WFS 3.0
RESTful Geospatial Services
Quality of Service
MapML
Earth Observation Platforms
Machine Learning & AI
Aviation Registries & Semantics
Web Services Security
Swath Data and CF Convention
3D LIDAR for First Responders
Federated Clouds & Workflows
OGC Compliance Testing
Information modelling
Augmented Reality
Geospatial Data Portrayal
Modeling & Simulation
TESTBED 14
Open Geospatial Consortium Innovation Program 2017/18

Testbed-14 Call for Participation (CFP)

Version 2.4.1 - 21 March, 2018

Table of Contents

Corrigenda

The following table identifies all corrections that have been applied to this CFP compared to the original release. Minor editorial changes (spelling, grammar, etc.) are not included.

Section	Description
Introduction	[20 December] Added link to *Part 2 ITT* solicitation and revised the language in the Note to past tense.
Summary of Testbed 14 Deliverables	[20 December] Added Note reminding the reader that details of the ITT bidding requirements are provided earlier in the document. Added ***ITT*** deliverables to *CFP Deliverables and Funding Status* table.
CityGML and Augmented Reality	[20 December] Last paragraph removed. Previous paragraph modified to include ARML.
B.9.2. Deliverables	[20 December] Descriptions of D136 and D137 corrected. Both referenced D108/D109 erroneously. Now references point to D134 and D135.
EOC Deliverables	[17 January] Added EOC Deliverables to *Appendix B*.
Master Schedule	[18 January] Revised Kickoff dates to 10-12 April.
Master Schedule	[4 February] Synchronized Final DER delivery date to coincide with TC 3-week rule.
Exploitation Platform	[21 March] Repaired ExploitationPlatform link.

Abbreviations

The following table lists all abbreviations used in this CFP.

3DPS	3D Portrayal Service
ABI	Activity Based Intelligence
AOI	Area of Interest
AMQP	Advanced Message Queuing Protocol
AR	Augmented Reality
AtomPub	Atom Publishing Protocol
AVI	Aviation
BBOX	Bounding Box
BPMN	Business Process Model and Notation
CDR	Content Discovery and Retrieval
CIS	Coverage Implementation Schema
CITE	Compliance Interoperability and Testing
CF	Climate and Forecasting
CFP	Call for Participation
CMD	Command Center
CSMW	Community Sensor Model Working Group
CSW	Catalog Service Web
CTL	Compliance Testing Language
DAP	Data Access Protocol
DCAT	Data Catalog Vocabulary
DDIL	Denied, Degraded, Intermittent, or Limited Bandwidth
DGIWG	Defense Geospatial Information Working Group
DISA	Defense Information System Agency
DWG	Domain Working Group
EO	Earth Observation
EOWCS	Earth Observation Profile Web Coverage Service
ER	Engineering Report
EXI	Efficient XML Interchange format
FGDC	Federal Geographic Data Committee
FIXM	Flight Information Exchange Model
FO	Field Operations
GDAL	Geospatial Data Abstraction Library
GEOINT	Geospatial intelligence
GeoXACML	Geospatial XACML
GIBS	Global Imagery Browse Services
GML	Geography Markup Language

GUI	Graphical User Interface
HDF	Hierarchical Data Format
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext transfer protocol secure
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
JSON-LD	JSON Linked Data
KML	Keyhole Markup Language
LiDAR	Light detection and ranging
MEP	Mission Exploitation Platform
MTOM	Message Transmission Optimization Mechanism
NASA	National Aeronautics and Space Administration
NetCDF	Network Common Data Form
NetCDF-CF	NetCDF Climate Forecasting
NSG	National System for Geospatial Intelligence
OAuth	Open Authorization
OBP	Object Based Production
OGC	Open Geospatial Consortium
OPeNDAP	Open-source Project for a Network Data Access Protocol
PKI	Public Key Infrastructure
PMT	(ShapeChange) Profile Management Tool
POI	Points-of-interest
PubSub	Publication Subscription
RDF	Resource Description Framework
SAML	Security Assertion Markup Language
SOS	Sensor Observation Service
SPARQL	SPARQL Protocol and RDF Query Language
SSO	Single Sign On [https://en.wikipedia.org/wiki/Single_sign-on]
SWAP	Size, Weight, and Power
SWE	Sensor Web Enablement
SWG	Standards Working Group
T13	Testbed-13
TEAM	Test, Evaluation, And Measurement Engine
TEP	Thematic Exploitation Platform [https://tep.eo.esa.int]
TSPI	Time-Space-Position-Information Standard
TWMS	Tiled Web Mapping Service
US	United States
UML	Unified Modeling Language
USGS	U.S. Geological Survey

W3C	World Wide Web Consortium
WCPS	Web Coverage Processing Service
WCS	Web Coverage Service
WFS	Web Feature Service
WIS	Web Integration Service
WKT	Well Known Text
WMS	Web Mapping Service
WMTS	Web Mapping Tile Service
WPS	Web Processing Service
WS	Web Service
WSDL	Web Services Description Language
XACML	eXtensible Access Control Markup Language
XOP	XML-binary Optimized Packaging
XXE	XML External Entity Injection

Introduction

The Open Geospatial Consortium (OGC®) is releasing this ***Call for Participation*** ("CFP") to solicit proposals for the OGC Testbed-14 ("the Testbed") initiative.

The Testbed 14 solicitation is being issued in two parts due to specialized sponsor procurement requirements:

- Part 1 - this CFP document ("Part 1 CFP" or just "CFP")
- Part 2 - an Invitation To Tender [http://emits.sso.esa.int/emits/owa/emits_external.showsum?typ1=8039&user=Anonymous&ctr=LOGICAUK] pack ("Part 2 ITT" or just "ITT") under the European Space Agency (ESA) Electronic Mailing Invitation to Tender System [http://emits.sso.esa.int/emits/owa/emits.main] (EMITS). The *Part 2 ITT* can also be accessed manually in EMITS by selecting *Entities* → *CGI IT UK LTD* → *Open Invitations to Tender*.

Part 2 ITT is described separately from this document and has distinct response requirements. So any Bidder wishing to respond to ***both*** Part 2 ITT (described externally) and the *Part 1 CFP* (described herein) should deliver two separate proposals, one for each set of response requirements. A Bidder wishing to respond to one or the other (but not both) would submit only one proposal.

IMPORTANT

The *Part 2 ITT* was released on 14 December, 2017. To provide a complete description of Testbed-14 scope, this CFP documentation was extended through *Corrigenda* to combine all deliverables from both parts (*Part 1 CFP* and *Part 2 ITT*). All work items marked as belonging to *Part 2 ITT* have distinct response requirements that are provided as part of the ITT. Bidding on these ITT requirements and work items requires an independent proposal! ITT requirements and work items are only copied into the *Part 1 CFP* solicitation documents for completeness.

The *response requirements* described in the here in the CFP body and in Appendix A Management Requirements apply to *Part 1 CFP* requirements and deliverables only.

Under this *Part 1 CFP*, the OGC will provide cost-sharing funds on behalf of sponsoring organizations ("Sponsors") to partially offset expenses uniquely associated with the Testbed. This CFP requests proposals from organizations ("Bidders") wishing to participate in delivery and, in some cases, to receive cost-sharing funds. Any Bidder interested in Testbed participation should respond by submitting a proposal per the instructions provided herein.

OGC intends to involve as many technology developers and providers ("Participants", to be selected from among the Bidders) as possible to the extent that each Participant can contribute to and benefit from initiative outcomes. Not all proposals are required to request cost-sharing funds. While the majority are expected to include a combination of cost-sharing request and in-kind contribution, responses offering solely in-kind contributions (i.e., requesting no cost-sharing funds whatsoever) are also welcomed.

The only offers that should be included in formal proposals are those directly addressing express CFP requirements. Proposals covering additional requirements should be submitted separately for independent consideration.

The OGC Innovation Program [http://www.opengeospatial.org/ogc/programs/ip] provides global, hands-on, collaborative prototyping for rapid development and delivery of proven candidate specifications to the OGC Standards Program, where these candidates can then be considered for further action. In Innovation Program initiatives, Participants collaborate to examine specific geo-processing interoperability questions posed by the initiative’s Sponsors. These initiatives include Testbeds, experiments, pilots, and plugfests – all designed to foster the rapid development and adoption of open, consensus-based standards.

The OGC recently reached out to potential initiative sponsors to review the OGC technical baseline [http://www.opengeospatial.org/specs/?page=baseline], discuss results of prior initiatives, and identify current Testbed requirements.

IMPORTANT

Discussions are ongoing for potential sponsorship of additional requirements that are not yet firm enough for inclusion in this CFP. Should funding for these additional requirements be provided later, a follow-on CFP with a compressed response timeline might also be issued if the work can be included without interfering with the original Master Schedule or Appendix B Technical Architecture.

Benefits of Participation

In general, Bidders should propose specifically against the list of deliverables described under the Summary of Testbed Deliverables section below. But Bidders may go beyond *funded* deliverables to propose in-kind contributions that will address *unfunded* requirements as well. Participants should note, however, that Sponsors have committed to fund only those deliverables identified as being funded.

This Testbed provides a business opportunity for stakeholders to mutually define, refine, and evolve service interfaces and protocols in the context of hands-on experience and feedback. The outcomes are expected to shape the future of geospatial software development and data publication. The Sponsors are supporting this vision with cost-sharing funds to partially offset the costs associated with development, engineering, and demonstration of these outcomes. This offers selected Participants a unique opportunity to recoup a portion of their Testbed expenses.

Testbed Policies and Procedures

This CFP incorporates the following additional documents:

- CFP Appendix A Management Requirements,
- CFP Appendix B Technical Architecture,
- OGC Innovation Program Policies and Procedures [http://www.opengeospatial.org/ogc/policies/ippp],
- OGC Principles of Conduct [http://www.opengeospatial.org/ogc/policies/conduct], and
- OGC Intellectual Property Rights Policy [http://www.opengeospatial.org/about/ipr].

This Testbed will be conducted in accordance with OGC Innovation Program Policies and Procedures [http://www.opengeospatial.org/ogc/policies/ippp].

OGC Principles of Conduct [http://www.opengeospatial.org/ogc/policies/conduct] will govern all personal and public interactions in this initiative.

One Testbed objective is to support the OGC Standards Program [http://www.opengeospatial.org/ogc/programs/spec] in the development and publication of open standards. Each Participant will be required to allow OGC to copyright and publish documents based in whole or in part upon intellectual property contributed by the Participant during Testbed performance. Specific requirements are described under the "Copyrights" clauses of the OGC Intellectual Property Rights Policy [http://www.opengeospatial.org/about/ipr].

Initiative Roles

The roles generally played in any OCG Innovation Program initiative are defined in the OGC Innovation Program Policies and Procedures [http://www.opengeospatial.org/ogc/policies/ippp], including Sponsors, Bidders, Participants, Observers, and the Innovation Program Team ("IP Team").

Additional explanations of the roles of Sponsors, Bidders, Participants, and Observers are provided in the Tips for New Bidders.

The IP Team for this Testbed will include an Initiative Director, an Initiative Architect, and multiple Thread Architects. Unless otherwise stated, the Initiative Director will serve as the primary point of contact (POC) for the OGC.

Thread Architects will work with Participants, Sponsors, and each other to ensure that Testbed activities and deliverables are properly assigned and performed. They are responsible for scope and schedule control, as well as for within-thread and cross-thread communications. They will also provide timely escalation to the Initiative Director regarding any severe issues or risks that happen to arise.

Chapter 1. General Proposal Submission Guidelines

This section presents general guidelines for submitting a CFP proposal. Detailed instructions for submitting a response proposal using the *Bid Submission Form* web page can be found in Appendix A Management Requirements.

NOTE

The *Narrative Response Template* Word document and the *Financial Response Template* Excel spreadsheet that were required for proposal submissions under prior testbeds are no longer being used in this Testbed.

Proposals must be submitted before the appropriate response due date indicated in the Master Schedule.

Bidders responding to this CFP must be OGC members and must be familiar with the OGC mission, organization, and process. Proposals from non-members will be considered *provided that* a completed application for OGC membership (or a letter of intent to become a member) is submitted prior to (or with) the proposal.

Information submitted in response to this CFP will be accessible to OGC and Sponsor staff members. This information will remain in the control of these stakeholders and will not be used for other purposes without prior written consent of the Bidder. Once a Bidder has agreed to become a Testbed Participant, they will be required to release proposal content (excluding financial information) to all Testbed stakeholders. Commercial confidential information should not be submitted in any proposal and should generally not be disclosed during Testbed execution.

Bidders will be selected to receive cost sharing funds on the basis of adherence to the requirements stipulated in this CFP and the overall quality of their proposal. The general Testbed objective is for the work to inform future OGC standards development with findings and recommendations surrounding potential new specifications. Bidders are asked to formulate a path for producing executable interoperable prototype implementations that meet the stated CFP requirements, and for documenting the findings and recommendations arising from those implementations. Bidders not selected for cost sharing funds may still be able to participate in addressing the stated CFP requirements on a purely in-kind basis.

However, to help maintain a manageable process, Bidders are advised to avoid attempts to use the Testbed as a platform for introducing new requirements not included in Appendix B Technical Architecture. Any additional in-kind scope should be offered outside the formal bidding process, where an independent determination can be made as to whether it should be included in Testbed scope or not. Items deemed out-of-Testbed-scope might be more appropriate for inclusion in another OGC Innovation Program initiative.

Each selected Participant (including pure in-kind Participants) will be required to enter into a Participation Agreement contract ("PA") with the OGC. The reason this requirement applies to pure in-kind Participants is that other Participants will be relying upon their delivery to show component interoperability across organizational boundaries.

Each PA will include a statement of work ("SOW") identifying Participant roles and responsibilities. The purpose of the PAs is to encourage and enable Participants to work together to realize Testbed goals for the benefit of the broader OGC community.

1.1. Questions and Clarifications

Once the original CFP has been published, ongoing authoritative updates and answers to questions can be tracked by monitoring the CFP Clarifications page. Instructions for accessing this page are included under Proposal Submission Procedures.

Bidders may submit questions via timely submission of email(s) to the OGC Technology Desk (techdesk@opengeospatial.org [mailto:techdesk@opengeospatial.org]). Question submitters will remain anonymous, and answers will be regularly compiled and published on the CFP clarifications page.

OGC may also choose to conduct a Bidder’s question-and-answer webinar to review the clarifications and invite follow-on questions.

Chapter 2. Proposal Evaluation Criteria

Proposals will be evaluated according to criteria that can be divided into two areas: *Management* and *Technical*.

2.1. Management Criteria

- Bidder willingness and ability to comply with Appendix A Management Requirements,
- Feasibility of proposed solution utilizing proposed resources,
- Proposed in-kind contribution in relation to proposed cost-share funding request.

2.2. Technical Criteria

- Understanding of and compliance with requirements as stated in Appendix B Technical Architecture,
- Quality and suitability of proposed design, and
- Where applicable, proposed solutions are OGC-compliant.

Chapter 3. Master Schedule

The following table details the major Testbed milestones and events:

Mileston e	Date	Event
M01	8 December 2017	CFP Release (ITT released 14 Dec.)
M02	5 January 2018	Final Bidder Questions Due (CFP-only)
M03	12 January 2018	(CFP-only) Bidders Q&A Webinar (tentative)
M04	21 January 2018	CFP Proposal Submission Deadline (11:59pm U.S. Eastern time)
M05	2 February 2018	First Round of Bidder Notifications Started
M06	16 February 2018	Second Round of Bidder Notifications Started
M07	31 March 2018	All CFP Participation Agreements Signed
M08	***10-12 April 2018***	Kickoff Workshop Event (date updated on 18 Jan.)
M09	31 May 2018	Initial Engineering Reports (IERs)
M10	30 June 2018	Component Implementation Designs
M11	31 July 2018	IER-to-DER status check; TIE Connectivity Test; Early implementations of any component on which another component depends
M12	31 August 2018	TIE Readiness Review
M13	30 September 2018	TIE-Tested Component Implementations completed; Preliminary DERs complete & clean, ready for Testbed internal reviews
M14	31 October 2018	Ad hoc TIE demonstrations (as requested during the month) & Demo Assets posted to Portal; Near-Final DERs posted to Pending & WG review requested
M15	***21 November 2018***	Final DERs (incorporating WG feedback) posted to Pending to meet 3-week rule for vote at December TC Meeting (date updated on 4 Feb.)
M16	30 November 2018	Participant Final Summary Reports
M17	[Date TBD] December 2018	Demonstration Event

Table 1. Master schedule

3.1. Sequence of Events, Phases, and Milestones

The following diagram provides a notional schedule of major Testbed events and milestones, and their approximate sequence of occurrence. The Testbed will use rolling-wave project management whereby more detailed scheduling will take place as each milestone draws near.

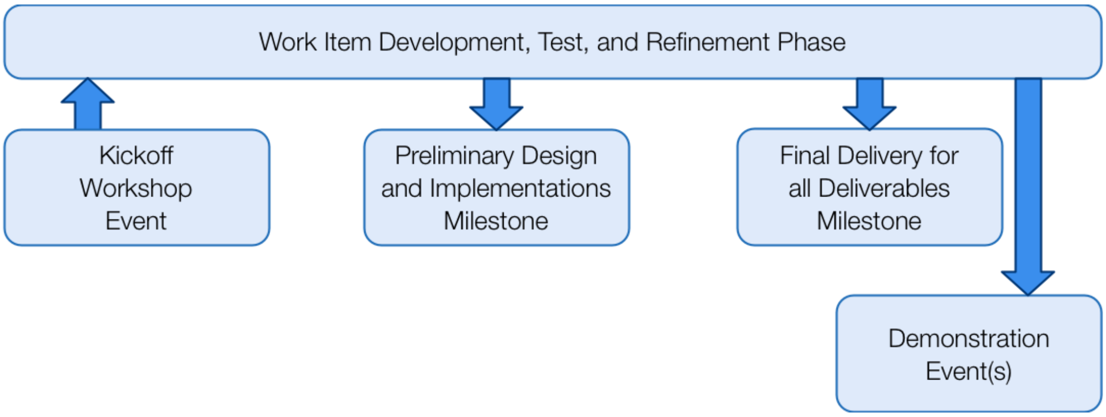

Work Item Development, Test, and Refinement Phase
Kickoff Workshop Event
Preliminary Design and Implementations Milestone
Final Delivery for all Deliverables Milestone
Demonstration Event(s)

Figure 1. Overview of events and milestones

Participant Selection and Agreements:

Once the original CFP has been published, ongoing authoritative updates and answers to questions can be tracked by monitoring the CFP Clarifications page. Instructions for accessing this page are included under Proposal Submission Procedures.

Bidders may submit questions via timely submission of email(s) to the OGC Technology Desk (techdesk@opengeospatial.org [mailto:techdesk@opengeospatial.org]). Question submitters will remain anonymous, and answers will be regularly compiled and published in the CFP clarifications document.

OGC may also choose to conduct a Bidder's question-and-answer webinar to review the clarifications and invite follow-on questions.

Following the closing date for submission of proposals, OGC will evaluate received proposals, review recommendations with Sponsors, and negotiate Participation Agreement (PA) contracts, including statements of work (SOWs), with selected Bidders. Participant selection will be complete once PA contracts have been signed with all Participants.

Kickoff Workshop: A Kickoff Workshop ("Kickoff") is a face-to-face meeting where Participants, guided by thread architects, will refine the Testbed architecture and settle upon specific interface models to be used as a baseline for prototype component interoperability. Participants will be ***required*** to attend the Kickoff, including the breakout sessions of each thread for which they were selected. Participants will be expected to use these breakouts to collaborate with other Participants and confirm intended *Component Interface Designs*.

After the face-to-face Kickoff, most Testbed activities will be conducted remotely via web meetings and teleconferences until the Demonstration Event near the end of Testbed execution.

Development of Engineering Reports, Change Requests, and Other Document Deliverables: Development of Engineering Reports (ERs), Change Requests (CRs) and other document deliverables will commence during or immediately after Kickoff. Participants will deliver an Initial Engineering Report (IER) plus several iterations of a Draft Engineering Report (DER). Full process details can be found in the OGC ER Process [https://portal.opengeospatial.org/files/?artifact_id=73351].

Under the Participation Agreement (PA) contracts to be formed with selected Bidders, ***ALL*** Participants will be responsible for contributing content to the ERs. But the ER Editor role will assume the duty of being the primary ER author.

IMPORTANT

As compared to the ER Editor role in Testbed-13, ER Editors in Testbed-14 must adopt a more ***proactive approach to consuming and documenting the knowledge*** being generated during component implementation. In other words, ER Editors will serve as primary authors in addition to their editor role. All participants are required to support the editors by making necessary documentation material available.

Component Design, Development, and Preliminary Testing: Participants will continue documenting detailed and final *Component Interface Designs* in the Testbed wiki. This documentation will allow task Participants to confirm a mutual understanding of each other's interfaces so that subsequent interoperability testing can take place. A preliminary *Technology Integration Experiment* ("TIE", sometimes also referred to as a "Technology Interoperability Experiment") *Connectivity Test* milestone will be used to ensure that Testbed service endpoints can be reached by Testbed clients.

IMPORTANT

Each proposed *service* component implementation deliverable should be accompanied by identification of one or more particular datasets that would be suitable for the proposed service. Details are provided under Data Requirements for Proposing Service Component Implementation Deliverables.

TIE Readiness Review: A TIE Readiness Review will be conducted with a Thread Architect to confirm that each TIE Participant is prepared to start conducting TIE testing with counterpart Participants.

Component Interoperability Testing, and Acceptance: Participants should deliver ***completed and fully TIE-tested*** component implementations no later than the 30 September milestone (unless an earlier milestone is identified in Appendix B Technical Architecture). The primary acceptance criterion for a component implementation deliverable is the conduct and recording of the TIE test. This test can also prove useful in accelerating the development of Demonstration Event assets such as video recordings.

Draft Engineering Reports: Participants should also deliver ***complete and clean*** Draft

Engineering Report (DERs) by the 30 September milestone. A *complete* DER is one for which all major clauses have been populated with meaningful content. A *clean* is one where all known editorial defects have been repaired. This milestone will impact ***ALL*** Testbed Participants, even component implementers, who will be responsible for making necessary documentation material available to the ER Editor for use in authoring the ER. Testbed Sponsors and Thread Architects will review these DERs in the weeks following delivery.

DER Rework and SWG/DWG Reviews: Participants will be required to perform rework based on the reviews from Sponsors and Thread Architects. The ER editor will then make a request to the selected OGC SWG or DWG to perform its review and to consider making a request that the DER be voted on by the OGC Technical Committee (TC) for publication. The OGC 3-week rule must be followed if the DER is to receive a vote at a TC Meeting. The DER must be posted to the members-only *OGC Pending Documents* directory to enable this TC vote. Participants will likely have to perform one final round of rework based on SWG/DWG feedback. The further along the DER is when submitted to the WG, the less rework will be required.

Final Summary Reports and Demonstration Event: Participants Final Summary Reports will constitute the close of funded activity. Further development work might take place to prepare and refine assets to be shown at the Demonstration Event.

Assurance of Service Availability: Participants selected to implement *service* components must maintain availability for a period of no less than one year after the Final Delivery Milestone. OGC might be willing to entertain exceptions to this requirement on a case-by-case basis.

Detailed requirements for meeting all these milestones are provided in Appendix A Management Requirements.

Chapter 4. Summary of Testbed 14 Deliverables

The following tables show the full set of Testbed 14 deliverables, including ID, deliverable name, task, and funding status.

A deliverable’s funding status can be funded ("F"), unfunded ("U"), or under negotiation ("Un-Neg"), depending on the current state of sponsor funding.

- For a deliverable with a funding status of "F", sponsor funding has already been confirmed.
- A deliverable with a funding status of "U" is within CFP scope, but has a lower priority and does not have any sponsor funding. It’s possible that a Sponsor might later agree to provide funding. But no Sponsor funding is currently available, and any bids for these deliverables would necessarily be purely in-kind.
- A deliverable with a funding status of "Un-Neg" is one for which a sponsor intends to provide funding, but a final commitment of this funding is still pending.

NOTE

A deliverable with a funding status of "***ITT***" can only be proposed under the *Part 2 ITT* solicitation. See the Main Body Introduction for details regarding how to bid on these deliverables.

Please note that each deliverable indicated as "F" or "Un-Neg" would be funded at most once. No deliverable should be interpreted as offering multiple instances. For any deliverable still under negotiation ("Un-Neg"), if funding for that deliverable ends up *not* being committed, any bid for cost-sharing on that deliverable will be dismissed. The deliverable would change to an unfunded "U" status, and a Bidder could potentially choose to contribute it purely in-kind.

IMPORTANT

Discussions are ongoing for potential sponsorship of additional requirements that are not yet firm enough for inclusion in this CFP. Should funding for these additional requirements be provided later, a follow-on CFP with a compressed response timeline might also be issued if the work can be included without interfering with the original Master Schedule or Appendix B Technical Architecture.

Testbed deliverables are organized into *threads*, which are described in detail in Appendix B Technical Architecture.

All Participants are ***required*** to provide at least some level of ***in-kind contribution*** (i.e., activities requesting no cost-share compensation). As a rough guideline, a proposal should include at least one dollar of in-kind contribution for every dollar of cost-sharing compensation requested. All else being equal, ***higher levels of in-kind contributions will be considered more favorably*** during evaluation.

