

Heterogeneous Mission Accessibility -Follow-On - Online Data Access (HMA-FO_ODA)

Technical Specification (TS) -Software Requirements Specification (SRS)

EOX Doc.Ref:	HMA-FO_ODA-TS-SRS_EOX	
Issue:	1.1	
Date:	2010-04-19	
Authors:	C.Schiller, S.Krause, S.Meissl	
Responsible for Document:	C.Schiller	
Prime Contractor	OX	Thurngasse 8/4 1090 Wien Österreich / Austria office@eox.at
Subcontractors		www.eox.at
geographic information management	ACOBS SPOT	IBAN AT171400024510821897 BIC BAWAATWW Registered Vienna Commercial chamber Vienna FN 312068z UID ATU64265057

Commercial in Confidence

Approved by	Organisation	Date	Signature
Gerhard Triebnig	EOX	2010-04-19	
Pier-Giorgio Marchetti	ESA		

Release Track

Release No.	Date	Remarks
1.0	2010-01-27	Initial release as part of HMA-FO_ODA-TS_EOX_20100127
1.1	2010-04-19	changes of the document structure and adoption of content according to ECSS-E-ST-40C.

Distribution List:

Authors:	All
EOX:	G. Triebnig
ESA:	Pier-Giorgio Marchetti
GIM:	S. Desender
SPOT:	D. Giacobbo, P.Merigot
Spacebel:	Y. Coene

Table of Content

1 Introduction	4
1.1 Scope	4
1.2 Structure of Document	5
2 Applicable and Reference Documents	6
2.1 Applicable Documents	6
2.2 Reference Documents	6
3 Abbreviations	8
4 Software Overview	9
4.1 Function and purpose	9
4.2 Environmental considerations	11
4.3 Relation to other systems	12
4.4 Constraints	13
5 Requirements	14
5.1 General	15
5.2 Functional requirements	16
5.3 Performance requirements	16
5.4 Interface requirements	16
5.4.1 Human-Machine-Interface (HMI)	17
5.4.2 Security gateway	19
5.4.3 WCS EO Application Profile	19
5.4.4 Interface towards Catalogue services	20
5.4.5 Interface towards Ordering services	20
5.4.6 Interfaces to other services and/or service chains (e.g. WPS, WCPS	5, WCTS,)
	20
5.4.7 External Interfaces	20
5.4.7.1 General	20
5.4.7.2 GML 3.2.1 Application Schema for WCS	20
5.4.7.3 WCS 2.0 Core	20
5.4.7.4 WCS 2.0 Extension: KVP protocol	20
5.4.7.5 WCS 2.0 Extension: XML/POST protocol	21
5.4.7.6 WCS 2.0 Extension: SOAP protocol	21
5.4.7.7 Additional WCS 2.0 Extensions:	21
5.5 Operational requirements	

5.6 Resources requirements	22
5.6.1 Hardware resources	22
5.6.2 Description of the computer software	23
5.6.2.1 Background analysis of Open Source software packages	23
5.6.2.2 Software development approach	26
5.7 Design requirements and implementation constraints	29
5.8 Security and privacy requirements	30
5.9 Portability requirements	31
5.10 Software quality requirements	31
5.11 Software reliability requirements	31
5.12 Software maintainability requirements	31
5.13 Software safety requirements	32
5.14 Software configuration and delivery requirements	32
5.15 Data definition and database requirements	32
5.16 Human factors related requirements	32
5.17 Adaptation and installation requirements	32
6 Validation requirements	33
7 Traceability	34
8 Logical model description	35

List of Tables

Table 1: Trade-off Criteria for	• existing OWS OS Software	Implementations25
---------------------------------	----------------------------	-------------------

List of Figures

Figure 1: WCS 2.0 Draft Overview	9
Figure 2: WebMapViewer on the SSE Portal	
Figure 3: MapServer Architecture from [RD7]	27

1 Introduction

The Heterogeneous Mission Accessibility – Follow On – Task 3, Online Data Access (HMA-FO_ODA) project is funded by the European Space Agency ESA. The Task 3 of the HMA project aims at analyzing GMES relevant projects and services (e.g. Fast Track Service, Core Service, geoland2, MyOcean, GSCDA, GEOSS, and many more) for their requirements to access EO data online as their input source. Based on these needs use cases for the online access to EO data shall be derived and the requirements to satisfy them should be presented.

The use and the processing of data collected by EO sensors requires knowledge of the metadata describing the circumstances of the acquisition and the processing applied. The EO Community uses a variety of formats for raster image data, various metadata formats, and a wide range of methods for their distribution – be it online, or offline. Additionally, users of EO products typically require EO information products from multiple providers and strongly desire that the information services at these different providers all behave in a comparable, if not in the same way for homogeneous access and use.

WCS is supposed to be such a standard and unique Web based service mechanism. But when applied to EO products it turns out that the WCS standard still leaves degrees of freedom which result in unwanted differences in service behaviors in different providers' implementations. This hampers interoperability.

Therefore, a WCS Application Profile for EO Sensors (EO WCS) shall be developed providing the specification of a service with a defined behavior, tailored to the exploitation of the characteristic information inherent in EO products. Additionally, an Open-Source Software WCS package shall be enhanced to demonstrate the usage of the Application Profile for EO Sensors.

1.1 Scope

This document (HMA-FO_ODA-TS-SRS_EOX) represents the Technical Specification (TS) -Software Requirements Specification (SRS) for the Heterogeneous Mission Accessibility – Follow On – Online Data Access (HMA-FO_ODA) project and is the result of the work performed under the ESA ESRIN/Contract No. 22507/09/I-LG. It describes the functional and non functional requirements applicable to the software items.

The document compiles information in response to the Requirement Baseline Document – Software System Specification (HMA-FO_ODA-RB-SSS_EOX). It contains the Software Requirements Specification covering Function, Performance, Product and Software Quality, Security, Data definition and Database requirements as well as the Interface Control Document (ICD) for the external interfaces. The Technical Specification (TS) is the basis against which the detailed design, the coding, the unit and integration testing, and the validation is performed.

The Technical Specification, contains a precise and coherent definition of functional and performance requirements and the specification for interactions with external systems, which may be other software products or other systems. All significant trade-offs,

feasibility analyzes, make/buy decisions and supporting technical assessments are documented in the Technical Specification.

This document is a living document during the course of the HMA-FO project. It will be updated according to the status of the maturity of the specification and developments.

1.2 Structure of Document

This is Clause 1 which presents the introduction, the scope and the structure of the document.

Clause 2 lists the Applicable and Reference Documents.

Clause 3 contains the Abbreviations listing.

Clause 4 contains provides an overview of the ODA system and of the components used to build it.

