameter ............................................................................................................................. 46

7.1.2.4 Preliminary and extensible list of ordering options .................................................................... 47

7.1.3 Order Specification ....................................................................................................................... 48

7.1.3.1 DeliveryInformationType ........................................................................................................... 51

7.1.3.2 FTPAddressType ....................................................................................................................... 52

7.1.3.3 DeliveryAddressType ............................................................................................................... 53

7.1.4 Order Item ..................................................................................................................................... 54

7.1.4.1 CommonOrderItemType .......................................................................................................... 54

7.1.4.2 SceneSelection Type ................................................................................................................ 58

7.1.5 Order Quotation .......................................................................................................................... 59

7.1.6 Order Monitoring Specification .................................................................................................. 63

7.1.6.1 CommonOrderMonitorSpecification ....................................................................................... 64

7.1.7 Order Item Monitoring Specification ........................................................................................ 66

7.1.7.1 CommonOrderStatusItemType .............................................................................................. 67

8 EXTERNAL INTERFACES .................................................................................................................... 69

8.1 INTERFACE SPECIFICATIONS ....................................................................................................... 72

8.1.1 Shared parameters ........................................................................................................................ 72

8.1.1.1 statusNotification element ..................................................................................................... 72

8.1.2 GetCapabilities Operation ........................................................................................................... 72

8.1.2.1 GetCapabilities input message: GetCapabilities element .................................................. 73

8.1.2.2 GetCapabilities output message: Capabilities element ....................................................... 74

8.1.2.2.1 OperationsMetadata section standard contents ................................................................. 76

8.1.2.2.2 Contents section ................................................................................................................... 77

8.1.2.3 Exceptions .............................................................................................................................. 78
8.3 Other Issues

8.3.1 Distributed Orders implementation............................................................................ 110
8.3.1.1 Get Options scenario.............................................................................................. 111
8.3.1.2 Get Quotation scenario.......................................................................................... 112
8.3.1.3 Submit scenario..................................................................................................... 113
8.3.1.4 Status notification scenario.................................................................................... 114
8.3.1.5 Get Status scenario.............................................................................................. 115
8.3.1.6 Cancel scenario.................................................................................................... 116
8.3.2 Semantic issues......................................................................................................... 118
8.3.3 Technical issues......................................................................................................... 118
8.3.4 Other Issues............................................................................................................... 118
Figures

Figure 5-1: UML Class Diagram notations................................................................. 29
Figure 5-2: UML Sequence Diagrams Notations.................................................... 30
Figure 6-1: Sequence of steps generally performed for ordering products from EO Catalogue......35
Figure 6-2: Sequence of steps generally performed for subscribing to EO products. ............38
Figure 7-1: CommonOrderOptionsType diagram.................................................... 42
Figure 7-2: ParameterDescriptorType diagram.................................................... 45
Figure 7-3: sps:Parameter diagram........................................................................... 47
Figure 7-4: CommonOrderSpecification diagram.................................................. 49
Figure 7-5: OrderSpecification diagram............................................................... 50
Figure 7-6: DeliveryInformationType diagram..................................................... 51
Figure 7-7: FTPAddressType diagram................................................................. 52
Figure 7-8: DeliveryAddressType diagram............................................................ 53
Figure 7-9: CommonOrderItemType diagram...................................................... 55
Figure 7-10: SceneSelectionType diagram............................................................ 58
Figure 7-11: OrderQuotation diagram................................................................. 60
Figure 7-12: CommonOrderMonitorSpecification diagram................................. 64
Figure 7-13: CommonOrderStatusItemType diagram........................................... 67
Figure 8-1: Asynchronous requests scenario....................................................... 71
Figure 8-2: GetCapabilities request diagram..................................................... 73
Figure 8-3: Capabilities diagram............................................................................ 75
Figure 8-4: GetOptions element diagram............................................................ 80
Figure 8-5: GetOptionsResponse diagram......................................................... 82
Figure 8-6: GetQuotation diagram........................................................................ 85
Figure 8-7: GetQuotationAck element diagram.................................................. 87
Figure 8-8: GetQuotationResponse diagram....................................................... 89
Figure 8-9: GetQuotationResponseAck diagram................................................ 90
Figure 8-10: Submit diagram................................................................................ 92
Figure 8-11: SubmitAck diagram......................................................................... 93
Figure 8-12: SubmitResponse diagram............................................................... 95
Figure 8-13: SubmitResponseAck diagram. ................................................................. 96
Figure 8-14: GetStatus diagram. ............................................................................. 98
Figure 8-15: GetStatusResponse element diagram. ............................................. 100
Figure 8-16: DescribeResultAccess diagram. ....................................................... 102
Figure 8-17: DescribeResultAccessResponse diagram .......................................... 103
Figure 8-18: Cancel diagram. ................................................................................. 106
Figure 8-19: CancelAck diagram. .......................................................................... 106
Figure 8-20: CancelResponse diagram. ................................................................. 109
Figure 8-21: CancelResponseAck diagram. .......................................................... 109
Figure 8-22: Get Options Scenario. ................................................................. 111
Figure 8-23: Get Quotation Scenario. ............................................................... 112
Figure 8-24: Submit Scenario. ............................................................................. 113
Figure 8-25: Update Status Scenario. ............................................................... 114
Figure 8-26: Get Status Scenario. ................................................................. 115
Figure 8-27: Cancel Scenario. ................................................................. 116
Figure 8-28: Retrieval of on-line available data scenario .................................... 117
Tables

Table 7-1: CommonOrderOptionsType description. ................................................................. 43
Table 7-2: ParameterDescriptorType description. ................................................................. 46
Table 7-3: sps:Parameter description..................................................................................... 47
Table 7-4: Order options extensible list................................................................................ 48
Table 7-5: CommonOrderSpecification description. ............................................................ 50
Table 7-6: OrderSpecification description............................................................................. 51
Table 7-7: DeliveryInformationType description. ................................................................. 52
Table 7-8: FTPAddressType description................................................................................ 52
Table 7-9: DeliveryAddressType description......................................................................... 54
Table 7-10: CommonOrderItemType description................................................................. 57
Table 7-11: SceneSelectionType description......................................................................... 59
Table 7-12: OrderQuotation description............................................................................... 61
Table 7-13: OrderItemGroupPrice description.................................................................... 62
Table 7-14: OrderItemPrice description............................................................................... 63
Table 7-15: CurrencyType description.................................................................................. 63
Table 7-16: CommonOrderMonitorSpecification description............................................. 65
Table 7-17: CommonOrderStatusItemType description....................................................... 68
Table 8-1: Operation request encoding................................................................................. 69
Table 8-2: Implementation of parameters in GetCapabilities operation request................ 74
Table 8-3: Section name values and content........................................................................ 76
Table 8-4: Mandatory Order Service operations.................................................................. 76
Table 8-5: Optional Order Service operations....................................................................... 77
Table 8-6: Description of Contents section of Capabilities document.................................. 78
Table 8-7: GetOptions element description.......................................................................... 81
Table 8-8: GetOptionsResponse description......................................................................... 83
Table 8-9: GetQuotation description..................................................................................... 86
Table 8-10: GetQuotationAck description............................................................................. 88
Table 8-11: GetQuotationResponse description................................................................. 90
Table 8-12: GetQuotationResponseAck description............................................................. 90
Table 8-13: Submit description................................................................................................. 93
Table 8-14: SubmitAck description.......................................................................................... 94
Table 8-15: SubmitResponse description.................................................................................. 96
Table 8-16: SubmitResponseAck description.......................................................................... 96
Table 8-17: GetStatus description........................................................................................... 99
Table 8-18: GetStatusResponse description............................................................................ 101
Table 8-19: DescribeResultAccess description..................................................................... 102
Table 8-20: DescribeResultAccessResponse description....................................................... 104
Table 8-21: DescribeResultAccessResponse description....................................................... 105
Table 8-22: Cancel description............................................................................................... 106
Table 8-23: CancelAck description........................................................................................ 107
Table 8-24: CancelResponse description................................................................................ 109
Table 8-25: CancelResponseAck description........................................................................... 110
i. Preface

This best practices document describes a profile to order Earth Observation data products. This document expands on the work presented in “Best Practices for Earth Observation Products” OGC-05-057r4, separating the order services from the catalogue services which have been presented in 06-079 and in 06-131. The final goal being to agree to a coherent set of interfaces for ordering 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.

ii. Submitting organisations

The following organisations will submit the original document or its revisions to the OGC™ Ordering WG:

- ESA – European Space Agency
- Spacebel s.a.
- EUSC
- Spot Image

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:

- ASI
- CNES
- DLR
- Eumetsat
- MDA

Note: 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:

<table>
<thead>
<tr>
<th>Contact</th>
<th>Company</th>
<th>Email</th>
</tr>
</thead>
<tbody>
<tr>
<td>Daniele Marchionni</td>
<td>ELSAG DATAMAT</td>
<td>Daniele.marchionni &lt;at&gt; elsagdatamat.com</td>
</tr>
<tr>
<td>Jolyon Martin</td>
<td>ESA</td>
<td>Jolyon.Martin &lt;at&gt; esa.int</td>
</tr>
</tbody>
</table>

iv. Revision history

<table>
<thead>
<tr>
<th>Date</th>
<th>Internal version</th>
<th>Editor</th>
<th>Primary clauses modified</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>21/06/2006</td>
<td>1.0.0 Draft</td>
<td>D. Marchionni</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td>Internal version</td>
<td>Editor</td>
<td>Primary clauses modified</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>------------------</td>
<td>-----------------------</td>
<td>------------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>26/06/2006</td>
<td>1.0.1 Draft</td>
<td>D. Marchionni</td>
<td></td>
<td>Updated title and XSD</td>
</tr>
<tr>
<td>11/08/2006</td>
<td>1.0.2 Draft</td>
<td>D. Marchionni</td>
<td></td>
<td>Updated all sections.</td>
</tr>
<tr>
<td>23/08/2006</td>
<td>1.0.3 Draft</td>
<td>D. Marchionni</td>
<td></td>
<td>Updated all sections.</td>
</tr>
<tr>
<td>25/08/2006</td>
<td>1.0.4 Draft</td>
<td>D. Marchionni</td>
<td></td>
<td>Updated all sections.</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>§7.1.3.1: updated definition adding scenePolygon and albumExtract fields</td>
<td>HMA CDR RID: CDR#02</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Added Subscription scenario in §6.1, §6.2.1; §7.1.1: Added optionType element to: ProductServiceOptionType\productOrderOptions\option §7.1.1: optionValueDefinition has been updated to optional §7.1.2: priority made optional §7.1.3: updated optionSelectedValues element adding time and polygon values; ProductOrderItemType\productId\identifier has been made optional §8.1.3, §8.1.3.1: updated description. §8.1.5: updated description §8.1.6.2: added optional priority element; §8.1.6.2.1: removed OrderStatusItemType\sceneCentre because not needed; OrderStatusItemType\productId\identifier has been made optional; OrderStatusItemType\options updated with the new definition §8.1.7: updated description of the operation.</td>
<td>HMA CDR RID: CDR#10</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>§8.1.6.2.1: clarified that the status can be left empty when no information are available.</td>
<td>HMA CDR RID: CDR#14</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Added time stamp to each request messages apart GetCapabilities. Updated §8.1.3.1, §8.1.4.1, §8.1.5.1, §8.1.6.1, §8.1.7.1, §8.1.8.1</td>
<td>HMA CDR RID: CDR#33</td>
</tr>
<tr>
<td>Date</td>
<td>Internal version</td>
<td>Editor</td>
<td>Primary clauses modified</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>------------------</td>
<td>-----------------</td>
<td>------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>§7.1.1: added identifier element in ProductServiceOptionType\productOrderOptions\option</td>
<td>HMA CDR RID: CDR#53</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>§8.1.3.1: added identifier element in the GetOptions request message.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>§8.1.3.2: added status and errorMessage element to the response message</td>
<td></td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>§7.1.1: updated definition removing the ProductServiceOptions level</td>
<td>HMA CDR RID: CDR#54</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>§81.3.2: same as previous one</td>
<td></td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Added §5.3</td>
<td>HMA CDR RID: CDR#70</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated §7.1.1, §7.1.3, §8.1.3.2</td>
<td>HMA CDR RID: CDR#71</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>§7.1.3: updated description of identifier element.</td>
<td>HMA CDR RID: CDR#72</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>See CDR#02</td>
<td>HMA CDR RID: CDR#73</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>§81.6: updated description of the operation.</td>
<td>HMA CDR RID: CDR#75</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>§8.1.6.2, §8.1.6.2: added missionSpecificStatusInfo element in OrderStatusItemType.</td>
<td>HMA CDR RID: CDR#78</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated §7.1.2.1: added e-mail field</td>
<td>HMA CDR RID: CDR#80</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated §8.1.5.2</td>
<td>HMA CDR Comment: the orderId has to be made mandatory and set to a special value in case of failure.</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated §8.1.5, §8.1.5.1</td>
<td>HMA CDR Comment: quotationId (s) can be optionally provided in Submit</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>§8.1.6</td>
<td>Clarified the purpose of GetStatus operation.</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>§8.1.6.2: status element references the global element</td>
<td>The status element of GetStatusResponse to refer the already defined global element.</td>
</tr>
<tr>
<td>Date</td>
<td>Internal version</td>
<td>Editor</td>
<td>Primary clauses modified</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>------------------</td>
<td>-------------------</td>
<td>------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated §7.1.3.1, Appendix A</td>
<td>Defined type (xs:dateTime) for: orderItem\sceneSelection\TemporalSelection\startTime &amp; endTimeTime &amp; endDateTime</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated §7.1.3, Appendix A</td>
<td>Defined type (xs:string) for: orderItem\qualityOfService</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated §7.1.2.1, Appendix A</td>
<td>Defined type (xs:string) for: orderSpecification\deliveryInformation\ftp-pull &amp; ftp-push</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated §8.1.6.2.1, Appendix A</td>
<td>Defined type (xs:string) for OrderMonitorResponse \ orderSpecification \ orderItem \ deliveryMethod &amp; qualityOfService</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Appendix A</td>
<td>Removed unused types</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Appendix A</td>
<td>Updated schema</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Appendix B</td>
<td>Updated WSDL Updated also SOAP Action</td>
</tr>
<tr>
<td>21/11/2006</td>
<td>1.1.0 Draft</td>
<td>D. Marchionni</td>
<td>Appendix C</td>
<td>Updated examples</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated request message of all operations.</td>
<td>To add version in requests.</td>
</tr>
<tr>
<td>Date</td>
<td>Internal version</td>
<td>Editor</td>
<td>Primary clauses modified</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>------------------</td>
<td>---------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Option element within GetOptionResponse has been redefined using sps &amp; swe types. Options element within orderItem has been redefined using the sps &amp; swe types. Quality of service element is now managed with this dynamic list of options (both in GetOptions and in OrderSpecification).</td>
<td>HMA AR RID: AR#11</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated definition of OrderOptionsResponseType, SubmitProductOrderRequestType, GetStatusResponseType, GetQuotationRequestType, UpdateStatusRequestType</td>
<td>HMA AR RID: AR#12</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Added new operation for sending quotation. Removed delivery of quotation by e-mail. Added payments information, in order options and order submit. Moved orderAccount from order specification to order item.</td>
<td>HMA AR RID: AR#14</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated table 8-1</td>
<td>HMA AR RID: AR#15</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Added table 7-2</td>
<td>HMA AR RID: AR#20</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated qualityOfService element within the GetOptionsResponse. qualityOfService moved to the list of dynamic options.</td>
<td>HMA AR RID: AR#21</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>All strings have been bounded.</td>
<td>HMA AR RID: AR#22</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Added fax number (18 chars)</td>
<td>HMA AR RID: AR#23</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Added parameter in table 7-2</td>
<td>HMA AR RID: AR#24</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Added parameter in table 7-2</td>
<td>HMA AR RID: AR#25</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Options definition performed via sps &amp; swe types. Updated example files including the band selection option for colour composition</td>
<td>HMA AR RID: AR#26</td>
</tr>
<tr>
<td>Date</td>
<td>Internal version</td>
<td>Editor</td>
<td>Primary clauses modified</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>------------------</td>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Options definition performed via sps &amp; swe types. Updated example files including sub-band selection option.</td>
<td>HMA AR RID: AR#27</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Added Rectangle for scene selection options, but used GML type.</td>
<td>HMA AR RID: AR#28</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>None</td>
<td>HMA AR RID: AR#49</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated Order Id definition.</td>
<td>HMA AR RID: AR#50</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>None</td>
<td>HMA AR RID: AR#53</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Added “grouping” element to GenericOption type.</td>
<td>HMA AR RID: AR#72</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>None</td>
<td>HMA AR RID: AR#91</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Order quotation improved.</td>
<td>HMA AR RID: AR#92</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Used gml:PolygonType, gml:RectangleType, gml:PointType to specify geographic elements</td>
<td>HMA AR RID: AR#95</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Removed duplicated element definitions; added new delivery methods</td>
<td>HMA AR RID: AR#96</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Added number of copies in the orderItem element.</td>
<td>HMA AR RID: AR#97</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>ftp-pull removed from DeliveryInformationType; ftp-pull put in DeliveryInformationTypeExt which inherits from the previous one.</td>
<td>HMA AR RID: AR#98</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>None</td>
<td>HMA AR RID: AR#99</td>
</tr>
<tr>
<td>19/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Updated definition of orderId</td>
<td>HMA AR RID: AR#100</td>
</tr>
<tr>
<td>09/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Added itemId and packaging to the Submit operation.</td>
<td>Inputs from E-OA project</td>
</tr>
<tr>
<td>09/02/2007</td>
<td>1.2.0 Draft</td>
<td>D. Marchionni</td>
<td>Added DescribeResultAccess operation</td>
<td>Inputs from E-OA project</td>
</tr>
<tr>
<td>Date</td>
<td>Internal version</td>
<td>Editor</td>
<td>Primary clauses modified</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>------------------</td>
<td>-----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------</td>
</tr>
</tbody>
</table>
| 09/02/2007 | 1.2.0 Draft      | D. Marchionni | Updated delivery options:  
• deliveryMethod includes FTP, WCS;  
• packageMedium has been made an enumeration.  
• Removed FTP push / pull from DeliveryInformation because the on-line delivery is performed via DescribeResultAccess operation. | Inputs from E-OA project                          |
| 09/02/2007 | 1.2.0 Draft      | D. Marchionni | Call-back operation for asynchronous requests (UpdateStatus) has been replaced by SubmitResponse and CancelResponse.                                                                                                   | Alignment to SPS EO Profile                      |
| 04/05/2007 | 1.2.1 Draft      | D. Marchionni | Updated OR7 to the latest available SWE Architecture document                                                                                                                                                           | HMA FP RID: FP#9                                 |
| 04/05/2007 | 1.2.1 Draft      | D. Marchionni | Updated figure reference at §7.1.2                                                                                                                                                                                   | HMA FP RID: FP#11                                 |
| 04/05/2007 | 1.2.1 Draft      | D. Marchionni | Added sftp and ftps as delivery methods (table 7-1)                                                                                                                                                                   | HMA FP RID: FP#12                                 |
| 04/05/2007 | 1.2.1 Draft      | D. Marchionni | Corrected definition of “use” attribute in §7.1.2                                                                                                                                                                     | HMA FP RID: FP#13                                 |
| 04/05/2007 | 1.2.1 Draft      | D. Marchionni | Updated §7.1.2 and table 7-2 specifying the difference between the [OR6] and [NR11] definitions.                                                                                                                     | HMA FP RID: FP#14                                 |
| 04/05/2007 | 1.2.1 Draft      | D. Marchionni | Corrected table 7-4.                                                                                                                                                                                                   | HMA FP RID: FP#16                                 |
| 04/05/2007 | 1.2.1 Draft      | D. Marchionni | Clarified albumExtract definition in Table 7-15                                                                                                                                                                       | HMA FP RID: FP#17                                 |
| 04/05/2007 | 1.2.1 Draft      | D. Marchionni | Updated §7.1.6, Figure 7-16, table 7-20, Figure 7-17, Figure- 7-18, Figure 7-19, Table 7-23, Figure 7-20, Figure 7-21  
OrderStatusType renamed to StatusType  
OrderStatusType/orderState renamed to StatusType/status | HMA FP RID: FP#18                                 |
<table>
<thead>
<tr>
<th>Date</th>
<th>Internal version</th>
<th>Editor</th>
<th>Primary clauses modified</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>04/05/2007</td>
<td>1.2.1 Draft</td>
<td>D. Marchionni</td>
<td>Updated table 8-1 explaining the sync / async usage of GetQuotation operation.</td>
<td>HMA FP RID: FP#19</td>
</tr>
<tr>
<td>04/05/2007</td>
<td>1.2.1 Draft</td>
<td>D. Marchionni</td>
<td>Table 8-7: clarified the meaning of “monitoring” attribute.</td>
<td>HMA FP RID: FP#20</td>
</tr>
<tr>
<td>04/05/2007</td>
<td>1.2.1 Draft</td>
<td>D. Marchionni</td>
<td>Updated Table 8-17, §6.1, §8.1.4</td>
<td>HMA FP RID: FP#21</td>
</tr>
<tr>
<td>04/05/2007</td>
<td>1.2.1 Draft</td>
<td>D. Marchionni</td>
<td>Added expirationDate in Figure 8-18 and Table 8-21</td>
<td>HMA FP RID: FP#22</td>
</tr>
<tr>
<td>04/05/2007</td>
<td>1.2.1 Draft</td>
<td>D. Marchionni</td>
<td>Removed third bullet of §8.2.4</td>
<td>HMA FP RID: FP#24</td>
</tr>
<tr>
<td>04/05/2007</td>
<td>1.2.1 Draft</td>
<td>D. Marchionni</td>
<td>Table 7-10</td>
<td>HMA FP RID: FP#26</td>
</tr>
<tr>
<td>04/05/2007</td>
<td>1.2.1 Draft</td>
<td>D. Marchionni</td>
<td>Figure 7-2, Table 7-2, Figure 7-3, Table 7-3, Figure 7-11, Table 7-12</td>
<td>Alignment with SWE Common 0.0.0 (going to be approved as 1.0.0) and with SPS EO 0.9.4</td>
</tr>
<tr>
<td>04/05/2007</td>
<td>1.2.1 Draft</td>
<td>D. Marchionni</td>
<td>Figure 7-12, Table 7-13, Figure 7-14, Table 7-15, Figure 7-16, Table 7-19, Figure 7-21, Table 7-25, Figure 8-4, Table 8-7, Figure 8-5, Table 8-8, Figure 8-7</td>
<td>Integration with SPS EO for future product ordering.</td>
</tr>
<tr>
<td>04/05/2007</td>
<td>1.2.1 Draft</td>
<td>D. Marchionni</td>
<td>Figure 8-4, Table 8-7</td>
<td>To return list of supported collections within the Capabilities document.</td>
</tr>
<tr>
<td>04/05/2007</td>
<td>1.2.1 Draft</td>
<td>D. Marchionni</td>
<td>§8.1.1.1, Figure 8-2, Table 8-2</td>
<td>Removed password from userInformation element</td>
</tr>
<tr>
<td>04/05/2007</td>
<td>1.2.1 Draft</td>
<td>D. Marchionni</td>
<td>Figure 7-17, Table 7-21, Figure 7-18, Table 7-22, Figure 7-19, Table 7-23, Figure 8-15, Table 8-18, Figure 8-16, Figure 8-21</td>
<td>Enhanced get order list functionality of GetStatus operation.</td>
</tr>
<tr>
<td>14/08/07</td>
<td>BP final</td>
<td>Carl Reed</td>
<td>Various to ready document for posting.</td>
<td></td>
</tr>
<tr>
<td>14/09/2007</td>
<td>1.3</td>
<td>D. Marchionni</td>
<td>IMPR#32</td>
<td>Updated description of sceneSelection element within ProductOrderItemType (Error! Reference source not found.).</td>
</tr>
<tr>
<td>Date</td>
<td>Internal version</td>
<td>Editor</td>
<td>Primary clauses modified</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>------------------</td>
<td>------------------</td>
<td>--------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>14/09/2007</td>
<td>1.3</td>
<td>D.Marchionni</td>
<td>IMPR#37</td>
<td>Added regionOfInterest parameter at Table 7-4); Added GetOptions &amp; GetOptionResponse examples for subscriptions.</td>
</tr>
<tr>
<td>14/09/2007</td>
<td>1.3</td>
<td>D.Marchionni</td>
<td>IMPR#67</td>
<td>§3.2 Removed references to ESA documents.</td>
</tr>
<tr>
<td>14/09/2007</td>
<td>1.3</td>
<td>D.Marchionni</td>
<td>IMPR#72</td>
<td>Added reference to OGC 07-118 [OR8] for references to all Identity Management aspects.</td>
</tr>
<tr>
<td>14/09/2007</td>
<td>1.3</td>
<td>D.Marchionni</td>
<td>IMPR#73</td>
<td>Attributes have been aligned with the HMA User Identity Management ICD:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• telNumber fields replaced with telephoneNumber (DeliveryAddress Type)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• faxNumber replaced with facsimileTelephoneNumber (DeliveryAddress Type)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• country attributes shall be compliant with ISO 3166</td>
</tr>
<tr>
<td>14/09/2007</td>
<td>1.3</td>
<td>D.Marchionni</td>
<td>IMPR#82</td>
<td>Updated priority definition (Table 7-8, 7-22); Updated status definition (Table 7-21, 7-24)</td>
</tr>
<tr>
<td>14/09/2007</td>
<td>1.3</td>
<td>D.Marchionni</td>
<td>IMPR#83</td>
<td>Same as IMPR#37</td>
</tr>
<tr>
<td>14/09/2007</td>
<td>1.3</td>
<td>D.Marchionni</td>
<td>IMPR#90</td>
<td>Same as IMPR#72</td>
</tr>
<tr>
<td>14/09/2007</td>
<td>1.3</td>
<td>D.Marchionni</td>
<td></td>
<td>Updated examples and schema (Appendix A, B, C)</td>
</tr>
<tr>
<td>27/09/2007</td>
<td>0.9.1</td>
<td>D. Marchionni</td>
<td>Removed user identifier</td>
<td>Updated schema, examples, picture.</td>
</tr>
<tr>
<td>Date</td>
<td>Internal version</td>
<td>Editor</td>
<td>Primary clauses modified</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------</td>
<td>-----------------</td>
<td>---------------------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>26/05/2008</td>
<td>0.9.2</td>
<td>D. Marchionni</td>
<td>• Removed hierarchy for order options returned by GetOptions;</td>
<td>DAIL-RD-ESA-EN-0001</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Removed hierarchy for orders specified in Order Quotation</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Removed hierarchy for orders specified in Order Submission</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Removed hierarchy for orders specified in Order Monitoring</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Added ftp element within DeliveryInformation</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Order options at order level: not included in this issue.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Added OrderReference to the list of order search criteria</td>
<td></td>
</tr>
<tr>
<td>26/05/2008</td>
<td>0.9.2</td>
<td>D. Marchionni</td>
<td>• Updated DescribeResultAccess Response for returning additional information about the resource to be accessed</td>
<td>DAIL-RD-ASU-EN-0006</td>
</tr>
<tr>
<td>26/05/2008</td>
<td>0.9.2</td>
<td>D. Marchionni</td>
<td>• Removed second sentence in nextReady field of table 8-19</td>
<td>DAIL-RD-ASU-EN-0007</td>
</tr>
<tr>
<td>26/05/2008</td>
<td>0.9.2</td>
<td>D. Marchionni</td>
<td>Multiplicity of URL field of DescribeResultResponse updated to 0..N.</td>
<td>DAIL-RD-ASU-EN-0008</td>
</tr>
<tr>
<td>26/05/2008</td>
<td>0.9.2</td>
<td>D. Marchionni</td>
<td>• Added “description” element to CommonOrderOption sType</td>
<td>DAIL-RD-SPB-EN-0011</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Added “productOrderOption sId” element to CommonOrderItemType</td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td>Internal version</td>
<td>Editor</td>
<td>Primary clauses modified</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>------------------</td>
<td>-----------------</td>
<td>------------------------------------------------------------------------------------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>26/05/2008</td>
<td>0.9.2</td>
<td>D. Marchionni</td>
<td>Updated Definition of DescribeResultAccess: allReady and nextReady are now enumerative values.</td>
<td>DAIL-RD-SPB-EN-0015</td>
</tr>
<tr>
<td>26/05/2008</td>
<td>0.9.2</td>
<td>D. Marchionni</td>
<td>Table 7-6: qualityOfService is applicable also to Subscription orders.</td>
<td>JM e-mail 23/04/2008</td>
</tr>
<tr>
<td>05/09/2008</td>
<td>0.9.3</td>
<td>D.Marchionni</td>
<td>§8.1.3.1: collectionId, identifier and taskingRequestId are in a single choice. Updated schema (Appendix A).</td>
<td>DAIL-RD-SPB-EN-0007</td>
</tr>
<tr>
<td>05/09/2008</td>
<td>0.9.3</td>
<td>D.Marchionni</td>
<td>Added exception specification for all operations (added paragraphs §8.1.X.3). Updated WSDL (Appendix B).</td>
<td>DAIL-RD-SPB-EN-0014</td>
</tr>
<tr>
<td>05/09/2008</td>
<td>0.9.3</td>
<td>D.Marchionni</td>
<td>Added FAILED and TERMINATED order statuses (updated tables 7-16, 7-17). Updated schema (Appendix A).</td>
<td>DAIL-RD-SPB-EN-0016</td>
</tr>
</tbody>
</table>