Some participation may be fully in-kind. However, to help maintain a manageable process, Bidders are advised to avoid attempts to use the Testbed as a platform for introducing new requirements not included in Appendix B Technical Architecture. Any additional in-kind scope should be offered outside of the formal bidding process, where an independent determination can be made as to whether it should be included in Testbed scope or not. Items deemed out-of-Testbed-scope might be more appropriate for inclusion in another OGC Innovation Program initiative.

Any item proposed as a fully in-kind contribution to meet a requirement already included in Appendix B Technical Architecture will likely be accepted if it meets all the other evaluation criteria.

In the tables below, document deliverables are numbered D001 and increasing, and component implementation deliverables are numbered D100 and increasing.

The thread abbreviations are as follows:

- CITE: Compliance
- EOC: Earth Observation & Clouds
- MoPoQ: Modeling, Portrayal, and Quality of Service
- NextGen: Next Generation Services

4.1. CFP Deliverables and Funding Status

Additional technical details can be found in Appendix B Technical Architecture.

ID	Document / Component	Thread	Funding Status
D003	Secure Client Test ER	CITE	F
D027	Compliance ER	CITE	F
D112	Secure Client Tests and Implementations	CITE	F
D152	DGIWG CAT 2.0.2 Tests	CITE	F
D153	DGIWG CAT 2.0.2 Reference Implementation	CITE	F
D154	WFS3.0 Compliance Tests	CITE	F
—	—	—	—
D007	Swath Coverage Engineering Report	EOC	F

ID	Document / Component	Thread	Funding Status
D008	Application Package ER	EOC	***ITT***
D009	ADES & EMS Results and Best Practices ER	EOC	***ITT***
D010	Authorization, Authentication, & Billing ER	EOC	***ITT***
D100	Secured ADES Implementation I	EOC	***ITT***
D101	Secured ADES Implementation II	EOC	***ITT***
D102	Secured EMS Implementation I	EOC	***ITT***
D103	Secured EMS Implementation II	EOC	***ITT***
D104	TEP Client I	EOC	***ITT***
D105	TEP Client II	EOC	***ITT***
D106	Application Package and App Development I	EOC	***ITT***
D107	Application Package and App Development II	EOC	***ITT***
D108	Billing Service Implementation I	EOC	***ITT***
D109	Billing Service Implementation II	EOC	***ITT***
D134	WCS for Swath Coverage v1.1 & REST	EOC	F
D135	WCS for Swath Coverage v1.1	EOC	F
D136	Client to WCS-EO 1.1 for Swath Coverage	EOC	F
D137	Client to WCS-EO 1.1 & REST for Swath Coverage	EOC	F
—	—	—	—
D001	SWIM Information Registry ER	MoPoQ	F
D002	Semantically Enabled Aviation Data Models ER	MoPoQ	F
D011	WMS QoSE ER	MoPoQ	F
D012	MapML ER	MoPoQ	F
D013	PointCloud Data Handling ER	MoPoQ	F
D022	NAS ER	MoPoQ	F
D029	Symbology Engineering Report	MoPoQ	F
D030	Machine Learning ER	MoPoQ	F
D115	Multiple WMS with QoSE support	MoPoQ	F
D116	Client with WMS QoSE support	MoPoQ	F
D117	WMS QoSE test suite	MoPoQ	F
D118	MapML Browser Extension	MoPoQ	F
D119	MapML Cloud Proxy	MoPoQ	F
D132	Image Repository and Feature Store	MoPoQ	F
D133	Client to Knowledge Base	MoPoQ	F
D141	Machine Learning Validation Client	MoPoQ	F
D144	NAS Implementation Shape Change	MoPoQ	F
D145	NAS Ontologies and Samples	MoPoQ	F
D160	Portrayal Ontology	MoPoQ	F

ID	Document / Component	Thread	Funding Status
D161	GeoPackage with Portrayal Support	MoPoQ	F
D162	WMS with Portrayal Support	MoPoQ	F
D163	WMTS with Portrayal Support	MoPoQ	F
D164	Machine Learning Knowledge Base	MoPoQ	F
D165	Machine Learning System	MoPoQ	F
—	—	—	—
D021	Secure Resource Oriented Geospatial Data Handling ER	NextGen	F
D023	Federated Clouds Engineering Report	NextGen	F
D024	Security ER	NextGen	F
D025	WPS-T ER	NextGen	F
D026	Workflow ER	NextGen	F
D028	CityGML and AR ER	NextGen	F
D040	Complex Feature Handling ER	NextGen	F
D041	Complex Feature Handling Study Support	NextGen	F
D113	Next Generation Service Implementation 2	NextGen	F
D140	Next Generation Service Implementation 1	NextGen	F
D142	Next Generation Client Implementation 1	NextGen	F
D143	Next Generation Client Implementation 2	NextGen	F
D146	NGA Analysis Tool WPS	NextGen	F
D147	Security Mediation Service WPS	NextGen	F
D148	BPMN Service	NextGen	F
D149	WPS-T Implementation	NextGen	F
D150	CSW for WPS Processes & Analytic Metadata	NextGen	F
D151	OAuth2.0 Authorization Server	NextGen	F
D155	CityGML and AR Service I	NextGen	F
D156	CityGML and AR Service II	NextGen	F
D157	CityGML and AR Client I	NextGen	F
D158	CityGML and AR Client II	NextGen	F
D159	CityGML and AR Content	NextGen	F
—	—	—	—

Table 2. CFP Deliverables and Funding Status

< end of main body >

Appendix A: Management Requirements

This appendix presents detailed CFP management requirements for submitting a bid. It also covers procedures for participation during Testbed execution. The Appendix B Technical Architecture presents the detailed CFP technical requirements.

All potential Bidders, ***even experienced Testbed Participants***, should read this appendix carefully from end-to-end (and notify OGC immediately if any defects are discovered). There are significant differences as compared to prior testbed solicitations. Bidders who are new to OGC Testbeds are also encouraged to review the Tips for New Bidders.

The following sections describe the processes for a Bidder to submit a proposal and for a Participant (i.e., a selected Bidder) to perform against a Participation Agreement (PA) contract. The order of topics roughly parallels the sequence described in the Master Schedule. In general, the term "activity" is used as a verb describing work to be performed and the term "deliverable" is used as a noun describing artifacts to be memorialized and delivered for inspection and use.

A.1. Proposal Submission Procedures

The process for completing a Testbed 14 proposal is essentially embodied in the online Bid Submission Form [https://proposal.opengeospatial.org/]. A summary is provided here for the reader's convenience.

Once an online account has been created, the user will be taken to a home page indicating the "Status of Your Proposal." If any defects in the form are discovered, this page includes a link for notifying OGC. The user can return to this page at any time by clicking the *OGC logo* in the upper left corner.

IMPORTANT

Because the *Bid Submission Form* is new for Testbed 14, it might contain some areas that are still brittle or in need of repair. Please notify OGC of any discovered defects. Periodic version updates will be provided as needed.

Please consider making **backup local copies** of all inputs in case any of them need to be re-entered.

Please also note that this form has already gone "live" as of the CFP release date. Any submitted bids will be treated as earnest submissions, even those submitted well before the response deadline. Be certain that you intend to submit your proposal before you click the *Submit* button on the *Review* page.

Clicking on the *Clarifications* link will navigate to the CFP clarifications page.

On the far right, the *Review* link navigates to a page summarizing all the deliverables the Bidder is proposing.

> **TIP**
>
> Consider regularly creating printed output copies of this *Review* page at various checkpoints during proposal creation in case an input error is made later.

Once the final version of the information for all the proposed deliverables has been entered, the Bidder can submit the completed proposal to OGC by clicking the *Submit* button at the bottom.

> **TIP**
>
> In general, up until the time that the user clicks this *Submit* button, the proposal may be edited as many times as the user wishes. However, this initial version of the form contains no "undo" capability, so please use caution in over-writing existing information.

The user is provided an opportunity under *Attached Documentation* at the bottom of this page to attach collateral documentation (one document per proposal). This document could conceivably contain any specialized information that wasn’t suitable for entry into a *Proposed Contribution* field under an individual deliverable. It should be noted, however, that this additional documentation will only be read on a ***best-effort basis***. There is no guarantee it will be used during evaluation to make selection decisions; rather, it could optionally be examined if the evaluation team feels that it might help in understanding any specialized (and particularly promising) contributions.

The *Propose* link takes the user to the first page of the proposal entry form. This form contains fields to be completed once per proposal such as names and contact information.

It also contains an optional *Organizational Background* field where Bidders (particularly those with no experience participating in an OGC testbed) may provide a description of their organization. It also contains a click-through check box where each Bidder will required (before entering any data for individual deliverables) to acknowledge its understanding and acceptance of the requirements described in this appendix.

Clicking the *Update and Continue* button then navigates to the form for submitting deliverable-by-deliverable bids. On this page, existing deliverable bids can be modified or deleted by clicking the appropriate icon next to the deliverable name. Any attempt to delete a proposed deliverable will require scrolling down to click a *Confirm Deletion* button.

To add a new deliverable, the user would scroll down to the *Add Deliverable* section and click the *Deliverable* drop-down list to select the particular deliverable of interest.

The user would then enter the required information for each of the following fields (*for this deliverable only*). Required fields are indicated by an asterisk ("*"):

- Estimated *Projected Labor Hours** for this deliverable
- *Funding Request**: total U.S. dollar cost-share amount being requested for this deliverable (to cover burdened labor only)
- Estimated *In-kind Labor Hours** to be contributed for this deliverable
- Estimated *In-Kind Contribution*: total U.S. dollar estimate of the in-kind amount to be contributed for this deliverable (including all cost categories)

Cost-sharing funds may ***only*** be used for the purpose of offsetting burdened labor costs of development, engineering, and demonstration of Testbed outcomes related to the Participant's assigned deliverables. By contrast, the costs used to formulate the Bidder's in-kind contribution may be much broader, including supporting labor, travel, software licenses, data, IT infrastructure, etc.

Finally, a Bidder proposing to deliver a *Service Component Implementation* should use the *Proposed Contribution (Please include any proposed datasets)* field to identify what suitable ***datasets*** would be contributed (or what data should be acquired from another identified source) to support the proposed service. Full details of this requirement can be found under Data Requirements.

This field *Proposed Contribution (Please include any proposed datasets)* could also be used to provide a succinct description of what the Bidder intends to deliver for this work item to meet the requirements expressed in Appendix B Technical Architecture. This language could potentially include a brief elaboration on how the proposed deliverable will contribute to advancing the OGC technical baseline [http://www.opengeospatial.org/specs/?page=baseline], or how implementations enabled by the specification embodied in this deliverable could add specific value to end-user experiences.

However, to help maintain a manageable process, Bidders are advised to avoid attempts to use the Testbed as a platform for introducing new requirements not included in Appendix B Technical Architecture. Any additional in-kind scope should be offered outside the formal bidding process, where an independent determination can be made as to whether it should be included in Testbed scope or not. Items deemed out-of-Testbed-scope might be more appropriate for inclusion in another OGC Innovation Program initiative.

The statements of work (SOWs) in the *Participation Agreement* (PA) contracts will ultimately require performance during testbed *execution* against the deliverables as described in Appendix B Technical Architecture plus any subsequent *Corrigenda*.

A single bid may propose deliverables arising from any number of Testbed threads. To ensure that the full set of sponsored deliverables are made, OGC might negotiate with individual Bidders to drop and/or add certain deliverables from their proposals.

A.2. Conditions for Participation

A selected Bidder will become a Testbed Participant by executing a *Participation Agreement* (PA) contract with OGC, which will require the following conditions:

- Each Bidder should use the *Bid Submission Form* to make their proposal.
- Proposals from non-OGC-members will be considered *provided that* a completed application for OGC membership (or a letter of intent to become a member) is submitted prior to (or with) the proposal.
- No work facilities will be provided by OGC. Each Participant will be required to perform its PA obligations at its own provided facilities and to interact remotely with other Testbed stakeholders.
- In general, a proposed component deliverable based on a product that has earned OGC Certification will be evaluated more favorably than one which has not.
- Bidders proposing to build interoperable components should be prepared to test and demonstrate interoperability with components supplied by other Participants. In particular, server-endpoints must be accessible over the public Internet during Testbed TIEs.
- Participants selected to implement component deliverables will be expected to participate in the full course of interface and component development, TIEs, and demonstration support activities throughout initiative execution. Participants selected as Editors will also be expected to participate in the full course of activities throughout the initiative, documenting implementation findings and recommendations and ensuring document delivery.
- Participants should remain aware of the fact that the Testbed components will be developed across many organizations. To maintain interoperability, each Participant should diligently adhere to the latest technical specifications so that other Participants may rely on the anticipated interfaces during the TIEs.
- All Selected Participants (both cost-share and pure in-kind) must send ***at least one technical representative per assigned thread to the Kickoff Workshop***. Participants are also encouraged to send at least one technical representative to the *Demonstration Event* tentatively scheduled for December, 2018.

A.3. Proposal Evaluation and Invitations to Selected Bidders

Specific proposal evaluation criteria were presented earlier. Several process steps to be conducted by the IP Team are presented below to aid readers in understanding the overall process.

Soon after the proposal submission deadline, the IP Team and Sponsors will begin reviewing proposals, examining suitability of the proposed deliverables in light of the CFP requirements. During this analysis, the IP Team might need to contact Bidders to obtain clarifications and better understand what is being proposed.

At the *Decision Technical Evaluation Meeting I* (TEM I), the IP Team will present Sponsors with draft recommendations regarding which parts of which proposals should be offered cost-share funding. Sponsors will use this opportunity to suggest modifications.

The IP Team will notify each Bidder of their selection, including a description of the particular deliverables being offered, and OGC's intent to enter into a *Participation Agreement* (PA) with them. Selected Bidders ***must be available for these contacts to be made*** to enable confirmation of continued interest. Bidder acceptance of the offer essentially ***locks in the scope*** that will be used to form the SOW to be attached to the forthcoming PA. A Bidder may reject the offer (for example, if the offer is to deliver only a small subset of its proposed deliverables).

A *Decision Technical Evaluation Meeting II* (TEM II) meeting will be conducted to review the outcomes of initial offers to selected Bidders, and any modifications that were needed due to Bidder rejections.

Following TEM II, the IP Team will develop the formal SOW and attach it to the full PA for each selected Bidder. Selected Bidders ***must be available*** to enable finalization of their PA contract.

Each PA will include a final identification of all assigned deliverables. The specific requirements that these deliverables must meet will be those documented in Appendix B Technical Architecture (plus any subsequent corrigenda).

Because Testbeds tend to deal with the lowest maturity level of OGC standards technology, they operate under the principle that "interoperable running code wins" (based loosely on an Internet Engineering Task Force founding belief [https://www.ietf.org/tao.html#rfc.section.2]).

In contrast with ordinary commercial contracts, which often focus on delivering a single-customer solution, the *Participation Agreement* contracts will encourage Participants to work in the ***broader interests of the OGC community***. These interests are generally represented in the OGC Standards Working Groups (SWGs) and Domain Working Groups (DWGs). Testbed sponsorship enables a selected subset of these interests to be brought to bear on the production of *interoperable running code* and the reporting back of findings and recommendations as Change Requests (CRs) and document deliverables under the OGC Engineering Report Process [https://portal.opengeospatial.org/files/?artifact_id=73351].

In general, at least one Participant will be assigned for each client-server pair as an *Engineering Report* (ER) Editor, who will be responsible for authoring the ER, though there might be exceptions. See General Requirements for Proposing Deliverables for details. An *ER Template* will be provided to assist in the breakdown of content into multiple chapters describing specific findings and recommendations, a summary chapter, and supporting chapters (e.g., containing references and revision history).

IMPORTANT

As compared to the ER Editor role in Testbed-13, ER Editors in Testbed-14 must adopt a more proactive approach to consuming and documenting the knowledge being generated during component implementation. In other words, ER Editors will serve as primary authors in addition to their editor role. All participants are required to support the editors by making necessary documentation material available.

OGC members who who are not selected to make any deliverables may still observe all Testbed activities by visiting the (members-only) OGC Observer Agreements [https://portal.opengeospatial.org/?m=public&orderby=default&tab=6] page and registering to become a Testbed-14 Observer. The Observer Agreement will require that Observers follow the same intellectual property protection rules as apply to other Testbed stakeholders (e.g., to avoid sharing other stakeholder’s confidential information with individuals outside the Testbed).

Testbed stakeholders should also avoid making any representations that any of the intermediate technical decisions made internally during Testbed execution reflect any sort of endorsed position of the OGC. Formal Testbed recommendations will become "official" once the containing ER has received approval to be made public.

A.4. Kickoff Workshop

Testbed execution will commence with a Kickoff Workshop event ("Kickoff"). In general, the Kickoff ensures a common Participant understanding of the overall Testbed architecture and of all interfaces relevant to their assigned deliverables. Under the Testbed’s rapid pace, and guided by the Thread Architect, periods of development will be followed by synchronization among component developers, so that the likelihood of interoperation is maximized at each iteration. To maintain interoperability, each Participant should diligently adhere to the latest agreed specifications so that other Participants may rely on the anticipated interfaces in subsequent TIE testing.

Since Participants will be relying on one another to deliver interoperable components, all Participation Agreement (PA) contracts should be settled by the start of Kickoff. Any Participant that has not yet executed a PA by start of Kickoff will be required to ***attest to its commitment*** to a PA Statement of Work (SOW). The full PA must then be executed with

OGC ***no later than Kickoff completion*** (two days after Kickoff start).

The Kickoff will include both plenary and thread-based breakout sessions. Thread breakouts will be used to ensure a mutual understanding of the detailed interfaces that will support component interoperability. The Kickoff will also present an opportunity to finalize regularly scheduled thread and subthread teleconferences ("telecons").

All selected Participants must send ***at least one technical representative per assigned thread to the Kickoff Workshop***, and a representative should be present at the breakout sessions for ***every thread*** for which the Participant has been assigned a deliverable.

IMPORTANT

In-person attendance at these breakout meetings has been found to be critical to later Testbed success. Remote attendance will not be an option.

These requirements will also apply to Participants not requesting any cost-share funding (i.e., purely in-kind contribution) since their deliverables are still being ***relied upon*** by other Participants.

A.5. Participant Communication and Reporting

A.5.1. Points of Contact

Each selected Participant should designate a primary point of contact ("Primary POC") who shall remain available throughout Testbed execution for communications regarding status. The POC should identify at least one alternative point of contact to support the Primary POC as needed. The POCs should provide contact information including their e-mail addresses and phone numbers.

All proposals should include a statement attesting to the POCs' understanding and acceptance of the duties described herein. This statement will be enabled by a simple check-box in the *Bid Submission Form*.

A.5.2. Kickoff Status Report

Selected Participants should provide a one-time Kickoff Status Report that includes a list of personnel assigned to support the initiative and assurance that the Participants understand the schedule for all of its deliverables. This report should be submitted in electronic form to a designated email address no later than the last day of the Kickoff event.

A.5.3. Monthly Progress Reports

Participant Business and Technical POCs should provide, respectively, monthly *business* and *technical* progress and status reports due on the 3rd of each month (or the first working day thereafter) and covering the previous month.

Monthly Technical Reports will be accessible to all Testbed Stakeholders and should contain no confidential information (e.g., labor rates). *Monthly Business Reports* may contain confidential information and should be shared only with identified OGC POCs.

Any Participant who has a reliable forecast of what will take place in the remaining days of any particular reported month may submit its report early and subsequently report any urgent, last-minute updates via a follow-on email.

Detailed monthly reporting requirements and procedures will be provided during PA contract negotiation. It is likely that the Portal-based *Actions* feature (used in Testbed-13) will continue to be used for Testbed-14 monthly technical reporting.

The IP Team Thread Architects will review action item status on a weekly basis with assigned Participants, who should be available for these contacts to be made. Additional details regarding project management activities during Testbed execution can be found at Project Monitoring and Control.

As a preview, monthly technical reports will likely require the following information for each deliverable:

- Deliverable ID and Name
- Health: G, Y, R (green, yellow, red)
- %complete (0%-100%)
- Work accomplished in reporting period (last month)
- Work anticipated in reporting period+1 (current month)
- Any known risks or issues (see definitions under Project Monitoring and Control)
- Response to mitigate the risk or remedy the issue

A.5.4. Final Summary Report

Participants should provide a *Final Summary Report* near the end of Testbed execution. Detailed requirements and procedures will be provided during PA contract negotiation. These reports will likely require the following information:

- Describe, in detail, the work completed to fulfill the PA SOW items,

- Summarize Participant’s overall contribution to the project, and
- Present recommendations for future OGC Innovation Program and Standards Program efforts.

Any timely Participant who is up-to-date with all the Monthly Technical Status Reports in the Portal will likely be permitted to optionally satisfy the first "in detail" requirement by merely referencing the detail already contained in the Monthly Technical Reports. Timely Participants may also be permitted choose to build their Final Summary Report in the body of an email if desired (or, alternatively, submitted as a document attachment).

A.5.5. Regular Teleconferences

At least one of Participant Technical POC should be available for regularly scheduled telecons for *each* thread in which it is participating. Weekly thread telecons will be conducted. Records will be kept in meeting minutes posted to the Wiki and/or GoToMeeting recordings uploaded to the Portal.

These telecons are intended to accelerate understanding and action regarding relevant Testbed activities, particularly status of any risks or issues that might block or are blocking progress toward timely delivery.

Participants assigned as ER Editors may, with permission from the relevant Thread Architect, lead the substantive discussions during any ad hoc or regularly scheduled sub-thread telecons.

A.5.6. Correspondence and Collaboration

Participants should be able to utilize the following correspondence and collaboration tools:

- Send and receive emails to/from thread-based email broadcast lists,
- Contribute technical content using the OGC Member Wiki [https://portal.opengeospatial.org/wiki/System/BeginnersStartHere] tool,
- Participate in telecons using the GoToMeeting [https://www.gotomeeting.com/] tool,
- Edit ER source files in the Asciidoc [http://asciidoc.org/] format using the OGC ER Template [https://github.com/opengeospatial/er_template],
- Upload ER source files to the OGC Testbed-14 GitHub [https://github.com/] repository.

A.6. General Requirements for Proposing Deliverables

In general, Participants will be required to make deliverables meeting the requirements stated in CFP Appendix B Technical Architecture.

Under the Participation Agreement (PA) contracts to be formed with selected Bidders, all Participants will be responsible for contributing content to the ERs.

But the *ER Editor* role will assume the duty of being the ***primary*** ER author. As compared to the ER Editor role in Testbed-13, ER Editors in Testbed-14 must adopt a more ***proactive approach to consuming and documenting the knowledge*** being generated during component implementation. In other words, ER Editors serve as primary authors in addition to their editor role. All other participants are required to support the editors by making necessary documentation material available.

In addition, *component implementers* (Participants selected to deliver component implementations) will be responsible for contributing ***completed*** technical artifacts for inclusion as ER annexes. These include UML diagrams, XML schemas and instances, and abstract test suites, all of which have their own specialized clauses in the ER Template. Component implementers will also be responsible for contributing demo assets such as video recordings of component operation.

ER Editors will be responsible for observing the component implementation work and documenting (in the assigned ER) all findings and recommendations (except the technical artifacts identified above). They will also be responsible for ensuring the document's overall integrity, authoring summary material and ensuring that global sections (e.g., References, Terms, Revision History, Bibliography) have all been completed.

ER Editors will also be responsible for enabling all required ER reviews (and subsequent rework), including internal reviews by Thread Architects and Sponsors, as well as requests for reviews by an appropriate OGC Standard Working Group [http://www.opengeospatial.org/projects/groups/swg] (SWG) or Domain Working Group [http://www.opengeospatial.org/projects/groups/wg] (DWG).

ER Editors will also be responsible for creating (and referencing) any ***Change Requests*** (CRs) arising from the work described in the assigned ER. CRs can be created using the OGC CR system [http://ogc.standardstracker.org/].

A.7. Specific Requirements for Proposing *Document* Deliverables

The primary acceptance criterion for a document deliverable (e.g., an ER) will be approval from a Thread Architect to post the document to an OGC (members-only) folder called the *OGC Pending directory*. Permission to post will not be granted until the full OGC Engineering Report Process [https://portal.opengeospatial.org/files/?artifact_id=73351] has been followed, including the step to request a review from an OGC SWG or DWG. All appropriate tooling should also have been utilized (ER Template, Asciidoc, GitHub), and the document should have achieved a satisfactory level of consensus among all assigned Participants.

A.8. Specific Requirements for Proposing *Component Implementation* Deliverables

In general, prototype *component implementation* deliverables (including services, clients, datasets, and tools) will be provided by methods suitable to its type and stated requirements. A service (e.g. a WFS instance or even an intermediary component) will be delivered by deployment for use in the Testbed via an accessible URL. A client will be used to exercise a service to test and demonstrate interoperability. Components will be developed and deployed in all threads for integration and interoperability testing in support of agreed-upon thread scenario(s) and technical architecture. The Kickoff will provide the opportunity to establish many of the agreed-upon details. These technical artifacts could also be utilized for cross-thread scenarios in demonstration events.

More specifically, the first acceptance criterion for any *component implementation* deliverable will be a successful demonstration of capability in a *Technology Integration Experiment* ("TIE") test.

The next acceptance criterion will be assurance that the work-package ER Editor has access to sufficient documentation material to record all findings and recommendations in the ER and associated CRs.

Another criterion will be evidence that the implementer has *directly authored* full ER content for any relevant Abstract Test Suite (ER Template Annex A), XML Schema Document (ER Template Annex B), or UML Model (ER Template Annex C).