Clause 5 provides the HMA-FO specification derived from analysis of the Requirements Baseline Software System Specification, the Specification Dependencies and the EO Metadata Requirements documents.

Clause 6 describes the validation approach applicable to each requirement.

Clause 7 contains the traceability matrix allowing tracing of requirements from SSS to SRS and from SRS to SSS.

Clause 8 presents the Logical Model describing the relationships of the entities of the system in a view independent of the implementation.

2 Applicable and Reference Documents

2.1 Applicable Documents

- [AD1] ECSS Standard: Space project management Project planning and implementation, Ref. ECSS-M-ST-10C Rev. 1, 6 March 2009
- [AD2] ECSS Standard: Space project management Configuration and information management, Ref. ECSS-M-40C Rev.1, 6 March 2009
- [AD3] ECSS Standard: Space project management Cost and schedule management, Ref. ECSS–M–60C, 31 July 2008
- [AD4] ECSS Standard: Space Engineering Software, Ref. ECSS-E-ST40C, 6 March 2009
- [AD5] ECSS Standard: Space product assurance Software product assurance, Ref. ECSS-Q-ST-80C, 6 March 2009
- [AD6] HMA-FO_ODA Requirements Baseline Document Software System Specification (HMA-FO_ODA-RB-SSS_EOX, v1.1, 2010-04-10)
- [AD7] HMA-FO_ODA Requirements Baseline Document Technical Note (HMA-FO_ODA-RB-TN_EOX, v1.1, 2010-04-10)
- [AD8] HMA-FO_RB: EO Metadata Requirements –Requirements Baseline Document (HMA-FO_ODA-EOMR-RB_SPOT, v1.0, 2010-03-01)
- [AD9] HMA-FO_TN: Specification Dependencies Technical Note (hmafo-tn-0001-spbv11-draft.doc)
- [AD10] GML 3.2.1 Application Schema for WCS (including XML Schema) (OGC 09-146)
- [AD11] WCS 2.0 Core (including XML Schema) (OGC 09-110)
- [AD12] WCS 2.0 Extension: KVP protocol (OGC 09-147)
- [AD13] WCS 2.0 Extension: XML/POST protocol (OGC 09-148)
- [AD14] WCS 2.0 Extension: SOAP protocol (OGC 09-149)
- [AD15] WCS 2.0 Overview: Core and Extensions (Best Practice) (OGC 09-153)
- [AD16] OGC 07-118r5, User Management Interfaces for Earth Observation Services, version 0.1, 2010-03-05

2.2 Reference Documents

- [RD1] http://www.mapserver.org/, MapServer Homepage
- [RD2] http://geoserver.org/display/GEOS/Welcome, GeoServer Open Source
- [RD3] http://www.deegree.org/, deegree Homepage
- [RD4] http://www.unidata.ucar.edu/projects/THREDDS/, THREDDS Data Server

Homepage

- [RD5] http://www.osgeo.org/, OSGeo Homepage
- [RD6] http://www.osgeo.org/node/812, News MapServer Incubation Graduation
- [RD7] http://mapserver.org/introduction.html, MapServer Introduction
- [RD8] http://www.swig.org/, SWIG Homepage
- [RD9] http://www.python.org/, Python Homepage
- [RD10] http://svn.osgeo.org/mapserver/trunk/, MapServer development source (svn)
- [RD11] http://subversion.tigris.org/, Subversion Homepage
- [RD12] http://trac.osgeo.org/mapserver/, MapServer issue tracker (trac)
- [RD13] http://trac.edgewall.org/, Trac Homepage
- [RD14] http://www.mapserver.org/community/lists.html, MapServer mailings lists
- [RD15] http://www.gnu.org/software/mailman/, Mailman Homepage
- [RD16] ttp://www.mapserver.org/community/irc.html, MapServer IRC
- [RD17] http://sourceforge.net/, SourceForge
- [RD18] http://code.google.com/, Google Code
- [RD19] http://zeus.pin.unifi.it/projects/wcsClientLite/, WCS Client of the University of Florence

3 Abbreviations

AP	Application Profile	OGC	Open Geospatial Consortium
CDS	Coordinated Data access System (GSCDA)	OGR	OGR Simple Features Library
EO	Earth Observation	OSGeo	Open Source Geospatial Foundation
EO-WCS	EO Application Profile for WCS		
EP	Extension Package	OSS	Open Source Software
ESA	European Space Agency	RB	Requirements Baseline
GCM	GMES Contributing Mission	SOAP	Simple Object Access Protocol,
GDAL	Geospatial Data Abstraction Library	SOS	Sensor Observation Service (OGC)
GMES	Global Monitoring for Environment and Security	SSE	Service Support Environment
GML	Geographic Markup Language (OGC)	TDS	THREDDS Data Server
GPL	GNU General Public License	THREDDS	Thematic Realtime Environmental Distributed Data Services (TDS)
GSC-DA	GMES space component - data access	TN	Technical Note
НМА	Heterogeneous Mission Accessibility	TS	Technical Specification
HMA-E	HME – ESA	WCPS	Web Coverage Processing Service (OGC)
HMA-FO	HMA – Follow On	WCS	Web Coverage Service (OGC)
HMA-I	HMA - Initial	WCS-T	Web Processing Service – Transactional (OGC)
HME-T	HMA- Testbed	WCTS	Web Coordinate Transformation Service (OGC)
ICD	Interface Control Document	WFS	Web Feature Service (OGC)
KVP	Key-Value Pair	WMS	Web Mapping Service(OGC)
НМІ	Human Machine Interface	WPS	Web Processing Service (OGC)
ODA	Online Data Access	XML	Extended Markup Language

4 Software Overview

This section describes the function and purpose of the Online Data Access system (ODA) as well as components which will be used to build it. The major ingredients, which need to be developed, are the WCS EO Application Profile (EO-WCS) and the adaptations to implement the WCS 2.0 standard into an Open Source Software package for demonstration purposes.

4.1 Function and purpose

During an international process and by participation of many members of the OGC the Web Coverage Service protocol has been re-defined, now achieving the version 2.0. This Standardization process has now reached maturity and voting for the release of the WCS 2.0 as an official OGC standard is on its way.

WCS 2.0 introduces a modularized concept. A rather slender standards core specifications can be enlarged by developing extension for specialized functionalities. Such extension may represent special data models, file format encodings (e.g. GeoTIFF; Jpeg2000, NetCDF), service models projection regulations (e.g. CRC, scaling), protocols (e.g. GET/KVP, SOAP), or usability modules (e.g. multi-linguality)

In Figure 1 an overview of the relationship and the interactions between different domain specific WCS Application Profiles, the WCS extensions, the WCS core specification and the WCS foundation is presented.





The WCS 2.0 versions of the documents listed below have recently been submitted to

OGC for voting.