v. Changes to the OGC™ Abstract Specification

To be confirmed as result of ongoing work.

vi. Future work

None.

vii. Foreword

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

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

The service proposed in this implementation specification is intended to support the ordering of Earth Observation (EO) data products either from previously identified data set collections via a typical catalogue interaction or from future acquisitions specified via a Programming service [OR6]. The intent of this specification is to describe an interface that can be supported by many data providers (satellite operators, data distributors, etc.), most of whom have existing (and relatively complex) facilities for the management of these data.
1 Scope

This proposed implementation specification describes the interfaces, bindings and encodings required to order Earth Observation (EO) products.
2 Conformance

Conformance will be tested by the HMA-T project.
3 References

This document references several external standards and specifications as dependencies:

4. WSDL, Web Services Description Language (WSDL) 1.1, http://www.w3.org/TR/2001/NOTE-wsdl-20010315

3.1 Normative References

[NR7] WSDL, Web Services Description Language (WSDL) 1.1. Available [online]: http://www.w3.org/TR/wSDL
[NR9] OWS Common Implementation Specification, May 2005 OGC 05-008c1

3.2 Other References


[OR4] OGC™ CSW-ebRIM Registry Service – Part 2 : Basic extension package, version 1.0.0, 2008/02/29 -OGC 07-144r2

[OR5] OGC™ CSW-ebRIM Registry Service - Part 1: ebRIM profile of CSW, version 1.0.0, 2008/02/29. OGC 07-110r2

[OR6] OpenGIS® Sensor Planning Service - Application Profile for EO Sensors OGC 07-018 V0.9.4a


[OR8] OpenGIS® User Management Interfaces For Earth Observation Services OGC 07-118
4 Terms and definitions

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

4.1. Application 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]

4.2. client
software component that can invoke an operation from a server

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

4.4. 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]

4.5. dataset series (dataset collection1)
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.

4.6. geographic dataset
dataset with a spatial aspect [ISO 19115]

4.7. geographic information
information concerning phenomena implicitly or explicitly associated with a location relative to the Earth [ISO 19128 draft]

1 Due to historical reasons we’ll mainly use the term ‘dataset collection’ in this document although the term ‘dataset series’ is used in the ISO/TC211 Terminology Maintenance Group.
4.8. georesource
gеographic information of a specific type (e.g. geographic dataset, geographic application, geographic service)

4.9. identifier
а 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

4.10. interface
named set of operations that characterise the behaviour of an entity [ISO 19119]

4.11. metadata dataset (metadataset)
metadata describing a specific dataset [ISO 19101]

4.12. 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].

4.13. metadata schema
conceptual schema describing metadata

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

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

4.15. operation
specification of a transformation or query that an object may be called to execute [ISO 19119]

4.16. parameter
variable whose name and value are included in an operation request or response
4.17. **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]

4.18. **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].

4.19. **request**
invocation of an *operation* by a *client*

4.20. **response**
result of an *operation*, returned from a *server* to a *client*

4.21. **schema**
formal description of a model [ISO 19101, ISO 19103, ISO 19109, ISO 19118]

4.22. **server**
**service instance**
a particular instance of a *service* [ISO 19119]

4.23. **service**
distinct part of the functionality that is provided by an entity through interfaces [ISO 19119]
capability which a service provider entity makes available to a service user entity at the interface between those entities [ISO 19104 terms repository]

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

4.25. **service metadata**
metadata describing the *operations* and *geographic information* available at a *server* [ISO 19128 draft]
4.26. state
condition that persists for a period

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

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

4.28. version
version of an Implementation Specification (document) and XML Schemas to which the requested operation conforms

NOTE An OWS Implementation Specification version may specify XML Schemas against which an XML encoded operation request or response must conform and should be validated.
5 Symbols and abbreviations

5.1 Symbols (and abbreviated terms)
Some frequently used abbreviated terms:

API Application Program Interface
COTS Commercial Off The Shelf
CQL Common Query Language
CRS Coordinate Reference System
CSW Catalogue Service-Web
DCE Distributed Computing Environment
DC Dublin Core
DCMI Dublin Core Metadata Initiative
DCP Distributed Computing Platform
DDS Data Dissemination Service
EO Earth Observation
HMA Heterogeneous Missions Accessibility
HTTP Hyper Text Transport Protocol
ISO International Organisation for Standardisation
OGC Open GIS Consortium
SOAP Simple Object Access Protocol
SQL Structured Query Language
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
5.2 **UML notation**

5.2.1 **UML Class Diagrams**

Some of the diagrams in this document are presented using the Unified Modeling Language (UML) static structure diagram. The UML notations used in this document are described in Figure 5-1, below.

**Association between classes**

<table>
<thead>
<tr>
<th>Association Name</th>
<th>Class #1</th>
<th>Class #2</th>
</tr>
</thead>
<tbody>
<tr>
<td>role-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>role-2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Association Cardinality**

<table>
<thead>
<tr>
<th></th>
<th>Class #1</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Only one</td>
<td></td>
<td>One or more</td>
<td></td>
</tr>
<tr>
<td>0..*</td>
<td></td>
<td>Specific number</td>
<td></td>
</tr>
<tr>
<td>0..1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Aggregation between classes**

<table>
<thead>
<tr>
<th>Aggregate Class</th>
<th>Component Class #1</th>
<th>Component Class #2</th>
<th>Component Class #n</th>
</tr>
</thead>
</table>

**Class Inheritance (subtyping of classes)**

<table>
<thead>
<tr>
<th>Superclass</th>
<th>Subclass #1</th>
<th>Subclass #2</th>
<th>Subclass #n</th>
</tr>
</thead>
</table>

**Figure 5-1: UML Class Diagram notations.**

In these UML class diagrams, the class boxes with a light background are the primary classes being shown in this diagram, often the classes from one UML package. The class boxes with a gray background are other classes used by these primary classes, usually classes from other packages.

In this diagram, the following stereotypes of UML classes are used:

<<Interface>> A definition of a set of operations that is supported by objects having this interface. An Interface class cannot contain any attributes.

<<Type>> A stereotyped class used for specification of a domain of instances (objects), together with the operations applicable to the objects. A Type class may have attributes and associations.

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

<<CodeList>> A flexible enumeration that uses string values for expressing a list of potential values. If the list alternatives are completely known, an enumeration shall be used; if the only likely alternatives are known, a code list shall be used.
<<Enumeration>> A data type whose instances form a list of alternative literal values. Enumeration means a short list of well-understood potential values within a class.

In this document, the following standard data types are used:
CharacterString – A sequence of characters
Boolean – A value specifying TRUE or FALSE
Integer – An integer number
Identifier – Unique identifier of an object
URI – An identifier of a resource that provides more information
URL – An identifier of an on-line resource that can be electronically accessed

5.2.2 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.

![Sequence Diagram Example](image)

Figure 5-2: UML Sequence Diagrams 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 subclause.

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

- **Optional single element without child elements**
  ![Optional single element without child elements](image)

- **Optional single element with child elements**
  ![Optional single element with child elements](image)

- **Mandatory single element.**
  ![Mandatory single element](image)

- **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 multiple element containing child elements](image)

- **Mandatory single element with containing simple content (e.g. text) or mixed complex content (e.g. text with xhtml markup).**
  ![Mandatory single element with containing simple content or mixed complex content](image)

- **A sequence of elements. The elements must appear exactly in the sequence in which they appear in the schema diagram.**
  ![A sequence of elements](image)

- **A choice of elements. Only a single element from those in the choice may appear at this position.**
  ![A choice of elements](image)

- **Types. If an element refers to a complex global type, the type is shown with a border and yellow background.**
  ![A choice of elements](image)

- **Complex Type. The following figure illustrates the use of a complex type for defining an XML element**
  ![Complex Type](image)
5.4 Document terms and definitions

This document uses the specification terms defined in Subclause 5.3 of [NR9].
6 System context

This section focuses on the purpose, scope and policies of ordering services that comply with this specification. It documents special requirements and describes the context of use.

6.1 Application domain

The ordering service described in this specification has the objective of supporting the following types of orders of Earth Observation products:

- Order from catalogues of EO products.
  
  This service allows the preparation and the submission of an order including products identified via a search in a catalogue of EO products ([OR3]).

- Order of products derived from a programming request.
  
  This service allows the submission of an order including products which can be derived from a set of future acquisition segments specified via an external Programming Service ([OR6]).

- Subscription to EO products.
  
  This service allows users the periodical reception of products of interest on the areas of interest.

For these purposes, the Ordering Service for Earth Observation Products specifies the following operations:

- **GetCapabilities**, allows a client to request and receive service metadata (or Capabilities) documents that describe the abilities of the specific server implementation.

- **GetOptions**, allows clients to retrieve the options for issuing an order: in case of product ordering it returns the options for ordering a specific type of product; in case of subscriptions it returns the possible parameters to set for specifying the scope of the subscription (e.g.: area of interest, expiration date, etc.).

- **GetQuotation**, which allows the client to get a quotation either of the order that is going to be submitted or of the subscription going to be subscribed. This operation, in order to support the wider set of clients and service providers, supports several interaction models:
  
  - Synchronous quotation;
  
  - Asynchronous via Notification: the client has to implement a call-back operation (**GetQuotationResponse**) which is called by the server when the quotation is available.
  
  - Asynchronous via Monitoring: after the first activation, in which the client specifies all order parameters, the client has to call it again (referencing the first call) until the server is able to return the quotation.
  
  - Off-line: the quotation is not provided by on-line interaction but via mail / e-mail.

- **Submit**, which allows either submitting an order of products (from EO catalogue or from a programming request) or for subscribing to a subscription. This operation is asynchronous and then the client has to implement a call-back operation (**SubmitResponse**) for receiving the result of the operation.

- **DescribeResultAccess**, which allows accessing the products ordered with on-line delivery.
• **GetStatus**, which allows to retrieve either the status of submitted orders or the status of subscribed subscriptions.

• **Cancel**, which allows either to ask the cancellation of an already submitted order or to unsubscribe a subscription. This operation is asynchronous and then the client has to implement a call-back operation (**CancelResponse**) for receiving the result of the operation.

### 6.2 Essential Use-cases

#### 6.2.1 Ordering from catalogue of EO Products

The following figure shows the nominal usage of the Order Service operations for ordering products from EO Catalogues:
Figure 6-1: Sequence of steps generally performed for ordering products from EO Catalogue.

In the scenario the following entities are specified:

- Web Service Client, which represents the user submitting requests to the Order Service;
- Order Service Instance: it is the server providing the order service;
The typical scenario is:

- The list of products to be ordered has been prepared on client side by querying EO Catalogues.
- The client gets the list of supported operations from the Ordering Service instance (GetCapabilities).
- The list of ordering options is retrieved for each product to be ordered (GetOptions).
- Then the order is prepared on the client side choosing the requested options for each of the products to order.
- The quotation of the just prepared order can be asked by calling the GetQuotation operation. The quotation can be received either synchronously or via asynchronous notification depending on client and server capabilities.
- In case the quotation is accepted, the order can be submitted to the Order Service (Submit). The Order Service returns back an acknowledgement and start the execution of the order.
- After the order has been submitted the following events are possible:
  - The client asks the status of the order to verify the progress of the order (GetStatus);
  - The client is notified of possible status updates (SubmitResponse).
  - The client can ask the cancellation of the order (Cancel)
- When the order processing is completed, the ordered products are either delivered to the user or can be retrieved on-line by calling DescribeResultAccess operation depending on the selected delivery method.

### 6.2.2 Order of Future Products derived from tasking requests

For future products the scenario is very similar to the previous one: instead of getting the catalogue identifier of the products to order, the client has to identify the future acquisition segments needed for generate the products to order. This step is performed accessing an external programming service ([OR6] – Sensor Planning Service Earth Observation Application Profile). These future acquisition segments are identified within the products order via the identifier of the corresponding tasking request returned by the Programming service.

Then the submission of this type of order can be summarized with the following steps:

- Identification of the necessary future acquisition segments and require them to the external Programming Service ([OR6] - SPS EO). As a result the programming request identifiers of all needed acquisitions are available on client side.
- Get order options for the products that can be ordered from the identified acquisitions: call GetOptions specifying as input parameters the programming identifiers.
- The order is prepared on client side choosing the options necessary for getting the needed products from the identified acquisitions.
- The quotation of the just prepared order can be asked by calling the GetQuotation operation. The quotation can be received either synchronously or via asynchronous notification depending on client and server capabilities.
• In case the quotation is accepted, the order can be submitted to the Order Service (Submit). The Order Service returns back an acknowledgement and start the execution of the order.

• After the order has been submitted the following events are possible:
  o The client asks the status of the order to verify the progress of the order (GetStatus);
  o The client is notified of possible status updates (SubmitResponse).

• The client can ask the cancellation of the order (Cancel)

• When the order processing is completed, the ordered products are either delivered to the user or can be retrieved on-line by calling DescribeResultAccess operation depending on the selected delivery method.

6.2.3 **Subscribe to EO Products**

This section explains the steps to follow for subscribing to published EO products subscriptions.
Figure 6-2: Sequence of steps generally performed for subscribing to EO products.

In the scenario the following entities are specified:

- Web Service Client, which represents the user submitting requests to the Order Service;
- Order Service Instance: it is the server providing the order service;

The typical scenario is:
• The client gets the list of supported operations from the server (GetCapabilities).

• The list of subscriptions to subscribe has been prepared on client side querying a catalogue storing the advertised subscriptions, and then the next step is to retrieve the list of possible subscription options (GetOptions). Possible examples options are:
  - region of interest, which allows to receive of the whole available products only the one overlapping this area;
  - type of area coverage, specifying how the products to be returned are spatially related to the specified area (e.g. overlap, inclusion, etc.)
  - expiration date, which specifies the limit date & time of validity of the subscription;
  - repetition, which specify the number of time the base observation period is repeated.
  - the number of products per observation.

• The subscription order is prepared on the client side setting the available subscription options.

• The quotation of the just prepared order can be asked by calling the GetQuotation operation. The quotation can be received either synchronously or via asynchronous notification depending on client and server capabilities.

• In case the quotation is accepted, the subscriptions are subscribed to the Order Service by calling Submit operation. The Order Service returns back an acknowledge confirming the activation of the subscription.

• After the order has been submitted the following events are possible:
  - The client asks the status of his / her subscriptions (GetStatus).
  - The progress of the subscriptions are directly notified to the client (SubmitResponse)
  - The client is allowed to unsubscribe the subscriptions (Cancel).

• When some products are ready, they can be retrieved by DescribeResultAccess operation if ordered with on-line delivery.
7 Information models

As previously anticipated, this specification supports different type of orders:

- Product Orders
  - Order from EO Catalogue ([OR3]): the products to be ordered are identified from catalogue performing a catalogue search. The retrieved identifiers are used for building the items within the order.
  - Future product orders via programming service ([OR6] SPS EO)
    For building a future product order 2 steps are necessary:
      - A programming request has to be defined in order to specify the needed acquisition segments. This step is performed via the SPS EO instance linked to the Ordering service which support future products ordering.
        The tasking request deals only with the parameters needed for the acquisition e.g.: start & stop time / orbit or area to cover, polarization, incidence angles, sensor mode, etc. At this stage the level of product, the format, the delivery options are not specified.
      - A product order has to be specified, referencing the tasking request, for setting all parameters needed to deliver products to the user e.g.: required product type, product format, media, delivery address, accounting and billing information.

- Subscription
  For subscribing a subscription it is sufficient specifying the appropriate collection identifier, the area and the temporal extension of the subscription.

In the following sub-sections the data structures for modelling these three type of orders are described.

7.1 Information model for EO product ordering

The information models section deals with the information item managed through the Order Service operations, which are:

- Order options (§7.1.2)
- Order Specification (§7.1.3)
- Order Item (§7.1.4)
- Order Quotation (§7.1.5)
- Order Monitoring (§7.1.6)
- Order Item Monitoring (§7.1.7)
7.1.1 XML schema approach

This specification deals mainly with 2 different types of requests: product orders and subscription orders. These requests share several common parameters and have some differences, and then to model them in a XML schema we envisaged the following options:

- Put all parameters together making optional the specific ones;
- Put all parameters together putting choice elements for the specific ones;
- Define a hierarchy with an abstract type grouping the common parameters and use type substitution.

The first approach is the most simple one, but does not allow strict schema checks, because all the specific parameters are optional and cannot be verified whether the parameters needed to a specific type of order are provided or not.

The second approach implies the usage of lot of choices in the schema and if a new request has to be managed, all these choices have to be updated.

The third approach allows schema checks for the different type of requests and is more extensible than the previous one because it is sufficient to add the new request to the substitution group and all places where the substitution group was used don’t have to be updated. However it adds complexity to the schema due to substitution groups, which are often source of problems for XML data binding tools, and to the hierarchy of types, which is needed to regroup common elements without replicating the definitions in several places.