A Participant assigned to deliver a *client component implementation* should, in addition, ***record TIE test execution*** (as evidence of delivery) and create ***demo assets***, brief (several-minute) video recordings of client operation reflecting the newly enabled capabilities. These assets will become candidates for incorporation into executive-level briefings to be shown at the *Demonstration Event* near the end of the Testbed.

A Participant assigned to deliver a *service component implementation* should, in addition, provide assurance that the service endpoint will remain available to OGC and Sponsors for a period of no less than one year after Testbed execution. OGC might be willing to entertain exceptions to this requirement on a case-by-case basis.

A Participant assigned to deliver a *service component implementation* should, in addition, identify of one or more particular datasets that would be suitable for the proposed service. Details are provided under Data Requirements for Proposing Service Component Implementation Deliverables.

Some requirements might request delivery of a packaged prototype component (e.g., using a virtual machine or container). Consult the specific language in the Appendix B Technical Architecture for details regarding these requirements.

A.8.1. Data Requirements for Proposing *Service Component Implementation* Deliverables

A Bidder proposing to deliver a prototype *service component implementation* should provide ***detailed information*** regarding what data would be contributed (or what data should be acquired from another identified source) to support the proposed service. Some Testbed Sponsors might provide data, but it might also be necessary to complement these with additional datasets.

As an illustration, in the Testbed-12 initiative, the San Francisco Bay served as the primary area of interest (AOI) for service component implementation data. Full test dataset details are described in the document Testbed-12 Test Dataset Implementation with Documentation (16-136) [http://docs.opengeospatial.org/guides/16-136.html]. One of the primary sources for Testbed-12 data was the Homeland Infrastructure Foundation-Level Data (HIFLD) [https://hifld-geoplatform.opendata.arcgis.com/].

A similar pattern could be followed for a Testbed 14 *service component implementation* by proposing one or more HIFLD datasets for an assumed AOI covering the area of south Florida impacted by Hurricane Irma in September, 2017. Specific HIFLD data matching these criteria can be found at the HIFLD Hurricane Irma Response [https://respond-irma-geoplatform.opendata.arcgis.com/] site.

Alternatively, a Bidder could identify a non-HIFLD source (and even an alternative AOI) if another option would be more suitable to meet the technical requirement stated in the Appendix B Technical Architecture.

For example, one of the Sponsors identified the NGA Data Depot [https://www.geoint.community/data-depot] from the GEOINT Solutions Marketplace [https://www.geoint.community/] as a potential alternate source of appropriate data.

As another example, Testbed-12 [http://docs.opengeospatial.org/guides/16-136.html] reused OpenStreetMap feature data (which had actually originated in Testbed 11).

A.9. Project Monitoring and Control

Testbed execution monitoring and control will follow an encapsulation principle. As long as a Participant continues [1] supporting activities on which other Testbed stakeholders are relying (e.g., telecons, TIEs, ER inputs) and [2] making timely delivery of their own deliverables, the question of what's "going on" inside the Participant organization will likely not be raised.

Otherwise, any issues (or risks) that block (or threaten to block) timely delivery will enter a *remediation* status requiring the following responses:

- Development, implementation, and reporting of a credible mitigation or recovery response to get back on track,
- Recording in the Monthly Participant Technical Status Report a "Red Flag" Action Status indicating that the deliverable is in remediation (plus either "Red" or "Yellow" health in the Action Description),
- Notification of the relevant Thread Architect(s), who will review the recovery plan and report status to the Initiative Director and Sponsors.

The following definitions will be used to communicate and track deliverable status:

- For purposes of the Testbed, a *risk* can be defined as any uncertain future event that has a sufficiently high exposure to threaten Participants' ability to make on-time delivery of the promised deliverable scope.
- An *issue* is a risk that has actually occurred (no longer any uncertainty surrounding whether it will occur or not).
- A *mitigation* response is one that reduces a risk's exposure (e.g., finding an alternative source for meeting a data dependency).
- A *recovery* response is one that remedies the damage caused by the issue (e.g., getting the deliverable back on schedule or being excused from delivery of some particular requirement).
- A deliverable whose timely delivery is threatened should have an overall status of being *in remediation* and should reflect a *Red* or *Yellow* health.
- A *Red* deliverable health is appropriate when the response has not yet been created and agreed, or it has been created and agreed but is not believed to be sufficient to make timely delivery.
- A *Yellow* deliverable health is appropriate when the response has been created and agreed, and is in the process of execution (but not yet fully completed).
- Once the response has been fully completed, the deliverable health should return to *Green* and the deliverable would no longer be considered in remediation.

In general, risks and issues that are contained within a particular thread will be monitored and managed by the assigned Thread Architect, who will report ongoing status to other stakeholders. Any risks or issues that cross threads should be brought to the attention of all relevant Thread Architects and the Initiative Director.

A.10. Tips for New Bidders

Bidders who are new to OGC Testbeds are encouraged to review the following tips, extracted and updated from the *Testbed-13 Clarifications* document.

- The roles generally played in any OCG Innovation Program initiative are defined in the OGC Innovation Program Policies and Procedures [http://www.opengeospatial.org/ogc/policies/ippp], from which the following definitions are derived and extended.
  - **Sponsors** are OGC member organizations that contribute financial resources to steer Testbed requirements toward rapid development and delivery of proven candidate specifications to the OGC Standards Program. These requirements take the form of the deliverables described herein. Sponsors representatives help serve as "customers" during Testbed execution, helping ensure that requirements are being addressed and broader OGC interests are being served.
  - **Bidders** are organizations who submit proposals in response to this CFP. A Bidder selected to participate will become a Participant through the execution of a Participation Agreement contract with OGC. Most Bidders are expected to propose a combination of cost-sharing request and in-kind contribution (though solely in-kind contributions are also welcomed).
  - **Participants** are OGC member organizations that generate empirical information through the definition of interfaces, implementation of prototype components, and documentation of all related findings and recommendations in Engineering Reports, Change Requests and other artifacts. They also make substantial in-kind contributions to an initiative. Participants assign business and technical representatives to represent their interests throughout Testbed execution.
  - **Observers** are individuals from OGC member organizations that have agreed to OGC intellectual property requirements in exchange for the privilege to access Testbed communications and intermediate work products. They may contribute recommendations and comments, but the IP Team has the authority to table any of these contributions if there’s a risk of interfering with any primary Testbed activities.
  - The **Innovation Program Team** (IP Team) is the management team that will oversee and coordinate the initiative. This team is comprised of OGC staff, representatives from member organizations, and OGC consultants. The IP Team communicates with Participants and other stakeholders during Testbed execution,

provides Testbed scope and schedule control, and assists stakeholders in understanding OGC policies and procedures.

    - The term **Stakeholders** is a generic label that encompasses all Testbed actors, including representatives of Sponsors, Participants, and Observers, as well as the IP Team. Testbed-wide email broadcasts will often be addressed to "Stakeholders".
    - **Suppliers** are organizations (not necessarily OGC members) who have offered to supply specialized resources such as capital or cloud credits. OGCs role is to assist in identifying an initial alignment of interests and performing introductions of potential consumers to these suppliers. Subsequent discussions would then take place directly between the parties.

- Non-OGC member organizations must become members in order to be selected as Participants.
- Any individual wishing to gain access to Testbed intermediate work products in the restricted area of the Portal (or attend the private Testbed working meetings / telecons) must be a member-approved user of the OGC Portal system.
    - The reason for this restriction is that members are bound by the OGC bylaws, which offer protections to other members. Affording the same privileges to non-members could have a chilling effect on what are intended to be free and innovative Testbed discussions.
- Individuals from any OGC member organization that does not become a Testbed Sponsor or Participant may still (as a benefit of membership) quietly observe all Testbed activities by registering as a Testbed *Observer*.
- Prior Testbed participation is not a direct bid evaluation criterion.
    - However, prior participation could accelerate and deepen a Bidder’s understanding of the information presented in the CFP.
- All else being equal, preference will be given to proposals that include a larger proportion of in-kind contribution.
- All else being equal, preference will be given to proposed components that are certified OGC compliant.
- All else being equal, a proposal addressing all of a deliverable’s requirements will be favored over one addressing only a subset.
    - Each Bidder is at liberty to control its own proposal, of course. But if it does choose to propose only a subset for any particular deliverable, it might help if the Bidder prominently and unambiguously states precisely what subset of the deliverable requirements are being proposed.
- Sponsors will be given an opportunity to review selection results and offer advice, but ultimately the Participation Agreement (PA) contracts will be formed bilaterally between OGC and each Participant organization. No multilateral contracts will be

formed.

  - Beyond this, there are no restrictions regarding how a Participant chooses to accomplish its deliverable obligations so long as the Participant's obligations are met in a timely manner (e.g., with or without contributions from third-party subcontractors).

- In general, only one organization will be selected to receive cost-share funding per deliverable, and that organization will become the *Assigned Participant* upon which other Participants will rely for delivery.

  - Optional in-kind contributions may be made provided that they don't disrupt delivery of the required, reliable contributions from *Assigned Participants*.

- A Bidder may propose against any or all Testbed threads.

  - Participants in past Testbeds have often been assigned to make only a single deliverable. At the other extreme, there is theoretically no upper limit on the number of deliverables a single organization could be assigned to make.

- The period of performance for PAs is expected to run through December.

- In general, the PAs will not require delivery any component source code to OGC.

  - What is delivered instead is the behavior of the component in the TIEs, and the corresponding documentation of findings, recommendations, and technical artifacts in the related ER(s).

  - In some instances, a Sponsor might expressly require a component to be developed under open-source licensing, in which case the source code would become publicly accessible outside the Testbed as a by-product of implementation.

- Results of other recent OGC initiatives can be found in the OGC Public Engineering Report Repository [http://docs.opengeospatial.org/per/].

- A Bidders Q&A Webinar will likely be conducted in January. The webinar will be open to the public, but prior registration will be required.

< end of appendix >

Appendix B: Technical Architecture

B.1. Introduction

This Annex B provides background information on the OGC baseline, describes the Testbed-14 architecture and thread-based organization, and identifies all requirements and corresponding work items. For general information on Testbed-14, including deadlines, funding requirements and opportunities, please be referred to the Testbed-14 CFP Main Body.

Each thread aggregates a number of tasks. Each task specifies a number of requirements that need to be fulfilled by work items. The work items are funded by different sponsors.

An overview of all threads and assigned tasks is provided further below.

B.2. Testbed Baseline

B.2.1. Types of Deliverables

OGC Testbed-14 threads require several types of deliverables. Deliverable indications "funded" or "unfunded" are provided in the Testbed-14 CFP Main Body. Please make sure you understand your tasks and provision requirements according to Annex A.

Documents

Engineering Reports (**ER**) and **Change Requests** (**CR**) will be prepared in accordance with OGC published templates. Engineering Reports will be delivered by posting on the OGC Portal Pending Documents list when complete and the document has achieved a satisfactory level of consensus among interested participants, contributors and editors. Engineering Reports are the formal mechanism used to deliver results of the Innovation Program to sponsors and to the OGC Standards Program [http://www.opengeospatial.org/ogc/programs] and OGC Standard Working [http://www.opengeospatial.org/projects/groups/swg] Group or Domain Working Groups [http://www.opengeospatial.org/projects/groups/wg] for consideration.

IMPORTANT	It is emphasized that participants delivering Engineering Reports must also deliver Change Requests that arise from the documented work.

Implementations

Services, **Clients**, **Datasets** and **Tools** will be provided by methods suitable to its type and stated requirements. For example, services and components (e.g. a WFS instance) are delivered by deployment of the service or component for use in the Testbed via an accessible URL. A Client software application or component may be used during the Testbed to exercise services and components to test and demonstrate interoperability; however, it is most often not delivered as a license for follow-on usage. Implementations of services, clients and data instances will be developed and deployed in all threads for integration and interoperability testing in support of the agreed-up thread scenario(s) and technical architecture. The services, clients, and tools may be invoked for cross-thread scenarios in demonstration events.

B.2.2. OGC Reference Model

The OGC Reference Model (ORM) [http://www.opengeospatial.org/standards/orm] version 2.1, provides an architecture framework for the ongoing work of the OGC. Further, the ORM provides a framework for the OGC Standards Baseline. The OGC Standards Baseline consists of the member-approved Implementation/Abstract Specifications as well as for a number of candidate specifications that are currently in progress.

The structure of the ORM is based on the Reference Model for Open Distributed Processing (RM-ODP), also identified as ISO 10746. This is a multi-dimensional approach well suited to describing complex information systems.

The ORM is a living document that is revised on a regular basis to continually and accurately reflect the ongoing work of the Consortium. We encourage respondents to this RFQ to learn and understand the concepts that are presented in the ORM.

This Annex B refers to the RM-ODP approach and will provide information on some of the viewpoints, in particular the Enterprise Viewpoint, which is used here to provide the general characterization of work items in the context of the OGC Standards portfolio and standardization process, i.e. the enterprise perspective from an OGC insider.

The Information Viewpoint considers the information models and encodings that will make up the content of the services and exchanges to be extended or developed to support this Testbed. Here, we mainly refer to the OGC Standards Baseline, see section Standards Baseline.

The Computational Viewpoint is concerned with the functional decomposition of the system into a set of objects that interact at interfaces – enabling system distribution. It captures component and interface details without regard to distribution and describes an interaction framework including application objects, service support objects and infrastructure objects. The development of the computational viewpoint models is one of the first tasks of the Testbed, usually addressed at the kick-off meeting.

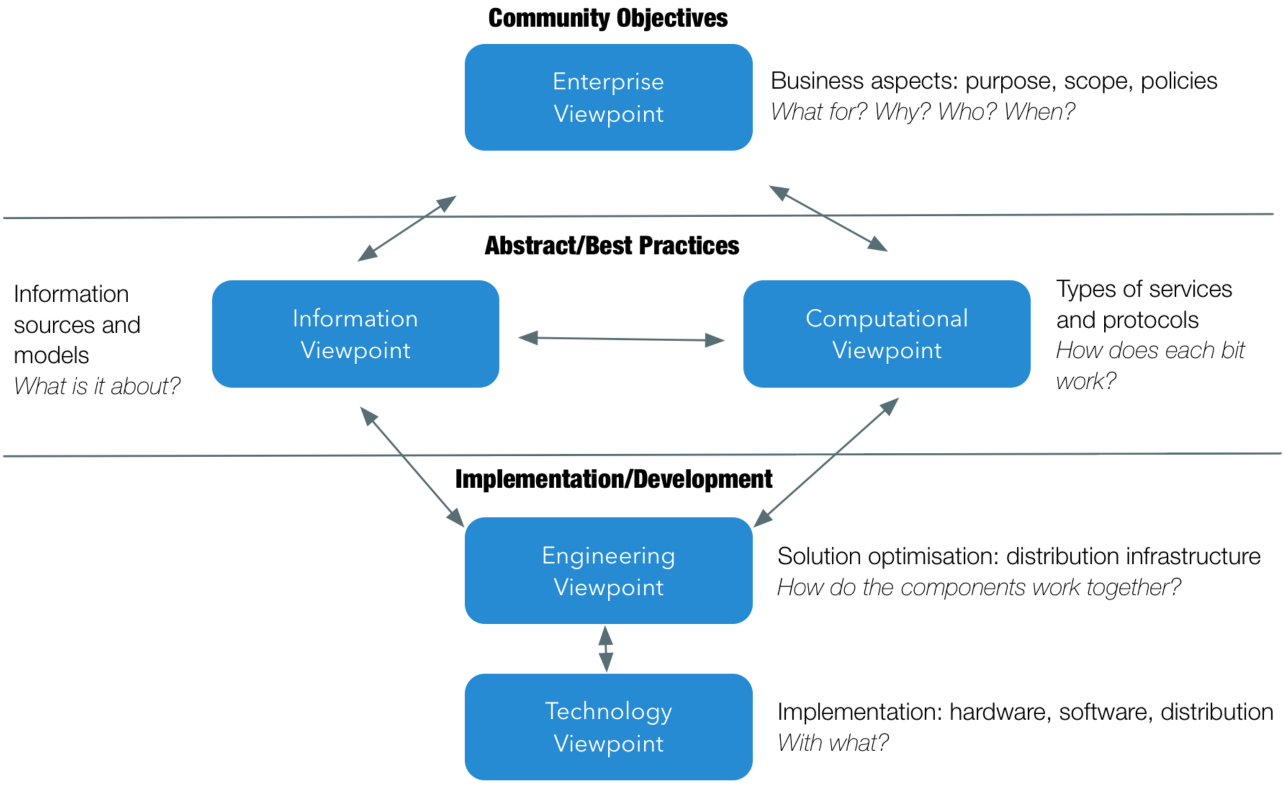

Community Objectives
Enterprise Viewpoint
Business aspects: purpose, scope, policies
What for? Why? Who? When?
Abstract/Best Practices
Information sources and models
What is it about?
Information Viewpoint
Computational Viewpoint
Types of services and protocols
How does each bit work?
Implementation/Development
Engineering Viewpoint
Solution optimisation: distribution infrastructure
How do the components work together?
Technology Viewpoint
Implementation: hardware, software, distribution
With what?

Figure 2. Reference Model for Open Distributed Computing

The Engineering Viewpoint is concerned with the infrastructure required to support system distribution. It focuses on the mechanisms and functions required to:

a. support distributed interaction between objects in the system, and

b. hides the complexities of those interactions.

It exposes the distributed nature of the system, describing the infrastructure, mechanisms and functions for object distribution, distribution transparency and constraints, bindings and interactions. The engineering viewpoint will be developed during the Testbed, usually in the form of TIEs, where Testbed participants define the communication infrastructure and assign elements from the computational viewpoint to physical machines used for demonstrating the Testbed results.

B.2.3. OGC Standards Baseline

The OCG Standards Baseline [http://www.opengeospatial.org/standards] is the complete set of member approved Abstract Specifications [http://www.opengeospatial.org/docs/as], Standards [http://www.opengeospatial.org/docs/is] including Profiles [http://www.opengeospatial.org/docs/profile] and Extensions [http://www.opengeospatial.org/docs/is], and Community Standards [http://www.opengeospatial.org/standards/community].

OGC standards are technical documents that detail interfaces or encodings. Software developers use these documents to build open interfaces and encodings into their products and services. These standards are the main "products" of the Open Geospatial Consortium and have been developed by the membership to address specific interoperability challenges. Ideally, when OGC standards are implemented in products or online services by two different software engineers working independently, the resulting components plug and play, that is, they work together without further debugging. OGC standards and supporting documents are available to the public at no cost. OGC Web Services (OWS) are OGC standards created for use in World Wide Web applications. For this Testbed, it is emphasized that all OGC members have access to the latest versions of all standards. If not otherwise agreed with the Testbed architects, these shall be used in conjunction with - in particular - Engineering Reports resulting from previous Testbeds.

Any Schemas (xsd, xslt, etc.) that support an approved OGC standard can be found in the official OGC Schema Repository [http://schemas.opengis.net].

The OGC Testing Facility [http://cite.opengeospatial.org/teamengine/] Web page provides online executable tests for some OGC standards. The facility helps organizations to better implement service interfaces, encodings and clients that adhere to OGC standards.

B.2.4. OGC Best Practices and Discussion Papers

OGC also maintains other documents relevant to Innovation Program initiatives, including Engineering Reports [http://www.opengeospatial.org/docs/er], Best Practice Documents [http://www.opengeospatial.org/docs/bp], Discussion Papers [http://www.opengeospatial.org/docs/discussion-papers], and White Papers [http://www.opengeospatial.org/docs/whitepapers].

B.2.5. Data

All participants are encouraged to provide data that can used to implement the various scenarios that will be developed during the Testbed. A number of Testbed sponsors will provide data, but it might be necessary to complement these with additional data sets. Please provide detailed information if you plan to contribute data to this Testbed.

B.2.6. Services in the Cloud

Participants are encouraged to provide data or services hosted in the cloud. There is an overarching requirement to provide cloud-hosting capabilities to allow thread participants to move services and/or data to the cloud.

B.2.7. Dependencies between Threads

Dependencies between threads have been minimized. In some rare cases one thread uses components from another thread. In this case, the thread providing components used by another thread need to provide the relevant components in time. No cross-thread dependencies exist where component developers from disjunct threads need to agree on a common interface. The interface is always defined within a single thread exclusively.

B.3. Testbed Threads

Testbed-14 is organized in a number of threads. Each thread combines a number of tasks that are further defined in the following chapters. The threads integrate both an architectural and a thematic view, which allows to keep related work items closely together and to remove dependencies across threads.

- **Thread 1**: Modeling, Portrayal, and Quality of Service (**MoPoQ**)
    - Information Registries & Semantic Enablement
    - Application Schema Modeling and Conversion
    - Portrayal
    - MapML
    - Quality of Service & Experience (QoSE)
    - Machine Learning, Deep Learning & Artificial Intelligence
    - LiDAR Point Cloud Data Handling
- **Thread 2**: Earth Observation & Clouds (**EOC**)
    - Swath Data and the Climate Forecast Convention
    - Exploitation Platform
- **Thread 3**: Next Generation Services (**NextGen**)
    - Next Generation OGC Web Services, Federated Clouds, Security & Workflows
    - Complex Feature Handling
    - CityGML and Augmented Reality
- **Thread 4**: Compliance (**CITE**)
    - Compliance and Interoperability Testing

B.4. Tasks

Each of the following sections provides a detailed description of a particular task.

NOTE Please note that a few links of the links below to Testbed-13 Engineering Reports will result in “404 Not Found” results as some these reports are still being prepared for publication.

B.5. Machine Learning, Deep Learning & Artificial Intelligence

One of the most interesting recent developments in Machine Learning (ML) has been in Deep Learning, a specialization of Artificial Neural Networks.

Deep Learning (DL) can be defined as a neural network approach that involves a large number of parameters and layers in one of four fundamental network architectures: unsupervised pre-trained networks, convolutional neural networks, recurrent neural networks and recursive neural networks. Automatic feature extraction is one of the areas where DL has shown great promise.

OGC Web services play a pivotal role in many geospatial work flows. Facading any type of geospatial data store and processing system, they can be used to support *Machine Learning, Deep Learning & Artificial Intelligence* systems. The goal of this task is to develop a holistic understanding and to derive best practices for integrating Machine Learning, Deep Learning, and Artificial Intelligence tools and principles into OGC Web service contexts. Even though an image analysis workflow is used here to describe interactions between components and interface requirements, it is emphasized that this task is not constraint to image analysis but should work with all types of data including vector information!

Essentially, the following two questions shall be answered:

1. How to best support Machine Learning and Artificial Intelligence using OGC Web Services?
2. How to best publish inputs to and outputs from Machine Learning and Artificial Intelligence using OGC Web Services?

To some extent, these questions can build on previous work, which allows to integrate machine learning tools behind OGC Web service interfaces by revising OGC Web Image Classification Service Discussion Paper (OGC 05-017) to support image classification using Machine Learning and Artificial Intelligence.

Previous Work OGC previously prototyped a Web Image Classification Service (WICS) interface (OGC 05-017 [https://portal.opengeospatial.org/files/?artifact_id=8981&version=2]) that defines these operations:

- GetClassification
- TrainClassifier
- DescribeClassifier Operation

Using these operations, the WICS provides three types of classifications:

a. Unsupervised classification: getCapabilities, describeClassifier, and getClassification.

b. Supervised classification using server archived trained classifiers: getCapabilities, describeClassifier, and getClassification.

c. Supervised classification using classifiers trained based on client supplied training data.

The previous work was not done with a Deep Learning Classifier and with focus on gridded imagery data, but that may not affect the Interface operations that are independent of algorithm type. The WICS operations directly match the functions shown in Figure 3 of the CFP.

Implementation of Components and Services

On the implementation side, this task shall support an image analysis and processing scenario. Imagine satellite imagery is stored in an image archive that is available to human image analysts and Machine Learning tools. An image analyst identifies a set of feature types on the images and makes this data available to the data store. These annotated data are available to both consumers and Machine Learning tools that use that data for training purposes. Depending on the type of available imagery data, Testbed-14 may differentiate two scenarios. In the first scenario, the image analyst uses prior- and post event data to identify for example areas destroyed by a natural disaster such as a hurricane. In the second scenario, the image analyst identifies feature types that do not take any change detection into account. The identified features could be compliant with Geography Mark-up Language (GML) application schemas such as the e.g. Multinational Geospatial Co-production Program (MGCP) Application Schema [https://portal.dgiwg.org/files/?artifact_id=5442&format=doc] and the NSG Application Schema (NAS).

The Image Analyst queries image data (or is informed about new data by push notifications), identifies specific feature types, and provides the results of the analysis back to the system. These data include the identified feature types, the associated snippets (set of pixels in an image), as well as additional metadata as defined in Testbed-14.

A Machine Learning tool uses the Image Analysts' results as input training data to improve its learning algorithm. Rather than concentrating on the quality of the feature identification results, Testbed-14 shall explore the best workflows, service interaction patterns, an application schemata to express both analysis results and associated metadata such as trust levels and confidence, associated body of evidence, associated imagery, and optimized learning capabilities. The general workflow is illustrated in the figure below. The "Learning" box in the lower right corner shall illustrate the iterative nature of the workflow that uses results from the "Machine Learning and Artificial Intelligence" box to further train the model (e.g. by using external processes or data).