Submitted are:

- GML 3.2.1 Application Schema for WCS (including XML Schema) (OGC 09-146)
- WCS 2.0 Core (including XML Schema) (OGC 09-110)
- WCS 2.0 Extension: KVP protocol (OGC 09-147)
- WCS 2.0 Extension: XML/POST protocol (OGC 09-148)
- WCS 2.0 Extension: SOAP protocol (OGC 09-149)
- WCS 2.0 Overview: Core and Extensions (Best Practice) (OGC 09-153)

In addition 20 XML schema & example files and a draft template for coverage format extensions (Best Practice) are under preparation and will be provided to OGC soon.

For the demonstration implementation of WCS server and client the following issues need further discussion:

- file formats (GeoTIFF mandatory; JPEG2000 optional)
- CRC & Scaling Projections (EPSG: 3035, EPSG: 4326).
 - Note: There are approx. 50 UTM zones required to cover the European territory at local scale. The exact implementation effort to cover all these UTM zones has to be further evaluated. Based on this evaluation an implementation plan (phase 1, phase 2, phase 3) will be presented during the next project phase.
- WCS-EO Application Profile

Additional extensions to be written to enable the demonstration of the full functionality of the WCS concept for widespread user communities include:

- Nil values (e.g. masking processing)
- NetCDF (e.g for MetOcean the CF/NetCDF)
- WCS-T
- WCPS

The definition of the WCS 2.0 OGC standard is focused on wide applicability for various application domains. This requires that it is to be based on a much broader concept as needed for the access to Earth Observation data. Therefore, a WCS EO Application Profile (EO WCS) is developed to define the exact usage and application of the WCS 2.0 standard for the Earth Observation domain. This shall allow users and especially software developers (by enabling their products) to directly access EO data online when applying the EO WCS.

Based on the high increase in connectivity due the general availability of broad band connection, the increase in computing power, users are more and more interested in receiving information products in a timely manner directly onto their computers. But providing information products implies that the EO products taken by satellite sensors need to be evaluated, processed, analyzed, and transformed into information products. Since the experts performing this work are often not associated with the data provider nor with the information user, EO products have to be transferred from the Ground

Station to an EO expert and then to the customer. So far this has been undertaken either by postal shipment of physical media (Tapes, CDROM, DVD, portable Disks) or by transfer of files utilizing the FTP protocol. FTP, however, does not allow for previews of the data, for subsetting or any preprocessing.

WCS on the other hand, enables exactly these features (and many more) prior data transfer from the supplier to the user. And by chaining WCS with other Web Service the possibilities of processing and information generation seems to be endless. Finally, a user would be able to receive exactly the information type, area, and projection requested and all the hassle about processing and data preparation, would be performed transparent to the user.

The combination of WCS 2.0 and EO WCS shall benefit satellite data providers, data refiner, and user alike by tearing down barricades hindering the online access to EO data. The easy and direct delivery of coverages, based on user requirements should be a commendable target.

4.2 Environmental considerations

This section describes the hardware and software resources used to define a target environment.

For the demonstration of the applicability of a WCS in combination with the EO WCS to enable online data access to EO data holdings for users and providers no special hardware requirements are postulated.

A low-end server system with a broadband internet connectivity should be sufficient. Requirements for the memory equipment of the server are moderate (e.g. 4 to 8GB). However, it has to be stated that the system has to process large amounts of data and therefore increased memory availability will result in shorter processing times.

Hard disk storage requirements are mainly for the operating system, the WCS software and a geo-enabled database. The demonstration scenario is based on the assumption that the test data sets are not stored directly on the system but are accessed through a network connection (TBC) providing a more realistic scenario. A direct storage of the test data set on the system in question is, however, possible.

For performance reasons a RAID-1 (or RAID-10) is recommended, providing improved access speed for reading operations and redundancy for stability.

Hardware resources:

CPU: low-end server machine;

RAM: 4 to 8 GB

Graphic Adapter: no specific needs.

Disks: RAID-10 Array (size depending on test data set storage location)

Software resources:

Operating System: Linux, preferably Debian

Mapserver

PostgreSQL & PostGIS extension

Python and possibly Django/GeoDjango GDAL OGR misc. software development libraries

For the demonstration system, no special environmental consideration are required.

4.3 Relation to other systems

In general the ODA system will be designed to work as a stand-alone system. This means that any Web Service shall be able to utilize the ODA functionality if following the standards defined for the EO WCS.

However, several relationships with other systems are envisaged, especially for the usage of the HMI client interface planned for the demonstration system. Additionally interactions with HMA EO catalogs, the HMA order handling system, HMA security system, and HMA WMS instances are subject of interactions with the ODA system.

The following is the list of systems having relationship with the ODA system. However, they are not prerequisites for the general functioning of the ODA system.

SSE (Service Support Environment)

SSE provides the so called "HM Services", heterogeneous multi-mission services, to its clients as an integration layer between the GMES contributing missions. It is in charge of splitting and routing requests towards the GMES contributing missions and packing together the received responses. It relies on the contributing missions for the actual execution of requests. It further enables the creation of service chains enabling the combination of different "atomic" services to create new services with more complex functionalities.

WebMapViewer

The WebMapViewer, provided by GIM, represents the HMI (Human Machine Interface) between ODA and an interactive user. For the HMI the WebMapViewer which is also incorporated at ESA's SSE portal has been chosen. This is a logical choice for a demonstration client platform as the ESA SSE portal with its integrated WebMapViewer already acts as a client to OGC Web Map-, Web Feature-, Web Coverage-, and Sensor Observation Services. The SSE WebMapViewer is quite unique in the sense that it offers a thin web based graphical user interface to Web Coverage Services, hereby not requiring any browser extensions. Not only will this choice for well-established technology provide a quick start for developments within this project but will also allow ESA to integrate the client side results of this project in its existing environments.

OGC EO Catalogs

The requirements state interaction with EO catalogs. However, the interactions are not in

the form of full catalog queries or request but rather represent a limited set of information exchange regarding provision of metadata for coverages in the WCS and linking of catalog entries to coverage resources. The way of information exchange is currently under definition in the HMA-FO project.

Ordering & Programming Gateway (OPGW)

Interactions of ODA with the HMA Ordering & Programming Gateway are limited to two forms of information exchange.

- the OPGW forwards an order including all the information and processing parameters necessary to process the data and provide access to the final product
- the WCS forwards information about a coverage, AOI, timeframe, etc. to the OPGW for further processing of the actual order with the OPGW sending back an affirmation for final data access provisioning.

HMA Security Gateway (SGW)

The ODA system will interact with the HMA security system in such way that it accepts and honors Tokens and will act accordingly.