The third approach was the one chosen until the release 0.9.2; starting from 0.9.3 the schema has been simplified adopting instead the first approach.

7.1.2 Order Options

Order options specify all possible valid combinations of options for ordering products of a specified dataset collection or for subscribing to a subscription.

7.1.2.1 CommonOrderOptionsType

The following figure represents the common order options.
Figure 7-1: CommonOrderOptionsType diagram.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>CommonOrderOptionsType</td>
<td>It contains the common information about the different order options available for the order.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>productOrderOptionsId</td>
<td>Identifier of the specific order option group. Type: Not empty string (max 40 char) Example: “on-line retrieval”</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Tag Name</td>
<td>Tag Description</td>
<td>Product Ordering</td>
<td>Subscription</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------------------------------------------------------------------------</td>
<td>------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>Description</td>
<td>Free text description of the order option group.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: string (max 255 char)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>orderType</td>
<td>This tag specifies whether the order option is related to a subscription order or to a product order.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: string</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Permitted Values: PRODUCT_ORDER, SUBSCRIPTION_ORDER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>option</td>
<td>Extensible list of ordering options (see Table 7-2)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>productDeliveryOptions</td>
<td>Delivery Options for the order item.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>deliveryMethod</td>
<td>Delivery methods valid for the delivery medium.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Permitted Values: mail, ftp, sftp (SSH File Transfer Protocol), ftps (FTP over SSL), P2P, wcs, wms, email, dds</td>
<td></td>
<td></td>
</tr>
<tr>
<td>packageMedium</td>
<td>Identification of a delivery Medium and a Medium Formatting Option.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: Not empty string (max 40 chars)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Permitted Values: NTP, DAT, Exabyte, CD-ROM, DLT, D1, DVD, file (for ftp, sftp, ftps, wcs, P2P, DDS delivery), HDDVD, BD (Blu-ray)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>orderOptionInfoURL</td>
<td>Pointer to external information about the current order option.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: xs:anyURI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>paymentOptions</td>
<td>Payment options</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>paymentMethod</td>
<td>Payment methods.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: string (max 40 char)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Example: quota, invoice, prepay (to be indicated for free products), deposit account, credit card, credit card previously supplied</td>
<td></td>
<td></td>
</tr>
<tr>
<td>paymentOptionInfoURL</td>
<td>Pointer to external information about the current payment option.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: xs:anyURI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sceneSelectionOption</td>
<td>Identifies a scene selection option available for an order option group.</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>sceneType</td>
<td>Identifier for the scene type specific to the selected product. Detailed characteristics of the scene type are expected to be specified under link “orderOptionsInfoURL”</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: Not empty string (max 20 chars)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7-1: CommonOrderOptionsType description.
7.1.2.2 ParameterDescriptorType

The ParameterDescriptorType ([NR11] §11.2.1, [OR6] §8.1) defines the input a client has to provide for setting ordering options. One instance of this type represents one order option to be set for the specified order item.

The ParameterDescriptorType contains the mandatory attributes:

- “parameterID” which will be used to reference a specific parameter in other requests, e.g. Submit / GetQuotation.
- The “use” attribute defines if the parameter can or shall be provided by the client. It enumerates the strings “optional” and “required”. If required, a Submit or GetQuotation request will not be validated as true if this parameter is missing.
- The third attribute “updateable” is defined in the SPS specification, but in this context has to be set always true because update operation is not supported in ordering service.

The sps:ParameterDescriptorType defines four elements: three optional elements and one mandatory. The three optional elements can be used to provide further description about the order option (gml:description), to allow the presetting of possible values (restriction) (e.g. “yes, no” or “day, night” or “1, 2, 3”), and for defining the cardinality of possible input elements (cardinality). Cardinality is restricted to positive integers (excluding zero) and the string value “unbounded”.

The data structure of the input elements that shall be provided by the client is defined in the mandatory and unbounded “sps:definition” element. This element serves as an entry point to parsers to find the data block definition that has to be matched by the input data. It is followed by either a sps:commonData element (see SWECommon for further information), a “sps:taskMessageDefinition” element which is a link to an external definition of the data block or a “sps:GeometryDefinition” or multiple sps:ParameterDescriptor.

The “sps:definition” element has been made unbounded in [OR6] with respect the [NR11] base specification in order to allow the definition of the same parameters more times e.g.: a regionOfInterest may be a gml:Polygon OR a Circle.

“sps:GeometryDefinition” is of type QName and is restricted to the GML elements gml:Point, gml:Line, gml:Polygon. It is assumed that clients “know” how to encode those basic elements.

The sps:ParameterDescriptorType has been enhanced in [OR6] w.r.t [NR11] adding recursion in order to allow the definition of complex parameters as list, aggregates, list of aggregates, etc.
Figure 7-2: ParameterDescriptorType diagram.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>ParameterDescriptorType</td>
<td>See [NR11] §11.2.1, [OR6]§8.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>parameterID</td>
<td>It is the identifier of the order option to use when the order is submitted.</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: xs:ID</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>use</td>
<td>Defines if the parameter shall be provided by the client or not. It enumerates the strings “optional” and “required”. If required, a Submit or GetQuotation request will not be validated as true if this parameter is missing. Type: String Allowed values: required, optional</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Tag Name</td>
<td>Tag Description</td>
<td>Product Ordering</td>
<td>Subscription</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>updateable</td>
<td>It is optional and defines if a parameter can be updated (defined in SPS, but not supported in Ordering Service).</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>sps:Description</td>
<td>Additional textual description of the order option.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>sps:definition</td>
<td>This element describes the structure of the order option.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>swe:commonData</td>
<td>see SWECommon [OR7]</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>sps:TaskMessageDefinition</td>
<td>It is a link to an external definition of the data block</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>sps:GeometryDefinition</td>
<td>Enumerative value: gml:Point, gml:Line, gml:Polygon</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>sps:ParameterDescriptor</td>
<td>Recursive parameter description definition in charge of representing parameters having complex nested structure.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>sps:restriction</td>
<td>It specifies a set of predefined values the client has to specify for the ordering parameter.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>sps:cardinality</td>
<td>Cardinality of parameter to insert</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>identifier</td>
<td>Product identifier element.                                                                --------------------------------------------------------------------------------------------------------------------------------</td>
<td>X</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>grouping</td>
<td>Additional identifier for grouping correlated order options.</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Table 7-2: ParameterDescriptorType description.

7.1.2.3 sps:Parameter

The Parameter Element is used to provide the value for a specific order option. The encoding follows the description that is part of the definition element of an sps:ParameterDescriptor Element. The Parameter
Element is therefore rather simple in its definition. It just has to provide the mandatory parameterID attribute to link the values to the specific order option. The values itself are replacing the any-Element.

In case of order option with nested definition, the value has to be set by setting appropriately the nested sps:Parameter element.

![Figure 7-3: sps:Parameter diagram](image)

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
<td>See [NR11] §11.2.1, [OR6]§8.1</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>parameterID</td>
<td>It is the identifier of the order option to use when the order is submitted.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: xs:ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sps:value</td>
<td>It is the value of the order option in case of non-nested definitions.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: any</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sps:Parameter</td>
<td>Recursive declaration of the sps:Parameter.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: sps:ParameterType</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 7-3: sps:Parameter description.**

7.1.2.4 Preliminary and extensible list of ordering options

The ordering parameters include all parameters to be specified for generating the product needed to the user.

The first column specifies the name of the parameter to put in the parameterID attribute of ParameterDescriptorType; the second column describes it; the “Product Orders” column specifies whether the parameter is applicable for product orders; the last column specifies whether the parameter is applicable for Subscriptions.
<table>
<thead>
<tr>
<th>Tasking Parameter Name</th>
<th>Description</th>
<th>Product Orders</th>
<th>Subscription Orders</th>
</tr>
</thead>
<tbody>
<tr>
<td>processingLevel</td>
<td>Level of processing required on the data to acquire. E.g.; ASA_IMM_1P, MER_RR__2P, etc. Type: ENUMERATED_STRING</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>format</td>
<td>Product format e.g.: CEOS, ESA, etc. Type: ENUMERATED_STRING</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>regionOfInterest</td>
<td>This parameter is meaningful only for subscriptions allowing users restricting the scope of the subscription. Type: GeometryDefinition (polygon)</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>startDate</td>
<td>UTC time of the starting visiting period. Type: TIME</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>completionDate</td>
<td>UTC time of the finishing visiting period. Type: TIME</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>numberOfObservations</td>
<td>It specifies how many times the revisiting has to be performed. Type: NUMERIC</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>observationGap</td>
<td>Number of days between 2 successive observations. Type: NUMERIC</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>compression</td>
<td>Type of compression applied to the delivered products. E.g.: none, zip, gzip, bzip2, Type ENUMERATED_STRING</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>qualityOfService</td>
<td>Quality of service available Examples: Standard, Rush, NRT Type: ENUMERATED_STRING</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

Table 7-4: Order options extensible list.

7.1.3 Order Specification

This section defines all parameters a client has to specify for submitting an order (products / subscription).

For the definition of orders the following hierarchy of complex types has been declared:

- CommonOrderSpecification, which is the root of the hierarchy and includes all parameters common to order specification and order monitoring;
- OrderSpecification, inherited from CommonOrderSpecification and add the parameters specific for submitting orders;
These types are described in the following figures and tables:

**Figure 7-4: CommonOrderSpecification diagram.**

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>CommonOrderSpecification</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>orderReference</td>
<td>User defined name assigned to that order.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td><strong>Type</strong>: Not empty string (max 30 chars)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>orderRemark</td>
<td>Textual remark on the order.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Type</strong>: Not empty string (max 255 chars)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>deliveryInformation</td>
<td>Delivery Information element.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Type</strong>: DeliveryInformationType (§7.1.3.1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>invoiceAddress</td>
<td>Invoice Address element.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Type</strong>: DeliveryAddressType (§7.1.3.3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>packaging</td>
<td>This element allows packing all ordered items in the same file.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Type</strong>: string <strong>Allowed values</strong>: zip, tar, tgz</td>
<td></td>
<td></td>
</tr>
<tr>
<td>priority</td>
<td>Priority of the order</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Type</strong>: string <strong>Allowed Values</strong>: STANDARD, FAST_TRACK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tag Name</td>
<td>Tag Description</td>
<td>Product Ordering</td>
<td>Subscription</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------</td>
<td>-----------------</td>
<td>--------------</td>
</tr>
<tr>
<td>orderType</td>
<td>This tag specifies whether the order option is related to a subscription order or to a product order. Type: string Permitted Values: PRODUCT_ORDER, SUBSCRIPTION_ORDER</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Table 7-5: CommonOrderSpecification description.

Figure 7-5: OrderSpecification diagram.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
</table>
Table 7-6: OrderSpecification description.

7.1.3.1 DeliveryInformationType

The following figure gives a graphical representation of DeliveryInformationType:

Figure 7-6: DeliveryInformationType diagram.
## 7.1.3.2 FTPAddressType

![FTPAddressType diagram]

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>serverAddress</td>
<td>Address of the user FTP server the ordering service has to put the data.</td>
</tr>
<tr>
<td>Type:</td>
<td>Not empty string</td>
</tr>
<tr>
<td>userId</td>
<td>Identifier of the user for accessing the FTP server.</td>
</tr>
<tr>
<td>Type:</td>
<td>Not empty string (max 8 chars)</td>
</tr>
<tr>
<td>userPassword</td>
<td>User password for accessing the FTP server.</td>
</tr>
<tr>
<td>Type:</td>
<td>Not empty string</td>
</tr>
<tr>
<td>directory</td>
<td>Directory file path name where the ordering service has to put the products.</td>
</tr>
<tr>
<td>Type:</td>
<td>Not empty string (max 1024 chars)</td>
</tr>
</tbody>
</table>

Table 7-8: FTPAddressType description.
7.1.3.3 DeliveryAddressType

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeliveryAddressType</td>
<td></td>
</tr>
<tr>
<td>recipient</td>
<td>Identification of the receiving person.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> Not empty string (max 40 chars)</td>
</tr>
<tr>
<td>companyRef</td>
<td>Identification of the receiving entity.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> Not empty string (max 40 chars)</td>
</tr>
<tr>
<td>postalAddress</td>
<td>Postal Address of the user.</td>
</tr>
<tr>
<td>streetAddress</td>
<td>Street Address element.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String (max 40 chars)</td>
</tr>
<tr>
<td>city</td>
<td>City element.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String (max 40 chars)</td>
</tr>
<tr>
<td>state</td>
<td>State element.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String (max 40 chars)</td>
</tr>
<tr>
<td>postalCode</td>
<td>Postal Code element.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String (max 12 chars)</td>
</tr>
</tbody>
</table>

Figure 7-8: DeliveryAddressType diagram.
Table 7-9: DeliveryAddressType description.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>postalBox</td>
<td>Postal Box element. Only number part, only digits allowed</td>
</tr>
<tr>
<td>telephoneNumber</td>
<td>Telephone number of the receiving person.</td>
</tr>
<tr>
<td>facsimileTelephoneNumber</td>
<td>FAX number of the receiving person.</td>
</tr>
</tbody>
</table>

7.1.4 Order Item

This section defines all parameters a client has to specify for one item within an order (products / subscription).

7.1.4.1 CommonOrderItemType
Figure 7-9: CommonOrderItemType diagram.
<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>CommonOrderItemType</td>
<td>Common parameters of an ordered item.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>itemId</td>
<td>Unique identifier of the item within the order.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: non empty string (max 80 char)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>productOrderOptionsId</td>
<td>Identifier of chosen order option group</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>orderItemRemark</td>
<td>Textual remark on the order item put by the XML issuer.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> Not empty string (max 255 chars).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>options</td>
<td>It specifies the options to be applied to the ordered item. The list of values</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>specified in this element shall comply with the one returned by GetOptionsResponse.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sps:Parameter</td>
<td>X X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>parameterID</td>
<td>Name of order option.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: xs:QName</td>
<td></td>
<td></td>
</tr>
<tr>
<td>value</td>
<td>Value of the order option.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: any</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sps:Parameter</td>
<td>Value of order option in case of recursive definitions.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: sps:ParameterType</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sceneSelection</td>
<td>This field has two purposes:</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>- It specifies the selection of the scene from the product that is to be</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>delivered.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Then it is used from the processing chain for extracting the scene from the</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>parent product.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- It is the footprint of the product that has to be delivered.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Then this value is used along the ordering chain for e.g.:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>rendering on map, checking spatial restrictions, etc.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>This field has to be <strong>always</strong> provided unless the product has not a</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>spatial coverage.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Its type is described in paragraph 7.1.4.2.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>deliveryMethod</td>
<td>Delivery methods valid for the delivery medium.</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Permitted Values:</strong> mail, ftp, P2P, wcs, e-mail, dds</td>
<td></td>
<td></td>
</tr>
<tr>
<td>packageMedium</td>
<td>Identification of a delivery Medium and a Medium Formatting Option.</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Permitted values:</strong> NTP, DAT, Exabyte, CD-ROM, DLT, D1, DVD, file, HDDVD, BD (Blu-ray)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>numberOfCopies</td>
<td>Specifies how many copies of the same order item have to be delivered.</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> integer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>payment</td>
<td>User selected payment information.</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>If specified, it overrides the possible payment method and payment info stored</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>in the user profile.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>paymentMethod</td>
<td>Selected payment method: quota, credit card, etc.</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> non empty string (max 40 char)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tag Name</td>
<td>Tag Description</td>
<td>Product</td>
<td>Ordering</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------</td>
<td>----------</td>
</tr>
<tr>
<td>orderAccount</td>
<td>In case of payment by quota, this field specifies the account under which the user is authorised to order from the specific provider e.g. a project or service name.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>creditCardInfo</td>
<td>In case of payment by credit card it specifies the credit card information. TBD encoding of credit card information within the message.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>productId</td>
<td>This field is set for ordering items from Catalogue. It identifies the target product on which the order item is based.</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>identifier</td>
<td>Product identifier element. Depending on the selected options, this item is either: • the ordered product • the parent product from which the product required by the client has to be prepared by applying the selected options.</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>collectionId</td>
<td>Search space for the product requested. It is referenced as parentIdentifier in the [OR3].</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>taskingRequestId</td>
<td>This element is set for ordering products coming from a tasking request issued to SPS instance linked to the Ordering Service.</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>sps:ID</td>
<td>Identifier for the feasibility study / tasking request.</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>subscriptionId</td>
<td>It identifies the target subscription on which the order item is based.</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>collectionId</td>
<td>Search space for the product requested. It is referenced as parentIdentifier in the [OR3].</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

Table 7-10: CommonOrderItemType description.
7.1.4.2 SceneSelection Type

The following figure gives a graphical representation of the SceneSelectionType:

![SceneSelectionType diagram]

**Figure 7-10: SceneSelectionType diagram**

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>SceneSelectionType</td>
<td></td>
<td>X</td>
<td>Not applicable</td>
</tr>
<tr>
<td>sceneType</td>
<td>Identifier for the scene type specific to the selected product. <strong>Type:</strong> Not empty string (max 20 chars)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sceneCoordinates</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>gml:Point</td>
<td>Scene center element. <strong>Type:</strong> gml:PointType</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gml:Polygon</td>
<td>Bounding polygon of the selected scene. <strong>Type:</strong> gml:PolygonType</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gml:Rectangle</td>
<td>Bounding rectangle of the selected scene. <strong>Type:</strong> gml:RectangleType</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tag Name</td>
<td>Tag Description</td>
<td>Product Ordering</td>
<td>Subscription</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>albumExtract</td>
<td>Source data extract defined by its bounding box with album catalogue coordinates (impossible for mosaic or cropping). Album catalogue is a low resolution image of the acquired datastrip stored within the catalogue. In Pleiades case, it corresponds to a 4-bands 22.4m resolution raw image (i.e. un-projected). On these images it is possible to select a sub-image by specifying its row / column range within the parent.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>firstRow</td>
<td>First row</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: Integer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>nbRow</td>
<td>Number of rows</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: Integer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>firstCol</td>
<td>First column</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: Integer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>nbCol</td>
<td>Number of columns</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: Integer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>temporalSelection</td>
<td>Temporal selection element</td>
<td></td>
<td></td>
</tr>
<tr>
<td>startDateTime</td>
<td>Start time for the temporal selection in the following format: CCYY-MM-DDThh:mm:ss.ccZ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: xs:DateTime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>endDateTime</td>
<td>Stop time for the temporal selection in the following format CCYY-MM-DDThh:mm:ss.ccZ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: xs:DateTime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>scenePosition</td>
<td>Provider specific system to define the position of the scene. Examples: “frame 1234”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: Not empty string (max 40 chars)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sceneSize</td>
<td>Provider specific system to define the size of the scene. Examples: “frames: 3”, “40x50 km”, “26 seconds”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: Not empty string (max 40 chars)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7-11: SceneSelectionType description.

7.1.5 Order Quotation

This section describes the information provided into the order quotation.
Figure 7-11: OrderQuotation diagram.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrderQuotation</td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
**Tag Name** |
--- |
quotationId | Identifier of the whole order quotation. URI conventions are followed for encoding this identifier. Type: anyURI |
validityTime | Time until the quotation of the whole order is valid. It is the minimum of the validity time of all orderItem groups. Type: xs:dateTime |
price | Total price of the order. Is not provided when the different orderItem groups have different payment methods (e.g. quota and credit card (USD)). Type: CurrencyType (Table 7-15) |
orderItemGroupPrice | Price information of order items groups. The ordered items are grouped because: • All items managed by the same provider have the same quotation rules; • Discounts can be applied when several items are ordered together; • Different provider can support different payment methods. Type: OrderItemGroupPrice (Table 7-13) |
contractInformation | Textual description of rights and conditions applied to the whole order. Type: string (max 1024 chars) |

**Table 7-12: OrderQuotation description.**

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrderItemGroupPrice</td>
<td>Price information related to a group of ordered items.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>provider</td>
<td>Provider which accepted the request for quotation for this group of items</td>
<td></td>
<td></td>
</tr>
<tr>
<td>serviceName</td>
<td>Service Name Type: String (max 40 chars)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>organization</td>
<td>Provider’s organization name. Type: string (max 40 chars)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tag Name</td>
<td>Tag Description</td>
<td>Product Ordering</td>
<td>Subscription</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>quotationId</td>
<td>Identifier of the order group quotation. This identifier is optional because can be returned when the quotation of the ordered items group is performed by an organization different from the one providing the Ordering service itself. URI conventions are followed for encoding this identifier. Type: anyURI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>validityTime</td>
<td>Time until the quotation of the order item group is valid. Type: xs:dateTime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>price</td>
<td>Price of the orderItemGroup. Type: CurrencyType (Table 7-15)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>balance</td>
<td>In case of providers supporting payment by quota, this field returns the balance of the quota considering the price of this orderItemGroup. Type: CurrencyType (Table 7-15)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>orderItemPrice</td>
<td>Price of the each item of the group. Type: OrderItemPrice.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>contractInformation</td>
<td>Textual description of rights and conditions applied to the ordered items. Type: string (max 1024 chars)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 7-13: OrderItemGroupPrice description.**

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrderItemPrice</td>
<td>Price information of a single order item.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>itemId</td>
<td>Unique identifier of the order item within the order. It is the same identifier specified in the order. Type: string (max 80 chars)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>productId</td>
<td>This field is set for ordering items from Catalogue. It identifies the target product on which the order item is based. Type: ProductIdType</td>
<td>X</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>taskingRequestId</td>
<td>This element is set for ordering products coming from a tasking request issued to SPS instance linked to the Ordering Service.</td>
<td>X</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>subscriptionId</td>
<td>It identifies the target subscription on which the order item is based. Type: SubscriptionIdType</td>
<td>Not Applicable</td>
<td>X</td>
</tr>
</tbody>
</table>
Tag Name | Tag Description | Product Ordering | Subscription
--- | --- | --- | ---
price | List price of the order item. Optional because the actual price is the one at group level. Type: CurrencyType |  |  |
contractInformation | Textual description of rights and conditions applied to the ordered item. Type: string (max 1024 chars) |  |  |

**Table 7-14: OrderItemPrice description.**

---

| Tag Name | Tag Description | Product Ordering | Subscription
--- | --- | --- | ---
CurrencyType |  | X | X
| value | value |  |  
|  | Type: double |  |  |
currency | Currency including ISO 4217 (e.g.: EUR, USD (US Dollar), CAD (Canada Dollar), AUD (Australia Dollar), GBP (United Kingdom Pounds), etc.) and also special values not conflicting with the ISO ones for representing quota. Type: string (max 10 chars) |  |  |

**Table 7-15: CurrencyType description.**

### 7.1.6 Order Monitoring Specification

This section defines all parameters returned to the client when getting the status of submitted orders.

For the definition of orders the following hierarchy of complex types has been declared:

- CommonOrderSpecification, which is the root of the hierarchy and includes all parameters common to order specification and order monitoring;
- CommonOrderMonitorSpecification, which includes all order monitoring parameters;

These types are described in the following sub-sections.
7.1.6.1 CommonOrderMonitorSpecification

Figure 7-12: CommonOrderMonitorSpecification diagram.