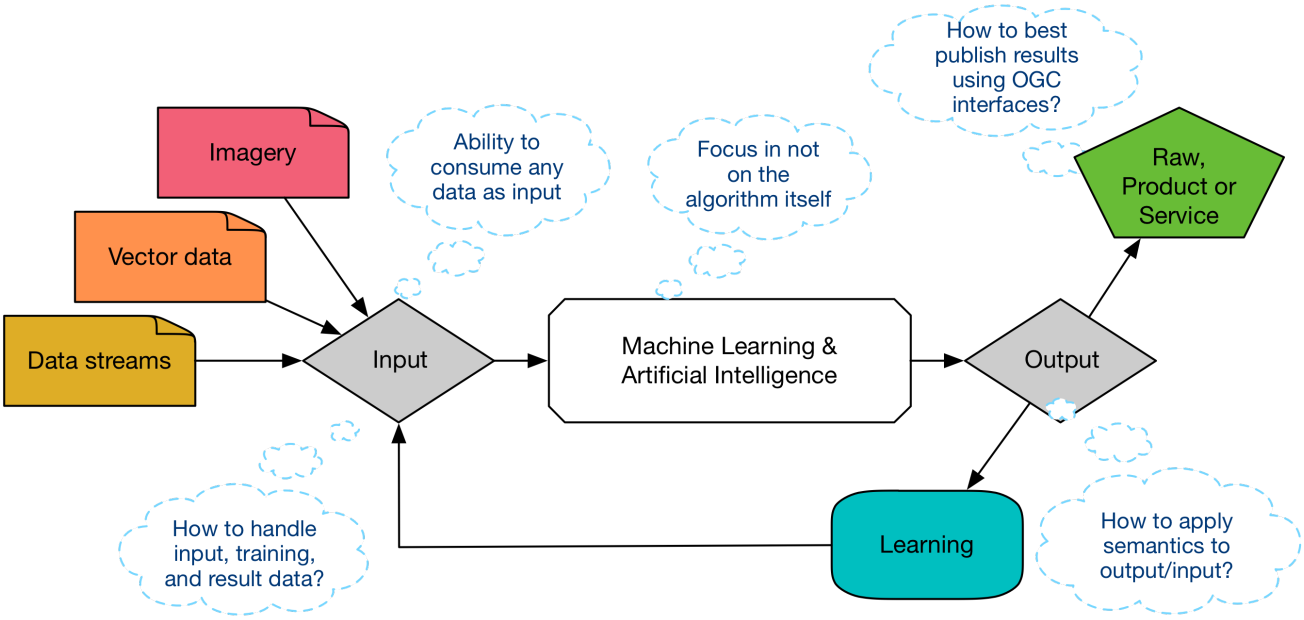

How to best publish results using OGC interfaces?
Imagery
Ability to consume any data as input
Focus in not on the algorithm itself
Raw, Product or Service
Vector data
Data streams
Input
Machine Learning & Artificial Intelligence
Output
How to handle input, training, and result data?
Learning
How to apply semantics to output/input?

Figure 3. Overview of the Machine Learning and Artificial Intelligence task

All data shall be made available at OGC data access services (e.g. WFS, WCS, SOS) as identified in responses to this Call for Participation. Additional catalog services might become part of the overall scenarios if necessary. This Call for Participation does not specify details on the involved service technology (i.e. participants can identify services types), but provides suggestions for possible implementations in the figure below.

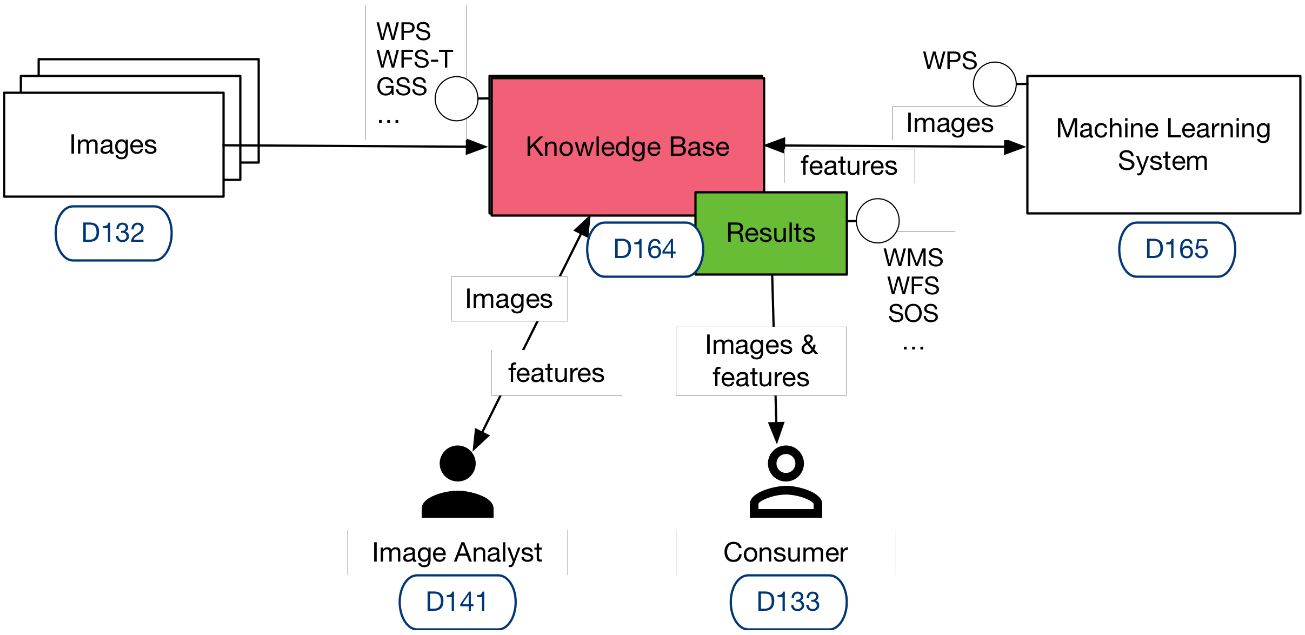


Figure 4. Possible implementation using transactional service types

Concept Study and Engineering Report

The Testbed-14 Machine Learning Engineering Report shall describe all experiences and lessons learned of the *Machine Learning, Deep Learning & Artificial Intelligence* implementations. In addition, it shall includes the results of a concept study that provides a truly holistic approach on the integration of machine learning and OGC Web services. This holistic approach may require some experimentation and shall include the following items:

- Receive and process metadata about the applied processes, quality parameters, links to original images and/or snippets, etc. Gal’s thesis on Uncertainty in Deep Learning [http://mlg.eng.cam.ac.uk/yarin/thesis/thesis.pdf] provides solid background on uncertainty and quality aspects.
- Describe training data handling
- Integrate machine learning tools behind OGC Web service interfaces
- Provide performance information about the AI tool
- Describe how to best support Machine Learning and Artificial Intelligence using OGC Web Services
- Describe How to best publish outputs from Machine Learning and Artificial Intelligence using OGC Web Services
- Reflect on Principles for Algorithmic Transparency and Accountability [https://www.acm.org/binaries/content/assets/public-policy/2017_usacm_statement_algorithms.pdf], as there is growing evidence that some algorithms and analytics can be opaque, making it impossible to determine when their outputs may be biased or erroneous.

- Make recommendations for new OGC standards or extending OGC standards to support AI/ML geospatial algorithms, e.g., Simple Features for Big Data [https://portal.opengeospatial.org/files/?artifact_id=75244] as proposed at OGC Big Data DWG, September 2017.
- Ideally make recommendations about use of tiles or grid structures for the analysis including DGGS that help integrating data from various sources.

In addition, the concept study shall analyze a geospatial feature extraction workflow that is slightly more complex than what can be realized in the implementation scenario described above. For simplicity, the process is outlined here in the form of **Input-Process-Output stages**. Issues that affect the **Input stage** include:

- Conversion of images from their original format into a format suitable for feature extraction and classification. This is an area where WPS could play a role.
- Filtering of images to remove those of very low quality (e.g. those mostly covered by clouds) that might skew the ML model. This is an area where a WCPS-enabled WPS could play a role.

Some of the issues that affect the **Process stage** of a geospatial feature extraction workflow include:

- Organizations that are innovating on the algorithms side do not always have access to the vast amounts of data required to develop a fully-fledged production-level ML model. One option is to apply the "move-the-algorithm-to-the-data" approach that was demonstrated in Testbed-12 and Testbed-13. This is an area where WPS could play a role. A possible outcome could be a WPS Profile for developing DL models for geospatial feature extraction. Of course, the "move-the-data-to-the-algorithm" approach could also be applied as well. The benefit would be faster innovation by enabling organizations to implement and try out their ML algorithms on infrastructure that can handle large realistic datasets. Note the relevance to the NSG Data Analytic Architecture Service [https://www.youtube.com/watch?v=4SWiDi4mMKg] (NDAAS). The benefit to image providers would be increased use of their imagery products.
- A well-trained artificial neural network has a parameter vector representing the weights and biases of the model being applied. These weights and biases are re-adjusted to increase the significance of some bits of information and minimize the significance of others. This enables the model to learn which features are tied to which outcomes. In practice however, other sources of information also play a role in the classification of features. The benefit of an integrative approach would be greater efficiency and effectiveness of knowledge sharing. The benefit to imagery and feature data providers would be increased use of their imagery exploitation services.

Some of the issues that affect the **Output stage** of a geospatial feature extraction workflow include:

- Provision of metadata in a format that adequately describes the provenance of the model and its configuration parameters. Note that configuration of neural networks is a key part of developing DL models. A possible outcome could be an ISO 19115 profile or extension (perhaps building on the Data Quality work of Testbed-12).
- Export of outputs in a form that allows the outputs to be re-ingested as inputs for further learning, in a closed feedback loop as described in the e-mail sent yesterday. A possible outcome could be profiles for WPS, WFS and OWS Context. Note that this concept is consistent with the NGA 2020 analysis technology plan [https://www.nga.mil/MediaRoom/PressReleases/Documents/NGA_Analysis_Tech_Plan.pdf] (page 6) which states that “the role of human analysts will evolve over the next six years as computers assume an active (vice today’s mainly passive) role in GEOINT exploitation: human analysts will spend more time interrogating and validating the machine’s work, exploiting the gaps and applying broader contextual awareness to the algorithmic outputs”. The concept is also related to Human-Agent Collectives [https://pdfs.semanticscholar.org/39d6/191ecccf29be360a9dc7900428d702c31faf.pdf?_ga=2.179511524.1327766577.1511169830-1796992056.1511169830]. Another important aspect to consider is the use of controlled vocabularies to allow consistent usage of results from Machine Learning tools. Options here include for example the Multinational Geospatial Co-production Program (MGCP) or the NSG Core Vocabulary [https://nsgreg.nga.mil/doc/view?i=4428&month=11&day=30&year=2017].

B.5.1. Work Items Overview

The following figure illustrates the *Machine Learning, Deep Learning & Artificial Intelligence* work items.

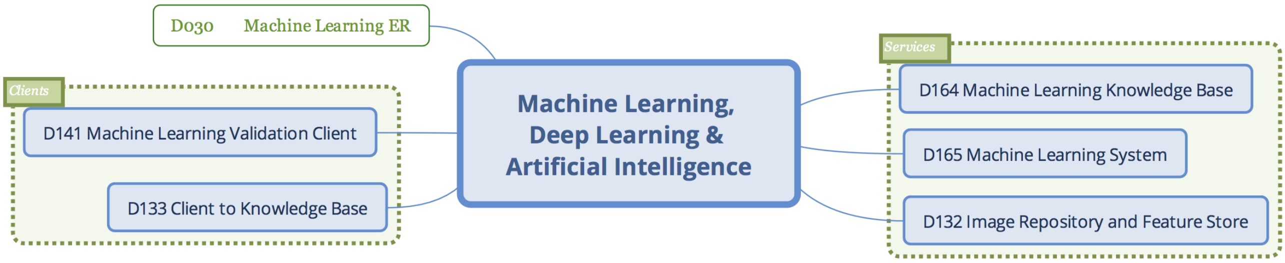

D030 Machine Learning ER
Clients
D141 Machine Learning Validation Client
D133 Client to Knowledge Base
Machine Learning, Deep Learning & Artificial Intelligence
Services
D164 Machine Learning Knowledge Base
D165 Machine Learning System
D132 Image Repository and Feature Store

Figure 5. Machine Learning, Deep Learning & Artificial Intelligence deliverables

B.5.2. Deliverables

To re-iterate, the aim of this task is to develop a holistic understanding and derive best practice for integrating Machine Learning, Deep Learning, Artificial Intelligence tools and practices into OGC Web Services. This is to be achieved through focusing on two key questions of How to best support Machine Learning and Artificial Intelligence and How best to publish outputs from Machine Learning and Artificial Intelligence both using OGC Web Services. The suite of deliverables should interact with each other to demonstrate this.

The following list identifies all deliverables that are part of this task. Additional requirements may be stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D030 Machine Learning Engineering Report** - that covers are results from the implementation scenarios and the detailed concept study described above. Additionally, it should also examine the implementations of Next Generation OGC Web Services, and how these might need to be addressed.
- **D132 Image Repository and Feature Store** - that makes all imagery and additional feature data available to Knowledege Base and AI System. Should support push of images into knowledge base.
- **D133 Client to Knowledge Base** - Visualization and exploration client that allows interacting with knowledge base and image repository. Allows to explore all imagery, processing results, quality information, machine learning validation results.
- **D141 Machine Learning Validation Client** - Client to the Machine Learning System and Knowledge base that will rate the correctness of the machine learning system output. Makes that data available at the knowledge base and as training input to the machine learning system.
- **D164 Machine Learning Knowledge Base** - Knowledge Base that contains all imagery, imagery snippets, analysis results etc. Exposes data access and transactions interfaces.
- **D165 Machine Learning System** - Actual machine learning system, exposes a WPS or similar interface. System will classify output using specified NAS/profile to suit use case.

B.6. Information Registries & Semantic Enablement

In the last decade, a technological framework known as SWIM [https://www.faa.gov/nextgen/programs/swim/] (System Wide Information Management) has been developed in the FAA [https://www.faa.gov/nextgen/programs/swim/], EUROCONTROL [https://www.eurocontrol.int/swim], and more recently the Asia-Pacific Region as a viable model for Air Traffic Management (ATM) applications. "SWIM is a Federal Aviation Administration (FAA) advanced technology program designed to facilitate greater sharing of Air Traffic Management (ATM) system information, such as airport operational status, weather information, flight data, status of special use airspace, and National Airspace System (NAS) restrictions. SWIM will support current and future NAS programs by providing a flexible and secure information management architecture for sharing NAS information. SWIM will use commercial off-the-shelf hardware and software to support a Service Oriented Architecture (SOA) that will facilitate the addition of new systems and data exchanges and increase common situational awareness." (wikipedia) [https://en.wikipedia.org/wiki/System_Wide_Information_Management]

SWIM addresses the communications and interoperability requirements of highly-distributed, loosely-coupled, and platform-independent components by consistently applying the principals of Service-Oriented Architecture (SOA). The major goal of SWIM is providing information to members of the civil aviation community. This information includes (but is not limited to) aeronautical, meteorological, and flight information, and it is rendered in a formal language such as XML.

The work in Testbed-14 is primarily motivated by the following aspects:

1. Lack of information discovery capabilities. Although SWIM supports service discovery by providing a searchable catalog (e.g., a service registry) of all SWIM-enabled services, information provided by the services can only be discovered in indirect and often inefficient ways. To establish whether a specific type of information is provided by SWIM, a user generally attempts to deduce what the "right" service is for the given type of information and then has to access and examine that service's meta-data in order to determine whether the service does in fact provide the needed information. In other words, there is no information discovery; the information can only be discovered by extension of a service discovery.
2. Deficiency of semantic representation of service meta-information. A description of information provided by a service usually is limited to documents that generally focuses on defining how the service input/output can be constructed and manipulated (e.g., XML schemas, WSDL files) and fail to capture enough semantics. Conversely, free-text human-consumable documents such as the FAA's Web Service Description Document (WSDD) support a sufficient amount of semantics but cannot be accessed and used by a software agent. Neither formal-language nor human-readable documents are suitable for automated information discovery and provide very limited support for semantic interoperability.

These aspects are subject of the following, simplified but realistic use case:

- Assume a registry about services already exist.
- Each service advertises its own features. These are presented as GML Application Schemas
- A client (user driven) interrogates the registry for specific information, not always using the semantics of the features registered in the registry. This interrogation happens several times, since users will be required to inspect various services until the user finds the information needed. For example, if searching for an "alternative destination airport" the user will use a service that provides some form of a flight plan.
- Once the right service is located, the user needs to study several service artifacts, for example, the Web Service Description Document (WSDD), the XML schema, and WSDL to determine whether the information about "alternative destination airport" is used and how it can be retrieved. During this process, the user may need to consult

subject matter experts to understand the data definitions contained in the XML schema.

Testbed-14 shall address the following requirements:

1. Testbed-14 shall provide recommendations for development of an "information registry", that is, a searchable catalog of information collectively provided by services in the context of a specific service inventory, such as FAA's SWIM. The results shall be captured in the *Information Registry Engineering Report* (D001).
2. Testbed-14 shall provide recommendations for making existing data models used in aviation industry (e.g., AIXM, WXXM, and FIXM) "semantically enabled". The data models shall be enabled to present their contents in formats suitable for adaptation by Semantic Web technologies, including considerations for role and applicability of ontologies and linked data approaches to complex information realms such as SWIM. The work shall provide a clear definition of next steps. The results shall be captured in the *Semantically Enabled Aviation Data Models Engineering Report* (D002).

B.6.1. Work Items Overview

The following figure illustrates the *Information Registries & Semantic Enablement* work items.

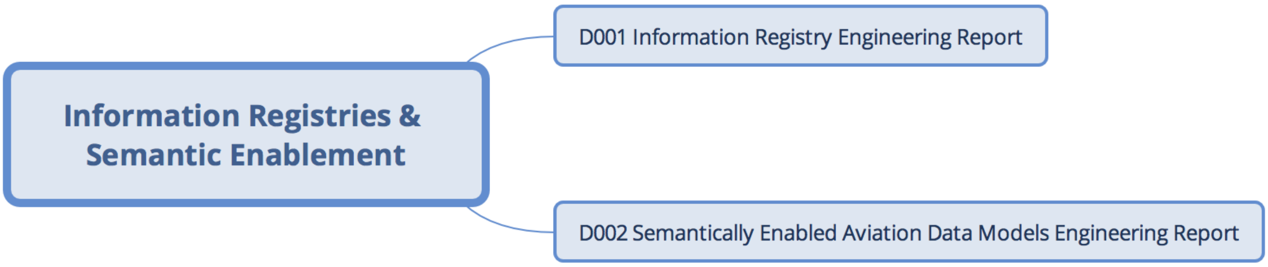

Information Registries & Semantic Enablement
D001 Information Registry Engineering Report
D002 Semantically Enabled Aviation Data Models Engineering Report

Figure 6. Information Registries & Semantic Enablement work items

B.6.2. Deliverables

The following list identifies all deliverables that are part of this task. Additional requirements are stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D001 Information Registry Engineering Report** - Engineering Report capturing all experiences and results of the information registry.
- **D002 Semantically Enabled Aviation Data Models Engineering Report** - Engineering Report capturing all experiences and results of the semantic enabling of aviation industry data models. The report shall cover at least AIXM, WXXM, and FIXM.

B.7. Next Generation OGC Web Services, Federated Clouds, Security & Workflows

This task combines a number of OGC technologies. It addresses both new capabilities at the service level as well as complex interaction patterns between services. In addition to the components described in this chapter, it makes use of WPS instances developed as part of the Exploitation Platform task.

The task consists of the four main topics WFS3.0, Federated Clouds, Security, and Workflows that make use of components provided by the Exploitation Platform task. The following figure provides an overview.

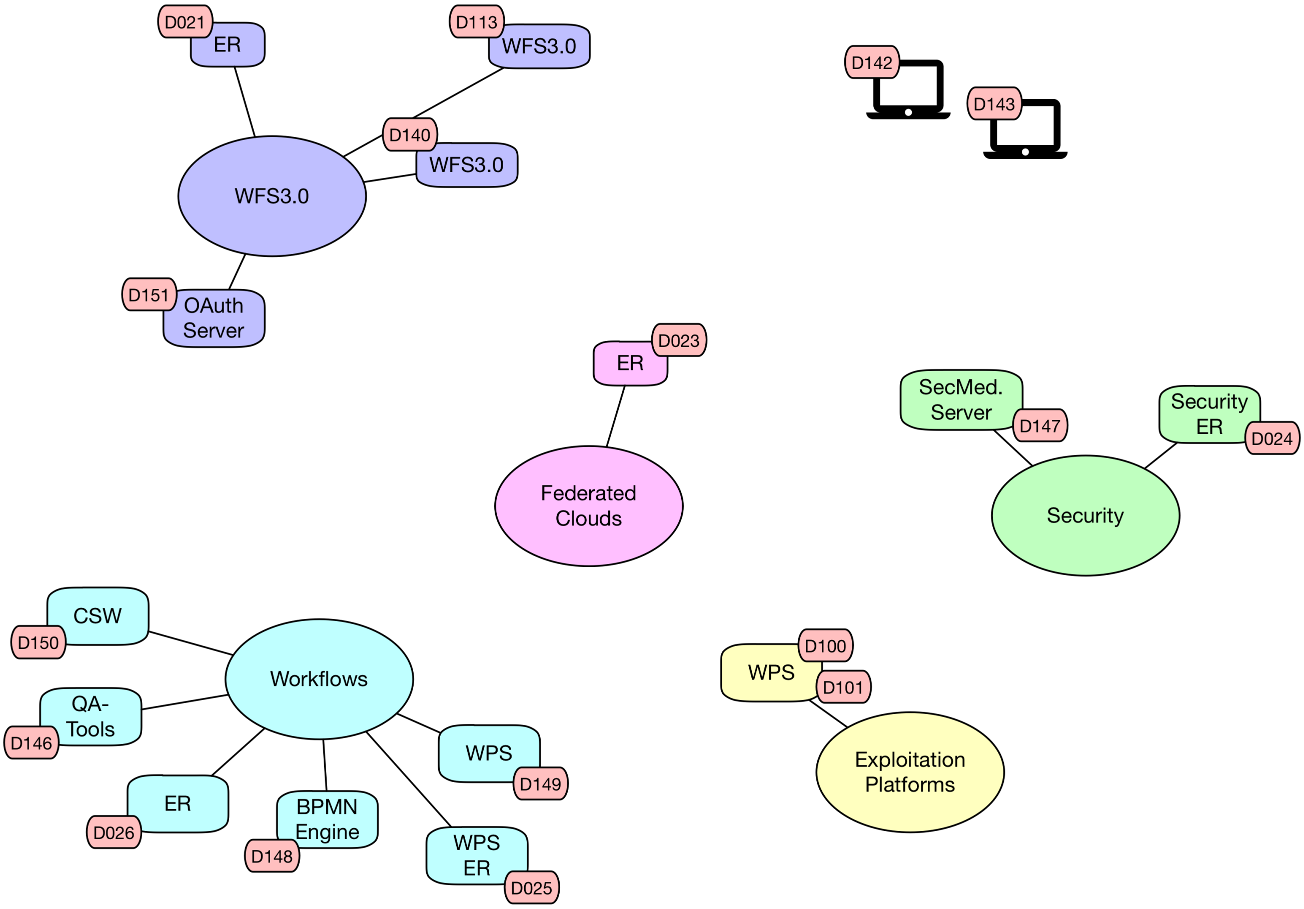


Figure 7. Work item overview with all deliverables. Purple, turquoise, and green: Topics with ERs and components, yellow: extract from Exploitation Platform task, red: deliverable identifiers.

- WFS3.0: Next Generation OGC RESTful Web service interfaces: Implementation of secured WFS3.0 service instances. Goal is to experiment with the new WFS3.0 specification and to add security mechanisms based on OAuth2.0.
- Federated Clouds: Integration of services that are hosted in different security environments, as it is often the case when services exist in different clouds. Using secured WFS3.0 instances from above and WPS instances from the Exploitation Platform work item, Federated Clouds shall explore mediation mechanisms to handle the different secured environments.

- Security: Goal is to better understand security mechanisms in modern Spatial Data Infrastructures. This part combines all security aspects in this task.
- Workflows: Creation and execution of secured workflows within the geospatial domain. Focus is on Business Process Model and Notation (BPMN)-powered workflows that handle even complex security settings, including new, previously unregistered processes added to transactional WPS.

The following diagram provides an overview of all work items of this task. Elements from this diagram are used repeatedly further below to emphasize the various foci of the different topics.

B.7.1. Next Generation OGC Web Services, Federated Clouds, Security & Workflows Topics

Next Generation OGC RESTful Web service interfaces

OGC Web Services were developed to implement a Service Oriented Architecture (SOA) architecture pattern. For this they work well. However, a new architecture pattern has seen wide adoption. Resource Oriented Architectures have become popular, particularly in browser-facing environments. This architecture pattern focuses on the exchange of resources. Services, where they exist at all, perform their job in the background.

Moving OGC services onto a resource oriented architecture requires more than just adding RESTful operations to the existing services. The whole concept of a service must be re-imagined. Participants in this task shall re-imagine OGC services in preparation to the production of a new generation of OGC Service standards.