The ODA system, however, will not manage access rules by itself or check if a user is restricted by certain access limitations (AOI, layers, timeframe, resolution, etc.). It will merely enforce the rules. The actual check for access restrictions has to be done elsewhere (e.g. SGW)

4.4 Constraints

Demonstration depends on availability of above described external systems.

Demonstration depends on availability of test data sets, catalogues, etc.

5 Requirements

The requirements in this document have a common table format with the following notation.

Identifier	Description	Ver.M.	Sig.
ODA_GEN_010	Here is the description of the requirement.	D	E-Y

Identifier Unique identifier composed of

"ODA_"	Online Data Access	
Section Code	e.g. GEN, FUN	
3-digit Number	Reference Number within	Section

Ver.M. Verification Method:

D Demonstration: This verification method may be used when actual conduct can verify achievement of requirements such as service and access, and when the requirements have been implemented.

T Test: A requirement may be verified by test alone if the form of the specification is such that the requirement can be directly measured.

A Analyzes: This verification method implies use of analytical techniques (such as system engineering analysis, statistics, mathematical modeling, and simulations) and shall be used to verify such requirements. This includes requirements which are not implemented in the demonstrator.

I Inspection: Verification by inspection is only done when testing is insufficient or inappropriate. This method of verification is for those requirements that are normally performed by some form of visual inspection.

R Review of Design: Verification by inspection is only done when testing is insufficient or inappropriate. This method of verification is for those requirements that are normally performed by some form of visual inspection.

Sig. Significance:

E_Y Requirement is considered as essential and will have to be implemented within the ODA demonstrator.

E_B Requirement is considered as essential but a decision on implementation cannot be taken yet. It will be implemented on a best effort basis.

E_N Requirement is considered as essential, it will have to be considered for the architectural design, but it is not going to be implemented in the ODA demonstrator.

D_Y Requirement is considered as desirable for the ODA demonstrator and will be implemented within the ODA demonstrator.

D_B Requirement is considered as desirable but a decision on implementation cannot be taken yet for the ODA demonstrator; it will be implemented on a best effort basis.

D_N Requirement is considered as desirable, it is not included in the ODA demonstrator implementation and shall only be considered for the architectural design on a best effort basis.

5.1 General

Identifier	Description	Ver.M.	Sig.
ODA_GEN_010	For the Online Data Access (ODA) a Web Service shall be used.	D	E_Y
ODA_GEN_020	The Web Service shall allow SOAP binding.	D	E_Y
ODA_GEN_030	The ODA service shall (somehow) allow subscriptions. Note: Subscription is not a real Online Data Access feature. It can be imagined that subscription is a separate service by its own which triggers ODA for data extraction and delivery.	A	D_N
ODA_GEN_040	The ODA service shall respect security and access control mechanisms based on [AD16]	D	E_Y
ODA_GEN_050	The ODA service shall respect the defined WCS EO Application Profile.	D	E_Y
ODA_GEN_060	Open Source software shall be operated to provide the downloading capabilities via Web Coverage Service (WCS).	D	E_Y
ODA_GEN_070	The ODA system shall be able to be configured to work in a stand-alone mode.	D	E_Y
ODA_GEN_080	The ODA service should be scalable.	Т	E_B
ODA_GEN_090	The ODA service should be able to be distributed on many machines.	A	E_N
ODA_GEN_100	The ODA service should be able to handle numerous simultaneous requests for access to the same EO data.	D	D_Y
ODA_GEN_110	The ODA service should provide fast access to EO data.	D	D_Y
REQ_GEN_120	ODA shall allow Near Real Time visualization. Access to	D	D_Y

	the data for visualization should be with a minimum delay (no delay greater than 2 second for one page).		
REQ_GEN_130	ODA shall provide simple access to the data content with sufficient set of metadata.	D	D_Y
REQ_GEN_131	Data extraction used for extracting thematic information should be done by using Image Classifier Tools or by manual editing.	A	D_N
REQ_GEN_140	ODA shall be flexible enough to be integrated within the End User environment. The data should be accessible on the End User application using either Web Service, API, or Add-on.	D	D_Y
REQ_GEN_160	ODA shall allow access that is not application dependent. Access to the data is using open standard as such provided by OGC for mainly backdrop use as with WMS. The access as WMS mode is fully independent of the Customer client solution (Web Browser, GEE, etc.)	D	E_Y

5.2 Functional requirements

ECSS: capabilities to be provided by the software, link between the requirements and the system states and modes. grouped by subject, in accordance with the logical model organization (e.g. per controlled subsystem). functional requirements related to software safety and dependability

Each requirement definition should be organized according to: a) General, b) Inputs, c) Outputs, d) Processing

5.3 Performance requirements

ECSS: list any specific requirement to the specified performance

5.4 Interface requirements

The external interfaces are covered by the OGC specifications, especially WCS 2.0, currently submitted into the OGC standardisation process, and EO WCS.

The internal interfaces are corresponding to interfaces between the WCS software implementation, namely Mapserver, PostgreSQL/PostGIS, GDAL, OGR, etc. and the operation system. There is a second group of "internal" interfaces namely those to

closely related software components like a catalogue running on the same data set and/or database as a WCS (see anticipated "Implementation Guidance" section in EO-WCS).

The platform for the EOX based demonstration implementation will be based on the freely available Debian-GNU/Linux Operating system. For this platform all required packages and source code for the MapServer, PostgreSQL/PostGIS database, libraries (GDAL, OGR, etc.) as well as general development tools are available and already in use by EOX.

For the demonstration instantiation the hardware interfaces will depend on the amount of the test data sets provided by ESA and other providers.

The HMI is represented by the interfaces between an interactive user and the selected WebMapViewer provided by GIM.

Identifier	Description	Ver.M.	Sig.
ODA_IF_010	The ODA system shall support the access via HTTP KVP.	D	E_Y
ODA_IF_020	The ODA system should support the access via HTTP KVP plus SOAP binding.	D	E_Y

5.4.1 Human-Machine-Interface (HMI)

For the HMI (Human-Machine-Interface) the WebMapViewer provided by GIM which is also incorporated at ESA's SSE portal has been chosen. This is a logical choice for a demonstration client platform as the ESA SSE portal with its integrated WebMapViewer already acts as a client to OGC Web Map, Web Feature, Web Coverage, and Sensor Observation Services. The SSE WebMapViewer is quite unique in the sense that it offers a thin web based graphical user interface to Web Coverage Services, hereby not requiring any browser extensions. Not only will this choice for well-established technology provide a quick start for developments within this project but will also allow ESA to integrate the client side results of this project in its existing environments.

We foresee the following evolutions to be undertaken to obtain a client application that is interoperable with the server-side implementations.