<p>| Tag Name       | Tag Description | Product Ordering | Subscription | Presentation |
|----------------|-----------------|------------------|--------------|--------------|--------------|</p>
<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
<th>Presentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CommonOrderMonitorSpecification</td>
<td>It specifies the common parameters returned into an order monitor XML document. Inherited from CommonOrderSpecification (Table 7-5)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>orderId</td>
<td>Unique identifier of the order for this service. Type: xs:anyURI</td>
<td>X</td>
<td>X</td>
<td>Brief &amp; full</td>
</tr>
<tr>
<td>orderStatusInfo</td>
<td>Contains the Order status information.</td>
<td></td>
<td></td>
<td>Brief &amp; full</td>
</tr>
<tr>
<td>status</td>
<td>Status of a product Order. Type: String</td>
<td>X</td>
<td>X</td>
<td>Brief &amp; full</td>
</tr>
<tr>
<td></td>
<td><strong>Current List of Valid Values:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Submitted (INTERMEDIATE status)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Accepted (INTERMEDIATE status)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• InProduction (INTERMEDIATE status)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Suspended (INTERMEDIATE status)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Cancelled (previously accepted item cancelled at request of the customer). FINAL status.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Completed (made available to the user as per defined DeliveryMethod). FINAL status.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Failed, when an error occurred during the processing of the order. FINAL status.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Terminated, when a subscription has been completed. FINAL status.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>additionalStatusInfo</td>
<td>Description associated with the Order status. Type: Not empty string (max 255 chars)</td>
<td>X</td>
<td>X</td>
<td>Brief &amp; full</td>
</tr>
<tr>
<td>missionSpecificStatusInfo</td>
<td>Additional text description where mission specific information can be put. Type: String (max 255 chars)</td>
<td>X</td>
<td>X</td>
<td>Brief &amp; full</td>
</tr>
<tr>
<td>orderDateTime</td>
<td>Date and Time of the order submission/update Type: xs:dateTime</td>
<td>X</td>
<td>X</td>
<td>Brief &amp; full</td>
</tr>
<tr>
<td>orderItem</td>
<td>This element reports the status information of the order item. Type: CommonOrderStatusItemType</td>
<td>X</td>
<td>X</td>
<td>full</td>
</tr>
</tbody>
</table>

Table 7-16: CommonOrderMonitorSpecification description.
7.1.7 Order Item Monitoring Specification

This section specifies the status information returned for product and subscription order items. For the definition of order item monitoring info, the following complex types have been defined:

- CommonOrderItemType, which regroups all attributes common to order submission and order monitoring;
- CommonOrderStatusItemType, which regroups the order status attributes;

These elements are described in the following sub-sections.
7.1.7.1 CommonOrderStatusItemType

Figure 7-13: CommonOrderStatusItemType diagram.
<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>CommonOrderStatusItemType</td>
<td>It regroups the common status attributes. Inherited from CommonOrderItemType (Table 7-10)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>orderItemStatusInfo</td>
<td>Status information at item level.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>Status of order item.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Current List of Valid Values:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Submitted (INTERMEDIATE status)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Accepted (INTERMEDIATE status)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• InProduction (INTERMEDIATE status)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Suspended (INTERMEDIATE status)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Cancelled (previously accepted item cancelled at request of the customer).</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Completed (made available to the user as per defined DeliveryMethod).</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Failed, when an error occurred during the processing of the order item.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Terminated, when a subscription has been completed.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>additionalStatusInfo</td>
<td>Description associated with the Order status.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> Not empty string (max 255 chars)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>missionSpecificStatusInfo</td>
<td>Additional text description where mission specific information can be put.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String (max 255 chars)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7-17: CommonOrderStatusItemType description.
8 External interfaces

This view describes the externally visible behaviour of the system, including the interfaces provided by its components and the supported protocol bindings. It defines the request and response message structures as part of the operation signatures.

All operations must support the embedding of requests and responses in SOAP messages. Only SOAP messaging (via HTTP/POST) with document/literal style has to be used. Messages must conform to SOAP 1.2 (http://www.w3.org/TR/SOAP/). The message payload will be in the body of the SOAP envelope.

Table 8-1 summarises the Ordering operations and their encoding methods that are applied in this document. The mandatory method bindings and data encodings are printed in bold.

<table>
<thead>
<tr>
<th>Ordering Operation</th>
<th>Request encoding</th>
<th>Sync / Async</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetCapabilities</td>
<td>XML/SOAP</td>
<td>Synchronous request</td>
<td></td>
</tr>
<tr>
<td>GetOptions</td>
<td>XML/SOAP</td>
<td>Synchronous request</td>
<td></td>
</tr>
</tbody>
</table>
| GetQuotation       | XML/SOAP         | Synchronous Asynchronous request | This operation can be used in different ways depending on the client and server ability:  
  - Synchronous, when the server is able to provide a real time response;  
  - Asynchronous via Notification, when the client is able to work as a server for getting the asynchronous notification  
  - Asynchronous via Monitoring: the client has to ask the server until it returns the quotation.  
  In case of Asynchronous via Notification usage, after the reception of this request, the order service calls the GetQuotationResponse operation provided by the client for sending the quotation. |
<table>
<thead>
<tr>
<th>Operation</th>
<th>SOAP/HTTP Method</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetQuotationResponse</td>
<td>XML/SOAP</td>
<td>Call-back for asynchronous request</td>
<td>This operation is called by the order service for sending the quotation in case of Asynchronous via Notification usage.</td>
</tr>
<tr>
<td>Submit</td>
<td>XML/SOAP</td>
<td>Asynchronous request</td>
<td>This operation is asynchronous and then, after the reception of this request, the order service calls (optionally) the SubmitResponse operation provided by the client.</td>
</tr>
<tr>
<td>SubmitResponse</td>
<td>XML/SOAP</td>
<td>Call-back for asynchronous request</td>
<td>This operation is called by the order service after the reception of Submit operations.</td>
</tr>
<tr>
<td>GetStatus</td>
<td>XML/SOAP</td>
<td>Synchronous request</td>
<td></td>
</tr>
<tr>
<td>Cancel</td>
<td>XML/SOAP</td>
<td>Asynchronous request</td>
<td>This operation is asynchronous and then, after the reception of this request, the order service calls (optionally) the CancelResponse operation provided by the client.</td>
</tr>
<tr>
<td>CancelResponse</td>
<td>XML/SOAP</td>
<td>Call-back for asynchronous request</td>
<td>This operation is called by the order service after the reception of Cancel operations.</td>
</tr>
<tr>
<td>DescribeResultAccess</td>
<td>XML/SOAP</td>
<td>Synchronous request</td>
<td></td>
</tr>
</tbody>
</table>

Regarding the asynchronous request and replies the WS-addressing ([NR10]) SOAP header extensions have been used:

- In the SOAP header of GetQuotation, Submit and Cancel operations request messages the following tags have to be included (**wsa** is the namespace of WS-addressing definitions):

  ```xml
  <wsa:ReplyTo>
  <wsa:Address> order service URI of the client </wsa:Address>
  </wsa:ReplyTo>
  <wsa:MessageID> unique identifier of the request </wsa:MessageID>
  ```

  In case the reply address is set to:

it means that the client does not have to be notified. It is useful for standard clients not having server capabilities. In this case the client has to ask the status of asynchronous requests calling dedicated operations (e.g. GetStatus for checking the status Submit and Cancel operations).

- In the SOAP header of GetQuotationResponse / SubmitResponse / CancelResponse operations request messages the following tags have to be included (wsa is the namespace of WS-addressing definitions):

```xml
<wsa:RelatesTo RelationshipType="wsa:Response">
Identifier of the GetQuotation, Submit or Cancel previously submitted requests
</wsa:RelatesTo>
```

The following sequence diagram explains the usage scenario of Submit, Cancel and GetQuotation operations:

![Sequence diagram](image)

**Figure 8-1: Asynchronous requests scenario.**

**User Identity Information**

Regarding the format and protocol of user information, which are needed for authenticating and authorizing ordering service requests, it is not covered by this ICD, but it is objective of [OR8] document.
8.1 Interface specifications

It gives formal, language-independent interface specifications that admit multiple programming language bindings and shows error conditions that can occur.

8.1.1 Shared parameters

8.1.1.1 statusNotification element

This element is used in asynchronous requests for specifying whether the client has to be notified and in which way:

- no notifications
- all status changes to be notified
- just the completion of the order submission to be notified

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>statusNotification</td>
<td>This element specifies how many status notifications are sent back to the client.</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td>Permitted Values:</td>
</tr>
<tr>
<td></td>
<td>• None (no status notification sent back)</td>
</tr>
<tr>
<td></td>
<td>• All (all status changes are notified)</td>
</tr>
<tr>
<td></td>
<td>• Final (only the completion of the order is notified)</td>
</tr>
</tbody>
</table>

8.1.2 GetCapabilities Operation

The mandatory 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 an Order Service. This section specifies the XML document that an Order Service server must return to describe its capabilities.
8.1.2.1 GetCapabilities input message: GetCapabilities element

The GetCapabilities operation request shall be as specified in Subclauses 7.2 and 7.3 of [NR9]. The value of the “service” parameter shall be “OS”. The allowed set of service metadata (or Capabilities) XML document section names and meanings shall be as specified in Tables 3 and 7 of [NR9].

The “Multiplicity and use” column in Table 1 of [NR9] specifies the optionality of each listed parameter in the GetCapabilities operation request. The following table specifies the implementation of those parameters by Order Service clients and servers.

<table>
<thead>
<tr>
<th>Name</th>
<th>Multiplicity</th>
<th>Client implementation</th>
<th>Server implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>service</td>
<td>One (mandatory)</td>
<td>The parameter shall be implemented by all clients, using the specified value</td>
<td>The parameter shall be implemented by all servers, checking that each parameter is received with specified value</td>
</tr>
<tr>
<td>AcceptVersions</td>
<td>Zero or one (optional)</td>
<td>Should be implemented by all software clients, using specified values</td>
<td>Shall be implemented by all servers, checking if parameter is received with specified value(s)</td>
</tr>
</tbody>
</table>

Figure 8-2: GetCapabilities request diagram.

The GetCapabilities operation request shall be as specified in Subclauses 7.2 and 7.3 of [NR9]. The value of the “service” parameter shall be “OS”. The allowed set of service metadata (or Capabilities) XML document section names and meanings shall be as specified in Tables 3 and 7 of [NR9].
Table 8-2: Implementation of parameters in GetCapabilities operation request

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default Implementation</th>
<th>Client Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sections</td>
<td>Each parameter may be implemented by each client</td>
<td>If parameter not provided, shall expect default response</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If parameter provided, shall allow default or specified response</td>
</tr>
<tr>
<td>updateSequence</td>
<td>Each parameter may be implemented by each server</td>
<td>If parameter not implemented or not received, shall provide default response</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If parameter implemented and received, shall provide specified response</td>
</tr>
<tr>
<td>AcceptFormats</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8.1.2.2 GetCapabilities output message: Capabilities element

The following figure provides a graphical representation of the Capabilities XML document.
Figure 8-3: Capabilities diagram.
The GetCapabilities response shall contain the Order Service sections specified in the following table. Depending on the values in the Sections parameter of the GetCapabilities operation request, any combination of these sections can be requested and shall be returned when requested.

<table>
<thead>
<tr>
<th>Section name</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceIdentification</td>
<td>Metadata about this specific server. The schema of this section shall be the same as for all OWSs, as specified in Subclause 7.4.3 and owsServiceIdentification.xsd of [NR9].</td>
</tr>
<tr>
<td>ServiceProvider</td>
<td>Metadata about the organization operating this server. The schema of this section shall be the same for all OWSs, as specified in Subclause 7.4.4 and owsServiceProvider.xsd of [NR9].</td>
</tr>
<tr>
<td>OperationsMetadata</td>
<td>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 as for all OWSs, as specified in Subclause 7.4.5 and owsOperationsMetadata.xsd of [NR9].</td>
</tr>
</tbody>
</table>

**Table 8-3: Section name values and content.**

In addition to these sections, each service metadata document shall include the mandatory “version” and optional updateSequence parameters specified in Table 6 in Subclause 7.4.1 of [NR9].

**8.1.2.2.1 OperationsMetadata section standard contents**

For the Order Service, the OperationsMetadata section shall be the same as for all OGC Web Services, as specified in Subclause 7.4.5 and owsOperationsMetadata.xsd of [NR9]. The mandatory values of various (XML) attributes shall be as specified in Table 8-4. Similarly, the optional attribute values listed in Table 8-5 shall be included or not depending on whether that operation is implemented by that server. In these tables the “Attribute name” column uses dot-separator notation to identify parts of a parent item. The “Attribute 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.

<table>
<thead>
<tr>
<th>Attribute name</th>
<th>Attribute value</th>
<th>Meaning of attribute value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation.name</td>
<td>GetCapabilities</td>
<td>Implemented by the server</td>
</tr>
<tr>
<td>Operation.name</td>
<td>GetOptions</td>
<td>Implemented by the server</td>
</tr>
<tr>
<td>Operation.name</td>
<td>Submit</td>
<td>Implemented by the server</td>
</tr>
</tbody>
</table>

**Table 8-4: Mandatory Order Service operations.**
Table 8-5: Optional Order Service operations.

### Contents section

This section provides additional information about the order quotation.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
</table>
| GetQuotationCapabilities | This element specifies how the quotation is supported by the Ordering service. In fact, depending on client and service capabilities different options are considered in this specification:  
  - Quotation can be supported or not;  
  - Quotation can be provided either synchronously or asynchronously;  
  - Quotation can be provided via operation interaction;  
  - In case of asynchronous quotation, the client can get it either via asynchronous notification or asking the quotation again until the service actually returns it.  
  The different attributes of GetQuotationCapabilities element specify which of these possibilities are supported by the service. | X                | X             |
<p>| supported              | Set to true whether the quotation is supported.                                                                                                                                                                |                  |              |
| Type: xs:boolean                            |                                                                                                                                                                                                                 |
| synchronous             | Set to true if the service is able to provide synchronous answer to GetQuotation.                                                                                                                                 |                  |              |
| Type: xs:boolean                            |                                                                                                                                                                                                                 |
| asynchronous            | Set to true if the service is able to provide an asynchronous answer to GetQuotation.                                                                                                                                 |                  |              |
| Type: xs:boolean                            |                                                                                                                                                                                                                 |</p>
<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
</table>
| monitoring          | This attribute is set to true when the client has to call GetQuotation several times until the server provides the quotation. Then in this case the quotation process works in this way:  
  - at the first call of GetQuotation the client specifies all order parameters;  
  - the server replies with the quotationId;  
  - then the client calls GetQuotation specifying the quotationId received at the previous call;  
  - if the quotation is available, then the server returns it, otherwise the quotationId is returned again.  
  - This process continues until the server returns the quotation.  
  Type: xs:boolean                                                                                                                                  |                  |              |
| off-line            | Set to true if the service is able to provide answer to quotation via mail / e-mail notification.                                                                                                           |                  |              |
| FutureProductsOrdering | This element specifies whether the ordering service supports also future product ordering and, if it is the case, the URL of the SPS instance to use with it.                                                                                          |                  | X            |
| supported           | Set to true when the Ordering Service supports future products ordering.                                                                                                                                       |                  |              |
| SPS_URL             | It is the SPS instance URL for submitting tasking requests for future products orders. This element shall be specified in case the “supported” flag is true.                                                                                     |                  |              |
| SupportedCollections | This element specifies the list of collections supported by this Ordering service. This list includes collections for products ordering and for subscriptions.                                                      |                  | X            |
| collectionId        | Collection identifier                                                                                                                                                                                        |                  | X            |
|                     | Type: non empty string max 62 chars                                                                                                                                                                           |                  |              |

Table 8-6: Description of Contents section of Capabilities document.

8.1.2.3 Exceptions

In case of errors during the processing of the request the service shall return an ows: ExceptionReport according to the clause 8 of [NR9].

78
8.1.3 GetOptions Operation

This operation allows getting the possible options for preparing an EO products order or for subscribing to EO products.

Depending on the input parameters, this operation works in different ways:

- Options by collection identifier
  - Because the collection identifier can refer either a product collection or a subscription then there are 2 cases:
    - Product collection:
      - In this case the operation returns options for ordering products from that collection. All products shall have the same options otherwise an exception is thrown.
    - Subscription:
      - Subscription options returned.

- Options by collection and product identifier
  - In this case the collection shall support product ordering and all options available for the specified product are returned.

- Options by tasking identifier
  - When the tasking identifier is specified (i.e. SPS:ID), which can refer to either a feasibility study or task submitted to SPS instance linked to this Ordering service, the operation returns the options available for ordering products starting from the acquisitions corresponding to the provided SPS:ID.
  - The SPS:ID can refer different type of requests:
    - Precisely identifier acquisitions, i.e. acquisitions are identified by their key attributes e.g.: orbit and start and stop equator crossing times OR UTC start and stop times etc.
    - Coverage request, i.e. acquisitions are specified via the definition of the area to cover and the related time frame.
    - Standing request, similar to the previous one, but the coverage is performed several times with a defined periodicity.

Returned options are structured in array of groups, each identified by the productOrderOptionsId and representing a valid combination of options (e.g. one group can specify the options for getting a level 1 product, another group specifies options for getting another level 1 product, another for level 2 product, etc.). Each group includes mainly the following attributes:

- Delivery options
- Payment options
- Scene selection options (not applicable for subscriptions)
- Extensible list of options e.g.: processing level, product format. This list can be product specific and then the identifier of the product can be optionally specified.
8.1.3.1 GetOptions input message: GetOptions

The type of GetOptions is OrderOptionsRequestType.

The following figure provides a graphical representation of this type.

![GetOptions element diagram](image)

**Figure 8-4: GetOptions element diagram.**

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetOptions</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>service</td>
<td>Service type identifier</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: non-empty string</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Allowed values: OS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>version</td>
<td>Specification version for operation</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: non-empty string, Format: &lt;x&gt;, &lt;y&gt;, &lt;z&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>timeStamp</td>
<td>It is the time when the request has been issued.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Type: xs:DateTime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>collectionId</td>
<td>It is the identifier of the dataset collection to get ordering options.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>It is referenced as parentIdentifier in the [OR3].</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type Value: string</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Permitted Values: Not empty string</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Syntax: The identifier format is not mandated by the specification.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Source: [OpenGIS], CG_QueryRequest</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

80
<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
</table>
| identifier | Product identifier element. Type Value: string Permitted Values: Not empty string This field is optional and multiple:  
  - if present then only the options related these products shall be returned.  
  - If not present all options applicable to the collection shall be returned. This identifier, if present, shall be owned by the collection. If the identifier is not present and the options depend always on it, the operation returns error. | X                |              |
| taskingRequest | This field is alternative to collectionId and identifier. This field has to be used in case of future product orders issued via both Ordering ICD and Programming ICD [OR6]. |                  |              |
| sps:ID    | Identifier for the feasibility study / tasking request. Type: xs:token                                                                         |                  | X            |

**Table 8-7: GetOptions element description.**

8.1.3.2 GetOptions output message: GetOptionsResponse

The following figure provides a graphical representation of this element.
Figure 8-5: GetOptionsResponse diagram

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetOptionsResponse</td>
<td>GetOptions Response element.</td>
</tr>
<tr>
<td>Tag Name</td>
<td>Tag Description</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>status</td>
<td>Completion result of the operation:</td>
</tr>
<tr>
<td></td>
<td>o success: operation successfully executed;</td>
</tr>
<tr>
<td></td>
<td>o partial: some error occurred during the processing of the request which lead to an incomplete response.</td>
</tr>
<tr>
<td></td>
<td>o failure: severe error occurred during the processing of the request. Request aborted.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String</td>
</tr>
<tr>
<td></td>
<td><strong>Permitted Values:</strong> success, partial, failure</td>
</tr>
<tr>
<td>errorMessage</td>
<td>Error Message element. Set if status different from success.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> Not empty string (max 255 chars).</td>
</tr>
<tr>
<td>orderOptions</td>
<td>Order Options</td>
</tr>
<tr>
<td></td>
<td>Set if status different from failure.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> either CommonOrderOptionsType</td>
</tr>
<tr>
<td></td>
<td>[Error! Reference source not found.]</td>
</tr>
</tbody>
</table>

Table 8-8: GetOptionsResponse description.

8.1.3.3 Exceptions

In case of errors during the processing of the request the service shall return an ows: ExceptionReport according to the clause 8 of [NR9].

8.1.4 GetQuotation Operation

This operation allows getting the quotation of the order specified as input parameter.

Order quotation can be performed in different ways depending on client and server capabilities:

- Synchronous quotation
  
  The server is able to provide the quotation in real time and then it is returned in the acknowledge message of GetQuotation operation.
  
  Synchronous quotation is performed when:
  
  - Capabilities/Content/GetQuotationCapabilities/synchronous is set to true;

- Asynchronous quotation with notification
  
  The server supports asynchronous quotation and the client is able to work as a server for getting the notification carrying on the quotation.
  
  The asynchronous quotation with notification is performed:
  
  - Capabilities/Content/GetQuotationCapabilities/asynchronous is set to true;
Then the quotation process is composed of the following interactions:

- The client calls GetQuotation of the server specifying its address in <wsa:ReplyTo> element;
- The server will call GetQuotationResponse operation of the client for sending the produced quotation.

**Asynchronous quotation with monitoring**

It is a variation of the previous one: the server support asynchronous quotation, but the client cannot work as a server. In this case the client will receive an id at the first call and then it has to call again GetQuotation providing the previously received id until the server will return the quotation in the acknowledge message of the operation.

The mechanism is performed when:

- Capabilities/Content/GetQuotationCapabilities/asynchronous is set to true;
- <wsa:ReplyTo> of GetQuotation message is set with the anonymous address: http://schemas.xmlsoap.org/ws/2003/03/addressing/role/anonymous

**Off-line quotation**

In this case the quotation is sent to the user via fax / mail to the address specified in the invoiceAddress element of the order specification.

Off line quotation is performed when:

- Capabilities/Content/GetQuotationCapabilities/off-line is set to true;
- <wsa:ReplyTo> is set with the anonymous address.
- No other mechanisms are possible.

**8.1.4.1 GetQuotation input message: GetQuotation**

The type of GetQuotation is GetQuotationRequestType.

The following figure provides a graphical representation of this type.
Figure 8-6: GetQuotation diagram.
### Table 8-9: GetQuotation description.