Participants in this task shall develop a prototype OGC standard for OGC Resource-Oriented Services. This standard shall be based on the WFS 2.0 capabilities, but is free to implement those capabilities in the most appropriate manner. Participants shall use the REST Users Guide [http://docs.opengeospatial.org/guides/16-057r1.html] and the REST Architecture Engineering Report [http://docs.opengeospatial.org/per/16-035.html] produced by Testbed-12. If there is a reason to deviate from these documents (in particular the REST Users Guide [http://docs.opengeospatial.org/guides/16-057r1.html]), then the editor of the WFS3.0 ER (D021) in cooperation with the WFS3.0 implementers shall revise the Testbed-12 documents!

In addition, editors and implementers shall survey existing Resource-Oriented APIs such as those found at http://www.pingable.org/the-top-15-web-apis-for-your-site.

The prototype standard shall be based on the most recent WFS3.0 definition that is available on Github [https://github.com/opengeospatial/WFS_FES/], maintained by the OGC WFS/FES SWG [http://www.opengeospatial.org/projects/groups/wfs/fesswg]. It shall be validated through two or more independent service implementations. All service implementations shall support HTTP over TLS (HTTPS) and HTTP Basic Authentication. In addition, all service implementations shall participate in the discussions about mutual authentication and delegated coarse granular access control with OAuth2 to explore issues and propose approaches to address those issues. An OAuth2 authorization server is available for Testbed-14.

Testbed-14 doesn't require fine grained policy-based access control, so appreciates any work done in this regard, in particular on the integration of (Geo-)XACML with OAuth2.

The following figure illustrates the work items included in the *Next Generation OGC RESTful Web service interfaces* topic.

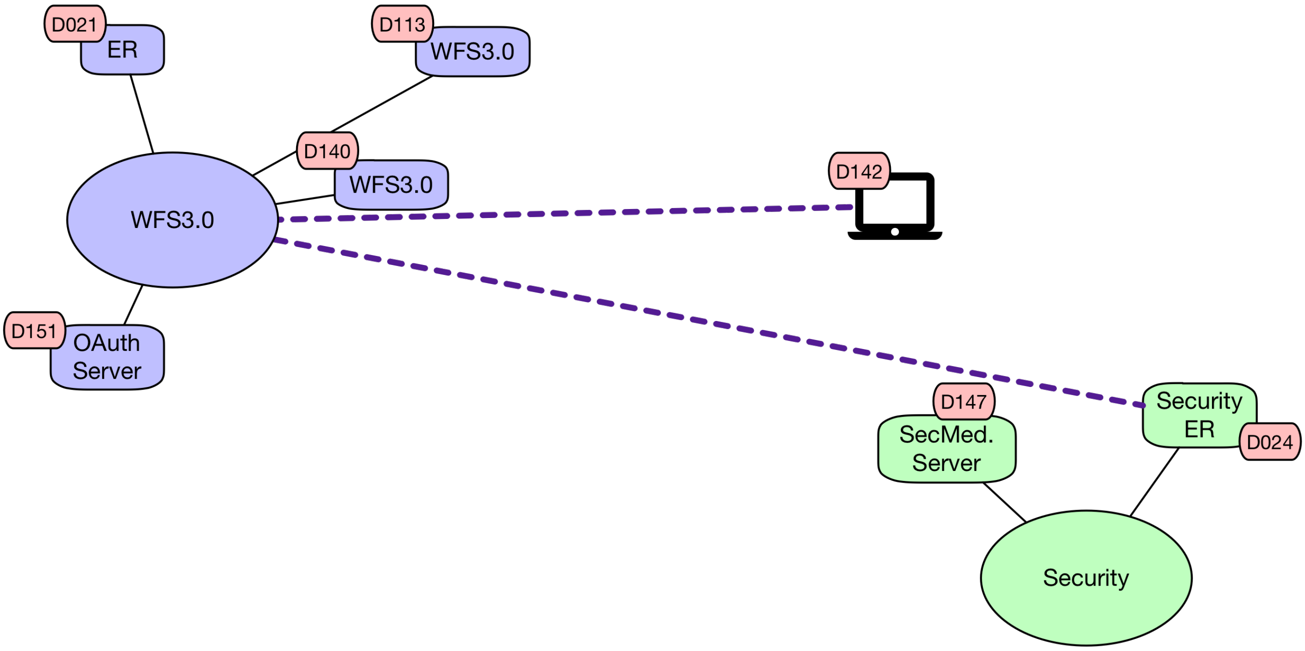


Figure 8. Work item overview of topic Next Generation OGC RESTful Web service interfaces

All WFS3.0 service instances shall be accompanied by a simplified client (a cURL [https://curl.haxx.se]-based client is sufficient). In addition, the service instances will be accessed by a dedicated client application which provides a GUI for user interactions and visualization capabilities (D142).

All WFS3.0 results need to be captured in the WFS3.0 Engineering Report (D021); appropriate Change Requests shall be submitted. This includes a summary of and references to the more detailed security discussion that is primarily captured in the Security Engineering Report (D024).

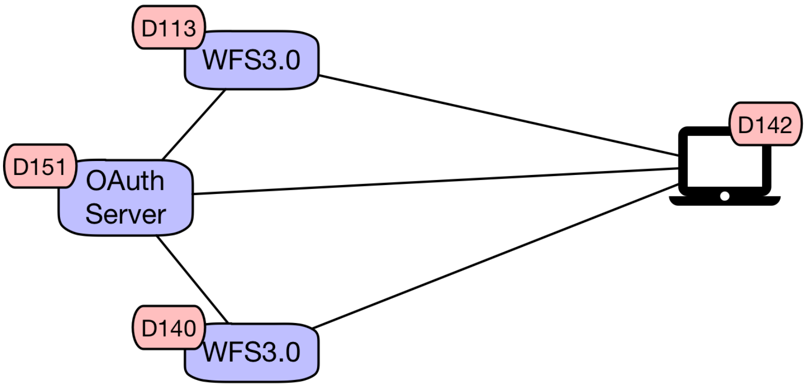


Figure 9. Client/server interactions of topic Next Generation OGC RESTful Web service interfaces

The different WFS3.0 prototypes shown in the figure above implement different features:

- At least one of these services shall include a façade, providing resource-oriented access to an existing WFS 2.0 service (D140).
- At least one other service one shall implement a number of Best Practices [https://www.w3.org/TR/sdw-bp/] as published by the joint OGC/W3C Spatial Data on the Web Working Group (D113). The service instance shall be developed as Free and Open Source Software (FOSS) and should be based on one of the leading WFS Open Source implementations (e.g. Geoserver). Whereas best practices 1 and 2 are mandatory, the service should support ideally all of the following best practices:
  - Best Practice 1 [https://www.w3.org/TR/sdw-bp/#globally-unique-ids]: Use globally unique persistent HTTP URIs for Spatial Things every resource has its own persistent URL
  - Best Practice 2 [https://www.w3.org/TR/sdw-bp/#indexable-by-search-engines]: Make your spatial data indexable by search engines. The implementation should take care that at a minimum each resource has a HTML + embedded schema.org landing page. The Testbed-14 Engineering Report shall provide information of the role of sitemaps for indexability and provide general advice on how to improve indexing of service offerings (and offered spatial data). Further on, the prototype implementation should support the following best practices:
  - Best Practice 3 [https://www.w3.org/TR/sdw-bp/#linking]: Link resources together to create the Web of data is dependent in BP 1 and BP 2
  - Best Practice 5 [https://www.w3.org/TR/sdw-bp/#describe-geometry]: Provide geometries on the Web in a usable way at least GML and GeoJSON
  - Best Practice 7 [https://www.w3.org/TR/sdw-bp/#bp-crs-choice]: Choose coordinate reference systems to suit your user’s applications at least WGS84
  - Best Practice 12 [https://www.w3.org/TR/sdw-bp/#convenience-apis]: Expose spatial data through 'convenience APIs'. Though WFS 3.0 is not necessarily a “convenience API”, the Testbed-14 implementation shall support:

- Use REST and document the API according to OpenAPI spec
- Use content negotiation as described here: https://www.w3.org/TR/dwbp/#Conneg
- Return data in chunks fit for use in Web applications and as useful sets of information.
- Simplify geometries

All OGC standards require a description of the conformance test approach and an Abstract Test Suite. The existing WFS conformance test approach is unlikely to be sufficient for the prototype services. Therefore, the Participants of this task shall support the Compliance task team and work in cooperation with the OGC CITE development team to document an appropriate conformance test approach for Resource Oriented services in the conformance section of their standard (D021).

While this task will concentrate on the development and implementation of the next generation of a Web Feature Service, the effort shall include an analysis of the implementation recommendations and their suitability for use in the next generation of Web Map Service and Web Coverage Service. The ability to form a core set of implementation requirements that will be supported by WFS, WCS and WMS is critical to the future success of these redesigned services (D021).

Federated Clouds

A big advantage of Cloud architecture is that it allows processing to go to where the data is combined with the ability to easily scale on demand. This works fine as long as the data and the processing reside in the same Cloud: "The Cloud Computing paradigm advocates centralized control over resources in interconnected data centers under the administration of a single service provider. This approach offers economic benefits due to supply-side economies of scale, reduced variance of resource utilization by demand aggregation, as well as reduced IT management cost per user due to multi-tenancy architecture." (Kurze et al, 2011 [http://www.aifb.kit.edu/images/0/02/Cloud_Federation.pdf]). Just, we currently see that industry is generating a proliferation of Clouds. Everything from the global scale commercial Clouds such as AWS or Azure, over private clouds down to personal Clouds which can run on a smart phone. The challenge is to bring the advantages of Cloud computing together while lowering the adverse effects of vendor or software lock-in.

A possible solution is described as Cloud Federations or **Federated Clouds**. "Cloud federation comprises services from different providers aggregated in a single pool supporting three basic interoperability features: resource migration, resource redundancy and combination of complementary resources resp. services." (Kurze et al, 2011 [http://www.aifb.kit.edu/images/0/02/Cloud_Federation.pdf]) The goal is to achieve the same platform transparency on the Cloud federation that has been achieved in single-vendor Cloud offerings. Testbed-14 focuses on the third aspect, the combination of resources and services.

Participants in this task shall build on the Testbed-13 Earth Observation Clouds thread to further advance Cloud federation. Testbed-13 results are documented in three Engineering Reports:

- The OGC Testbed-13: Application Package ER [http://docs.opengeospatial.org/per/17-023.html] defines a data model and serialization for Application Packages that can be used to efficiently deploy any type of containerized application onto a Cloud environment.
- The actual deployment and execution of these Application Packages is defined in the OGC Testbed-13: Application Deployment and Execution Service ER [http://docs.opengeospatial.org/per/17-024.html]
- The OGC Testbed-13: Cloud ER [http://docs.opengeospatial.org/per/17-035.html] describes the use of OGC Web Processing Service (WPS) for cloud architectures. The report addresses issues in lack of interoperability and portability of cloud computing architectures which cause difficulty in managing the efficient use of virtual infrastructure such as in cloud migration, storage transference, quantifying resource metrics, and unified billing and invoicing.

In Testbed-14, the work shall focus on the following issues:

1. Multi-level secure environments (federation across security levels and compartments).
2. Mediate security controls across US and non-US owned clouds.
3. Maintain confidentiality and integrity of information as it crosses Cloud boundaries.
4. Integration of Clouds on DDIL networks.

For each issue, participants shall provide a detailed definition of the problem, propose a solution, and demonstrate that solution using at least two independently developed services.

Participants should also be aware that the U.S. National Institute for Standards (NIST) and the IEEE have teamed together to develop the “Standard for Intercloud Interoperability and Federation” (SIIF) (see http://standards.ieee.org/news/2017/intercloud_interoperability_and_federation.html). The Charter for the SIIF is here http://sites.ieee.org/sagroups-2302/files/2017/07/P2302_PAR_Detail.pdf Participants in this thread shall monitor, and where possible participate in the NIST/IEEE effort representing the needs of the Geospatial Industry.

The federated cloud work embedded in this task makes use of components from the WFS3.0 and Security topics as well as services provided by the Exploitation Platform task. The following diagram illustrates all work items involved.

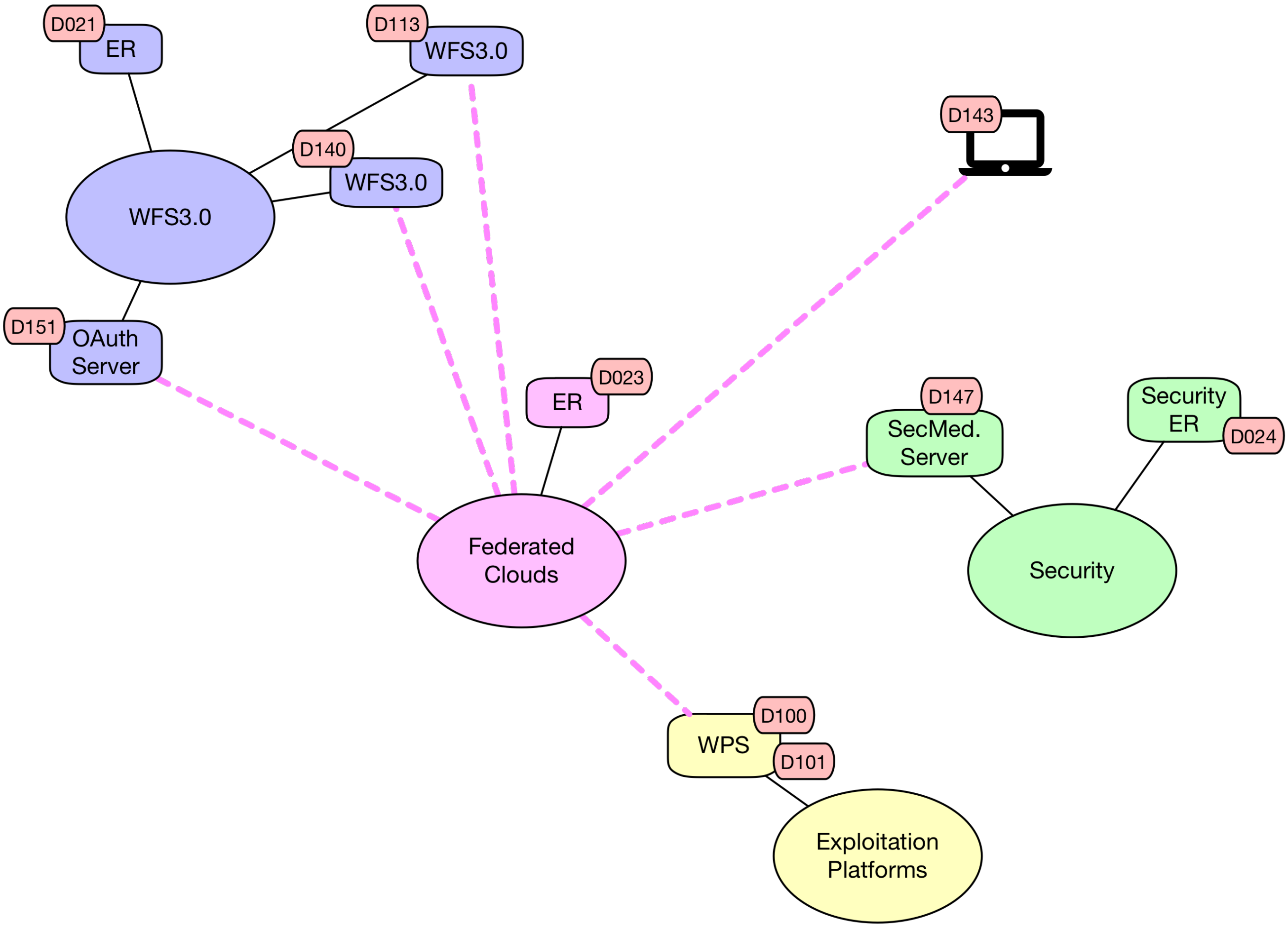


Figure 10. Work item overview of topic Federated Clouds

The dedicated client application (D143) interacts with a *WPS* service that acts as a workflow execution service (D149). The executed work flows include services that are hosted in two different security environments. It is supported by another WPS that acts a *Security Mediation Service* (D147).

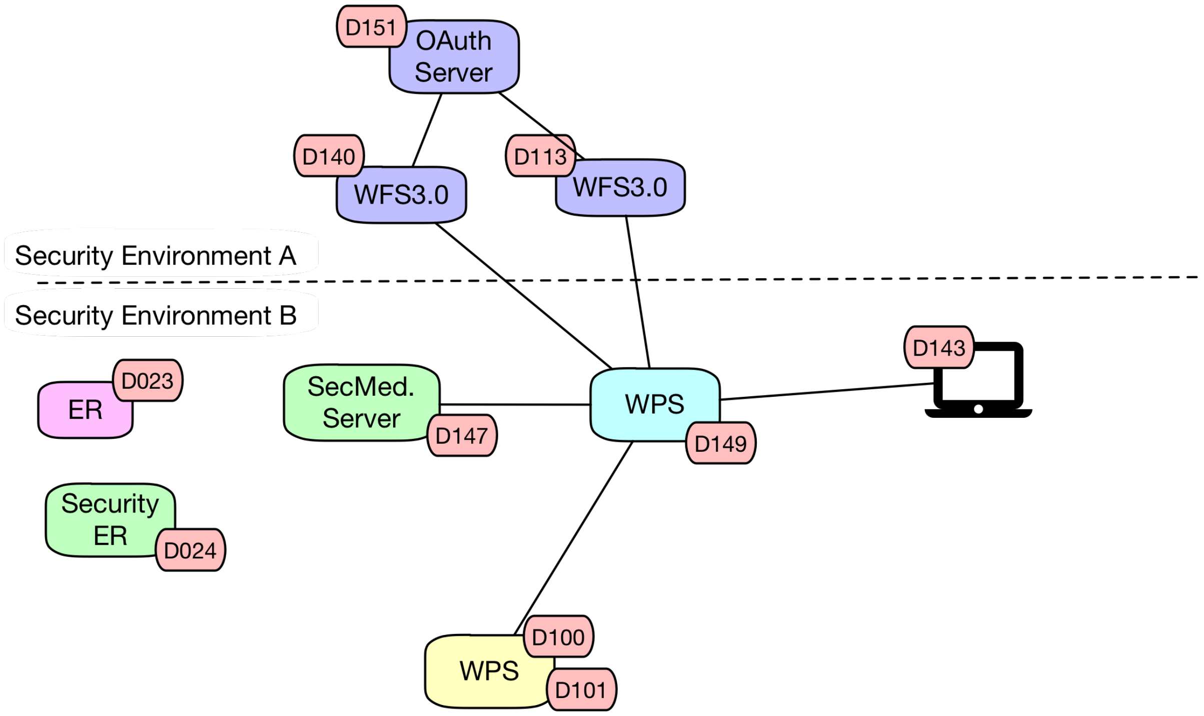


Figure 11. Client/server interactions of topic Federated Clouds

All *Federated Clouds* results need to be captured in the Federated Clouds Engineering Report (D023). This includes a summary of and references to the more detailed security discussion that is primarily captured in the Security Engineering Report (D024).

Security

Testbed-13 demonstrated the implementation of Identification and Authentication (I&A) and simple access controls by OGC Web Services. Results are documented in the OGC Testbed-13: Security ER [https://portal.opengeospatial.org/files/?artifact_id=76487&version=1]. Testbed-14 shall continue these efforts with focus on two topics: I&A Mediation and Security for Next Generation OGC RESTful Web service interfaces. Given that JSON and JSON Schema will play an important role in this context, Testbed participants shall consider the NSA document Security Guidance for the Use of JavaScript Object Notation (JSON) and JSON Schema [https://www.iad.gov/iad/library/supporting-documents/security-guidance-use-json-andjson-schemas.cfm] as an important source of information. The following diagram illustrates all work items involved.

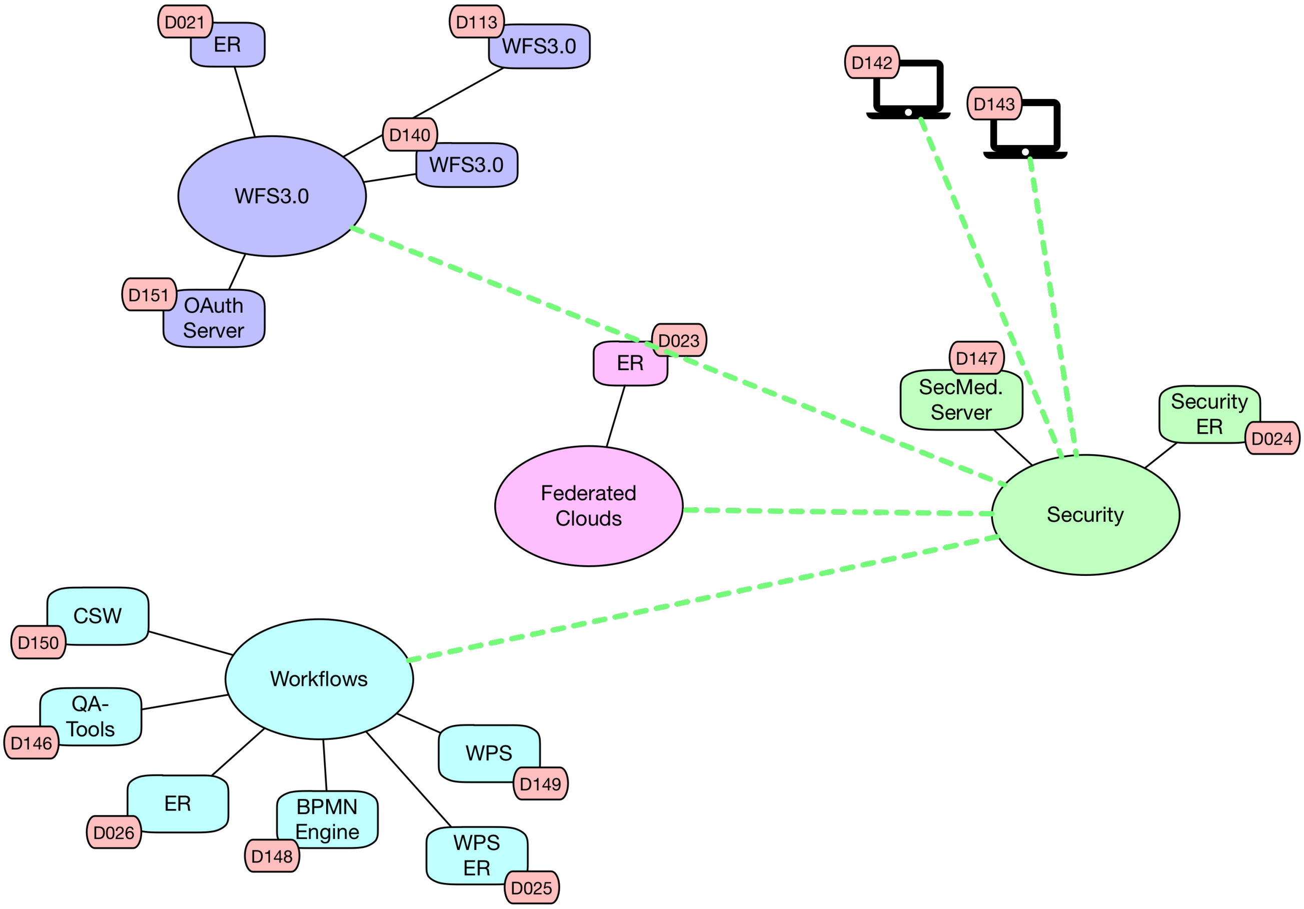


Figure 12. Work item overview of topic Security

The I&A Mediation Service (SecMed. Server (D147)) provides propagation of trusted identities across security domains with different Identification and Authentication protocols. Developers of the I&A Mediation Service shall demonstrate the ability to mediate e.g. between:

1. Basic authentication
2. OAuth2.0
3. HTTP with TLS
4. HTTP with TLS and mutual authentication

For that purpose, the I&A Mediation Service shall interact with the WFS3.0 services and the Exploitation Platform WPS 2.0 instances. These services are part of a Shibboleth environment, whereas the WFS3.0 services are part of an OAuth2/OpenID Connect environment. The following diagram illustrates the client/server interactions.

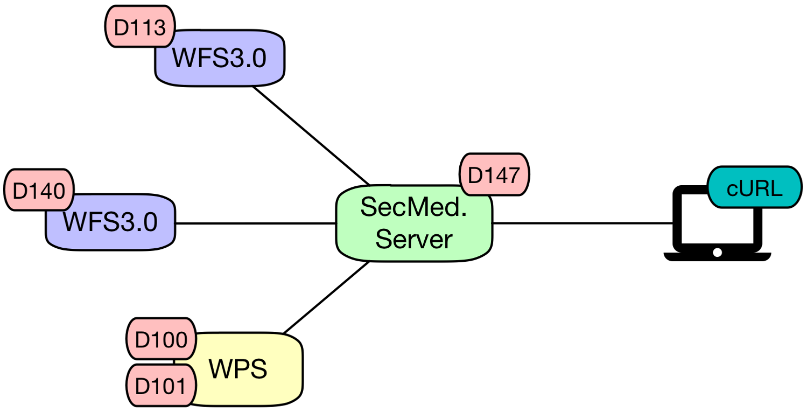


Figure 13. Client/server interactions of topic Security

The service implementer may decide to setup any number of additional services an use these instead of the WPS instances D100 and D101. In addition, the service implementer shall provide a simple client (a cURL [https://curl.haxx.se]-based client is sufficient) that allows exploring the I&A Mediation aspects.

As described in the Workflows section further below, this architecture shall be extended so that security settings can be managed as part of BPMN-powered workflows.