 Currently the SSE WebMapViewer supports version 1.0.0 of the WCS protocol (with a set of data formats as identified in the SSE ICD). The WebMapViewer will be enhanced by GIM to provide the necessary features which are defined in the requirement clause of [AD1]. The WebMapViewer will therefore be enabled to act as client to WCS 2.0 services covered by the EO WCS Application Profile, that this project will establish.

Changes relate to e.g. (listing incomplete):

- WCS 2.0 EO Application Profile compliant requests
- alignment with the OWS Common Specification
- adaptations to GML 3.2.1
- Multipart MIME responses to GetCoverage Requests
- Hierarchical coverage descriptions
- Use of GridCRS in descriptions and requests
- etc.
- Changes to Graphical User Interface following requirements analysis. Improvements would be required to handle display of metadata, more intuitive interface for dealing with coverage downloads, ...
- Specific improvements that come from the specifics defined within the WCS
 Application Profile for EO sensors



Figure 2: WebMapViewer on the SSE Portal

Figure 2 shows the GIM developed SSE WebMapViewer of the SSE Portal configured to serve an European raster data-set showing the degree of soil sealing for the area of Vienna.

The metadata identified in the EO metadata requirement work package will be provided by the server and shown by the client.

For interoperability demonstration, an additional WCS client like the one of the University of Florence [RD19] could be showcased. However, since also this client would first need

to be enhanced to work with the WCS 2.0 EO Application Profile this interoperability demonstration will only be performed if time and resources permit such work. This work may further be limited by the availability of the source code.

Identifier	Description	Ver.M.	Sig.
ODA_IF-CLI_030	The WebMapViewer shall support WCS 1.0.0 with the HTTP Get Binding.	D	E_Y
ODA_IF-CLI_040	The WebMapViewer should support WCS 2.0 Core with the HTTP Get Binding.	D	E_Y
ODA_IF-CLI_050	The WebMapViewer should support WCS 2.0 Core with the SOAP 1.2 Binding.	D	
ODA_IF-CLI_060	 The WebMapViewer shall support the following image formats: GeoTIFF (as per the limitations of SUN JAI libraries in handling TIFFs) Other coverage formats as supported by the Open Source GDAL library GDAL forms an optional component of the WebMapViewer installation. 	D	E_Y
ODA_IF-CLI_070	The WebMapViewer should support GML Rectified Grids as format in combination with the WCS 2.0 Core Profile.	D	E_Y
ODA_IF-CLI_080	The WebMapViewer should support the WCS EO Application Profile Extent of support to be confirmed after definition of extension if all aspects of this EO Extension can be covered.	D	E_Y

5.4.2 Security gateway

For details see :

OGC 07-118r1_User_Management_Interfaces_for_Earth_Observation_Services_0.0.4

5.4.3 WCS EO Application Profile

Is part of the current developments within this project.

5.4.4 Interface towards Catalogue services

For details see : HMA-FO Task-1

5.4.5 Interface towards Ordering services

For details see : HMA-FO Task-4 Note: what about Subscription service – interaction between catalog and ordering.

5.4.6 Interfaces to other services and/or service chains (e.g. WPS, WCPS, WCTS, ...)

For details see : TBD

5.4.7 External Interfaces

5.4.7.1 General

For details see : 09-153r1_WCS-overview_2010-04-14a.doc 10-017r1_WCS-2-0-Revision-Notes.doc

5.4.7.2 GML 3.2.1 Application Schema for WCS

For details see : 09-146r1_GML-Application-Schema-for-Coverages_2010-04-14c.doc

5.4.7.3 WCS 2.0 Core

For details see : 09-110r3_WCS-core_2010-04-14.doc

5.4.7.4 WCS 2.0 Extension: KVP protocol

For details see : 09-147r1_WCS-KVP_2010-04-14a.doc

5.4.7.5 WCS 2.0 Extension: XML/POST protocol

For details see :

09-148r1_WCS-XML-POST_2010-04-14a.doc

5.4.7.6 WCS 2.0 Extension: SOAP protocol

For details see : 09-149r1_WCS-XML-SOAP_2010-04-14a.doc

5.4.7.7 Additional WCS 2.0 Extensions:

5.4.7.7.1 Format encodings: GeoTIFF, Jpeg2000, NetCDF, etc.

For details see :

Each Format encoding represents an extension of the WCS 2.0 core specification. No format encoding extension is yet defined.

However, GeoTIFF is set mandatory for the EO WCS.

5.4.7.7.2 Service model: CRC – Projections

For details see :

CRC represents an extension of the WCS 2.0 core specification. The CRS extension is not yet defined.

5.4.7.7.3 Service model: WCS-T

For details see :

WCS-T represents an extension of the WCS 2.0 core specification. The WCS-T extension is not yet defined.

5.4.7.7.4 Service model: WCPS

For details see :

WCPS represents an extension of the WCS 2.0 core specification. The WCPS extension is not yet defined.

5.4.7.7.5 Data model: Nil values

For details see :

Nil values represents an extension of the WCS 2.0 core specification. The Nil values extension is not yet defined.

5.5 Operational requirements

This section lists specific requirements related to the operation.

Identifier	Description	Ver.M.	Sig.
ODA_SOR_010	The ODA system should be made available utilizing an instantiation of Linux (possibly Linux-HA -TBC) (version TBD).	D	E_Y
ODA_SOR_020	The PostgreSQL DB with the PostGIS extension shall be available (version TBD).	D	E_Y
ODA_SOR_030	The GDAL library shall be available (version TBD).	D	E_Y
ODA_SOR_040	Python programming language shall be available (version TBD).	D	E_Y
ODA_SOR_050	GCC compiler and Linux header files shall be available (version TBD).	D	E_Y
ODA_SOR_060	A series of EPSG codes to be supported, and which GDAL must be able to handle, shall be agreed upon.	D	E_Y

5.6 Resources requirements

All the resource requirements related to the software and the hardware requirements (target hardware on which the software is specified to operate); sizing and timing requirements shall be described.

Description of the computer software to be used with the software under specification or incorporated into the software item;

5.6.1 Hardware resources

For the demonstration of the applicability of a WCS in combination with the EO WCS to enable online data access to EO data holdings for users and providers no special hardware requirements are postulated.

A low-end server system with a broadband internet connectivity should be sufficient. Requirements for the memory equipment of the server are moderate (e.g. 4 to 8GB). However, it has to be stated that the system has to process large amounts of data and therefore increased memory availability will result in shorter processing times. Hard disk storage requirements are for mainly for the operating system, the WCS software and a geo-enabled database. The demonstration scenario is based on the assumption that the test data sets are not stored directly on the system but are accessed through a network connection (TBC) providing a more realistic scenario. A direct storage is of the test data set on the system in question is, however, possible.

For performance reasons a RAID-1 (or RAID-10) is recommended, providing access speed for reading and writing operations and redundancy for stability.