8.1.4.2 GetQuotation output message: GetQuotationAck

The following figure provides a graphical representation of this element.
Figure 8-7: GetQuotationAck element diagram.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetQuotationAck</td>
<td>Acknowledge to GetQuotation.</td>
</tr>
<tr>
<td>status</td>
<td>Completion result of the operation:</td>
</tr>
<tr>
<td></td>
<td>o success: operation successfully executed;</td>
</tr>
<tr>
<td></td>
<td>o partial: non blocking errors occurred during processing of the request.</td>
</tr>
<tr>
<td></td>
<td>o failure: severe error occurred during the processing of the request. Request aborted.</td>
</tr>
<tr>
<td>Type: String</td>
<td>Permitted Values: success, partial, failure</td>
</tr>
<tr>
<td>errorMessage</td>
<td>Error Message element. Set if status different from success.</td>
</tr>
<tr>
<td>Type: Not empty string (max 255 chars).</td>
<td></td>
</tr>
<tr>
<td>quotationId</td>
<td>Identifier of the requested quotation.</td>
</tr>
<tr>
<td></td>
<td>This field is set in case of asynchronous quotation.</td>
</tr>
<tr>
<td></td>
<td>Set if status different from failure.</td>
</tr>
<tr>
<td>Type: QuotationIdType</td>
<td></td>
</tr>
<tr>
<td>Tag Name</td>
<td>Tag Description</td>
</tr>
<tr>
<td>----------</td>
<td>----------------</td>
</tr>
<tr>
<td>quotation</td>
<td>This field is alternative to the previous one. It specifies the quotation of the order. It is set only in case of synchronous / asynchronous with Monitoring quotation requests. Set if status different from failure. Type: OrderQuotation.</td>
</tr>
</tbody>
</table>

**Table 8-10: GetQuotationAck description.**

8.1.4.3 Exceptions

In case of errors during the processing of the request the service shall return an ows: ExceptionReport according to the clause 8 of [NR9].

8.1.5 **GetQuotationResponse: call-back for GetQuotation operation.**

This operation has to be implemented by a client of Ordering service supporting asynchronous operations.

This operation allows the Ordering Service to send to the client the quotation of a specified order (8.1.4).

8.1.5.1 GetQuotationResponse input message: GetQuotationResponse

The following figure provides a graphical representation of this element.
Figure 8-8: GetQuotationResponse diagram.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetQuotationResponse</td>
<td>Message sending the order quotation to the client.</td>
</tr>
<tr>
<td>service</td>
<td>Service type identifier</td>
</tr>
<tr>
<td>version</td>
<td>Specification version for operation</td>
</tr>
<tr>
<td>status</td>
<td>Completion result of the quotation process:</td>
</tr>
<tr>
<td></td>
<td>o success: operation successfully executed;</td>
</tr>
<tr>
<td></td>
<td>o partial: some items have not been quoted.</td>
</tr>
<tr>
<td></td>
<td>o failure: severe error occurred during the processing of the request. Quotation aborted.</td>
</tr>
</tbody>
</table>

Type: String  
Permitted Values: success, partial, failure
<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorMessage</td>
<td>Error Message element. Set if status different from success.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> Not empty string (max 255 chars).</td>
</tr>
<tr>
<td>quotation</td>
<td>It specifies the quotation of the order. It is set if status different from failure.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> OrderQuotation.</td>
</tr>
</tbody>
</table>

**Table 8-11: GetQuotationResponse description.**

8.1.5.2 GetQuotationResponse output message: GetQuotationResponseAck

The following figure provides a graphical representation of this element.

![GetQuotationResponseAck diagram](image)

**Figure 8-9: GetQuotationResponseAck diagram.**

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetQuotationResponseAck</td>
<td>Acknowledge to GetQuotationResponse.</td>
</tr>
<tr>
<td>status</td>
<td>Completion result of the operation:</td>
</tr>
<tr>
<td></td>
<td>o success: operation successfully executed;</td>
</tr>
<tr>
<td></td>
<td>o partial: not applicable.</td>
</tr>
<tr>
<td></td>
<td>o failure: severe error occurred during the processing of the request. Request aborted.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String</td>
</tr>
<tr>
<td></td>
<td><strong>Permitted Values:</strong> success, partial, failure</td>
</tr>
<tr>
<td>errorMessage</td>
<td>Error Message element. Set if status different from success.</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> Not empty string (max 255 chars).</td>
</tr>
</tbody>
</table>

**Table 8-12: GetQuotationResponseAck description.**

8.1.5.3 Exceptions

In case of errors during the processing of the request the service shall return an ows: ExceptionReport according to the clause 8 of [NR9].
8.1.6 Submit Operation

This operation allows either to submit an EO products order or to subscribe EO products.

The order to submit can be specified in two different ways:

- Via quotation identifier
  
  When the server supports order quotation, then the order can be submitted specifying the quotation identifier received from the previously executed GetQuotation operation.

- Via order specification
  
  When the server does not support quotation or the user does not need it then the order can be submitted directly specifying all the order parameters.

Submit is asynchronous operation, but allows the client to specify the amount of notification to receive:

- None: no asynchronous notification is sent to the client.
  
  This kind of notification is activated by:
  
  - setting Submit/statusNotification to None.

- Final: the client is notified when the whole order has been completed. This mechanism is possible only for clients working as server and implementing the SubmitResponse operation.
  
  This kind of notification is activated by:
  
  - setting Submit/statusNotification has to Final;
  - setting <wsa:ReplyTo> of Submit message with the address where the client is listening to the notification and <wsa:MessageID> shall include a unique identifier of the request.

- Every change of the order status is notified to the client. This mechanism is possible only for clients working as server and implementing the SubmitResponse operation.
  
  This kind of notification is activated by:
  
  - setting Submit/statusNotification to All;
  - setting <wsa:ReplyTo> of Submit message with the address where the client is listening to the notification and <wsa:MessageID> shall include a unique identifier of the request.

8.1.6.1 Submit input message: Submit

The type of Submit is SubmitProductOrderRequestType.

The following figure provides a graphical representation of this type.
Figure 8-10: Submit diagram.
Tag Name | Tag Description
--- | ---
Submit | It contains the information to submit an order.
service | Service type identifier
version | Specification version for operation
timeStamp | It is the time when the request has been issued.
quotationId | QuotationId returned by GetQuotation operation,
statusNotification | This element specifies how many status notifications are sent back to the client (see 8.1.1.1)

Table 8-13: Submit description.

8.1.6.2 Submit output message: SubmitAck

The type of SubmitAck is SubmitProductOrderResponseType.

The following figure provides a graphical representation of this type.

![Figure 8-11: SubmitAck diagram](image)

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SubmitAck</td>
<td>Acknowledgment to order submission.</td>
</tr>
<tr>
<td>Tag Name</td>
<td>Tag Description</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| status        | Completion result of the operation:  
|               | o success: operation successfully executed;  
|               | o partial: non blocking errors occurred during processing of the request.  
|               | o failure: severe error occurred during the processing of the request. Request aborted.  
| Type:         | String  
| Permitted Values: | success, partial, failure.  
| errorMessage  | Error Message element. Set if status different from success.  
| Type:         | Not empty string (max 255 chars).  
| orderId       | Order identification number unique for this Provider. Set to “FAILURE” in case status field is set to failure  
| Type:         | xs:anyURI  
| orderReference| See Table 8-13                                                                                                                                                                                                                                                                     |

**Table 8-14: SubmitAck description.**

8.1.6.3 Exceptions

In case of errors during the processing of the request the service shall return an ows: ExceptionReport according to the clause 8 of [NR9].

8.1.7 **SubmitResponse: call-back for Submit operation.**

This operation has to be implemented by a client of Ordering service supporting asynchronous operations.

This operation allows the Ordering Service to send to the client notifications about the progress of submitted orders.

8.1.7.1 SubmitResponse input message: SubmitResponse

The following figure provides a graphical representation of this element.
Figure 8-12: SubmitResponse diagram.
Tag Name | Tag Description
--- | ---
SubmitResponse | Service type identifier
  service | Type: non-empty string
  Allowed values: OS
version | Specification version for operation
  Type: non-empty string. Format: <x>.<y>.<z>
timeStamp | It is the time when the request has been issued.
  Type: xs:DateTime
  The content of this element correspond to that returned by
  GetStatus with full presentation.
  Type: CommonOrderMonitorSpecification §7.1.6.1

Table 8-15: SubmitResponse description.

8.1.7.2 SubmitResponse output message: SubmitResponseAck
The following figure provides a graphical representation of this element.

![Figure 8-13: SubmitResponseAck diagram.](image)

Tag Name | Tag Description
--- | ---
SubmitResponseAck | Acknowledge to SubmitResponse.
  status | Status of reception of SubmitResponse message.
  Type: String
  Permitted Values: success, partial, failure
errorMessage | Error Message element.
  Type: Not empty string (max 255 chars).

Table 8-16: SubmitResponseAck description.
8.1.7.3 Exceptions
In case of errors during the processing of the request the service shall return an ows: ExceptionReport according to the clause 8 of [NR9].

8.1.8 GetStatus Operation
This operation is in charge of returning the status of submitted orders.
It can be used in different ways:
- Order search
  In this way the operation returns all orders matching the filtering criteria:
  - Last update: all orders updated after the specified date are returned;
  - Order status: all orders having the specified status are returned;
- Order retrieve
  Only the order matching the order identifier is returned.

The amount of returned information depends on the presentation:
- brief: only order level information are returned (no order items returned);
- full: whole information returned.

Both presentation values can be used for order search and order retrieve, but for the sake of efficiency the following usage is recommended:
- brief presentation to be used for order search;
- full presentation to be used for getting all details of some of the orders returned by the order search.

8.1.8.1 GetStatus input message: GetStatus
The type of GetStatus is GetStatusRequestType.
The following figure provides a graphical representation of this type.
Figure 8-14: GetStatus diagram.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetStatus</td>
<td>OrderMonitorRequest element.</td>
</tr>
<tr>
<td>service</td>
<td>Service type identifier</td>
</tr>
<tr>
<td></td>
<td>Type: non-empty string</td>
</tr>
<tr>
<td></td>
<td>Allowed values: OS</td>
</tr>
<tr>
<td>version</td>
<td>Specification version for operation</td>
</tr>
<tr>
<td></td>
<td>Type: non-empty string. Format: &lt;x&gt;,&lt;y&gt;,&lt;z&gt;</td>
</tr>
<tr>
<td>timeStamp</td>
<td>It is the time when the request has been issued.</td>
</tr>
<tr>
<td></td>
<td>Type: xs:DateTime</td>
</tr>
<tr>
<td>orderId</td>
<td>Identifier of the order to retrieve.</td>
</tr>
<tr>
<td></td>
<td>See Table 8-14</td>
</tr>
<tr>
<td>Tag Name</td>
<td>Tag Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| filteringCriteria | This element, alternative to the orderId, allows searching submitted orders. Supported search criteria are:  
  o Last update: only orders which status has been changed after the specified date and time are returned  
  o orderStatus: only orders having the specified statuses are returned.  
  o orderReference: only the orders having the specified order reference are returned. |
| lastUpdate    | Last update of the order. Type: Type: date in ISO 8601 format (CCYY-MM-DD)                                                                     |
| orderStatus   | Status of an order. Type: String (Table 7-16)                                                                                                  |
| orderReference | User defined string specified when the order has been submitted. Type String                                                                    |
| presentation  | This element specifies the amount of information to be returned by the GetStatus operation:  
  o brief: only information about the whole order are returned. No order items are returned.  
  o full: the whole order information are returned. Type: enumerated string Permitted values: brief, full |

*Table 8-17: GetStatus description.*

8.1.8.2 GetStatus output message: GetStatusResponse

The type of GetStatusResponse is GetStatusResponseType.

The following figure provides a graphical representation of this type.
Figure 8-15: GetStatusResponse element diagram.
<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
<th>Product Ordering</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetStatusResponse</td>
<td>OrderMonitorResponse element for submitted order.</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>status</td>
<td>Completion result of the operation:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>o success: operation successfully executed;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>o partial: some error occurred during the processing of the request which lead</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>to an incomplete response.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>o failure: severe error occurred during the processing of the request. Request</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>aborted.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type: String</td>
<td>Permitted Values: success, failure, incomplete</td>
<td></td>
<td></td>
</tr>
<tr>
<td>errorMessage</td>
<td>Error Message element. Set if status different from success.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type: Not empty string (max 255 chars).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>The amount of returned information depends on the presentation specified in the</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>request message. Mandatory if status &lt;&gt; failure.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: CommonOrderMonitorSpecification 7.1.6.1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 8-18: GetStatusResponse description.**

8.1.8.3 Exceptions

In case of errors during the processing of the request the service shall return an ows: ExceptionReport according to the clause 8 of [NR9].

8.1.9 **DescribeResultAccess operation**

This operation is in charge of returning the URL of products ordered specifying on-line delivery.
8.1.9.1 DescribeResultAccess input message: DescribeResultAccess

![Diagram of DescribeResultAccess](image)

**Figure 8-16: DescribeResultAccess diagram.**

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DescribeResultAccess</td>
<td>Service type identifier</td>
</tr>
<tr>
<td>service</td>
<td>Type: non-empty string Allowed values: OS</td>
</tr>
<tr>
<td>version</td>
<td>Specification version for operation</td>
</tr>
<tr>
<td>timeStamp</td>
<td>It is the time when the request has been issued. Type: xs:DateTime</td>
</tr>
<tr>
<td>orderId</td>
<td>See Table 8-14</td>
</tr>
<tr>
<td>subFunction</td>
<td>It is an enumerative string specifying the precise behaviour of the operation. Permitted values:</td>
</tr>
<tr>
<td></td>
<td>• allReady</td>
</tr>
<tr>
<td></td>
<td>Flag indicating if all the currently completed items are to be retrieved.</td>
</tr>
<tr>
<td></td>
<td>Calling DescribeResultAccess with this flag before at least one Item is in status “Completed” will return an empty list of URL(s).</td>
</tr>
<tr>
<td></td>
<td>• nextReady</td>
</tr>
<tr>
<td></td>
<td>Flag indicating if all the completed items since last call (or from the beginning of the processing anyway) are to be retrieved.</td>
</tr>
</tbody>
</table>

**Table 8-19: DescribeResultAccess description.**
8.1.9.2 DescribeResultAccess output message: DescribeResultAccessResponse

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DescribeResultAccessResponse</td>
<td>Completion result of the operation:</td>
</tr>
<tr>
<td>status</td>
<td>o success: operation successfully executed;</td>
</tr>
<tr>
<td></td>
<td>o partial: some error occurred during the processing of the request which lead to an incomplete response.</td>
</tr>
<tr>
<td></td>
<td>o failure: severe error occurred during the processing of the request. Request aborted.</td>
</tr>
<tr>
<td>Type: String</td>
<td>Permitted Values: success, failure, incomplete</td>
</tr>
<tr>
<td>errorMessage</td>
<td>Error Message element. Set if status different from success.</td>
</tr>
<tr>
<td>Type: Not empty string (max 255 chars).</td>
<td></td>
</tr>
<tr>
<td>URLs</td>
<td>itemld Order item identifier specified in the Submit request.</td>
</tr>
<tr>
<td>Type: non-empty string (max 80 chars)</td>
<td></td>
</tr>
</tbody>
</table>

Figure 8-17: DescribeResultAccessResponse diagram.
<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>itemAddress</td>
<td>This is the address of a single “item” to be retrieved; according to the settings of “packaging” in Submit an “item” might be the whole result of the Submit operation or a subset. If “allReady” flag is set to true, this list will contain the URL(s) of all the items currently available: if there are still items being processed these will NOT be available. According to the settings of packaging flag in Submit this might yield a single item or a bunch of items (packaging=None). If “nextReady” flag is set to true, this list will contain all the products available since last call (or from the beginning of the processing if called for the first time). According to the settings of packaging flag in Submit, this might yield a single item or a bunch of items (packaging=None). Type: OnLineAccessAddressType.</td>
</tr>
<tr>
<td>expirationDate</td>
<td>Date and time at which the URL will expire. Type: xs:dateTime</td>
</tr>
</tbody>
</table>

Table 8-20: DescribeResultAccessResponse description.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OnLineAccessAddressType</td>
<td>This type defines the full information for accessing an on-line ordered item. In particular it provides information about the ordered resource and also about the server that host it.</td>
</tr>
<tr>
<td>ServiceAddress</td>
<td>This element provides full information about the server hosting the order item. This element is useful e.g. in case of items ordered through a WCS / WMS / etc.</td>
</tr>
<tr>
<td>type</td>
<td>This field specifies the type of the service hosting the ordered item e.g.: WCS, WMS, etc. Type String</td>
</tr>
<tr>
<td>URL</td>
<td>URL of the server hosting the ordered item. Type: anyURI</td>
</tr>
<tr>
<td>info_URL</td>
<td>URL of a document providing metadata about the server. In case of OGC Web Services, it refers to the GetCapabilities operation (HTTP GET, KVP binding). Type: anyURI</td>
</tr>
<tr>
<td>Tag Name</td>
<td>Tag Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>infoRequest</td>
<td>In case the server information are returned via HTTP POST request, this element specifies the message to send (e.g. in case the service supports GetCapabilities with SOAP binding, then the GetCapabilities XML message is specified in this tag) Type: ##any</td>
</tr>
<tr>
<td>ResourceAddress</td>
<td>This element provides full information for accessing directly the ordered item.</td>
</tr>
<tr>
<td>URL</td>
<td>URL of the ordered item.</td>
</tr>
<tr>
<td>serviceRequest</td>
<td>In case the ordered item is accessible via HTTP POST protocol, this element specifies the message to send for getting the ordered item.</td>
</tr>
<tr>
<td></td>
<td>Type: ##any</td>
</tr>
</tbody>
</table>

Table 8-21: DescribeResultAccessResponse description.

8.1.9.3 Exceptions
In case of errors during the processing of the request the service shall return an ows: ExceptionReport according to the clause 8 of [NR9].

8.1.10 Cancel Operation
This operation allows cancelling a previously submitted order.
In case of EO product orders this operation triggers the cancellation of the order items; in case of subscriptions it means to unsubscribe from them.
The cancellation of product order items is not always possible, and then the operation returns the following results:
- “success”, if all items can be cancelled;
- “incomplete”, in case of partial cancellation
- “failure” in case the cancellation is not supported by the service or no item can be cancelled.
The cancellation is not a real-time process, and then the operation is asynchronous.

8.1.10.1 Cancel input message: Cancel
The type of Cancel is CancelRequestType.
The following figure provides a graphical representation of this type.
Order Services for Earth Observation Products

106

8.10.2 Cancel output message: CancelAck

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancel</td>
<td>Cancel Request element.</td>
</tr>
<tr>
<td>timeStamp</td>
<td>It is the time when the request has been issued. Type: xs:DateTime</td>
</tr>
<tr>
<td>orderId</td>
<td>See Table 8-14</td>
</tr>
<tr>
<td>statusNotification</td>
<td>This element specifies how many status notifications are sent back to the client (see 8.1.1.1)</td>
</tr>
</tbody>
</table>

**Table 8-22: Cancel description.**

![Cancel Request diagram](image)

**Figure 8-18: Cancel diagram.**

![CancelAck Request diagram](image)

**Figure 8-19: CancelAck diagram.**

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CancelAck</td>
<td>Cancel acknowledge for submitted order.</td>
</tr>
</tbody>
</table>
8.10.3 Exceptions

In case of errors during the processing of the request the service shall return an ows: ExceptionReport according to the clause 8 of [NR9].

8.11 CancelResponse: call-back for Cancel operation.

This operation has to be implemented by a client of Ordering service supporting asynchronous operations.

This operation allows the Ordering Service to send to the client notifications about the progress of cancellation of submitted orders.

8.11.1 CancelResponse input message: CancelResponse

Because the UpdateStatus operation carries the same information returned by the GetStatus operation, then the UpdateStatus element uses the OrderMonitorSpecification, which is used into the GetStatusResponse element.

The following figure provides a graphical representation of this element.
Figure 8-20: CancelResponse diagram.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CancelResponse</td>
<td>Service type identifier</td>
</tr>
<tr>
<td>service</td>
<td>Type: non-empty string</td>
</tr>
<tr>
<td></td>
<td>Allowed values: OS</td>
</tr>
<tr>
<td>version</td>
<td>Specification version for operation</td>
</tr>
<tr>
<td></td>
<td>Type: non-empty string</td>
</tr>
<tr>
<td></td>
<td>Format: &lt;x&gt;.&lt;y&gt;.&lt;z&gt;</td>
</tr>
<tr>
<td>timeStamp</td>
<td>It is the time when the request has been issued.</td>
</tr>
<tr>
<td></td>
<td>Type: xs:DateTime</td>
</tr>
<tr>
<td>orderMonitorSpecification</td>
<td>Order Monitor Specification element. The content of this element correspond to</td>
</tr>
<tr>
<td></td>
<td>that returned by GetStatus with full presentation.</td>
</tr>
<tr>
<td></td>
<td>Type: CommonOrderMonitorSpecification</td>
</tr>
</tbody>
</table>

Table 8-24: CancelResponse description.

8.1.11.2 CancelResponse output message: CancelResponseAck

Figure 8-21: CancelResponseAck diagram.

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CancelResponseAck</td>
<td>Status of reception of CancelResponse message.</td>
</tr>
<tr>
<td>status</td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td>Permitted Values: success, partial, failure</td>
</tr>
<tr>
<td>errorMessage</td>
<td>Error Message element.</td>
</tr>
<tr>
<td></td>
<td>Type: Not empty string (max 255 chars).</td>
</tr>
</tbody>
</table>
Table 8-25: CancelResponseAck description.

8.1.11.3 Exceptions

In case of errors during the processing of the request the service shall return an ows: ExceptionReport according to the clause 8 of [NR9].

8.2 Implementation guidance

The following section gives developers help when setting up an order service instance that complies with this interface specification. Any information provided here is non-normative or is a detailing of former descriptions.

8.2.1 Distributed Orders implementation

The Order Service operations have been defined to support also a multi-provider scenario where the client is allowed to build and submit orders involving Earth Observation products managed by different providers (e.g. ESA Multi-Mission Ground Segment, SPOT and Radarsat-2 CSA Ground Segment). Then in this scenario we have different order service instances with different roles:

- **façade Order Service Instance**, which is the intermediary element in charge of providing the clients a transparent access to the different providers and to orchestrate the access to them.

- **Delegated Order Service Instances**, which are the services instances running in the provider’s environment that are in charge of effectively carrying out the orders.

In the following sections the interactions occurring between the different service instances for executing the Order Service operations are described.
8.2.1.1 Get Options scenario

The order options are not centralised in the façade Order Service instance, but are distributed between the different providers.

- The client asks order options for **one** specific data set collection;
- The façade Order Service instance retrieves the information about the order services allowing to order that specific product / data set collection;
- The order options are asked to the right provider and then returned back to the client.

**Figure 8-22: Get Options Scenario.**
8.2.1.2 Get Quotation scenario

- The client has prepared the order with needed products and suitable options. Before submitting the order, it asks the quotation;
- The façade Order Service instance retrieves the information about the order services allowing to order that specific products / data set collections;
- The input order is split into different sub-orders each regrouping all items that can be managed by a single provider;
- The different identified providers are asked for the quotation of the different sub-orders. Each returns the identifier of the received quotation request. All the received quotation identifiers are stored locally and a global quotation id generated by the façade Order Service is sent to the client.
- The different providers send back the quotations to the façade calling the asynchronous call-back operation.
- When all quotations have been received, several options are possible:

Figure 8-23: Get Quotation Scenario.
If the client specified asynchronous quotation, the client is notified by calling the call-back operation;

If the client specified no notification, then the client has to monitor the status of the quotation by calling again GetQuotation operation specifying the quotationId received by the first call.