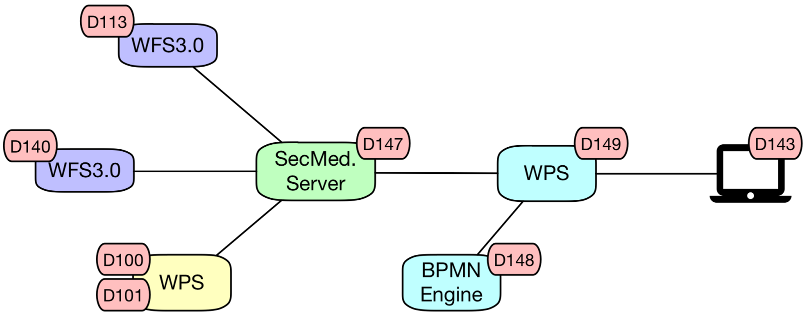


Figure 14. Client/server interactions of topic Security extended with BPMN-powered workflow services

In this case, the dedicated client (D143) uses a transactions WPS (D149) that allows registering new processes defined using BPMN. The WPS is supported by a BPMN Engine (D148). Further details are provided below.

The Security ER (D024) needs to capture all security related results of this task (including results from WFS3.0, Federated Clouds, and Workflows). As the various task topics explore various aspects of security in modern Spatial Data Infrastructures, the ER shall provide a consolidated view on that topic. The Security ER editor has a central role in coordinating and documenting the various aspects explored across this task.

Workflows

Testbed-14 shall explore five aspects as part of the Workflows topic. These include

- Business Process Model and Notation (BPMN) and OGC Services
- Workflow Security
- Transactional Workflows
- Analytic Quality Tool Analysis in Workflows
- Kubernetes, Docker and Cloud Foundry

The following diagram illustrates all work items involved. The Workflows topic uses security components from the WFS3.0 and Security topics.

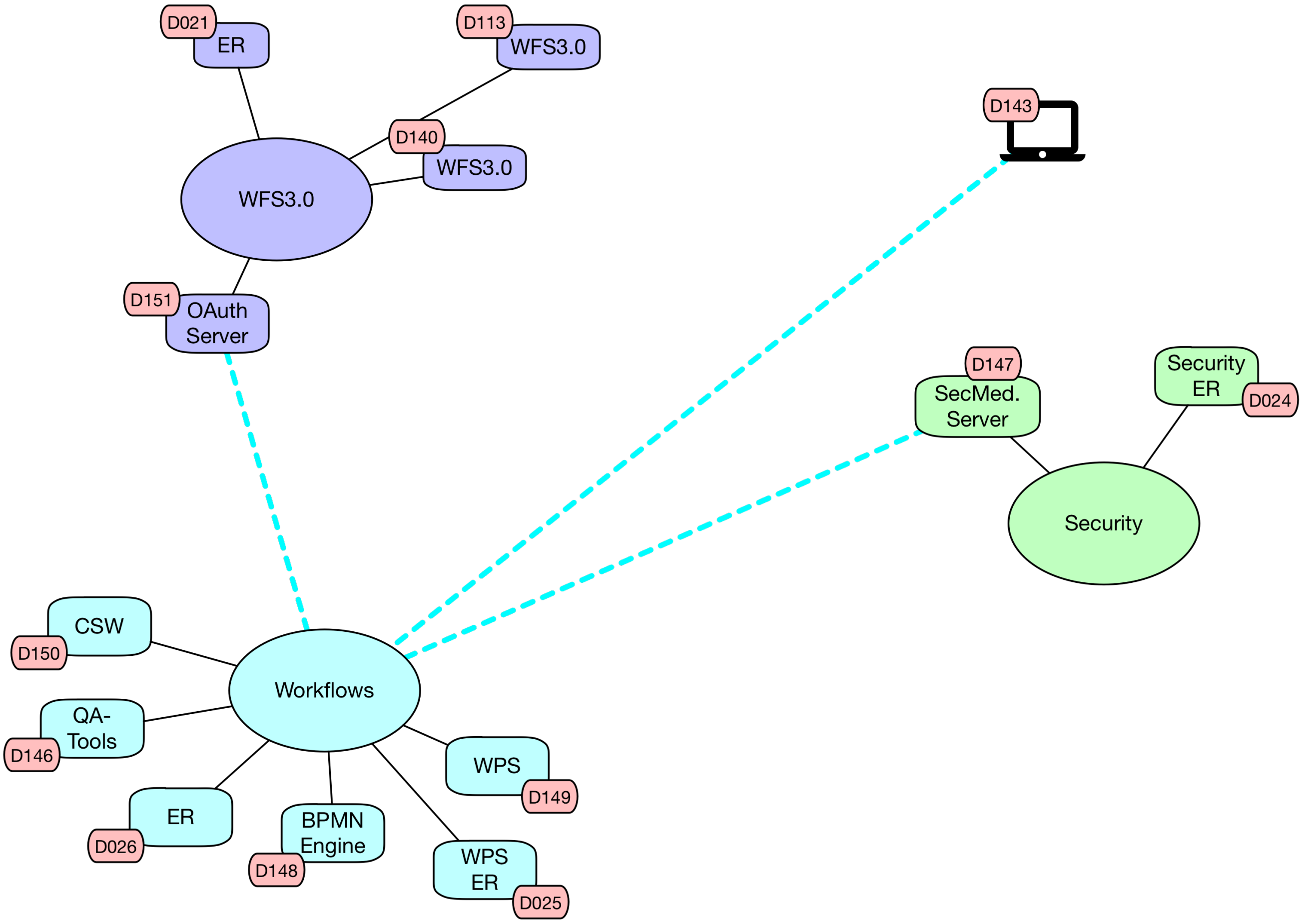


Figure 15. Work item overview of topic Workflows

The following figure illustrates the interactions between the components. A dedicated client (D143) interacts with a transactional WPS (D149) that serves as a frontend to a BPMN Engine (D148). The client can register new BPMN-based processes, for example to register new Analytic Quality processes. The BPMN descriptions shall include security settings that can be registered with an OAuth 2.0 Authorization server (D151).

While executing BPMN-processes, the transactional WPS (D149) makes use of Analytic Quality tools that are again facaded by a WPS interface (D146). A DGIWG-profile compliant CSW (D150) is used to store information about newly registered workflows and Analytic Quality results.

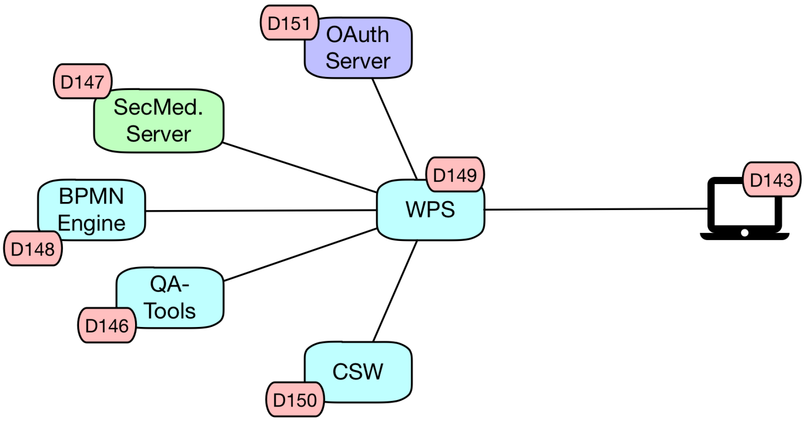


Figure 16. Client/server interactions of topic Workflows

Business Process Model and Notation (BPMN) and OGC Services

Testbed-13 showed that BPMN [http://www.omg.org/spec/BPMN/2.0/PDF] can be used to compose, discover and execute secured workflows in the geospatial domain. All details can be found in the OGC Testbed-13: Workflows ER [http://docs.opengeospatial.org/per/17-029.html], which evaluates earlier initiatives that mostly favored the Business Process Execution Language (BPEL) [https://en.wikipedia.org/wiki/Business_Process_Execution_Language] as workflow execution language and explains why more recent studies now recommend BPMN for describing and executing workflows comprised of OGC Web services (see in particular Rosser, 2015 [https://www.researchgate.net/profile/Julian_Rosser/publication/303881781_Full_Meta_Objects_for_flexible_geoprocessing_workflows_profiling_WPS_or_BPMN/links/5767941808ae421c448c52b2.pdf], OGC 16-091: BPMN 2.0 for Orchestrating OGC Services [https://portal.opengeospatial.org/files/?artifact_id=68879&version=1]).

Testbed-13 has demonstrated that the service task element (see Figure 10.12 - The Service Task class diagram in the BPMN 2.0.2 [http://www.omg.org/spec/BPMN/2.0/PDF] specification) can be used to specify OGC Web services. However, in Testbed-13 the participants had to create a client class in Java to do the actual communication with the services during the execution of the workflow. As the service task is a generic element, other BPMN engines will need a similar helper class. The Java class developed in Testbed-13 was specific to the Camunda BPMN Engine [https://camunda.org/] and cannot be used directly with other engines. Also, different BPMN engines handle the treatment of inputs and outputs to/from service tasks differently. The approach taken with Camunda will not work with other engines. Testbed-14 shall define a general approach for BPMN engines to communicate with OGC Web services, including the treatment of inputs and outputs for use across different workflow engines. In particular, the following aspects shall be addressed under the lead of the Workflow ER (D026) editor:

a. Investigate how client functionality to communicate with OGC Web services (e.g. XML en- and decoding) can be made usable by different workflow engines.

b. Investigate a common approach to handle inputs/outputs defined in the BPMN.

c. Incorporate the BPMN security extensions defined through Workflow Security in the proposed common approach.

d. Demonstrate the viability of the recommended approach through an implementation of an OGC compatible BPMN workflow engine.

Workflow Security

The requirements for Workflow Security have been described above. Goal is to demonstrate how security settings included in BPMN instances can be executed and enforced during the entire workflow from initial client (D143) interaction to final result provision.

Transactional Workflows

“The OGC Web Processing Service specification provides a means to perform distributed web-based processing on geodata. However, the specification does not provide the ability to dynamically deploy and undeploy processes” [Schaeffer, 2008 [https://wiki.52north.org/pub/Geoprocessing/TransactionalWPS/Schaeffer_-_Towards_a_Transactional_Web_Processing_Service.pdf]]. Testbed-14 shall develop a Transactional extension to WPS (WPS-T) based on the WPS 2.0 Interface Standard [http://docs.opengeospatial.org/is/14-065/14-065.html] and the WPS 2.0 Transactional Extension Discussion Paper [https://portal.opengeospatial.org/files/?artifact_id=74162] and suitable for use in a BPMN workflow. In addition, this sub-task shall perform the following activities in support of a WPS-T compatible workflow environment:

a. Investigate how the StatusInfo document of WPS 2.0 can be extended to support the transactional extension and for displaying intermediate results.

b. Investigate how newly created workflows/WPS processes can be discovered, e.g. considering the identifier, title, abstract and/or keywords.

c. Investigate how new resources can be dynamically used in the workflow, e.g. by using the OAuth 2.0 Authorization Code Grant Flow.

d. Provide an implementation of the OGC CSW ISO / DGIWG Profile for discovery of both newly deployed WPS processes and Analytic Quality metadata as described below.

The following figure shows all components being involved.

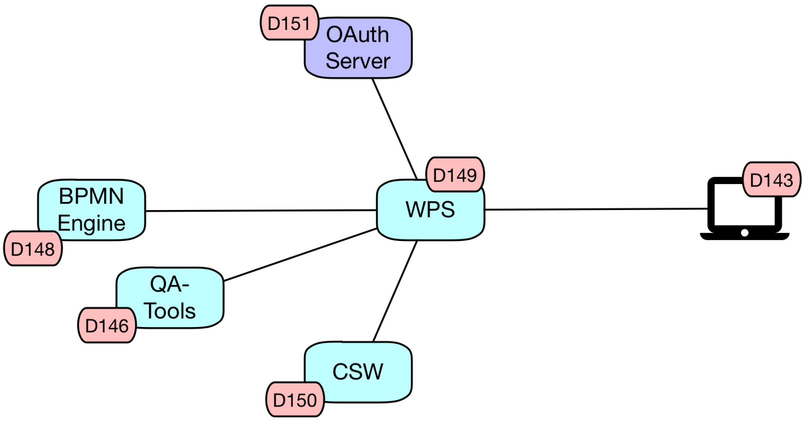


Figure 17. Client/server interactions for secure WPS transactions

The client application (D143) shall register new BPMN-based processes at the transactional WPS (D149). The WPS (D149) shall ensure that newly registered processes are executable using the BPMN engine (D148), the analytic quality tool WPS (D146) and any number of arbitrary OGC Web services. The WPS (D149) shall further ensure proper interaction with the OAuth Server (D151) for authorization settings of the new processes and register these at the catalog service (D150). The catalog service shall implement the CSW ISO/DGIWG Profile.

The transactional extension to WPS 2.0 will be captured in Engineering Report D025. This ER shall fulfill all requirements on new specifications as defined by the OGC Standards Program.

Participant implementers of the CSW ISO/DGIWG Profile shall coordinate with the development of the CITE Compliance capability for testing this profile implementation.

Analytic Quality Tool Analysis in Workflows

NGA seeks an expanded workflow and WPS wrapper development for additional geospatial analysis quality tools in development at NGA. NGA shall provide such tools directly or through GitHub. These tools will likely be in the form of APIs performing different functions related to the analysis of the quality of geospatial data. Modifications or updates to geospatial analysis quality tools shall be documented and incorporated into the tool baseline. Results of the quality analysis as performed in the workflow shall be used to populate an updated dataset metadata based on ISO 19115 and loaded to an OGC CSW ISO / DGIWG Profile.

Kubernetes, Docker and Cloud Foundry

A discussion has recently arisen around geospatial services (here "services" more in the sense of geospatial functions and processes, but not necessarily as in "Web services") and their availability in modern distributed architectures. Cloud platforms such as Cloud Foundry or OpenShift and container platforms such as Docker or Kubernetes lose their distinct borders as they integrate and interweave, partly by adding services such as e.g.

Docker Swarm and Compose, partly by extending their functionality beyond their original purpose. The Cloud Foundry Diego project integrates Docker into the Cloud Foundry PaaS [https://en.wikipedia.org/wiki/Platform_as_a_service]. Docker again can be integrated into various infrastructure tools such as Kubernetes, but plans native support for Kubernetes itself [https://youtu.be/h2B6mhDCw2o]. Remains the question what the geospatial community needs to do in order to support those integrated solutions in the best way. The Workflow Engineering Report (D026) shall contain a study on these aspects and answer questions such as

- What functionality needs to be provided as a software service?
- What is a platform function that shall be provided?
- Where are the borders between these two?

The study shall serve as a starting point for future architecture discussions. It does not need to be comprehensive in the sense that all tools or architectural aspects need to be analyzed. Bidders shall describe their focus and level of detail in their proposals.

B.7.2. Work Items Overview

The following figure illustrates the *Next Generation OGC Web Services, Federated Clouds, Security & Workflows* work items.

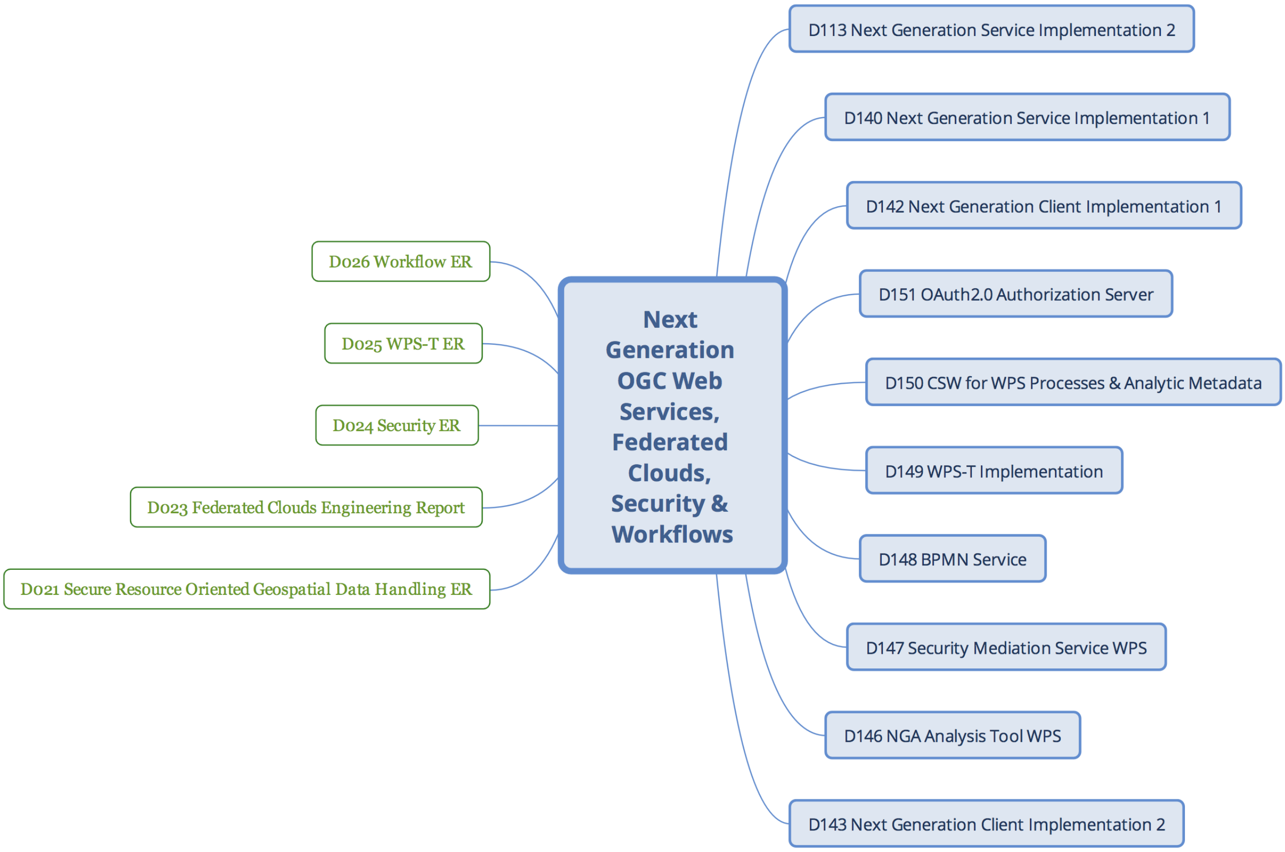

Next Generation OGC Web Services, Federated Clouds, Security & Workflows
D026 Workflow ER
D025 WPS-T ER
D024 Security ER
D023 Federated Clouds Engineering Report
D021 Secure Resource Oriented Geospatial Data Handling ER
D113 Next Generation Service Implementation 2
D140 Next Generation Service Implementation 1
D142 Next Generation Client Implementation 1
D151 OAuth2.0 Authorization Server
D150 CSW for WPS Processes & Analytic Metadata
D149 WPS-T Implementation
D148 BPMN Service
D147 Security Mediation Service WPS
D146 NGA Analysis Tool WPS
D143 Next Generation Client Implementation 2

Figure 18. Next Generation OGC Web Services, Federated Clouds, Security & Workflows work items

B.7.3. Deliverables

The following list identifies all deliverables that are part of this task. Additional requirements are stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D113 Next Generation Service Implementation 2** - WFS3.0 prototype and simple client (a cURL [https://curl.haxx.se]-based client is sufficient) that implements OGC/W3C Spatial Data on the Web Working Group best practices as defined above and security mechanisms. If Best Practices cannot be followed, the implementer shall develop a revised version (by adding the reasons why a given recommendation was changed and the new recommendation) of the REST User Guide [http://docs.opengeospatial.org/guides/16-057r1.html] together with the editor of D021.
- **D140 Next Generation Service Implementation 1** - WFS3.0 prototype that supports a facaded WFS2.0 instance with security support in addition to its native data access. The WFS2.0 instance shall be provided by D140. The service shall be complemented by a simple client (a cURL [https://curl.haxx.se]-based client is sufficient). If Best Practices cannot be followed, the implementer shall develop a revised version (by adding the reasons why a given recommendation was changed and the new recommendation) of the REST User Guide [http://docs.opengeospatial.org/guides/16-057r1.html] together with the editor of D021.
- **D142 Next Generation Client Implementation 1** - Client application that supports the WFS3.0 work described above. The client shall provide GUIs for user interactions and visualizations of results.
- **D143 Next Generation Client Implementation 2** - Client application that supports the Federated Clouds, Security, and Workflow work described above. The client shall provide GUIs for user interactions and visualizations of results.
- **D146 NGA Analysis Tool WPS** - WPS instance facading of NGA geospatial analysis quality tools.
- **D147 Security Mediation Service WPS** - WPS instance for providing Identification and Authentication mediation capabilities.
- **D148 BPMN Service** - BPMN Engine that can execute BPMN instances on behalf of the transactional WPS (D149)
- **D149 WPS-T Implementation** - Transactional WPS instance that allows registering new BPMN instances and interacts with authorization services (D151) and catalog services (D150).
- **D150 CSW for WPS Processes & Analytic Metadata** - Implementation of the OGC CSW ISO / DGIWG Profile for discovery of both newly deployed WPS processes and Analytic Quality metadata that is encoded compliant to ISO 19115.

- **D151 OAuth2.0 Authorization Server** - OAuth 2.0 server instance that supports all requirements stated above. The service shall allow registering new security settings for newly deployed processes. Ideally, the service supports OAuth 2.0 dynamic client registration.

- **D021 Secure Resource Oriented Geospatial Data Handling ER** - Engineering report that captures all WFS3.0 results. This includes a summary of and references to the more detailed security discussions that are primarily captured in the Security Engineering Report (D024). It includes also Change Requests against existing specifications and possible revisions of the REST User Guide [http://docs.opengeospatial.org/guides/16-057r1.html] if Best Practices cannot be followed. The latter will be developed in close cooperation with the WFS3.0 implementers D113 and D140.

- **D023 Federated Clouds Engineering Report** - Engineering report that captures all *Federated Clouds* results. This includes a summary of and references to the more detailed security discussion that is primarily captured in the Security Engineering Report (D024).

- **D024 Security ER** - Engineering report that captures all security related results of this task (including results from WFS3.0, Federated Clouds, and Workflows). As the various task topics explore various aspects of security in modern Spatial Data Infrastructures, the ER shall provide a consolidated view on that topic. The Security ER editor has a central role in coordinating and documenting the various aspects explored across this task.

- **D025 WPS-T ER** - Engineering report that specifies a transactional extension to WPS 2.0. This ER shall fulfil all requirements on new specifications as defined by the OGC Standards Program.

- **D026 Workflow ER** - Engineering report that captures all *Transactional Workflows* results. This includes a summary of and references to the more detailed security discussion that is primarily captured in the Security Engineering Report (D024) and the results of Kubernetes, Docker and Cloud Foundry study.

B.8. Complex Feature Handling

This task completes the Next Generation OGC Web Services, Federated Clouds, Security & Workflows task's section on WFS3.0. It is organized as a dedicated task because it reviews the work around WFS3.0 from a different angle. Here, focus is on complex 3D data or complex data schemas. The goal is to identify the best service solution for these particular needs as opposed to assuming they should be added to WFS 3.0 right away; which is a legitimate outcome of this task, but a full evaluation of options needs to precede.

The current work for a complete revision of WFS has focused on simple features and interaction, interrogation and extraction of these. However there are more complex use cases that require 3D data and/or more complex data schemas to achieve a users desired outcome. Some examples of these include:

- Queries for addresses, i.e all buildings on a road (addresses are nested within the building object in CityGML)
- Accessing simple LoD1 building models that are represented as solids.
- Display and filtering of analysis results on building level from solar, noise or other environmental analysis
- Accessing different versions (including historic representations) of features

To ensure that appropriate support is given to these use cases within the wider area of next-generation OGC Web services, Testbed-14 shall run a concept study addressing the following aspects:

1. Assessment of what use cases, especially those that utilise 3D or more complex data schemas, would require more complex interactions than the current WFS 3.0 spec would provide
2. Assessment of current OGC standards and community standards on the basis of both supporting the uses cases captured and alignment to next-gen OGC Web service approaches, including but not limited to a review around the interaction between 3DPS and WFS 2.0.
3. Recommendations for the most appropriate way to support these use cases within next-gen OGC services, whether by recommending an existing standard that is already aligned with the next-gen OGC web service approach, making appropriate extensions to WFS 3.0 or other standards, or whether a separate service should handle these use cases.

B.8.1. Work Items Overview

The following figure illustrates the *Complex Feature Handling* work items.