Identifier	Description	Ver.M.	Sig.
ODA_CRR_010	The ODA system shall operate on hardware platform utilizing a RAID-10 disk-array of appropriate size.	D	D_Y

5.6.2 Description of the computer software

5.6.2.1 Background analysis of Open Source software packages.

At Kickoff of the HMA-FO project it has been decided that the Online Data Access (Task 3) shall use the newly to be developed WCS 2.0 specification as the basis for the development of the WCS EO Application Profile (EO WCS) as well as for the development of a compliant test implementation using an Open Source Software Tool.

Therefore a analyzes of Open Source Software (OSS) packages implementing earlier WCS Interface Specification is performed. We are providing an evaluation of the pros and cons of each platform based on implementation requirements concerning the handling and storage of coverages and mosaics.

We are comparing these packages with respect to various criteria, most importantly:

- their support for WCS 1.1
- their flexibility to handle dynamically changing datasets
- their support for mosaic and image pyramid handling

Whereas the first criterion is obvious, the other two need explanation: The ODA software implementation must ensure that the WCS server configuration for available coverages and their metadata can be changed dynamically to handle incoming WCS-T transaction requests.

As EO product collections often contain many terabytes of EO product data, efficient manipulation and storage of large raster data collections and/or mosaics is crucial for productive use of a WCS(-T) implementation. For flexible and performant access to raster data, large coverages usually are stored as tile sets or image pyramids.

A tile set is obtained by splitting the original coverage at equally spaced perpendicular grid lines into smaller image files. An image pyramid consists of tile sets built from the same raster data resampled to ever diminishing image resolutions. Pyramids are especially useful to deliver imagery at a variety of different zoom levels with high performance. Tiles and pyramids can exist inside files (e.g. GeoTIF, JPEG200) or at filesystem level (i.e. tiles for each pyramid level are stored separately in the file system).

The technical pyramid metadata (often called tile index), i.e. the resolutions and locations of tiles within the file system, must be stored in a way enabling fast geospatial queries for performant coverage access. In practice, there are two solutions for the storage of tile indexes: ESRI shapefiles and spatially enabled databases.

Shapefiles are the more commonly supported storage method, whereas spatial databases allow for more flexibility and more performant geospatial indexing. The most comprehensive and elaborate open source spatial RDBMS implementation is PostgreSQL with its PostGIS extension, which serves as reference for the following comparison.

In Table 1.1.2 the basic properties of the following four software packages are listed:

- MapServer [RD1]
- GeoServer [RD2]
- deegree [RD3]
- THREDDS Data Server [RD4]

Compared Aspect	MapServer	GeoServer	deegree WCS	THREDDS Data Server
Maintaining Entity	Community project with backing of the active Open Source Geospatial Foundation (OSGeo)	Community project led by the company OpenGeo, USA	Community project led by lat/lon GmbH and University of Bonn, Germany	Unidata/UCAR
License	MIT-style	GPL Version 2	LGPL	LGPL
Implementation language	С	J2EE	Java	Java
Language bindings	Various through SWIG e.g. Python, Perl, Java, .NET, etc. and PHP	-	-	-
Database bindings	PostgreSQL/PostGI S, MySQL, Oracle	PostgreSQL/PostGI S, MySQL, DB2, Oracle, ArcSDE, SQL Server	Oracle	None
Raster formats	Various through GDAL e.g. GeoTiff, JPEG2000, HDF, netCDF, etc.	Native Java support for GeoTIFF, GTOPO30, ArcGrid, WorldImages, ImageMosiacs, and Image Pyramids Support for MrSID, ECW, JPEG2000, DTED, Erdas Imagine, and NITF through GDAL ImageIO Extension	GeoTIFF, ECW, GIF, TIFF, BMP, JPEG, PNG	GeoTIFF (only for output), NetCDF

Compared Aspect	MapServer	GeoServer	deegree WCS	THREDDS
				Data Server
Vector formats	PostGIS and	PostGIS, Shapefile,	_	-
	various through	ArcSDE, DB2 and		
	OGR e.g. Shapefile,	Oracle		
	GML, SQLite,			
	MySQL, Oracle			
	Spatial, etc.			
OWS compliance	WMS (1.0.0, 1.0.7,	WMS 1.1.1, WFS	WCS 1.0	WCS 1.0
	1.1.0, 1.1.1), WFS	(1.0 and 1.1,	(Reference	
	(1.0.0, 1.1.0), WCS	transactions and	implementation)	
	(1.0, 1.1), GML	locking) and WCS		
	(2.1.2, 3.1.0 Level	(1.0 and 1.1)		
	0 Profile), WMC	specifications, as		
	(1.0.0, 1.1.0),	tested by the CITE		
	Filter Encoding	conformance tests.		
	(1.0.0), SLD	GeoServer		
	(1.0.0), SOS	additionally serves		
	(1.0.0), OM	as Reference		
	(1.0.0), SWE	Implementation for		
	(1.0.1), OWS	WCS 1.1 and WFS		
	(1.0.0, 1.1.0)	1.0 and 1.1		
		GML (2.1.2, 3.1.1),		
		SLD		

Table 1: Trade-off Criteria for existing OWS OS Software Implementations

The **MapServer** development is now in the hands of the Open Source Geospatial Foundation (OSGeo) [RD5] and [RD6]OSGeo is an open development community that applies a responsible project governance model. The OSGeo seal provides added confidence in the viability and safety of the project. MapServer implements the WCS 1.1 interface.

MapServer excels at supporting a broad range of raster data sources by means of GDAL, an open source library for raster data manipulations written in C. It allows for a wide range of coordinate transformations by using the open source PROJ.4 library. Furthermore, it provides scripting language bindings (Python, PHP, and others) that allow to easily extend the functionality of the basic software.

MapServer supports various database bindings, most importantly to the geospatially enabled open source RDBMS PostgreSQL with its PostGIS extension. MapServer can be extended to use rasdaman, the raster image database developed by JUB.

MapServer's biggest drawback is that dynamic configuration is a relatively tricky thing to do. It relies on static Mapfiles for the configuration of an OGC Web Service which can be manipulated by means of the scripting language bindings. Database storage of dynamic configuration items, most importantly the metadata and file system paths of coverages uploaded with WCS-T, is not directly supported. It can be implemented using scripts or by extending the MapServer C code, however. MapServer's advantage is the support of fast-cgi, which is a protocol for interfacing interactive programs with a web server. This support provides a highly efficient implementation allowing MapServer to serve data, at

least via WMS, much faster then GeoServer.

GeoServer is a software package with capacities similar to MapServer implemented in Java. It is supported by an open source community as well. GeoServer is built on **Geotools**, an open source Java GIS toolkit. GeoServer is the reference implementation for WCS 1.1, but compliance with this standard has not yet been certified by the OGC.