8.2.1.3 Submit scenario

The client has prepared an order for precisely identified items. Before submitting the order, it has also asked the quotation; then the order is submitted.

In case the order is submitted providing the quotationId, the quotation id of the different sub-orders are retrieved from the local store and are used for actually submitting the sub-orders.

In case the order is submitted specifying all parameters:

- The façade Order Service instance retrieves the information about the order services allowing to order that specific product / subscription;
- The input order is split into different sub-orders each regrouping all items that can be managed by a single provider;
- The split sub-orders are sent to different providers identified in the second step.

Each provider creates and returns back the identifier of the sub-order that has been submitted to it.

The façade Order Service instance creates and returns back to the client the identifier of the whole order. Internally it has to manage the link between the “Client Order” and the different “Sub Orders”.

Figure 8-24: Submit Scenario.
8.2.1.4 Status notification scenario

Whenever a sub-order processed by a delegated order service instance changes its status and depending also on the values set in the statusNotification element, the façade Order Service Instance is notified through the SubmitResponse operation.

The SubmitResponse is called specifying the identifier of the sub-order, then the façade Order Service Instance has to retrieve the original order submitted by the client and then has to update the status of items specified in the notification.
8.2.1.5 Get Status scenario

- The client asks the status of a previously submitted order.
- The façade Order Service instance retrieves the identifiers of the sub-orders linked to the order to get the status.
- For each sub-order a GetStatus request is sent to the corresponding provider. The received statuses are put together and sent back to the client.

Figure 8-26: Get Status Scenario.
8.2.1.6 Cancel scenario

- The client requires the cancellation of a specific order.
- The façade Order Service instance retrieves the identifiers of the sub-orders linked to the order to be cancelled.
- For each sub-order a cancellation request is sent to the corresponding provider.

Figure 8-27: Cancel Scenario.
8.2.1.7 Retrieval of on-line available data scenario

The initial steps for preparing the order are not considered in this scenario.

An order with on-line delivery is prepared and submitted by the client asking final asynchronous notification.

When all providers had made available the data the SubmitResponse of the façade is called.

The façade notifies the client by calling the SubmitResponse operation of it.

The client can access the data by calling the DescribeResultAccess operation.

Depending on the selected delivery method, the client can download the data via FTP or can interact with it by using the WCS protocol, etc.

**Figure 8-28: Retrieval of on-line available data scenario**
8.2.2 Semantic issues

None.

8.2.3 Technical issues

- SOAP: Only SOAP messaging (via HTTP/POST) with document/literal style has to be used. Messages must conform to SOAP 1.2 (http://www.w3.org/TR/SOAP/). The message payload will be in the body of the SOAP envelope.

8.2.4 Other Issues

- Order modification due to the delegated order services.

  In some cases, the order services update the received orders. A possible example is the ESA Multi Mission Ground Segment, where the orders submitted by the users can be:

  - re-shuffled, because internally the order is organized in deliveries and then all order items having the same delivery type are managed together.
  - Updated, in case of problems on some ordered products, the order operator, upon agreement with the user, can replace them with possible alternatives.

  Then, to manage these possible modifications performed by the delegated order service instances, the UpdateStatus operation should be allowed also to update orders.

- Order Options management.

  The current approach is that order options are not centralized in the Façade Order Service Instance, and then the GetOptions request is forwarded to right Order Service instance owning the specified order options.

  A better approach is the centralized management of data, because the order options, similarly to the collection and service metadata, is almost static data and then once the data is stored in a registry, the possible updates are very rare.
APPENDIX A    XML Schema definitions (Normative)

<?xml version="1.0" encoding="UTF-8"?>
<!--
 :C Schema
 : Sep 2008
 :Order Schema
 :
 :Marchionni
 :9.4 -->
<xs:schema xmlns="http://earth.esa.int/hma/ordering" xmlns:xs="http://www.w3.org/2001/XMLSchema"
 xmlns:ows="http://www.opengis.net/ows" xmlns:gml="http://www.opengis.net/gml" xmlns:swe="http://www.opengis.net/swe/0.0"
 xmlns:sps="http://www.opengis.net/sps/0" targetNamespace="http://earth.esa.int/hma/ordering" elementFormDefault="qualified"
 attributeFormDefault="unqualified">
 <xs:import namespace="http://www.opengis.net/swe/0.0" schemaLocation="../swe/sweCommon/0.0.0/swe.xsd"/>
 <xs:import namespace="http://www.opengis.net/ows" schemaLocation="../ows/1.0.0/owsGetCapabilities.xsd"/>
 <xs:import namespace="http://www.opengis.net/sps/0" schemaLocation="Order_spsCommon.xsd"/>
 <xs:import namespace="http://www.opengis.net/gml" schemaLocation="../gml/3.1.1/base/gml.xsd"/>
 <xs:import namespace="http://www.opengis.net/ows" schemaLocation="../ows/1.0.0/owsExceptionReport.xsd"/>
 <xs:import namespace="http://www.opengis.net/ows" schemaLocation="/"/>
 <xs:element name="GetCapabilities">
  <xs:annotation>
   <xs:documentation>Request to a Order Service to perform the GetCapabilities operation. This operation allows a client to retrieve service metadata (capabilities XML) providing metadata for the specific Order server. In this XML encoding, no "request" parameter is included, since the element name specifies the specific operation. </xs:documentation>
  </xs:annotation>
  <xs:complexType>
   <xs:complexContent>
    <xs:extension base="ows:GetCapabilitiesType">
     <xs:attribute name="service" type="ows:ServiceType" use="required" fixed="OS"/>
    </xs:extension>
   </xs:complexContent>
  </xs:complexType>
 </xs:element>
 <xs:element name="Capabilities">
  <xs:annotation>
   <xs:documentation>XML encoded Order Service GetCapabilities operation response. This document provides clients with service metadata about a specific service instance. If the server does not implement the updateSequence parameter, the server shall always return the complete Capabilities document, without the updateSequence parameter. When the server implements the updateSequence parameter and the GetCapabilities operation request included the updateSequence parameter with the current value, the server shall return this element with only the "version" and "updateSequence" attributes. Otherwise, all optional elements shall be included or not depending on the actual value of the Sections parameter in the GetCapabilities operation request. </xs:documentation>
  </xs:annotation>
  <xs:complexType>
   <xs:complexContent>
    <xs:extension base="ows:CapabilitiesBaseType">
     <xs:sequence>
      <xs:element name="Contents" type="OrderingServiceContentsType" minOccurs="0"/>
     </xs:sequence>
    </xs:extension>
   </xs:complexContent>
  </xs:complexType>
 </xs:element>
 <xs:element name="GetOptions">
  <xs:annotation>
   <xs:documentation>GetOptions operation - request message </xs:documentation>
  </xs:annotation>
 </xs:element>
 <xs:element name="GetOptionsResponse">
  <xs:annotation>
   <xs:documentation>GetOptions operation - response message </xs:documentation>
  </xs:annotation>
 </xs:element>
 <xs:element name="GetQuotation">
  <xs:annotation>
   <xs:documentation>GetQuotation operation - request message </xs:documentation>
  </xs:annotation>
 </xs:element>
 <xs:element name="GetQuotationAck">
  <xs:annotation>
   <xs:documentation>GetQuotation operation - response message </xs:documentation>
  </xs:annotation>
 </xs:element>
</xs:schema>
<xs:element name="GetQuotationResponse" type="GetQuotationResponseRequestType">
    <xs:annotation>
        <xs:documentation>Async reply to GetQuotation - This message carries on the quotation.</xs:documentation>
    </xs:annotation>
</xs:element>

<xs:element name="GetQuotationResponseAck" type="GetQuotationResponseAckType">
    <xs:annotation>
        <xs:documentation>Response to acknowledge the receipt of quotation.</xs:documentation>
    </xs:annotation>
</xs:element>

<xs:element name="Submit" type="SubmitOrderRequestType"/>
<xs:element name="SubmitAck" type="SubmitOrderResponseType"/>
<xs:element name="SubmitResponse" type="StatusNotificationType"/>
<xs:element name="SubmitResponseAck" type="StatusNotificationAckType">
    <xs:annotation>
        <xs:documentation>Response to acknowledge the receipt of order status notification.</xs:documentation>
    </xs:annotation>
</xs:element>

<xs:element name="GetStatus" type="GetStatusRequestType"/>
<xs:element name="GetStatusResponse" type="GetStatusResponseType"/>
<xs:element name="DescribeResultAccess" type="DescribeResultAccessRequestType"/>
<xs:element name="DescribeResultAccessResponse" type="DescribeResultAccessResponseType"/>
<xs:element name="Cancel" type="CancelRequestType"/>
<xs:element name="CancelAck" type="CancelRequestAckType"/>
<xs:element name="CancelResponse" type="StatusNotificationType"/>
<xs:element name="CancelResponseAck" type="StatusNotificationAckType"/>

<!-- Elements Definition Section -->

<xs:element name="orderId" type="xs:anyURI"/>
<xs:element name="priority" type="PriorityType"/>

<xs:element name="statusNotification">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:enumeration value="None"/>
            <xs:enumeration value="Final"/>
            <xs:enumeration value="All"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>

<xs:element name="collectionId">
    <xs:annotation>
    </xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:minLength value="1"/>
            <xs:maxLength value="62"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>

<xs:complexType name="OrderRequestBaseType">
    <xs:annotation>
        <xs:documentation>XML encoded SPS operation request base, for all operations except Get Capabilities. In this XML encoding, no "request" parameter is included, since the element name specifies the specific operation.</xs:documentation>
    </xs:annotation>
    <xs:attribute name="service" type="xs:string" use="required" fixed="OS"/>
    <xs:attribute>Service type identifier. <xs:documentation>
        <xs:annotation>
            <xs:documentation>
                Service type identifier. <xs:documentation>
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="version" use="required"/>
</xs:complexType>
notifications are sent back to the caller of the Submit operation.
<xs:sequence>
  <xs:element ref="paymentMethod"/>
  <xs:choice>
    <xs:element name="orderAccount" minOccurs="0">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="1"/>
          <xs:maxLength value="20"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="creditCardInfo" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>This element should be managed in more secure way</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:choice>
</xs:sequence>

<!-- ====================================================================== -->
<!--=====================      Order Monitoring      ======================= -->
<!-- ====================================================================== -->

<xs:complexType name="GetStatusRequestType">
  <xs:complexContent>
    <xs:extension base="OrderRequestBaseType">
      <xs:sequence>
        <xs:element name="timeStamp" type="xs:dateTime" minOccurs="0"/>
        <xs:choice>
          <xs:element ref="orderId"/>
          <xs:element name="filteringCriteria" type="OrderSearchCriteriaType"/>
        </xs:choice>
        <xs:element name="presentation" type="PresentationType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="GetStatusResponseType">
  <xs:complexContent>
    <xs:extension base="OrderResponseBaseType">
      <xs:sequence>
        <xs:element ref="orderMonitorSpecification" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="CommonOrderMonitorSpecification">
  <xs:complexContent>
    <xs:extension base="CommonOrderSpecification">
      <xs:sequence>
        <xs:element ref="orderId"/>
        <xs:element name="orderStatusInfo" type="StatusType"/>
        <xs:element name="orderDateTime" type="xs:dateTime" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:element name="orderMonitorSpecification" type="CommonOrderMonitorSpecification" abstract="false"/>
<xs:complexType name="OrderSearchCriteriaType">
  <xs:sequence>
    <xs:element name="orderId"/>
    <xs:element name="orderDateTime" type="xs:dateTime" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="filteringCriteria" type="OrderSearchCriteriaType" abstract="false"/>
  </xs:sequence>
</xs:complexType>

order to perform GetStatus request with brief presentation. Of course orders without order items cannot be submitted. This element should be managed in more secure way.
<xs:element name="lastUpdate" type="DateTimeTypes" minOccurs="0"/>
<xs:element name="orderStatus" type="EnumStatusType" minOccurs="0" maxOccurs="unbounded"/>
<xs:element ref="orderReference" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<!-- ====================================================================== -->
<!--=====================      Order Item      ======================= -->
<!-- ====================================================================== -->
<xs:complexType name="CommonOrderItemType">
<xs:sequence>
<xs:element ref="itemId"/>
<xs:element ref="productOrderOptionsId"/>
<xs:element name="orderItemRemark" minOccurs="0">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:maxLength value="255"/>
</xs:restriction>
</xs:simpleType>
</xs:element>

<xs:element name="options" minOccurs="0" maxOccurs="unbounded">
<xs:complexType>
<xs:sequence>
<xs:element ref="sps:Parameter"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="sceneSelection" type="SceneSelectionType" minOccurs="0"/>
<xs:element name="deliveryMethod" type="DeliveryMethodType" minOccurs="0"/>
<xs:element name="packageMedium" type="PackageMedium" minOccurs="0"/>
<xs:element name="numberOfCopies" type="xs:int" minOccurs="0">
<xs:annotation>
<xs:documentation>In case of mail delivery, the number of copies to be delivered can
be specified.</xs:documentation>
</xs:annotation>
</xs:element>

<xs:element name="payment" type="PaymentOptionSelectedValue" minOccurs="0"/>
<xs:choice>
<xs:element ref="productId"/>
<xs:element ref="taskingRequestId"/>
<xs:element ref="subscriptionId"/>
</xs:choice>
</xs:sequence>
</xs:complexType>
<xs:complexType name="OrderItemIdType" abstract="false">
<xs:annotation>
<xs:documentation>Root type of the hierarchy of order item identifiers</xs:documentation>
</xs:annotation>
</xs:complexType>
<xs:complexType name="ProductIdType">
<xs:annotation>
<xs:documentation>Identifier of catalogued product</xs:documentation>
</xs:annotation>
<xs:complexContent>
<xs:extension base="OrderItemIdType">
<xs:sequence>
<xs:element ref="identifier"/>
<xs:element ref="collectionId" minOccurs="0"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:element name="productId" type="ProductIdType"/>
<xs:complexType name="SubscriptionIdType">
<xs:annotation>
<xs:documentation>Identifier of subscription</xs:documentation>
</xs:annotation>
<xs:complexContent>
<xs:extension base="OrderItemIdType">
<xs:sequence>
<xs:element ref="identifier"/>
<xs:element ref="collectionId" minOccurs="0"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:element name="subscriptionId" type="SubscriptionIdType"/>
<xs:complexType name="NumOfCopiesInPackageType">
<xs:annotation>
<xs:documentation>The number of copies to be included in the package</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element ref="numberOfCopies"/>
<xs:element ref="packageMedium"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="orderItemRemark" minOccurs="0" maxOccurs="0">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:maxLength value="255"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="options" minOccurs="0" maxOccurs="unbounded">
<xs:complexType>
<xs:sequence>
<xs:element ref="sps:Parameter"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="sceneSelection" type="SceneSelectionType" minOccurs="0"/>
<xs:element name="deliveryMethod" type="DeliveryMethodType" minOccurs="0"/>
<xs:element name="packageMedium" type="PackageMedium" minOccurs="0"/>
<xs:element name="numberOfCopies" type="xs:int" minOccurs="0">
<xs:annotation>
<xs:documentation>In case of mail delivery, the number of copies to be delivered can
be specified.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="payment" type="PaymentOptionSelectedValue" minOccurs="0"/>
<xs:choice>
<xs:element ref="productId"/>
<xs:element ref="taskingRequestId"/>
<xs:element ref="subscriptionId"/>
</xs:choice>
</xs:sequence>
</xs:complexType>
<xs:complexType name="OrderaddItemIdType" abstract="false">
<xs:annotation>
<xs:documentation>Root type of the hierarchy of order item identifiers</xs:documentation>
</xs:annotation>
</xs:complexType>
<xs:complexType name="ProductIdType">
<xs:annotation>
<xs:documentation>Identifier of catalogued product</xs:documentation>
</xs:annotation>
<xs:complexContent>
<xs:extension base="OrderaddItemIdType">
<xs:sequence>
<xs:element ref="identifier"/>
<xs:element ref="collectionId" minOccurs="0"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:element name="productId" type="ProductIdType"/>
<xs:complexType name="SubscriptionIdType">
<xs:annotation>
<xs:documentation>Identifier of subscription</xs:documentation>
</xs:annotation>
<xs:complexContent>
<xs:extension base="OrderaddItemIdType">
<xs:sequence>
<xs:element ref="identifier"/>
<xs:element ref="collectionId" minOccurs="0"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:element name="subscriptionId" type="SubscriptionIdType"/>
It can be set if status is incomplete. It shall be set when status is success. This element is not set when status is failure.
<xs:complexType name="ParameterDescriptorType">
  <xs:complexContent>
    <xs:extension base="sps:ParameterDescriptorType">
      <xs:sequence>
        <xs:element ref="identifier" minOccurs="0"/>
        <xs:element name="grouping" minOccurs="0">
          <xs:annotation>
            <xs:documentation>
              It identifies the group the option belong to e.g. processing option, etc. </xs:documentation>
          </xs:annotation>
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:maxLength value="40"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<!-- ====================================================================== -->
<!--============    Order Quotation Definitions    ========================-->
<!-- ====================================================================== -->

<xs:complexType name="GetQuotationRequestType">
  <xs:complexContent>
    <xs:extension base="OrderRequestBaseType">
      <xs:sequence>
        <xs:element name="timeStamp" type="xs:dateTime" minOccurs="0"/>
        <xs:choice>
          <xs:element ref="orderSpecification">
            <xs:annotation>
              <xs:documentation>
                This choice is set for getting the quotation of an order.
              </xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="quotationId" type="QuotationIdType">
            <xs:annotation>
              <xs:documentation>
                This choice is set when quotation monitoring is supported and a quotation request has been already submitted.
              </xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:choice>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="GetQuotationAckType">
  <xs:complexContent>
    <xs:extension base="OrderResponseBaseType">
      <xs:sequence minOccurs="0">
        <xs:annotation>
          <xs:documentation>
            The element is set when status is different from failure.
          </xs:documentation>
        </xs:annotation>
        <xs:choice>
          <xs:element name="quotationId" type="QuotationIdType">
            <xs:annotation>
              <xs:documentation>
                This choice is set in case of non sync quotations.
              </xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="quotation" type="OrderQuotation">
            <xs:annotation>
              <xs:documentation>
                This choice is set in case of synchronous quotations or as answer to quotation monitoring.
              </xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:choice>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
Order Services for Earth Observation Products

<xs:complexType name="GetQuotationResponseRequestType">
<xs:complexContent>
<xs:extension base="OrderRequestBaseType">
<xs:sequence>
<xs:element name="status" type="OrderResponseStatusType" nillable="false"/>
<xs:element name="errorMessage" minOccurs="0">
<xs:annotation>
<xs:documentation>This field is set when status element is different from success. It provides some information about the occurred problem.</xs:documentation>
</xs:annotation>
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:maxLength value="255"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="quotation" type="OrderQuotation"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

<xs:complexType name="GetQuotationResponseAckType">
<xs:complexContent>
<xs:extension base="OrderResponseBaseType"/>
</xs:complexContent>
</xs:complexType>

<xs:complexType name="OrderQuotation">
<xs:sequence>
<xs:element name="quotationId" type="QuotationIdType"/>
<xs:element name="validityTime" type="xs:dateTime" minOccurs="0"/>
<xs:element name="price" type="CurrencyType" minOccurs="0">
<xs:annotation>
<xs:documentation>Price of the whole order; mandatory unless the provider uses quota concept or products are free of charge.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="orderItemGroupPrice" type="OrderItemGroupPrice" maxOccurs="unbounded"/>
<xs:element ref="contractInformation" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

<xs:complexType name="OrderItemGroupPrice">
<xs:sequence>
<xs:element name="provider" type="ProviderType"/>
<xs:element name="quotationId" type="QuotationIdType" minOccurs="0"/>
<xs:element name="validityTime" type="xs:dateTime" minOccurs="0"/>
<xs:element name="price" type="CurrencyType"/>
<xs:annotation>
<xs:documentation>Price of the whole order item group.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="balance" type="CurrencyType" minOccurs="0"/>
<xs:element name="orderItemPrice" type="OrderItemPrice" maxOccurs="unbounded"/>
<xs:element ref="contractInformation" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

<xs:complexType name="OrderItemPrice">
<xs:sequence>
<xs:element ref="itemId">
<xs:annotation>
<xs:documentation>string identifying the order item within the whole order.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:choice>
<xs:element ref="productId"/>
<xs:element ref="taskingRequestId"/>
</xs:choice>
</xs:sequence>
</xs:complexType>
<xs:element ref="subscriptionId"/>
<xs:choice>
<xs:element name="price" type="CurrencyType" minOccurs="0">
<xs:annotation>
<xs:documentation>Price of the item; is optional if the price at group level is provided; not supported in case the products are free of charge.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element ref="contractInformation" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="CurrencyType">
<xs:sequence>
<xs:element name="value" type="xs:double"/>
<xs:element name="currency">
<xs:annotation>
<xs:documentation>Currency including ISO 4217 (e.g.: EUR, USD (US Dollar), CAD (Canada Dollar), AUD (Australia Dollar), GBP (United Kindom Pounds), etc.) and also special values for representing quota.</xs:documentation>
</xs:annotation>
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:maxLength value="10"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ProviderType">
<xs:sequence>
<xs:element name="serviceName">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:maxLength value="40"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="organization">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:maxLength value="40"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="QuotationIdType">
<xs:restriction base="xs:anyURI"/>
</xs:simpleType>
<!-- ====================================================================== -->
<!--============    DescribeResultAccess Definitions    ========================-->
<!-- ====================================================================== -->
<xs:complexType name="DescribeResultAccessRequestType">
<xs:complexContent>
<xs:extension base="OrderRequestBaseType">
<xs:sequence>
<xs:element name="timeStamp" type="xs:dateTime" minOccurs="0"/>
<xs:element ref="orderId"/>
<xs:element name="subFunction">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="allReady"/>
<xs:enumeration value="nextReady"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="DescribeResultAccessResponseType">
  <xs:complexContent>
    <xs:extension base="OrderResponseBaseType">
      <xs:sequence>
        <xs:element name="URLs" type="ItemURLType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<!-- Generic Definitions -->

<xs:simpleType name="EnumStatusType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Submitted"/>
    <xs:enumeration value="Accepted"/>
    <xs:enumeration value="Cancelled"/>
    <xs:enumeration value="Completed"/>
    <xs:enumeration value="InProduction"/>
    <xs:enumeration value="Suspended"/>
    <xs:enumeration value="Failed"/>
    <xs:enumeration value="Terminated"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="StatusType">
  <xs:sequence>
    <xs:element name="status" type="EnumStatusType"/>
    <xs:element name="additionalStatusInfo" minOccurs="0">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="1"/>
          <xs:maxLength value="255"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="missionSpecificStatusInfo" minOccurs="0">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:maxLength value="255"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="SceneSelectionType">
  <xs:sequence>
    <xs:element name="sceneType" type="xs:string" maxOccurs="1"/>
    <xs:element name="sceneCoordinates">
      <xs:complexType>
        <xs:choice>
          <xs:element ref="gml:Point"/>
          <xs:element ref="gml:Polygon"/>
          <xs:element ref="gml:Rectangle"/>
        </xs:choice>
      </xs:complexType>
    </xs:element>
    <xs:element name="albumExtract" minOccurs="0">
      <xs:annotation>
        <xs:documentation>source data extract defined by its bounding box with album catalog coordinates (impossible for mosaic or cropping)</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:complexType>
  <xs:sequence>
    <xs:element name="firstRow" type="xs:int"/>
    <xs:element name="nbRow" type="xs:int"/>
    <xs:element name="firstCol" type="xs:int"/>
    <xs:element name="nbCol" type="xs:int"/>
  </xs:sequence>
</xs:complexType>