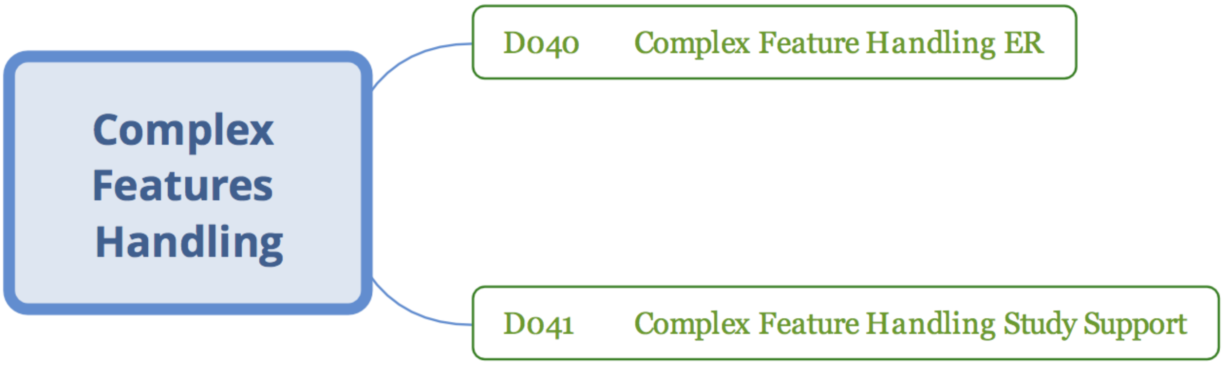

Complex Features Handling
D040 Complex Feature Handling ER
D041 Complex Feature Handling Study Support

Figure 19. Complex Feature Handling work items

B.8.2. Deliverables

The following list identifies all deliverables that are part of this task. Additional requirements may be stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D040 Complex Feature Handling ER** - This Engineering Report captures all results and experiences of this task.
- **D041 Complex Feature Handling Study Support** - Support role for the editor that shall help developing use cases and puts special emphasis on CityGML, 3DPS, and WFS 2.0 interactions.

B.9. Swath Data and the Climate Forecast Convention

OGC Testbed-14 supports the NASA Earth Science Data System (ESDS) Vision to

> make NASA's free and open earth science data interactive, interoperable and accessible for research and societal benefit today and tomorrow.
>
> — NASA

It contributes directly to ESDS mission to develop

> a more rigorous, detailed, and structured process to make interacting with data systems simpler, friendlier, and easier.
>
> — NASA

Historically, less emphasis at NASA was placed on getting swath data into desktop GIS clients than the level 3 and level 4 gridded data. Testbed-14 will address this aspect - taking the Climate Forecast (CF) Conventions for remote sensing data into account - and will conduct activities in the following categories:

1. Modeling swath data that use the Climate Forecast Conventions as coverages
2. Encodings for swath coverages
3. Provide swath data at WCSs hosting NASA data
4. Access and use CF-compliant Swath Data with GIS clients

Improving access to NASA data in Swath structure is a key objective in Testbed-14. A first step is to understand Swath Data in the context of OGC Coverages Implementation Schema

(CIS). The second step is to select a WCS Profile for a CIS defined CF-compliant Swath Data structure, before making swath coverage data available at WCS with visualization in GIS clients.

1. Modeling Swath Data that Use the Climate Forecast Conventions as Coverages

The Encoding of Swath Data in the Climate and Forecast Convention [http://gist.asciidoctor.org/?github-Unidata/EC-netCDF-CF//swath/swath.adoc] provides a proposal for an extension of the Climate and Forecast Metadata Convention [http://cfconventions.org/] (CF) to standardize the encodings of Earth Science swath data in the original instrument viewing geometry. The aim is that some form of the proposal be included in a future version of the CF convention. The proposal is based on the CF version 1.7 [http://cfconventions.org/Data/cf-conventions/cf-conventions-1.7/cf-conventions.html] (current release) although backward compatibility is not necessarily an objective.

In addition, the Draft Extension for Group Hierarchies [https://docs.google.com/document/d/1KK6IZ2ZmpaUTVgrw-GlFd6almppjvGz6D7nxVTO3BtI/edit] proposes enhancements called CF2-Group to support files with groups. Group files are those which utilize the "group" features of the Common Data Model (CDM) as implemented in NetCDF4 [http://www.unidata.ucar.edu/software/netcdf/]. Such files can contain hierarchical, directory-like structures not possible with the single, flat namespace described by the current convention.

The referenced documents should be taken as a starting point, though exploring possible mechanisms to express swath data compliant with the currently approved CF conventions version 1.7 and the latest working draft 1.8 [http://cfconventions.org/cf-conventions/cf-conventions.html] should be part of Testbed-14 as well.

When modeling swath data, different scanning methods resulting in different geometries shall be considered. The following graphic illustrates some typical situations, but is not exhaustive.

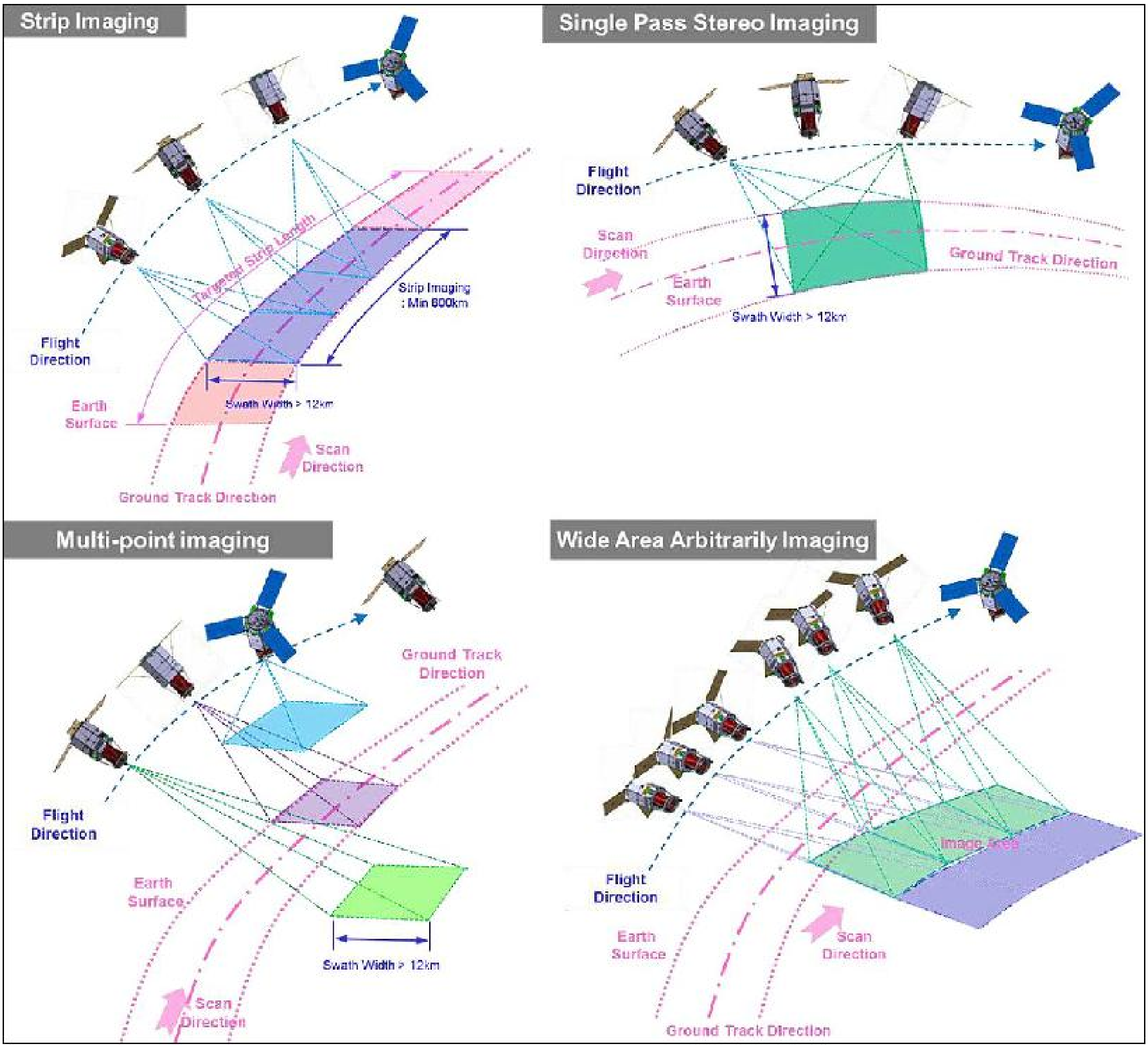


Figure 20. Example imaging/scanning modes (source: eoportal.org [https://directory.eoportal.org/web/eoportal/satellite-missions/k/kompsat-3a])

Others scanning modes that shall be addressed in Testbed-14 include for example vertical curtains. The Cloud-Aerosol LiDAR and Infrared Pathfinder Satellite Observations (CALIPSO) satellite mission provides vertical, curtain-like images of the atmosphere on a global scale using LiDAR. Probing the atmosphere with short pulses of laser light, CALIPSO allows scientists to determine precisely the altitudes of clouds and aerosol layers and the extent of layer overlap. In addition, it allows to identify the composition of clouds and to estimate the abundance and sizes of aerosols.

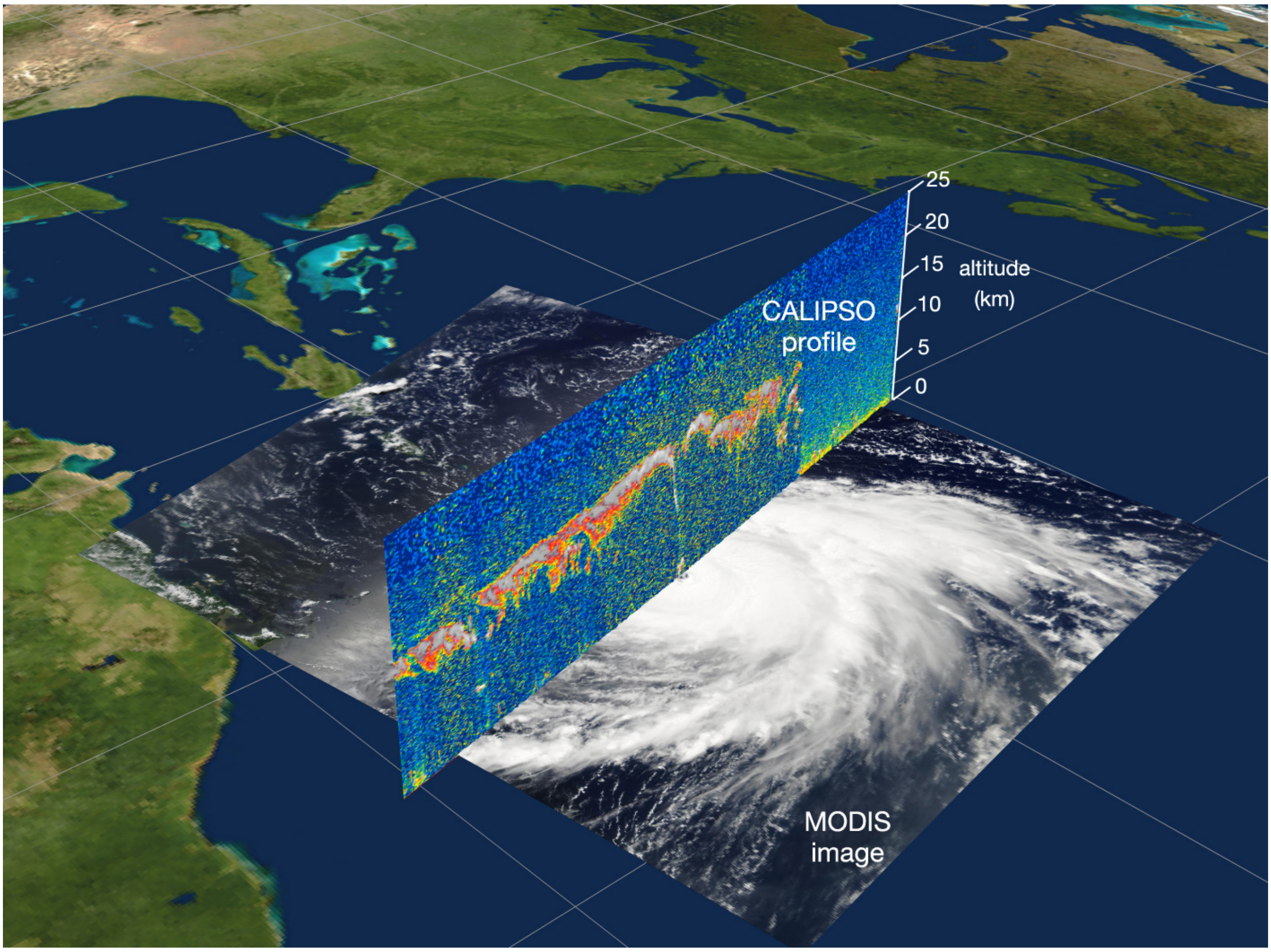


Figure 21. The CALIPSO LiDAR provides vertical, curtain-like images of the atmosphere on a global scale. (source: nasa.gov [https://www.nasa.gov/sites/default/files/thumbnails/image/diabar-calipso-full.jpg])

Another example for curtain-like scans are produced by the Cloud Profiling Radar (CPR), a 94-GHz nadir-looking radar which measures the power backscattered by clouds as a function of distance from the radar on board of CloudSat [https://www.nasa.gov/mission_pages/cloudsat/spacecraft/index.html]. The following figure shows the resulting scanning profile.

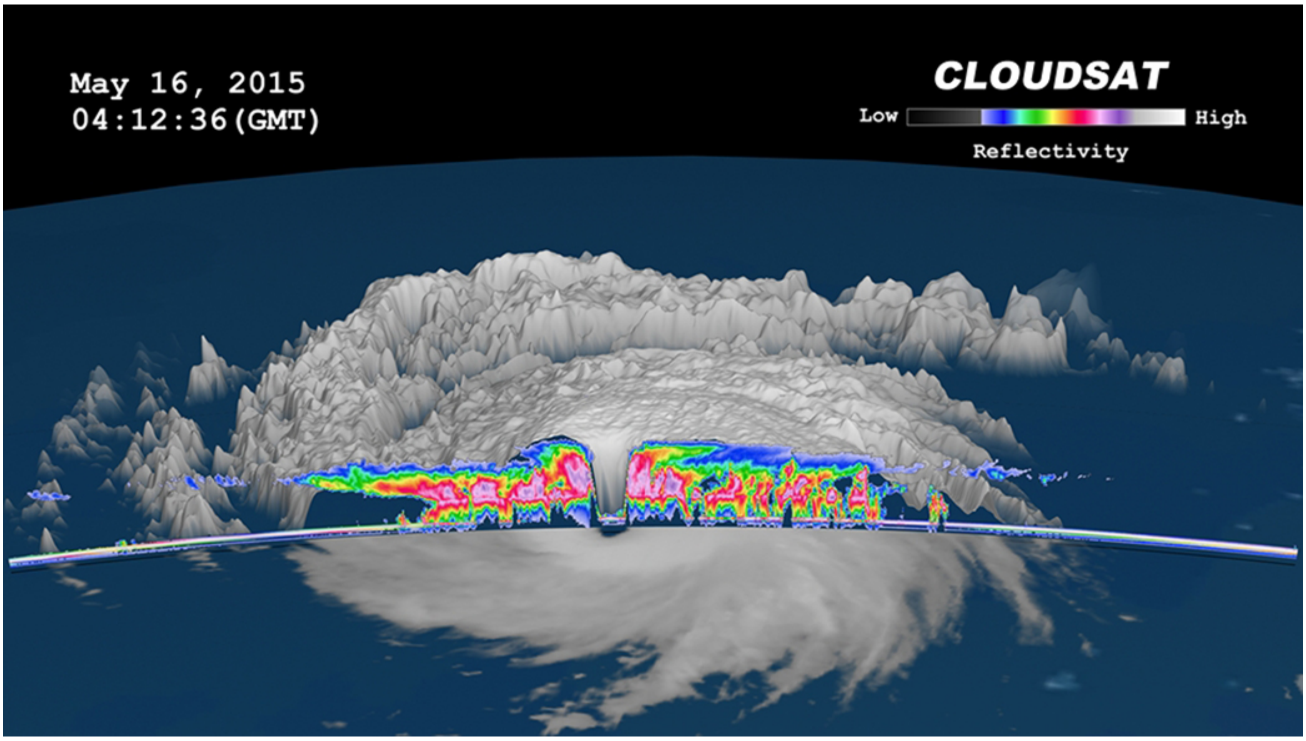


Figure 22. MITSTAT and CloudSat imagery of Typhoon Dolphin. (source: nasa.gov [https://www.nasa.gov/sites/default/files/thumbnails/image/cloudsat_-dolphin_0.jpg])

In summary, swath data from the following instruments and satellites shall be supported:

- Moderate Resolution Imaging Spectroradiometer (MODIS)
- Advanced Microwave Scanning Radiometer - Earth Observing System (AMSR-E)
- Visible Infrared Imaging Radiometer Suite (VIIRS) on the Suomi National Polar-Orbiting Partnership (Suomi NPP) satellite
- Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER)
- Multi-Imaging Spectroradiometer (MISR)
- Cloud-Aerosol Lidar and Infrared Pathfinder Satellite Observations (CALIPSO)
- Soil Moisture Active-Passive (SMAP)
- CloudSat

To support access to swath data using WCS, the swath data structure needs to be analyzed with respect to the OGC Coverage Implementation Schema (CIS) [http://docs.opengeospatial.org/is/09-146r6/09-146r6.html] version 1.1. CIS specifies a coverage model for concrete, interoperable, conformance-testable coverage structures. CIS is based on the abstract concepts of OGC Abstract Topic 6 (which is identical to ISO 19123). CIS is interoperable in the sense that coverages can be conformance tested, regardless of their data format encoding, down to the level of single "pixels" or "voxels". A key element of the *Swath Coverage Engineering Report* is a CIS Profile for CF-compliant Swath Data. Particular emphasis needs to be on the grid-irregular conformance class as described in section 8 of the OGC Coverage Implementation Schema (CIS) specification 09-146r6 [http://docs.opengeospatial.org/is/09-146r6/09-146r6.html].

Additional requirements may arise from the mandatory use of the OGC Web Coverage Service (WCS) Application Profile - Earth Observation v1.1, as described in the next sections.

2. Encodings for Swath Coverages

Coverages can be encoded in any suitable format (such as GML, JSON, GeoTIFF or NetCDF) and can be partitioned, e.g., for a time-interleaved representation. Encoding(s) of the swath coverages defined above need to be defined for access using WCS.

- GeoTIFF (floating point)
- NetCDF
- GeoPackage

3. Provide Swath Data at WCSs Hosting NASA Data

OGC Web Coverage Service (WCS) is a family of standards for Web service access to coverages. The recently developed OGC Web Coverage Service (WCS) Application Profile - Earth Observation v1.1 [https://portal.opengeospatial.org/files/?artifact_id=70317&version=1], short EO-WCS v1.1 (OGC 10-140r2), defines a profile of WCS 2.0 [https://portal.opengeospatial.org/files/09-110r4] (OGC 09-110r4) for use on Earth Observation data. Version 1.1 of the EO-WCS profile now supports 3D Datasets. A Dataset is an EO Coverage, which can represent, for example, a hyperspectral 2D satellite scene or a 3D atmospheric model.

Further background information on the latest EO profile updates is given in the Testbed-12 WCS Profile Update Engineering Report [http://docs.opengeospatial.org/per/] (OGC 16-033), which suggested the necessary extensions to the existing Web Coverage Service profiles, particularly the WCS Earth Observation Application Profile version 1.0 [https://portal.opengeospatial.org/files/42722] (OGC 10-140r1) to support multi-dimensional subsetting of 3D space and 1D time. The updated EO-WCS v1.1 dropped the 2D constraint and now explicitly allows coverages with more dimensions as long as they have a geographic footprint.

Testbed-14 shall analyse the EO-WCS profile version 1.1 and implement and describe two EO-WCS with the necessary extensions to cover the Swath Coverage modeling and encoding activities of the two prior steps. The two WCSs will be deployed for interoperability testing. Variations of the two WCS implementations will be determined based on evaluation of the responses to this Testbed-14 *Call for Participation*. Ideally, the first WCS supports the EO-profile v1.1 and offers an additional RESTful binding. This RESTful binding shall be described compliant to the OpenAPI specification [https://github.com/OAI/OpenAPI-Specification]. The OpenAPI Specification (OAS) defines a standard, programming language-agnostic interface description for REST APIs, which allows both humans and computers to discover and understand the capabilities of a service without requiring access to source code, additional documentation, or inspection of network traffic. The second implementation needs to be compliant to the EO profile version 1.1 with CF-compliant Swath Data support.

The WCSs will host NASA swath data. The Level 1 and Level 2 data to be hosted in the WCS are non-composited L1-L2 data for surface reflectance, surface temperature, soil moisture, fires, and aerosols.

4. Access and Use CF-Compliant Swath Data with GIS clients

To demonstrate access to the swath coverages, commonly available GIS clients will be used to access the WCS hosting NASA swath data. The clients will combine the swath coverages with other commonly available GIS data. The swath data will be visualized in the clients and combined with other data to demonstrate data integration with geospatial accuracy. These additional data sets need to be provided by the client developers.

A demonstration will be conducted that shows the value of WCS access to swath coverages. The demonstration will be scripted based on a scenario that will be defined at the Testbed-14 kick-off meeting. Responders to this call are encouraged to describe potential scenarios and relevant datasets in their proposals.

B.9.1. Work Items Overview

The following figure illustrates the *Swath Data and the Climate Forecast Convention* work items.

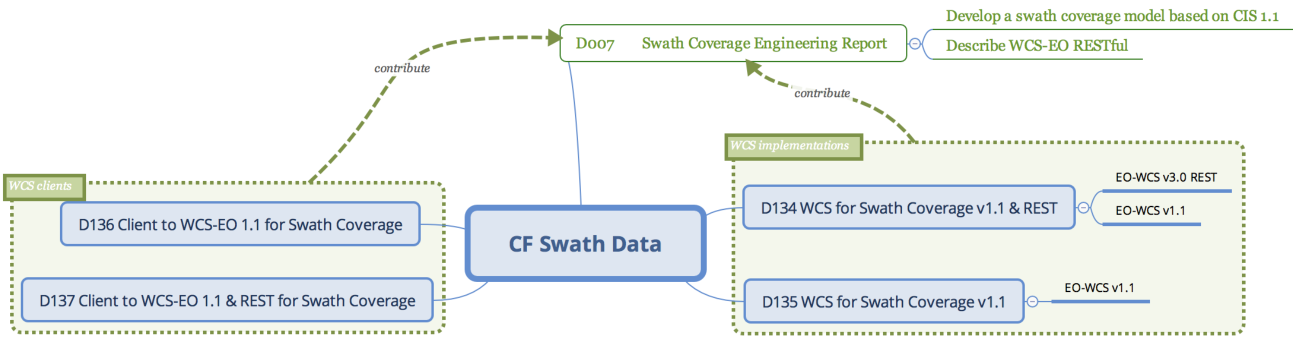

Develop a swath coverage model based on CIS 1.1
D007 Swath Coverage Engineering Report
Describe WCS-EO RESTful
contribute
contribute
WCS implementations
WCS clients
EO-WCS v3.0 REST
EO-WCS v1.1
D134 WCS for Swath Coverage v1.1 & REST
D136 Client to WCS-EO 1.1 for Swath Coverage
CF Swath Data
D137 Client to WCS-EO 1.1 & REST for Swath Coverage
EO-WCS v1.1
D135 WCS for Swath Coverage v1.1

Figure 23. CF-compliant Swath Data deliverables

B.9.2. Deliverables

The following list identifies all deliverables that are part of this task. Additional requirements may be stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D007: Swath Coverage Engineering Report** – Document results of the analysis of how swath data can be modeled as geospatial coverages compliant with the Climate and Forecasting Conventions. A data model will be defined using the elements in CIS and suitable for deployment in WCS. The ER will also include the results of the analysis of encodings suitable for CF-compliant Swath Data Coverages and document all implementation results of this task.
- **D134: WCS for Swath Coverage v1.1 & REST** – Deploy and make available an EO-WCS v1.1 that provides access to Swath Coverages as defined in the modeling activity. Host NASA swath data sets to be made accessible by WCS. Develop and provide a RESTful interface in addition. Describe the RESTful binding using OpenAPI [https://www.openapis.org] and contribute to the development of D007.
- **D135: WCS for Swath Coverage v1.1** – Deploy and make available an EO-WCS that provides access to Swath Coverages as defined in the modeling activity. Host NASA swath data sets to be made accessible by WCS. Contribute to the development of D007.
- **D136: Client to WCS-EO 1.1 for Swath Coverage** – Deploy and make available a client that accesses the WCSs 134 (non-RESTful binding only) and 135 hosting NASA Swath data. Client will also access other datasets using OGC Web Services. These services are known to the client. Contribute to the development of D007.
- **D137: Client to WCS-EO 1.1 & REST for Swath Coverage** – Deploy and make available a client that accesses the WCSs 134 (both RESTful binding and non-RESTful binding) and 135 hosting NASA Swath data. Client will also access other datasets using OGC Web Services. These services are known to the client. Contribute to the development of D007.

B.10. Application Schema Modeling and Conversion

The National System for Geospatial-Intelligence (NSG) Application Schema (NAS) is an ISO 19109 compliant application schema that defines the conceptual model for identifying and encoding feature data in the U.S. National System for Geospatial-Intelligence (NSG). NGA utilizes the open source software tool ShapeChange [http://shapechange.net/] as an integral piece in NAS development. This tool is used to take NAS-based UML models and create XML and RDF based schemas. Testbed-12 began development of capabilities for extracting profiles supporting specific mission functions from the full NAS content. Testbed-13 further refined the approach to NAS Profiling by investigating how a specific profile ("Urban Military Profile") can be processed in an automated way and used to derive implementation schemas for the OGC standards CDB [http://www.opengeospatial.org/standards/cdb] and CityGML [http://www.opengeospatial.org/standards/citygml].