GeoServer supports an equally wide range of raster formats and database backends as MapServer does. Among its benefits is an easy-to-use administration interface, which greatly simplifies configuration of data sources and service properties. Furthermore, it comes with an optional plug-in for image pyramid creation and manipulation which could be used for storing mosaics.

On the downside, GeoServer does only allow to store coverage and pyramid metadata (like the location of the tiles on the file system, and references to the original coverages a mosaic is composed of) in shapefiles on the file system. It needs to be extended to include metadata information stored in databases, which would be the method of choice for large collections of coverages or mosaics for reasons of performance, scalability and flexibility.

The **deegree** framework – written in Java, too – was the OGC Reference Implementation for WCS 1.0. However, the current version of the software does not yet support WCS 1.1, and thus we have to dismiss it from further consideration.

The **THREDDS Data Server** (TDS) is mentioned here for completeness, although it is not suitable for the purposes of the ODA project, as it provides only a very restricted implementation of WCS 1.0.

These considerations leave us with two viable open source software solutions for the implementation of online data access with respect to the criteria mentioned above: MapServer in combination with PostgreSQL/PostGIS for mosaic and pyramid storage and GeoServer.

- Given that coverage and mosaic handling using a combination of MapServer and PostgreSQL/PostGIS allows for great performance and flexibility;
- given MapServers performance advantage due to fast-cgi;
- given that GeoServer does not support this without modifications to the code of the distribution;
- given that EOX has ample experience with this open source software set-up and has successfully used it to implement raster data stores accessible through web interfaces;

we conclude that an approach combining MapServer and PostgreSQL/PostGIS is our first choice for implementation, GeoServer the second. Additionally, we propose to use rasdaman, an Open Source raster server usable as extension or side component to MapServer like GDAL, for the demonstration of advanced concepts.

5.6.2.2 Software development approach

We propose to implement changes by altering the source-code and trying to feed patches back to the MapServer community.

Figure 3 shows the high level architecture of MapServer. The part with brown background at the top of the figure shows the data input whereas the bottom parts in blue show the possible output.



Figure 3: MapServer Architecture from [RD7]

The implementation task will focus on the compliance to the WCS Application Profile for EO Sensors. However, as the Application Profile is still under development.

The profile will especially reflect on the new WCS-T specification which will be used to address the Use Case scenarios US4_1 to US4_3. as defined in the Requirement Baseline Document (HMA-FO_ODA-RB-SSS_EOX). From Figure 3 it can be seen that the data processed with MapServer currently flows only from the top to the bottom. The WCS-T specification foresees the data flow in the opposite direction. Currently no available Open Source implementation of the WCS-T specification is known to the consortium.

Currently there is no implementation of the WCS-T specification foreseen in the MapServer community. Due to the change of the target (WCS 1.1.1 \rightarrow WCS 2.0) extending the MapServer with the WCS-T interface, as stated in the proposal, will not anymore be the first implementation task.

The Transaction operation accepts a description of input coverages as parameter (where applicable), that contains references to the actual metadata and image files to be ingested. The operation can be synchronous or asynchronous; it returns an XML response confirming that the transaction has been processed by the server, or an exception report message in the case of failure.

The raster image data and metadata may be either attached to the request (e.g. via HTTP POST) or merely referenced to by a URL. In the latter case, the resources have to be made available by the user or the service that issues the request, using some file transfer mechanism (e.g. HTTP, FTP). Currently, the reference is expected to point to the data and metadata files themselves, but not to another WCS server.

Note that the terms client and server in that context do only refer to the roles of the participants with respect to the WCS-T protocol. If the data is not attached to the request, it has to be retrieved by the receiver of the WCS-T request by some other file transfer mechanism such as HTTP or FTP. In that case the originator of the request (the WCS-T client) usually will act as a server – the roles are reversed for actual data transfer.

The implementation will support insertion, update, partial update, and deletion of coverages on a server accessible via WCS to the extent the WCS specification allows it. Among the data formats supported will be at least GeoTiff and if possible JPEG2000. For these formats the implementation will be tested and demonstrated. However, various other file formats like HDF, NetCDF, JPEG, PNG, etc. could also be made easily available through the Geospatial Data Abstraction Library (GDAL).

The WCS interface is available at the output side, but not on the input. This means that MapServer can not act as WCS client. This functionality, however, which will be available with rather small effort when WCS-T is implemented. The consortium will propose this extension to the MapServer community after finishing the WCS-T implementations.

Besides the possibility to extend the C source-code of the OSS MapServer there exists the MapScript scripting interface. MapScript allows to directly create and access MapServer objects instead of interacting with MapServer through its Common Gateway Interface (CGI) and Mapfiles. The MapScript interface is generated by the Simplified Wrapper Interface Generator (SWIG) [RD8] library which provides language bindings to scripting languages like Python [RD9].

The current WCS specification foresees SOAP over HTTP as optional encoding. If it is decided to use these SOAP bindings within the EO-WCS Application Profile we propose to use the afore mentioned scripting interface for the implementation. This ensures the most efficient and flexible way to implement the support of the additional communication protocol layer SOAP.

MapServer currently supports WCS versions 1.0 and 1.1. The new developments for the MapServer based on WCS 2.0 will need, due to the increased development volume, to get a much closer involvement of the MapServer developer community to open a new development branch. Also the EO Application Profile will be based on WCS 2.0.

Special attention is drawn at the presentation of accompanying metadata. This includes client side implementations as well as the WCS server side which has to support the delivery of requested metadata be it in the GetCapabilities or DescribeCoverage response

Additional functionalities or at least configuration needs arising from the Application Profile which are not available in the current MapServer implementation. It has to be decided on a case by case basis in conjunction with the ESA technical officer if this is better implemented in the source-code or by using the MapScript scripting interface to MapServer.

MapServer is an official project of the Open Source Geospatial Foundation (OSGeo) [RD5] since it has graduated from incubation on 16th December 2008 [RD6] Thus the major

software configuration management tools are provided by OSGeo. These tools include a Subversion (SVN) repository [RD10], [RD11], an Issue and Bug Tracker and Wiki utilizing the software Trac [RD12], [RD13], and some mailing lists provided by the software Mailman [RD14], [RD15] For online discussions, which would be very ineffective via email, the MapServer community uses the Internet Relay Chat (IRC) [RD16].

All these software tools are licensed as Open Source software and are also used by EOX routinely.