<xs:element name="temporalSelection" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="startDateTime" type="xs:dateTime"/>
      <xs:element name="endDateTime" type="xs:dateTime"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="scenePosition" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="40"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="sceneSize" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="40"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="DeliveryInformationType">
  <xs:sequence>
    <xs:element name="ftp" type="FTPAddressType" minOccurs="0"/>
    <xs:element name="mail" type="DeliveryAddressType" minOccurs="0"/>
    <xs:element name="e-mail" minOccurs="0">
      <xs:annotation>
        <xs:documentation>E-mail address of the issuer of the request</xs:documentation>
      </xs:annotation>
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:maxLength value="40"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="receiverAddress" minOccurs="0">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:maxLength value="20"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="DeliveryAddressType">
  <xs:sequence>
    <xs:element name="recipient" minOccurs="0">
      <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="40"/>
      </xs:restriction>
    </xs:element>
    <xs:element name="companyRef" minOccurs="0">
      <xs:restriction base="xs:string"/>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:element name="postalAddress" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="streetAddress">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:maxLength value="40"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="city">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:maxLength value="40"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="state">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:maxLength value="40"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="postalCode">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:maxLength value="12"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="country">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:maxLength value="40"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="postBox">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:maxLength value="12"/>
          </xs:restriction>
          <xs:annotation>
            <xs:documentation>only number part, only digits
            </xs:documentation>
        </xs:simpleType>
      </xs:element>
      <xs:element name="telephoneNumber" minOccurs="0">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:maxLength value="18"/>
          </xs:restriction>
          <xs:annotation>
            <xs:documentation>including country code, prefix, without special sign or intermediate blanks</xs:documentation>
          </xs:annotation>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="facsimileTelephoneNumber" minOccurs="0">
  <xs:annotation>
    <xs:documentation>including country code, prefix, without special sign or intermediate blanks</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="18"/>
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:complexType name="CancelRequestType">
  <xs:complexContent>
    <xs:extension base="OrderRequestBaseType">
      <xs:sequence>
        <xs:element name="timeStamp" type="xs:dateTime" minOccurs="0"/>
        <xs:element ref="orderId"/>
        <xs:element ref="statusNotification"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="CancelRequestAckType">
  <xs:complexContent>
    <xs:extension base="OrderResponseBaseType"/>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="StatusNotificationType">
  <xs:complexContent>
    <xs:extension base="OrderRequestBaseType">
      <xs:sequence>
        <xs:element name="timeStamp" type="xs:dateTime" minOccurs="0"/>
        <xs:element ref="orderMonitorSpecification"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="StatusNotificationAckType">
  <xs:complexContent>
    <xs:extension base="OrderResponseBaseType"/>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="FTPAddressType">
  <xs:sequence>
    <xs:element name="serverAddress" type="xs:string"/>
    <xs:element name="userId">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:maxLength value="8"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="userPassword" type="xs:string"/>
    <xs:element name="directory">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:maxLength value="1024"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="OnLineAccessAddressType">
  <xs:sequence>
    <xs:element name="ServiceAddress" minOccurs="0">
      <xs:annotation>
        <xs:documentation></xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="ServiceAddress" minOccurs="0">
            <xs:simpleType>
              <xs:restriction base="xs:string">
                <xs:maxLength value="1024"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:element>
          <xs:element name="ServiceName">...
          <xs:element name="ServiceType">...
          <xs:element name="ServiceDescription">...
          <xs:element name="ServiceAccess">...
          <xs:element name="ServiceAvailability">...
          <xs:element name="ServiceSubscription">...
          <xs:element name="ServiceContact">...
          <xs:element name="ServiceLocation">...
          <xs:element name="ServiceUsage">...
          <xs:element name="ServiceLicense">...
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
Address information related to the Server hosting the item to be accessed.

- `<xs:documentation>` It specifies the type of the service hosting the resource e.g. WCS, WMS, etc. This field is optional, since full information about the server can be retrieved from info_URL and infoRequest elements.

- `<xs:documentation>` Address of the Server.

- `<xs:documentation>` Address for getting information about the server. In case of OGC Web Services it can refer to GetCapabilities operation with HTTP GET binding.

- `<xs:documentation>` In case the information can be retrieved by sending a request message, this field stores the message to be sent. In case of OGC Web Services supporting SOAP GetCapabilities, this field specifies the GetCapabilities message to send at the address specified in info_URL element.

- `<xs:documentation>` Address information of the resource to be accessed.

- `<xs:documentation>` URL for accessing the resource.

- `<xs:documentation>` In case the resource cannot be accessed simply via the URL, but a message needs to be sent, then this field specifies the message to send.
<xs:complexType name="ItemURLType">
  <xs:sequence>
    <xs:element ref="itemId"/>
    <xs:element name="itemAddress" type="OnLineAccessAddressType">
      <xs:annotation>
        <xs:documentation>This field specifies the full information for accessing the ordered item.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="expirationDate" type="xs:dateTime" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="EnumOrderType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="PRODUCT_ORDER"/>
    <xs:enumeration value="SUBSCRIPTION_ORDER"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="EnumPackagingType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="zip"/>
    <xs:enumeration value="tar"/>
    <xs:enumeration value="tgz"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="DateTimeTypes">
  <xs:union memberTypes="xs:date xs:dateTime"/>
</xs:simpleType>

<xs:simpleType name="OrderResponseStatusType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="success"/>
    <xs:enumeration value="partial"/>
    <xs:enumeration value="failure"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="PresentationType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="brief"/>
    <xs:enumeration value="full"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="PriorityType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="STANDARD"/>
    <xs:enumeration value="FAST_TRACK"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="CodeWithAuthorityType">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="codeSpace" type="xs:anyURI" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="OrderType">
  <xs:sequence>
    <xs:element name="identifier" type="CodeWithAuthorityType"/>
    <xs:element name="itemId">
      <xs:annotation>
        <xs:documentation>string identifying the order item within the order.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="itemId">
      <xs:annotation>
        <xs:documentation>string identifying the order item within the order.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="itemId">
      <xs:annotation>
        <xs:documentation>string identifying the order item within the order.</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:schema>

  <xs:element name="paymentMethod">
    <xs:annotation>
      <xs:documentation>Examples: invoice, prepay (to be indicated for free products), deposit account, credit card, credit card previously supplied</xs:documentation>
    </xs:annotation>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="40"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>

  <xs:element name="contractInformation">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:maxLength value="1024"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>

  <xs:element name="orderReference">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="30"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>

  <xs:element name="productOrderOptionsId">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="40"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>

</xs:schema>
APPENDIX B  WSDL Specification (Informative)

<?xml version="1.0" encoding="UTF-8"?>
xmlns:tns="http://earth.esa.int/hma/ordering" xmlns:ns1="http://www.w3.org/2001/XMLSchema"
xmlns:ns6="http://www.opengis.net/gml"
xmlns:ns7="http://www.opengis.net/ows" xmlns:ns8="http://www.opengis.net/sps/0"
xmlns:ns9="http://www.opengis.net/swe/0.0"
xmlns:ns10="http://www.opengis.net/swe/0.0/" xmlns:ns11="http://www.w3.org/2001/XMLSchema"
xmlns:ns12="http://www.w3.org/2001/XMLSchema" targetNamespace="http://earth.esa.int/hma/ordering">
<types>
<schema attributeFormDefault="qualified" elementFormDefault="qualified" targetNamespace="http://earth.esa.int/hma/ordering">
<import namespace="http://earth.esa.int/hma/ordering" schemaLocation="Order.xsd"/>
<import namespace="http://www.w3.org/2001/XMLSchema" schemaLocation="../ws-addressing/ws-address.xsd"/>
</schema>
</types>
<message name="GetCapabilitiesRequest">
<part name="parameter" element="tns:GetCapabilities"/>
</message>
<message name="GetCapabilitiesResponse">
<part name="parameter" element="tns:Capabilities"/>
</message>
<message name="getOptionsRequest">
<part name="GetOptionsRequestParameter" element="tns:GetOptions"/>
</message>
<message name="getOptionsResponse">
<part name="GetOptionsResponseParameter" element="tns:GetOptionsResponse"/>
</message>
<message name="getQuotationRequest">
<part name="GetQuotationRequestParameter" element="tns:GetQuotation"/>
</message>
<message name="getQuotationAck">
<part name="GetQuotationAckParameter" element="tns:GetQuotationAck"/>
</message>
<message name="getQuotationResponse">
<part name="GetQuotationResponseParameter" element="tns:GetQuotationResponse"/>
</message>
<message name="getQuotationResponseAck">
<part name="GetQuotationResponseAckParameter" element="tns:GetQuotationResponseAck"/>
</message>
<message name="submitRequest">
<part name="SubmitRequestParameter" element="tns:Submit"/>
</message>
<message name="submitAck">
<part name="SubmitAckParameter" element="tns:SubmitAck"/>
</message>
<message name="submitResponse">
<part name="SubmitResponseParameter" element="tns:SubmitResponse"/>
</message>
<message name="submitResponseAck">
<part name="SubmitResponseAckParameter" element="tns:SubmitResponseAck"/>
</message>
<message name="getStatusRequest">
<part name="GetStatusRequestParameter" element="tns:GetStatus"/>
</message>
<message name="getStatusResponse">
<part name="GetStatusResponseParameter" element="tns:GetStatusResponse"/>
</message>
<message name="cancelRequest">
<part name="CancelRequestParameter" element="tns:Cancel"/>
</message>
<message name="cancelAck">
<part name="CancelAckParameter" element="tns:CancelAck"/>
</message>
</definitions>
<message name="cancelResponse">
  <part name="CancelResponseParameter" element="tns:CancelResponse"/>
</message>
<message name="cancelResponseAck">
  <part name="CancelResponseAckParameter" element="tns:CancelResponseAck"/>
</message>
<message name="describeResultAccess">
  <part name="DescribeResultAccessParameter" element="tns:DescribeResultAccess"/>
</message>
<message name="describeResultAccessResponse">
  <part name="DescribeResultAccessResponseParameter" element="tns:DescribeResultAccessResponse"/>
</message>
<message name="StartHeader">
  <part name="MessageID" element="wsa:MessageID"/>
  <part name="ReplyTo" element="wsa:ReplyTo"/>
</message>
<message name="ContinueHeader">
  <part name="RelatesTo" element="wsa:RelatesTo"/>
</message>
<message name="ServiceExceptionReport">
  <part name="fault" element="ns7:ExceptionReport"/>
</message>
<portType name="SOAPport">
  <operation name="GetCapabilities">
    <input message="tns:GetCapabilitiesRequest"/>
    <output message="tns:GetCapabilitiesResponse"/>
    <fault name="ServiceExceptionReport" message="tns:ServiceExceptionReport"/>
  </operation>
  <operation name="GetOptions">
    <input message="tns:getOptionsRequest"/>
    <output message="tns:getOptionsResponse"/>
    <fault name="ServiceExceptionReport" message="tns:ServiceExceptionReport"/>
  </operation>
  <operation name="GetQuotation">
    <input message="tns:getQuotationRequest"/>
    <output message="tns:getQuotationAck"/>
    <fault name="ServiceExceptionReport" message="tns:ServiceExceptionReport"/>
  </operation>
  <operation name="GetQuotationResponse">
    <input message="tns:getQuotationResponse"/>
    <output message="tns:getQuotationResponseAck"/>
    <fault name="ServiceExceptionReport" message="tns:ServiceExceptionReport"/>
  </operation>
  <operation name="Submit">
    <input message="tns:submitRequest"/>
    <output message="tns:submitAck"/>
    <fault name="ServiceExceptionReport" message="tns:ServiceExceptionReport"/>
  </operation>
  <operation name="SubmitResponse">
    <input message="tns:submitResponse"/>
    <output message="tns:submitResponseAck"/>
    <fault name="ServiceExceptionReport" message="tns:ServiceExceptionReport"/>
  </operation>
  <operation name="GetStatus">
    <input message="tns:getStatusRequest"/>
    <output message="tns:getStatusResponse"/>
    <fault name="ServiceExceptionReport" message="tns:ServiceExceptionReport"/>
  </operation>
  <operation name="Cancel">
    <input message="tns:cancelRequest"/>
    <output message="tns:cancelAck"/>
    <fault name="ServiceExceptionReport" message="tns:ServiceExceptionReport"/>
  </operation>
  <operation name="CancelResponse">
    <input message="tns:cancelResponse"/>
    <output message="tns:cancelResponseAck"/>
    <fault name="ServiceExceptionReport" message="tns:ServiceExceptionReport"/>
<fault name="ServiceExceptionReport">
  <soap:fault name="ServiceExceptionReport" use="literal"/>
</fault>
</operation>

<operation name="SubmitResponse">
  <soap:operation soapAction="SubmitResponse"/>
  <input>
    <soap:body use="literal"/>
  </input>
  <output>
    <soap:body use="literal"/>
  </output>
  <fault name="ServiceExceptionReport">
    <soap:fault name="ServiceExceptionReport" use="literal"/>
  </fault>
</operation>

<operation name="GetStatus">
  <soap:operation soapAction="GetStatus"/>
  <input>
    <soap:body use="literal"/>
  </input>
  <output>
    <soap:body use="literal"/>
  </output>
  <fault name="ServiceExceptionReport">
    <soap:fault name="ServiceExceptionReport" use="literal"/>
  </fault>
</operation>

<operation name="Cancel">
  <soap:operation soapAction="Cancel"/>
  <input>
    <soap:body use="literal"/>
    <soap:header message="StartHeader" part="ReplyTo" use="literal"/>
    <soap:header message="StartHeader" part="MessageID" use="literal"/>
  </input>
  <output>
    <soap:body use="literal"/>
  </output>
  <fault name="ServiceExceptionReport">
    <soap:fault name="ServiceExceptionReport" use="literal"/>
  </fault>
</operation>

<operation name="CancelResponse">
  <soap:operation soapAction="CancelResponse"/>
  <input>
    <soap:body use="literal"/>
    <soap:header message="ContinueHeader" part="RelatesTo" use="literal"/>
  </input>
  <output>
    <soap:body use="literal"/>
  </output>
  <fault name="ServiceExceptionReport">
    <soap:fault name="ServiceExceptionReport" use="literal"/>
  </fault>
</operation>

<operation name="DescribeResultAccess">
  <soap:operation soapAction="DescribeResultAccess"/>
  <input>
    <soap:body use="literal"/>
  </input>
  <output>
    <soap:body use="literal"/>
  </output>
  <fault name="ServiceExceptionReport">
    <soap:fault name="ServiceExceptionReport" use="literal"/>
  </fault>
</operation>
<service name="HMA_OrderingService">
  <port name="HMA_OrderingServicePort" binding="tns:HMA_OrderingBinding">
    <soap:address location="http://earth.esa.int/"/>
  </port>
</service>

<!-- Modified import schema in order to define this WSDL as WS-I Basic Profile compliant -->

</definitions>
APPENDIX C   Examples

C.1      GetCapabilities

C.1.1   Request
<?xml version="1.0" encoding="UTF-8"?>
xml:namespace="http://www.opengis.net/ows"
SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">
<SOAP-ENV:Body>
<GetCapabilities
xmlns="http://earth.esa.int/hma/ordering"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:m0="http://www.opengis.net/ows">

<m0:AcceptVersions>
<m0:Version>1.0.0</m0:Version>
</m0:AcceptVersions>
</GetCapabilities>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

C.1.2   Response
<?xml version="1.0" encoding="UTF-8"?>
xml:namespace="http://www.opengis.net/ows"
SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">
<SOAP-ENV:Body>
<Capabilities
xmlns:ows="http://www.opengis.net/ows"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns="http://earth.esa.int/hma/ordering"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:m0="http://www.opengis.net/ows" version="0.00.00">
<ows:ServiceIdentification>
<ows:Title>ESA Order Service</ows:Title>
<ows:Abstract>Service for ordering Earth observation products</ows:Abstract>
<ows:Keywords>
<ows:Keyword>OS</ows:Keyword>
<ows:Type codeSpace="http://www.xmlspy.com">String</ows:Type>
</ows:Keywords>
<ows:ServiceType codeSpace="http://www.opengis.net" OGC:OS:ServiceType>
<ows:ServiceTypeVersion>0.01.00</ows:ServiceTypeVersion>
<ows:Fees>NONE</ows:Fees>
<ows:AccessConstraints>NONE</ows:AccessConstraints>
</ows:ServiceIdentification>
<ows:ServiceProvider>
<ows:ProviderName>ESA EECF</ows:ProviderName>
<ows:ProviderSite/>
<ows:ServiceContact>
<ows:IndividualName>John Smith</ows:IndividualName>
<ows:PositionName>EO Help Desk Operator</ows:PositionName>
<ows:ContactInfo>
<ows:Voice>+39 06 90 180 999</ows:Voice>
</ows:ContactInfo>
</ows:ServiceContact>
</ows:ServiceProvider>
<ows:OperationsMetadata>
<ows:Operation name="GetCapabilities">
<ows:DCP>HTTP</ows:DCP>
</ows:Operation>
</ows:OperationsMetadata>
</Capabilities>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
C.2   GetOptions

C.2.1   Product Order Request

<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
    <ows:Operation name="GetOptions">
      <ows:DCP>
        <ows:HTTP>
          <ows:Post xlink:href="http://earth.esa.int"/>
        </ows:HTTP>
      </ows:DCP>
    </ows:Operation>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
C.2.2 Product Order Response

<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
      xsi:schemaLocation="http://earth.esa.int/hma/ordering ..\Order.xsd">
      <status>success</status>
      <orderOptions>
        <productOrderOptionsId>Level 1, Product PRI (ASA_IMP)</productOrderOptionsId>
        <description>Order Options for producing ASAR IMP 1P product</description>
        <orderType>PRODUCT_ORDER</orderType>
        <options parameterID="processingLevel" use="required" updateable="false">
          <sps:Description>Product Level.</sps:Description>
          <sps:definition>
            <sps:commonData>
              <swe:Category>
                <swe:constraint>
                  <swe:AllowedTokens>
                    <swe:valueList>1B</swe:valueList>
                  </swe:AllowedTokens>
                </swe:constraint>
              </swe:Category>
            </sps:commonData>
            <sps:cardinality>1</sps:cardinality>
            <grouping>Processing Option</grouping>
          </options>
          <options parameterID="productType" use="required" updateable="false">
            <sps:Description>Product type.</sps:Description>
            <sps:definition>
              <sps:commonData>
                <swe:Category>
                  <swe:constraint>
                    <swe:AllowedTokens>
                      <swe:valueList>ASA_IMP_1P</swe:valueList>
                    </swe:AllowedTokens>
                  </swe:constraint>
                </swe:Category>
              </sps:commonData>
              <sps:cardinality>1</sps:cardinality>
              <grouping>Processing Option</grouping>
            </options>
            <options parameterID="qualityOfService" use="required" updateable="false">
              <sps:Description>Quality of Service</sps:Description>
              <sps:definition>
                <sps:commonData>
                  <swe:Category>
                    <swe:constraint>
                      <swe:AllowedTokens>
                        <swe:valueList>STANDARD NRT RUSH</swe:valueList>
                      </swe:AllowedTokens>
                    </swe:constraint>
                  </swe:Category>
                </sps:commonData>
                <sps:cardinality>1</sps:cardinality>
                <grouping>Processing Option</grouping>
              </options>
            </options>
          </options>
        </options>
      </GetOptionsResponse>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
</SOAP-ENV:Envelope>
to a band (R,G,B). This is to allow a user to form a composite colored image.
<sps:definition>
  <sps:commonData>
    <swe:Category>
      <swe:constraint>
        <swe:AllowedTokens>
          <swe:valueList>1B</swe:valueList>
        </swe:AllowedTokens>
      </swe:constraint>
    </swe:Category>
  </sps:commonData>
  <sps:cardinality>1</sps:cardinality>
  <grouping>Processing Option</grouping>
</options>

<options parameterID="productType" use="required" updateable="false">
  <sps:Description>Product type.</sps:Description>
  <sps:definition>
    <sps:commonData>
      <swe:Category>
        <swe:constraint>
          <swe:AllowedTokens>
            <swe:valueList>ASA_IMG_1P</swe:valueList>
          </swe:AllowedTokens>
        </swe:constraint>
      </swe:Category>
    </sps:commonData>
    <sps:cardinality>1</sps:cardinality>
    <grouping>Processing Option</grouping>
  </options>

  <options parameterID="qualityOfService" use="required" updateable="false">
    <sps:Description>Quality of Service.</sps:Description>
    <sps:definition>
      <sps:commonData>
        <swe:Category>
          <swe:constraint>
            <swe:AllowedTokens>
              <swe:valueList>STANDARD NRT RUSH</swe:valueList>
            </swe:AllowedTokens>
          </swe:constraint>
        </swe:Category>
      </sps:commonData>
      <sps:cardinality>1</sps:cardinality>
      <grouping>Processing Option</grouping>
    </options>
  </options>
</options>

<productDeliveryOptions>
  <deliveryMethod>mail</deliveryMethod>
  <packageMedium>CD-ROM</packageMedium>
</productDeliveryOptions>

<orderOptionInfoURL>http://www.provider.com/envisat/orderoptions.html</orderOptionInfoURL>

<orderOptions>
  <orderOptionInfoURL>http://www.provider.com/envisat/orderoptions.html</orderOptionInfoURL>
</orderOptions>

<productOrderOptionsId>Level 1,Product SLC (ASA_IMS)</productOrderOptionsId>
<description>Order Options for producing ASAR IMS 1P product</description>
<orderType>PRODUCT_ORDER</orderType>