EOX will actively participate in the community process to optimize the possible impact of the developed extensions. The developments and extensions will be fed into the main development branch of MapServer, if agreed by the community, hosted at OSGeo. Otherwise EOX will set-up an freely available SVN at their own premises with a development branch starting from the latest stable release of MapServer to develop and integrate the new extensions and also provide it as OSS. EOX will provide on a best effort basis upstream releases of the extensions as the MapServer community releases new stable versions. Patches for MapServer including the stable releases of the extensions will be made available either on SourceForge [RD17] or GoogleCode [RD18]. The choice is almost only a matter of taste since for software development the installation at EOX will be used and only stable patches will be fed to the chosen repository. However, the completely preferred solution is a close cooperation with the community which includes the usage of OSGeo repository and other tools.

Since GeoServer also is a community project comparable configurations can be found and equivalent mechanisms used if, in consultation with the responsible ESA technical officer, decision will be made to not use MapServer.

Based on the facts presented in the above comparison the decision has been in favour of MapServer as the OSS tool for the developments of the demonstration implementation.

5.7 Design requirements and implementation constraints

This section lists any requirements driving the design of the software item under specification and any identified implementation constraint. This may include software standards; design requirements; specific design methods to be applied; requirements relevant to numerical accuracy; specific constraints induced by reused software (e.g. COTS, free software and open source).

Identifier	Description	Ver.M.	Sig.
ODA_DRC_010	OGC standards shall be applied where applicable.	D	E_Y
ODA_DRC_020	Standards and Recommendations developed in the frame of the HMA project shall be applied on request (where applicable).	D	E_Y
ODA_DRC_030	In case JPEG-2000 datasets have to be created	D	D_Y

	utilizing proprietary transformation formats or libraries (e.g. ECW, MrSID, Kakadu) the license has to be provided as <i>Customer Furnished Item</i> .		
ODA_DRC_040	Computer and storage Hardware has to be provided as Customer Furnished Item.	D	E_Y

5.8 Security and privacy requirements

This section describes any security and privacy requirements.

Identifier	Description	Ver.M.	Sig.
ODA_SER_010	The ODA service shall respect security and access control mechanisms based on [AD16]	D	E_Y
ODA_TRA_020	The ODA service shall respect security and access control mechanisms for transactions.	D	E_Y
ODA_TRA_030	ODA shall maintain security on the dataset. Data shall be protected by copyright and security processes that block illegal use or downloading.	D	E_B
ODA_TRA_040	The data accessible online shall be protected for illegal downloading. The End User should have accesses to the data for discovery and visualization purpose. When he wants to process or download the data, the platform should identify the user and check his rights as defined on SLA.	D	E_B
ODA_TRA_050	The security constraint shall not be a barrier to the commercial activity. The End User accesses the data online and the security is applied only when necessary with a non significant impact of ease access to the data for authorized users.	D	E_B
ODA_TRA_060	The security level shall be in line with the threat. The End User accesses free online data (free sub sampling data of commercial product), no security applied for normal use. Nevertheless, in case of massive request that demonstrate no real user connected, the platform should apply a minimum security process as authoritarian disconnection and blacklisted user action.	D	E_B

5.9 Portability requirements

ECSS: list any portability requirement

5.10 Software quality requirements

This section lists any quality requirements.

Identifier	Description	Ver.M.	Sig.
ODA_QR_010	The ODA system shall be developed to operate under a Linux Operating System	D	E_Y

5.11 Software reliability requirements

This section lists any reliability requirements.

Identifier	Description	Ver.M.	Sig.
ODA_SR_010	The ODA system should be made available utilizing an instantiation of Linux (possibly Linux-HA) and the appropriate hardware (fail-over system).	D	D_Y

5.12 Software maintainability requirements

This section lists any maintainability requirements.

Identifier	Description	Ver.M.	Sig.
ODA_SMR_020	The developers and maintainers require a user account on the installation platform.	D	E_Y
ODA_SMR_020	The system shall provide access via a SSH connection for the developers and maintainers.	D	E_Y
ODA_SMR_030	Root access or superuser (sudo) access rights will be required for the developers and maintainers (at least during initial setup and operations).	D	E_Y

5.13 Software safety requirements

This section lists any safety requirements

Identifier	Description	Ver.M.	Sig.
ODA_SR_010	Access to the ODA Hardware shall be limited.	D	E_B

5.14 Software configuration and delivery requirements

ECSS: list any requirement applicable to the selected delivery medium and any software configuration applicable.

5.15 Data definition and database requirements

This section lists any requirement related to specific data format or structure to be exchanged with other systems or any database requirements allowing to take into account e.g. mission and product specific constraints.

Identifier	Description	Ver.M.	Sig.
ODA_DDD_010	PostgreSQL with the PostGIS extension shall be used.	D	E_Y

5.16 Human factors related requirements

ECSS: list any requirement applicable to:the personnel, manual operations, human-equipment interactions, constraints on personnel, concentrated human attention areas and that are sensitive to human errors and training, and human factors engineering.

5.17 Adaptation and installation requirements

ECSS: list any requirement applicable to adaptation data and to specific installation

6 Validation requirements

This section describes the validation approach applicable to each uniquely identified requirement, utilizing a validation matrix (requirements to validation approach correlation table).

To allow the evaluation of the EO-WCS specification and validation of the developed server and client side software a demonstration and test installation is foreseen in ODA. The final decision on which physical hardware this installation will be hosted will be taken together with the ESA Technical Officer during the project. However, we propose to install the server software at two different locations to:

- demonstrate the interoperability of the solution;
- simulate real life scenarios where two servers talk to each other (see e.g. HMA-FO_ODA-RB-SSS_EOX Use cases)

The first instantiation is to be installed at ESA premises to simulate the CDS component of the GSC-DA. The second installation shall be hosted by the consortium member Spot acting as GCM at their premises. This set-up allows the simulating of a possible adoption of the developed EO-WCS in the course of GSC-DA both at ESA and contributing missions.

7 Traceability

ECSS: traceability matrices:

1. from the upper level specification requirements to the requirements herein (forward traceability table), and

2. from the requirements contained herein to the upper level applicable specification (backward traceability table).

8 Logical model description

ECSS: include a top-down description of the logical model.

The method used to express the logical model shall be described. Diagrams, tables, data flows and explanatory text may be included. Diagrams, tables, data flows and explanatory text may be included. The behavioral view of the software logical model shall be also described in the SRS.

NOTE1: The logical model can be the result of an iterative verification process with the customer. It also supports the requirements capture, documents and formalizes the software requirements.

NOTE 2: A logical model is a representation of the technical specification, independent of the implementation, describing the functional behavior of the software product. The logical model is written with a formalized language and it can be possibly executable. Formal methods can be used to prove properties of the logical model itself and therefore of the technical specification. The logical model allows in particular to verify that a technical specification is complete (i.e. by checking a software requirement exists for each logical model element), and consistent (because of the model checking).

Heterogeneous Mission Accessibility - Follow-On - Technical Specification

---- End of Document ----