<options parameterID="processingLevel" use="required" updateable="false">
  <sps:Description>Product Level.</sps:Description>
  <sps:definition>
<sps:commonData>
<swe:Category>
<swe:constraint>
<swe:AllowedTokens>
<swe:valueList>1B</swe:valueList>
</swe:AllowedTokens>
</swe:constraint>
</swe:Category>
</sps:commonData>
</sps:definition>
<sps:cardinality>1</sps:cardinality>
<grouping>Processing Option</grouping>
</options>
<options parameterID="productType" use="required" updateable="false">
<sps:Description>Product type.</sps:Description>
<sps:definition>
<sps:commonData>
<swe:Category>
<swe:constraint>
<swe:AllowedTokens>
<swe:valueList>ASA_IMS_1P</swe:valueList>
</swe:AllowedTokens>
</swe:constraint>
</swe:Category>
</sps:commonData>
</sps:definition>
<sps:cardinality>1</sps:cardinality>
<grouping>Processing Option</grouping>
</options>
<options parameterID="qualityOfService" use="required" updateable="false">
<sps:Description>Quality of Service.</sps:Description>
<sps:definition>
<sps:commonData>
<swe:Category>
<swe:constraint>
<swe:AllowedTokens>
<swe:valueList>STANDARD NRT RUSH</swe:valueList>
</swe:AllowedTokens>
</swe:constraint>
</swe:Category>
</sps:commonData>
</sps:definition>
<sps:cardinality>1</sps:cardinality>
<grouping>Processing Option</grouping>
</options>
</productDeliveryOptions>
<deliveryMethod>mail</deliveryMethod>
<packageMedium>CD-ROM</packageMedium>
</productDeliveryOptions>
</orderOptionInfoURL>http://www.provider.com/envisat/orderoptions.html</orderOptionInfoURL>
</paymentOptions>
</paymentOptions>
<sceneSelectionOption>
<sceneType>Floating</sceneType>
</sceneSelectionOption>
</orderOptions>
<orderOptions>
<productOrderOptionsId>Level 1,Product MRI (ASA_IMM)</productOrderOptionsId>
<description>Order Options for producing ASAR IMM 1P product</description>
<orderType>PRODUCT_ORDER</orderType>
<options parameterID="processingLevel" use="required" updateable="false">
<sps:Description>Product level</sps:Description>
<sps:definition>
<sps:commonData>
<swe:Category>
<swe:constraint>
<swe:AllowedTokens>
<swe:valueList>
</swe:AllowedTokens>
</swe:constraint>
</swe:Category>
</sps:commonData>
</sps:definition>
<sps:cardinality>1</sps:cardinality>
<grouping>Processing Option</grouping>
</options>
<options parameterID="productType" use="required" updateable="false">
<sps:Description>Product type.</sps:Description>
<sps:definition>
<sps:commonData>
<swe:Category>
<swe:constraint>
<swe:AllowedTokens>
<swe:valueList>ASA_IMS_1P</swe:valueList>
</swe:AllowedTokens>
</swe:constraint>
</swe:Category>
</sps:commonData>
</sps:definition>
<sps:cardinality>1</sps:cardinality>
<grouping>Processing Option</grouping>
</options>
<options parameterID="qualityOfService" use="required" updateable="false">
<sps:Description>Quality of Service.</sps:Description>
<sps:definition>
<sps:commonData>
<swe:Category>
<swe:constraint>
<swe:AllowedTokens>
<swe:valueList>STANDARD NRT RUSH</swe:valueList>
</swe:AllowedTokens>
</swe:constraint>
</swe:Category>
</sps:commonData>
</sps:definition>
<sps:cardinality>1</sps:cardinality>
<grouping>Processing Option</grouping>
</options>
</productDeliveryOptions>
<deliveryMethod>mail</deliveryMethod>
<packageMedium>CD-ROM</packageMedium>
</productDeliveryOptions>
</orderOptionInfoURL>http://www.provider.com/envisat/orderoptions.html</orderOptionInfoURL>
</paymentOptions>
</paymentOptions>
<sceneSelectionOption>
<sceneType>Floating</sceneType>
</sceneSelectionOption>
</orderOptions>
<orderOptions>
<productOrderOptionsId>Level 1,Product MRI (ASA_IMM)</productOrderOptionsId>
<description>Order Options for producing ASAR IMM 1P product</description>
<orderType>PRODUCT_ORDER</orderType>
<options parameterID="processingLevel" use="required" updateable="false">
<sps:Description>Product level</sps:Description>
<sps:definition>
<sps:commonData>
C.2.3 Subscription with restrictions Request

<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
    <GetOptionsResponse>
      <productDeliveryOptions>
        <deliveryMethod>mail</deliveryMethod>
        <packageMedium>CD-ROM</packageMedium>
      </productDeliveryOptions>
      <orderOptionInfoURL>http://www.provider.com/envisat/orderoptions.html</orderOptionInfoURL>
      <paymentOptions>
        <paymentMethod>quota</paymentMethod>
      </paymentOptions>
      <sceneSelectionOption>
        <sceneType>Floating</sceneType>
      </sceneSelectionOption>
    </GetOptionsResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
C.2.4 Subscription with restrictions Response

<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
      <status>success</status>
      <orderOptions>
        <productOrderOptionsId>TerraSAR-X_SubscriptionOptions</productOrderOptionsId>
        <orderType>SUBSCRIPTION_ORDER</orderType>
        <options parameterID="regionOfInterest" use="required" updateable="false">
          <sps:Description>User area of interest on which the subscription shall be limited.</sps:Description>
          <sps:Definition>
            <sps:GeometryDefinition>gml:Polygon</sps:GeometryDefinition>
            <sps:Cardinality>1</sps:Cardinality>
            <grouping>Processing Option</grouping>
          </sps:Definition>
        </options>
        <options parameterID="startDate" use="required" updateable="false">
          <sps:Description>Definition of the base time period of interest.</sps:Description>
          <sps:CommonData>
            <swe:Time />
          </sps:CommonData>
        </options>
        <options parameterID="completionDate" use="required" updateable="false">
          <sps:Description>Definition of the base time period of interest.</sps:Description>
          <sps:CommonData>
            <swe:Time />
          </sps:CommonData>
        </options>
        <options parameterID="numberOfObservations" use="required" updateable="false">
          <sps:Description>Number of times the specified area shall be covered.</sps:Description>
          <sps:CommonData>
            <swe:Count />
          </sps:CommonData>
        </options>
        <options parameterID="observationGap" use="required" updateable="false">
          <sps:Description>Gap between the specified area shall be covered.</sps:Description>
          <sps:CommonData>
            <swe:Count />
          </sps:CommonData>
        </options>
      </orderOptions>
    </GetOptionsResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
C.3 GetQuotation

C.3.1 Request

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
    <GetQuotation xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://earth.esa.int/hma/ordering" xsi:schemaLocation="http://www.w3.org/2001/XMLSchema-instance" service="OS" version="1.2.0">
      <orderSpecification>
        <orderReference>example_0001</orderReference>
        <orderRemark>example</orderRemark>
        <deliveryInformation>
          <mail>
            <recipient>Mr. John Smith</recipient>
            <companyRef>DTMT</companyRef>
            <postalAddress>
              <streetAddress>Esrin Esa No. 1</streetAddress>
              <city>Frascati</city>
              <state></state>
              <postalCode>00100</postalCode>
              <country>IT</country>
            </postalAddress>
            <telephoneNumber>00390694180999</telephoneNumber>
          </mail>
        </deliveryInformation>
        <orderType>PRODUCT_ORDER</orderType>
        <orderItem>
          <itemId>item_0001</itemId>
          <productOrderOptionsId>Level 1, Product SLC (ASA.IMS)</productOrderOptionsId>
          <orderItemRemark>First product</orderItemRemark>
          <options>
            <sps:Parameter parameterID="processingLevel">
              <sps:value>
                <swe:Category>
                  <swe:value>1B</swe:value>
                </swe:Category>
              </sps:value>
            </sps:Parameter>
          </options>
        </orderItem>
      </orderSpecification>
    </GetQuotation>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```
C.3.2 Response

<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
      <status>success</status>
      <quote>
        <getQuotationId>urn:ESA:EECF:quotationId_0001</getQuotationId>
        <validityTime>2005-12-19T21:11:52.42</validityTime>
        <price>15</price>
        <currency>QUOTA</currency>
      </quote>
    </GetQuotationAck>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
C.4 Submit

C.4.1 Request

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Header>  
    <m:ReplyTo xmlns:m="http://www.w3.org/2005/08/addressing">  
      <m:Address>http://order_service_client.it</m:Address>  
    </m:ReplyTo>  
    <m:MessageID xmlns:m="http://www.w3.org/2005/08/addressing">000234359776845494</m:MessageID>  
  </SOAP-ENV:Header>  
  <SOAP-ENV:Body>  
    <Submit xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://earth.esa.int/hma/ordering" xsi:schemaLocation="http://earth.esa.int/hma/ordering ..\Order.xsd" service="OS" version="1.2.0">  
      <orderSpecification>  
        <orderReference>example_0001</orderReference>  
        <orderRemark>example</orderRemark>  
        <deliveryInformation>  
          <mail>  
            <recipient>Mr. John Smith</recipient>  
            <companyRef>DTMT</companyRef>  
            <postalAddress>  
              <streetAddress>Esrin Esa No. 1</streetAddress>  
              <city>Frassati</city>  
              <state>-</state>  
              <postalCode>00100</postalCode>  
              <country>IT</country>  
              <postBox>-</postBox>  
            </postalAddress>  
            <telephoneNumber>00390694180999</telephoneNumber>  
          </mail>  
        </deliveryInformation>  
        <orderType>PRODUCT_ORDER</orderType>  
        <orderItem>  
          <itemId>item_0001</itemId>  
          <productRef>identifier codeSpace="40282510510201144131I.3578391" EN1-05121921114570-3322.XI</productRef>  
          <collectionId>ESA,EECF,ENVISAT,ASA,IMx,xS</collectionId>  
          <price>  
            <value>15</value>  
            <currency>QUOTA</currency>  
          </price>  
          <balance>  
            <value>20</value>  
            <currency>QUOTA</currency>  
          </balance>  
        </orderItem>  
      </orderSpecification>  
    </Submit>  
  </SOAP-ENV:Body>  
</SOAP-ENV:Envelope>  
```
<productOrderOptionsId>Level 1, Product SLC (ASA_IMS)</productOrderOptionsId>

<orderItemRemark>First product</orderItemRemark>

<options>
  <sps:Parameter parameterID="processingLevel">
    <sps:value>
      <swe:Category>
        <swe:value>1B</swe:value>
      </swe:Category>
    </sps:value>
  </sps:Parameter>
</options>

<options>
  <sps:Parameter parameterID="productType">
    <sps:value>
      <swe:Category>
        <swe:value>ASA_IMS_1P</swe:value>
      </swe:Category>
    </sps:value>
  </sps:Parameter>
</options>

<options>
  <sps:Parameter parameterID="qualityOfService">
    <sps:value>
      <swe:Category>
        <swe:value>STANDARD</swe:value>
      </swe:Category>
    </sps:value>
  </sps:Parameter>
</options>

<sceneSelection>
  <sceneType>Floating Scene</sceneType>
  <sceneCoordinates>
    <gml:Point srsName="EPSG:4326"
      gml:id="DS_2044263010822">
      <gml:pos>2.374167 43.190833</gml:pos>
    </gml:Point>
  </sceneCoordinates>
</sceneSelection>

<deliveryMethod>mail</deliveryMethod>
<packageMedium>CD-ROM</packageMedium>
<payment>
  <paymentMethod>quota</paymentMethod>
  <orderAccount>project_10000</orderAccount>
</payment>

<productId>
  <identifier codeSpace="40282510510201144131.3578391">EN1-05121921114570-3322.XI</identifier>
  <collectionId>ESA.EECF.ENVISAT_ASA.IMx_xS</collectionId>
</productId>

</orderItem>
</orderSpecification>

<statusNotification>Final</statusNotification>
</Submit>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

C.4.2 Response

<?xml version="1.0" encoding="UTF-8"?>
  xmlns:SOAP-ENV:Body>
C.5 Submit Response

C.5.1 Request

<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Header>
    <m:RelatesTo xmlns:m="http://www.w3.org/2005/08/addressing" RelationshipType="http://www.w3.org/2005/08/addressing/reply">000234359776845494</m:RelatesTo>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <SubmitResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://earth.esa.int/hma/ordering" xsi:schemaLocation="http://earth.esa.int/hma/ordering ..\Order.xsd" service="OS" version="1.2.0">
      <timeStamp>2006-11-18T21:08:00.000</timeStamp>
      <orderMonitorSpecification>
        <orderReference>example_0001</orderReference>
        <orderRemark>example</orderRemark>
        <deliveryInformation>
          <mail>
            <recipient>Mr. John Smith</recipient>
            <companyRef>DTMT</companyRef>
            <postalAddress>
              <streetAddress>Esrin Esa No. 1</streetAddress>
              <city>Frascati</city>
              <state>-</state>
              <postalCode>00100</postalCode>
              <country>IT</country>
              <postBox>-</postBox>
            </postalAddress>
            <telephoneNumber>00390694180999</telephoneNumber>
          </mail>
        </deliveryInformation>
        <orderType>PRODUCT_ORDER</orderType>
        <orderId>123456-001</orderId>
        <orderStatusInfo>
          <status>Completed</status>
          <additionalStatusInfo>The order is accomplished</additionalStatusInfo>
        </orderStatusInfo>
        <orderItem>
          <itemId>item_0001</itemId>
          <productOrderOptionsId>Level 1,Product SLC (ASA_IMS)</productOrderOptionsId>
          <orderItemRemark>First product</orderItemRemark>
          <options>
            <sps:Parameter parameterID="processingLevel">
              <sps:value>
                <swe:Category>
                  <swe:value>1B</swe:value>
                </swe:Category>
              </sps:value>
            </sps:Parameter>
          </options>
        </orderItem>
      </orderMonitorSpecification>
    </SubmitResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
C.5.2 Response

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
    <SubmitResponseAck xsi:schemaLocation="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://earth.esa.int/hma/ordering:Order.xsd">
      <status>success</status>
    </SubmitResponseAck>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

C.6 Cancel

156
C.6.1 Request
  <SOAP-ENV:Header>
    <m:RelatesTo xmlns:m="http://www.w3.org/2005/08/addressing" RelationshipType="http://www.w3.org/2005/08/addressing/reply">000234359776845600</m:RelatesTo>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <Cancel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://earth.esa.int/hma/ordering" xsi:schemaLocation="http://earth.esa.int/hma/ordering ..Order.xsd" service="OS" version="1.2.0">
      <orderId>urn:ESA:EECF:order_id_0001</orderId>
      <statusNotification>Final</statusNotification>
    </Cancel>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

C.6.2 Response
  <SOAP-ENV:Body>
    <CancelAck xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://earth.esa.int/hma/ordering" xsi:schemaLocation="http://earth.esa.int/hma/ordering ..Order.xsd">
      <status>success</status>
    </CancelAck>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

C.7 CancelResponse
C.7.1 Request
  <SOAP-ENV:Header>
    <m:RelatesTo xmlns:m="http://www.w3.org/2005/08/addressing" RelationshipType="http://www.w3.org/2005/08/addressing/reply">000234359776845600</m:RelatesTo>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <CancelResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://earth.esa.int/hma/ordering" xsi:schemaLocation="http://earth.esa.int/hma/ordering ..Order.xsd" service="OS" version="1.2.0">
      <timeStamp>2006-11-18T21:08:00.000</timeStamp>
      <orderMonitorSpecification>
        <orderReference>example_0001</orderReference>
        <orderRemark>example</orderRemark>
        <deliveryInformation>
          <mail>
            <recipient>Mr. John Smith</recipient>
            <companyRef>DTMT</companyRef>
            <postalAddress>
              <streetAddress>Esrin Esa No. 1</streetAddress>
              <city>Frascati</city>
              <state></state>
              <postalCode>00100</postalCode>
              <country>IT</country>
            </postalAddress>
          </mail>
          <timeStamp>2006-11-18T21:08:00.000</timeStamp>
          <orderMonitorSpecification></orderMonitorSpecification>
        </deliveryInformation>
      </orderMonitorSpecification>
    </CancelResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
<Order Services for Earth Observation Products OGC 06-141r2

<deliveryInformation>
</deliveryInformation>

<orderType>PRODUCT_ORDER</orderType>
<orderId>123456-001</orderId>
<orderStatusInfo>
  <status>Cancelled</status>
  <additionalStatusInfo>The order is accomplished</additionalStatusInfo>
</orderStatusInfo>

<orderItem>
  <itemId>item_0001</itemId>
  <productOrderOptionsId>OPT1</productOrderOptionsId>
  <orderItemRemark>First product</orderItemRemark>
  <options>
    <sps:Parameter parameterID="processingLevel">
      <sps:value>
        <swe:Category>
          <swe:value>1B</swe:value>
        </swe:Category>
      </sps:value>
    </sps:Parameter>
  </options>
  <options>
    <sps:Parameter parameterID="productType">
      <sps:value>
        <swe:Category>
          <swe:value>ASA_IMS_1P</swe:value>
        </swe:Category>
      </sps:value>
    </sps:Parameter>
  </options>
  <options>
    <sps:Parameter parameterID="qualityOfService">
      <sps:value>
        <swe:Category>
          <swe:value>STANDARD</swe:value>
        </swe:Category>
      </sps:value>
    </sps:Parameter>
  </options>
  <sceneSelection>
    <sceneType>Floating Scene</sceneType>
    <sceneCoordinates>
      <gml:Point srsName="EPSG:4326">
        <gml:pos>2.374167 43.190833</gml:pos>
      </gml:Point>
    </sceneCoordinates>
    <temporalSelection>
      <startDateTime>2005-12-19T21:11:52.42</startDateTime>
      <endDateTime>2005-12-19T21:12:07.51</endDateTime>
    </temporalSelection>
    <scenePosition>frame 920</scenePosition>
  </sceneSelection>
</orderItem>

<deliveryMethod>mail</deliveryMethod>
<paymentMethod>quota</paymentMethod>
<orderAccount>project_10000</orderAccount>
<payment>
  <paymentMethod>quota</paymentMethod>
  <orderAccount>project_10000</orderAccount>
</payment>

<productId>
  <identifier codeSpace="40282510510201144131.3578391">EN1-05121921114570-3322.XI</identifier>
  <collectionId>ESA.EECF.ENVISAT_ASA.IMx_xS</collectionId>
  <productId>
  <orderItemStatusInfo>
    <status>Cancelled</status>
  </orderItemStatusInfo>
</productId>

<packageMedium>CD-ROM</packageMedium>
<payment>
  <paymentMethod>quota</paymentMethod>
  <orderAccount>project_10000</orderAccount>
</payment>

05121921114570-3322.XI<identifier codeSpace="40282510510201144131.3578391">EN1-05121921114570-3322.XI</identifier>
C.7.2 Response

<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
    <CancelResponseAck xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://earth.esa.int/hma/ordering" xsi:schemaLocation="http://earth.esa.int/hma/ordering ../Order.xsd">
      <status>success</status>
    </CancelResponseAck>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

C.8 GetStatus

C.8.1 Request

<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
    <GetStatus xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://earth.esa.int/hma/ordering" xsi:schemaLocation="http://earth.esa.int/hma/ordering ../Order.xsd" service="OS" version="1.2.0">
      <orderId>urn:ESA:EECF:order_id_0001</orderId>
      <presentation>full</presentation>
    </GetStatus>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

C.8.2 Response

<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
    <GetStatusResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://earth.esa.int/hma/ordering" xsi:schemaLocation="http://earth.esa.int/hma/ordering ../Order.xsd">
      <status>success</status>
      <orderMonitorSpecification>
        <orderReference>example_0001</orderReference>
        <orderRemark>example</orderRemark>
      </orderMonitorSpecification>
      <deliveryInformation>
        <mail>
          <recipient>Mr. John Smith</recipient>
          <companyRef>DTMT</companyRef>
          <postalAddress>
            <streetAddress>Esrin Esa No. 1</streetAddress>
            <city>Frascati</city>
            <state></state>
            <postalCode>00100</postalCode>
            <country>IT</country>
            <postBox></postBox>
          </postalAddress>
        </mail>
      </deliveryInformation>
    </GetStatusResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
<orderType>PRODUCT_ORDER</orderType>
<orderId>123456-001</orderId>
<orderStatusInfo>
  <status>Completed</status>
  <additionalStatusInfo>The order is accomplished</additionalStatusInfo>
</orderStatusInfo>
<orderItem>
  <itemId>item_0001</itemId>
  <productOrderOptionsId>Level 1, Product SLC (ASA_IMS)</productOrderOptionsId>
  <orderItemRemark>First product</orderItemRemark>
  <options>
    <sps:Parameter parameterID="processingLevel">
      <sps:value>
        <swe:Category>
          <swe:value>1B</swe:value>
        </swe:Category>
      </sps:value>
    </sps:Parameter>
  </options>
  <options>
    <sps:Parameter parameterID="productType">
      <sps:value>
        <swe:Category>
          <swe:value>ASA_IMS_1P</swe:value>
        </swe:Category>
      </sps:value>
    </sps:Parameter>
  </options>
  <options>
    <sps:Parameter parameterID="qualityOfService">
      <sps:value>
        <swe:Category>
          <swe:value>STANDARD</swe:value>
        </swe:Category>
      </sps:value>
    </sps:Parameter>
  </options>
  <sceneSelection>
    <sceneType>Floating Scene</sceneType>
    <sceneCoordinates>
      <gml:Point srsName="EPSG:4326" gml:id="DS_2044263010822">
        <gml:pos>2.374167 43.190833</gml:pos>
      </gml:Point>
    </sceneCoordinates>
    <temporalSelection>
      <startDateTime>2005-12-19T21:11:52.42</startDateTime>
      <endDateTime>2005-12-19T21:12:07.51</endDateTime>
    </temporalSelection>
    <scenePosition>frame 920</scenePosition>
  </sceneSelection>
</orderItem>
<deliveryMethod>mail</deliveryMethod>
<packageMedium>CD-ROM</packageMedium>
<payment>
  <paymentMethod>quota</paymentMethod>
  <orderAccount>project_10000</orderAccount>
</payment>
<productId>
  <identifier codeSpace="40282510510201144131.3578391">EN1-05121921114570-3322.XI</identifier>
  <collectionId>ESA.EECF.ENVISAT_ASA.IMx.xS</collectionId>
</productId>
<orderItemStatusInfo>
  <status>Completed</status>
</orderItemStatusInfo>
C.9 DescribeResultAccess

C.9.1 Request

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
    <DescribeResultAccess xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://earth.esa.int/hma/ordering" xsi:schemaLocation="http://earth.esa.int/hma/ordering ../Order.xsd" service="OS" version="1.2.0">
      <orderId>urn:ESA:EECF:order_id_0001</orderId>
      <subFunction>allReady</subFunction>
    </DescribeResultAccess>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

C.9.2 Response

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Body>
    <DescribeResultAccessResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://earth.esa.int/hma/ordering" xsi:schemaLocation="http://earth.esa.int/hma/ordering ../Order.xsd">
      <status>success</status>
      <URLs>
        <itemId>item_0001</itemId>
        <itemAddress>
          <ResourceAddress>
            <URL>ftp://user:online@eoaserver:2121/0c88028bf3aa6a6a143ed846f2be1ea4</URL>
          </ResourceAddress>
        </itemAddress>
      </URLs>
      <URLs>
        <itemId>item_0002</itemId>
        <itemAddress>
          <ResourceAddress>
            <URL>ftp://user:online@eoaserver:2121/0c88028bf3aa6a6a143ed846f2be1ea5</URL>
          </ResourceAddress>
        </itemAddress>
      </URLs>
      <URLs>
        <itemId>item_0003</itemId>
        <itemAddress>
          <ResourceAddress>
            <URL>ftp://user:online@eoaserver:2121/0c88028bf3aa6a6a143ed846f2be1ea6</URL>
          </ResourceAddress>
        </itemAddress>
      </URLs>
    </DescribeResultAccessResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```