Testbed-14 shall further extend the capabilities of ShapeChange in the following ways:

1. Modeling extensions

The following modeling aspects shall be addressed:

1. OCL Constraints: Extend ShapeChange to support transformation of NAS OCL Constraints into OWL, SHACL and/or SWRL (or other ontology rule language) encodings to be used with NEO;
2. Property Generalization: Simplification of the OWL property structure in NEO by automating the generalization of semantically equivalent UML attributes;
3. Property Enrichment: Addition of OWL properties in NEO to enhance automated reasoning, based on the inclusion of additional tagged values in the NAS; and
4. Implementation-driven Theories: Transformation of the NAS to an OWL ontology(ies) having different levels of complexity as needed to support different types of applications.
5. Extend ShapeChange capability to generate Ontologies from UML models: Demonstrate the enhanced generation of the NSG Enterprise Ontology (NEO) from the NSG Application Schema (NAS) UML models.
6. Extend ShapeChange to support generation of profiles of the NEO:
   a. Produce one ontology suitable for employment by a simple linked data application
   b. Produce one ontology suitable for employment by a complex analytic application
   c. Generate sample data-instances for each ontology
7. Extend ShapeChange to derive OWL/SHACL/SWRL from OCL Constraints in the NAS; including:
   a. Restriction of Allowed Code List Values
   b. Restriction of Allowed Geometries.
8. Extend ShapeChange to support the simplification of the OWL property structure in NEO by automating the generalization of semantically equivalent UML attributes.
9. Extend ShapeChange to support the generation of OWL properties in NEO from newly-included UML tagged values in the NAS.
10. Extend ShapeChange to create transformations of UML models (e.g., NAS) to OWL ontologies having different levels of complexity as needed to support different types of applications.
11. Coordinate on the development of prototype validation capabilities for NEO-conformant RDF-based geospatial data identified in the Compliance thread.

2. JSON Support

This task seeks to extend the ShapeChange tool to support the use of JSON Schema, JSON, and JSON-LD:

1. Extend ShapeChange to generate a JSON Schema that corresponds to the NEO.
2. Extend ShapeChange to generate a JSON-LD-based schema that corresponds to the NEO.
3. Coordinate on the development of prototype validation capabilities for NEO-based JSON/JSON-LD geospatial data identified in the Compliance thread.
4. Extend ShapeChange to generate a JSON Schema that corresponds to the NAS.
5. Extend ShapeChange to generate a JSON-LD-based schema that corresponds to the NAS.
6. Coordinate on the development of prototype validation capabilities for NAS-based JSON/JSON-LD geospatial data identified in the Compliance thread

3. RDF, JSON, and JSON-LD Conversion

1. Document techniques and recommendations for encoding NEO- and NAS-compliant geospatial data in RDF, JSON, and JSON-LD.

4. Enhancements as recommended in Testbed-13

Testbed 14 shall address enhancements recommended in OGC Testbed-13: NAS Profiling ER; specifically, Section 7.3.

1. In Testbed-13, only specific information of OCL constraints of the NAS profile were used to generate the GML-SF0 NAS profile schema. The workflow to generate the GML-SF0 schema extracts this information and then transforms all OCL constraints so that they are no longer available for generating Schematron assertions.
2. Testbed-14 shall perform a detailed analysis of the NAS OCL constraints, to determine which of them would be relevant for the creation of Schematron assertions for the GML-SF0 NAS profile schema. The analysis should also look at how such constraints could be transformed to be valid in the transformed schema.
3. As an example, consider constraints that limit a codelist-range during inheritance. If transformed appropriately when deriving the GML-SF0 schema, Schematron assertions could be generated to more thoroughly validate GML-SF0 XML instance data.
4. The NAS uses OCL constraints to restrict how the position(s) of a feature can be represented: by points, curves, surfaces, physical addresses, and location by identifier. One goal of the derivation of a GML-SF0 NAS profile schema researched in Testbed-13 was that the schema only contained feature types with a single geometry property. To

generate such feature types, restrictions on place representations were extracted from OCL constraints. To support this capability the development of an OCL evaluator to identify place restrictions is required.

5. NAS Profiling Users of the NAS have described it as too complex for mission-specific implementations. This task for ShapeChage development is to improve on the methodology developed/demonstrated in Testbed-13 for Profiling the NAS to achieve smaller, more narrowly-focused, profile outcomes.

1. Handling of association classes by the ShapeChange Profile Management Tool (PMT) should be improved. For example, if an association class is removed from the profile, the corresponding association (and roles) should be removed as well. Same for adding an association class.
2. The PMT should implement the design for profiling constraints which has been developed in Testbed 12 (see the Testbed 12 ShapeChange ER, chapter 7). Special cases like generating OCL constraints to specify that a property from a supertype is not allowed for a specific subtype could then be considered as well.
3. The profiling design developed in Testbed 12 contained additional profile parameters, which the PMT should implement, such as setting the abstractness of a class.
4. Additional mechanisms could be developed to simplify the process of defining a profile. Some possibilities include:
    a. Specific categories of classes could be hidden. A user would then be able to only work with, e.g. feature types or transportation-related classes.
    b. The UML package structure within an application schema could be hidden, instead simply directly listing all schema classes in the application schema.
    c. A new mechanism could be developed to allow the user to select one or more previously specified subsets of the application schema (e.g., NAS Views or View Groups) and include each in its entirety into the profile. For example, selecting all classes associated with the NAS Cultural View Group, the NAS Population View Group and a select number of feature types associated with the NAS Ports and Harbours View Group. These View Group Profiles shall be delivered as GML SF0 and JSON/JSON-LD based schemas

B.10.1. Work Items Overview

The following figure illustrates the *Application Schema Modeling and Conversion* work items.

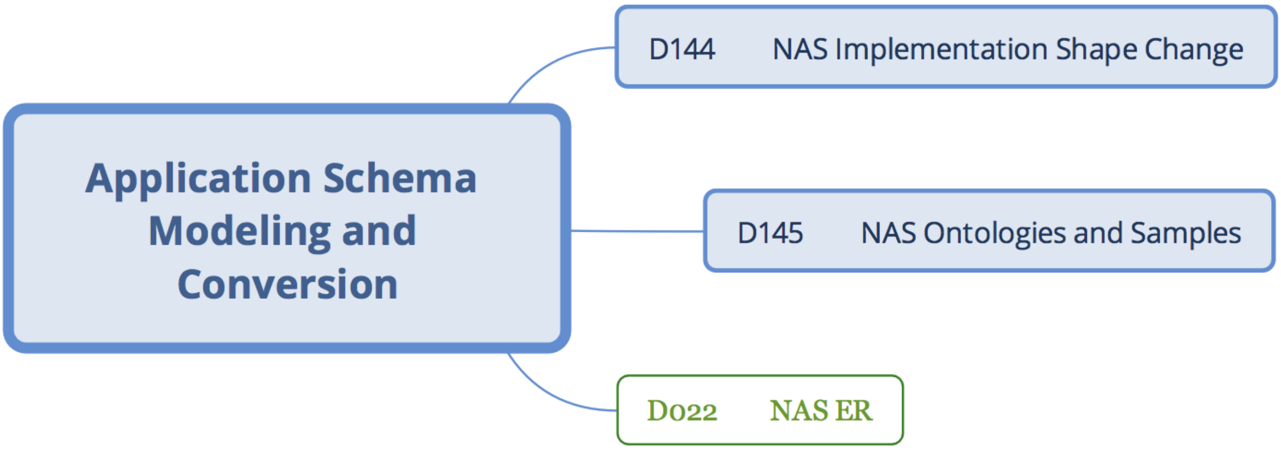

Application Schema Modeling and Conversion
D144 NAS Implementation Shape Change
D145 NAS Ontologies and Samples
D022 NAS ER

Figure 24. Application Schema Modeling and Conversion work items

B.10.2. Deliverables

The following list identifies all deliverables that are part of this task. Additional requirements are stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D144 NAS Implementation Shape Change** - All extensions and modifications as described above to the ShapeChange software.
- **D145 NAS Ontologies and Samples** - NAS ontologies and samples as described above.
- **D022 NAS ER** - Engineering Report capturing all result and experiences of the Application Schema Modeling and Conversion task.

B.11. LiDAR Point Cloud Data Handling

Natural Resources Canada's responsibility is to foster an environment for the development and use of Spatial Data Infrastructures (SDI) both within Canada and internationally. This task is specifically concerned with improving SDI basemap and applications empowered by LiDAR point cloud data.

GeoBase is a federal, provincial and territorial government initiative that is overseen by the Canadian Council on Geomatics (CCOG). It is undertaken to ensure the provision of, and access to, a common, up-to-date and maintained base of quality geospatial data for Canada.

One of GeoBase's current priorities is the National Elevation Strategy that consists in providing LiDAR point cloud and LiDAR-derived Digital Elevation Models (DEMs) for the southern part of Canada. These data products offer great potential for Canadians through a wide range of field, such as flood mapping, forestry and infrastructure. Considering the extent of the Canadian territory and the volume of such data, it is a large-scale project with potential complexities.

Point Clouds, in particular resulting from LiDAR and other sources of scanning that result in sets of vertices in multi-dimensional coordinate systems, play an important role in geospatial processes and applications. Currently, there is no consistent Point Cloud Ecosystem defined in the context of standardized geospatial services and encodings as defined by the Open Geospatial Consortium. The approach would need to be multi-faceted, including aspects such as point cloud data processing, storage, indexing, streaming, visualization, or indexing. Point cloud data would need to be served in a scalable way as raw data as well as in product form, ideally with links between the two.

On an abstract level, this work item shall address three key aspects:

- **Technological Challenges and Standards**: To overcome these challenges, GeoBase highly suggests continuing the development of point cloud management systems and point cloud streaming servers. Focus should be put notably on fast access open-source solutions, optimal indexation and compression efficiency, in order to deal with the complexities of managing point cloud over very large area.
- **State of the Art Solutions**: With a current (and continuously increasing) coverage of nearly 200 000 km² of heterogeneous LiDAR Point Cloud, GeoBase is interested in collaborations with the actual point cloud OGC stakeholders in order to test the current solutions and prototypes.
- **Point Cloud Standards**: Finally, GeoBase is interested in following the advancements on point cloud standards, such as LAS 1.4 or any other widely spread format.

Testbed-14 shall conduct a feasibility study that addresses the following aspects:

- **Point cloud management environment**: Architecture of an ideal management environment for Point Cloud data. The management environment is required for consistent and well communicated processing, storage, streaming, visualization, or indexing of Point Cloud data
- **Integration workflows**: Point Cloud data-set combination and integration from and across multiple vendors
- **Scalability**: How to scale Point Cloud processes in a standardized environment
- **Standards landscape**: Which OGC standards cover which role in the context of Point Cloud data handling and how do OGC standards relate to other standardization efforts such as e.g. LAS1.4. The OGC services need to cover a rich set of functionality, including
    - processing services for e.g. filtering, feature estimation, surface reconstruction, registration, model fitting and segmentation
    - catalog services for data and service registration
    - access services for data access including sub-setting

    - visualization and streaming services

- **Product delivery**: Current efforts address mostly the access and delivery of raw points. This aspect has to be extended to full products, including access services, metadata standards, registries etc.

- **Installation tests**: Several organizations have invested into Point Cloud services. These need to be analyzed in order to extract best practices and to optimize local environments in the context of international standardization.

To ensure a maximum quality Engineering Report, the participant shall foresee and document extensive consultations with OGC activities and other related activities, working groups (such as e.g. Land and Infrastructure DWG [https://portal.opengeospatial.org/?m=projects&a=view&project_id=421] and SWG [https://portal.opengeospatial.org/?m=projects&a=view&project_id=419] and Marine DWG [https://portal.opengeospatial.org/?m=projects&a=view&project_id=514]) and personnel. The participant with the support of the thread architect shall invite other OGC members to join this project on an in-kind basis.

B.11.1. Work Items Overview

The following figure illustrates the *LiDAR Point Cloud Data Handling* work items.

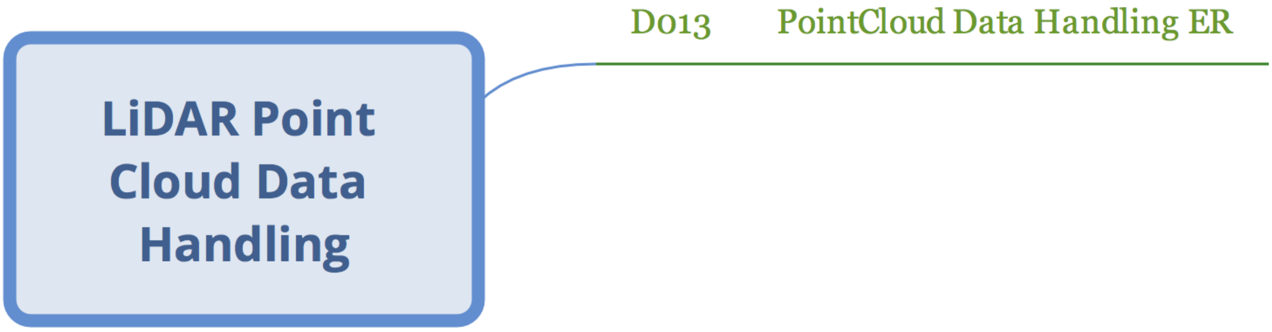


Figure 25. LiDAR Point Cloud Data Handling deliverables

B.11.2. Deliverables

The following list identifies all deliverables that are part of this task. Additional requirements may be stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D013 PointCloud Data Handling ER** - Engineering Report capturing all results and experiences of the feasibility study described above.

B.12. CityGML and Augmented Reality

The Architecture, CAD, and Geospatial industries have been converging on a body of standards to support an integrated view of urban spaces. This view covers the geography and topography, infrastructure and utilities, indoor spaces, and detailed building models. The Future City Pilot project was one of the collaborative efforts to advance this convergence. An example of what is possible can be obtained from the Future City Pilot project video that is available online [https://www.opengeospatial.org/projects/initiatives/fcp1].

This type of information would be very valuable for urban warfighters. This thread seeks to build on industrial capabilities to provide the highly detailed, fully integrated view of urban spaces required for the planning and execution of modern urban missions.

Testbed-14 shall evaluate the possible methods for integrating Augmented Reality (AR) content into CityGML content. The effort shall include an analysis of whether Augmented Reality content should be integrated into the CityGML data or whether the data content (features) should be linked for visualization purposes.

Testbed-14 shall demonstrate client side support for visualization of streamed CityGML with Augmented Reality (AR) content. Bidders shall include suggestions for suitable AR data in their proposals and shall investigate the role and applicability of OGC Augmented Reality Markup Language 2.0 (ARML 2.0) [http://docs.opengeospatial.org/is/12-132r4/12-132r4.html], the proposed Community Standard 3D Tiles [https://github.com/AnalyticalGraphicsInc/3d-tiles/blob/master/README.md], and OGC Community Standard I3S [http://docs.opengeospatial.org/cs/17-014r5/17-014r5.html].

B.12.1. Work Items Overview

The following figure illustrates the *CityGML and Augmented Reality* work items.

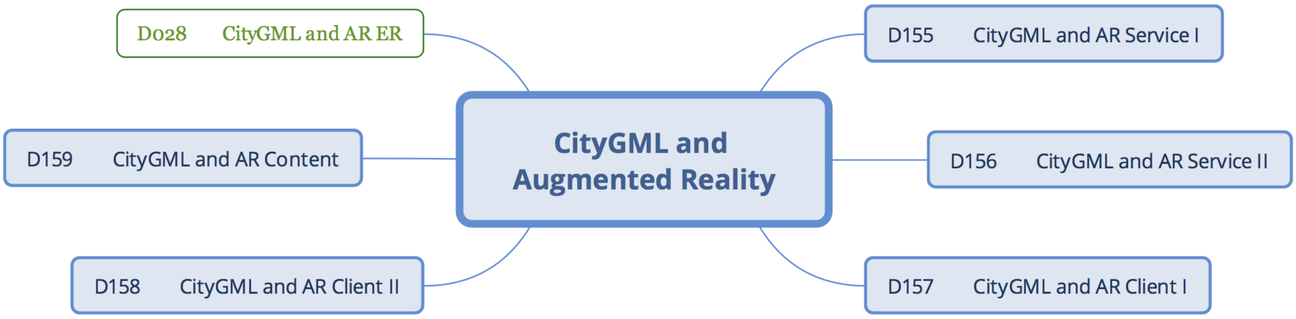

D028 CityGML and AR ER
D155 CityGML and AR Service I
D159 CityGML and AR Content
CityGML and Augmented Reality
D156 CityGML and AR Service II
D158 CityGML and AR Client II
D157 CityGML and AR Client I

Figure 26. CityGML and Augmented Reality work items

B.12.2. Deliverables

The following list identifies all deliverables that are part of this task. Additional requirements are stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D155 CityGML and AR Service I** - CityGML and Augmented Reality data streaming service with support for 3DTiles and/or i3s. The service instance shall be tested with both client applications.
- **D156 CityGML and AR Service II** - CityGML and Augmented Reality data streaming service with support for 3DTiles and/or i3s. The service instance shall be tested with both client applications.
- **D157 CityGML and AR Client I** - CityGML and Augmented Reality data streaming client application with support for 3DTiles and/or i3s. The client instance shall be tested with both servers.
- **D158 CityGML and AR Client II** - CityGML and Augmented Reality data streaming client application with support for 3DTiles and/or i3s. The client instance shall be tested with both servers.
- **D159 CityGML and AR Content** - CityGML and Augmented Reality data that can be integrated. Bidders shall make suggestions for the most appropriate AR data formats.
- **D028 CityGML and AR ER** - Engineering Report capturing all results and experiences from this task.

B.13. Portrayal

Style Layer Descriptors (SLD) were introduced in OWS-1.2 (2002). The currently available version 1.1 of the OpenGIS Styled Layer Descriptor Profile of the Web Map Service Implementation Specification [http://portal.opengeospatial.org/files/?artifact_id=22364] was published in 2007.

Over the past 10 years we have not seen an enthusiastic adoption of this technology, however it is still the underlying method to style layers in GeoServer [http://geoserver.org]. Furthermore, the common availability of 2.5 and 3-D visualization tools have taken symbology requirements far beyond those envisioned in 2002.

Testbed-14 shall investigate other renderer outputs such as JSON encoding of the portrayal information, so they can be handled on the client side in HTML5.x Canvas or other rendering libraries such as D3.js [https://d3js.org]. Participants in this task shall perform an Analysis of Alternatives, e.g., Cascading Style Sheets (CSS), YAML SLD, MBStyle, and CartoDB, to identify and assess alternatives to the OGC Style Layered Descriptions/Symbol Encoding (SLD/SE) standard as implemented in OGC GeoPackage and OGC WMS/WMTS. Participants shall recommend a path forward and if a revision of SLD/SE is the best alternative or if there are better options available, such as a GeoCSS standard. Standardized portrayal continues to be an issue for the end user as proprietary implementations are inconsistent between vendors.

NOTE At the time of writing, HTML5.1 2nd Edition [https://www.w3.org/TR/html51/] has been published as a W3C Recommendation, indicating the most mature stage of development. Nevertheless, version HTML5.2 [https://www.w3.org/TR/html52/] is a W3C Proposed Recommendation as of 02 November 2017 and likely to supersede HTML5.1 2nd Edition in the near future. Testbed participants should consider the latest recommendation during the Testbed.

So far, the emphasis on developing the portrayal ontologies (Testbeds-12/13) has been on modeling and representing portrayal information for feature data.

The Testbed-12 Semantic Portrayal, Registry and Mediation Engineering Report [http://docs.opengeospatial.org/per/16-059.html] specifies semantic information models and REST APIs for Semantic Registry, Semantic Mediation and Semantic Portrayal Services. It introduces the Semantic Registry Information Model (SRIM), a superset of the W3C DCAT ontology. SRIM can accommodate registry items other than dcat:Dataset, such as Service description, Schema and Schema Mapping description, and Portrayal Information (Styles, Portrayal Rules, Graphics and Portrayal Catalog), Layer, and Map Context. The Semantic Registry Service is used as an integration solution for federating and unifying information produced by different OGC Catalog Service information by providing a simplified access through hypermedia-driven API, using JSON-LD, Linked Data and HAL-JSON. During the Testbed, the Semantic Registry was used to store information about geospatial datasets and services, schemas and portrayal information. The Semantic Mediation Service was used to validate and perform transformation between schemas, including transformation chaining. The Semantic Portrayal Service was used as a convenience API to access Portrayal Registry information and perform rendering of geospatial data to different graphic representation (SVG, Raster and other pluggable formats).

The Testbed-12 GeoPackage Routing and Symbology Engineering Report [http://docs.opengeospatial.org/per/16-029r1.html] describes the results of experiments to perform routing and styling on a GeoPackage. It includes a proposed approach for persisting static and dynamic styling information inside a GeoPackage.

The OGC Testbed-13: Portrayal ER [http://docs.opengeospatial.org/per/17-045.html] captures the requirements, solutions, models and implementations of the Testbed-13 Portrayal Package. This effort leverages the work on Portrayal Ontology development and Semantic Portrayal Service conducted during Testbed-10, -11 and -12. The objective of Testbed-13 was to identify and complete the gaps in the latest version of the portrayal ontology defined in Testbed-12, complete the implementation of the Semantic Portrayal Service by adding rendering capabilities and perform a demonstration of the portrayal service that showcases the benefits of the proposed semantic-based approach.

This Testbed-14 task should extend the ontology to accommodate more complex symbols such as composite symbols and symbol templates to describe more advanced symbology standards such as the family of MIL-2525 symbols.

B.13.1. Work Items Overview

The following figure illustrates the *Portrayal* work items.

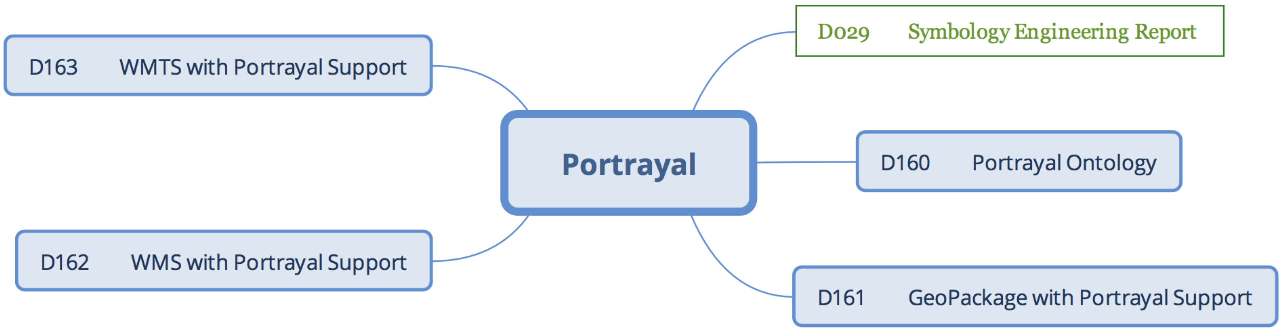

D163 WMTS with Portrayal Support
D029 Symbology Engineering Report
Portrayal
D160 Portrayal Ontology
D162 WMS with Portrayal Support
D161 GeoPackage with Portrayal Support

Figure 27. Portrayal work items

B.13.2. Deliverables

The following list identifies all deliverables that are part of this task. Additional requirements are stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D029 Symbology Engineering Report** - Engineering Report capturing all results and experiences of this task with particular emphasis on the topics shown in the Portrayal work items figure. Goal is to provide clear recommendations on how to best proceed with portrayal information encodings.
- **D160 Portrayal Ontology** - Extend the portrayal ontology developed in previous Testbeds to represent composite symbols and symbol templates.
- **D161 GeoPackage with Portrayal Support** - Implementation of GeoPackage that implements the approach recommended in D029.
- **D162 WMS with Portrayal Support** - Implementation of WMS instance that implements the approach recommended in D029.
- **D163 WMTS with Portrayal Support** - Implementation of WMTS instance that implements the approach recommended in D029.

B.14. MapML

Natural Resources Canada's responsibility is to foster an environment for the development and use of Spatial Data Infrastructures (SDI) both within Canada and internationally. This task is specifically concerned with improving SDI usability by non-traditional geospatial content users and producers including mainstream geospatial users, Web developers and Web browsers' trunk.

Standards are the backbone of every important technology in society. Maps and spatial data are no different. The central challenge for standards-makers is to develop standards which are suitable for the target 'audience'. As an example, consider electricity. While electricity, and the scientific and technical concepts behind its safe and effective generation, transmission and usage is beyond the knowledge of most people, it is also true that most people benefit from being able to easily use electricity. Most electrical appliances are built to consume electricity which conforms to standards at many levels, right up to use by consumer appliances via plugs and sockets. Information is not substantially different from electricity in many respects, in particular with respect to the need for standards.

Today, there exists a global network of hardware and software devices which produce and consume information of virtually any type: Hyper Text Transfer Protocol (HTTP) servers and Hyper Text Markup Language (HTML) browsers. Further, a huge supply of map and spatial content, in the form of Spatial Data Infrastructures (SDIs) already exists, created through the sustained and substantial investments of taxpayers across the globe. The objective of designing Map Markup Language is to improve integration of SDIs with the global HTTP application network (aka the "World Wide Web", or just "the Web") by strategically applying the architectural style with which the Web was designed: hypertext and media types. Enabling existing Spatial Data Infrastructures to be used with Map Markup Language will facilitate maps and spatial information to more readily fulfill their essential role in addressing society's problems and issues both large and small, from global to local, through the ubiquitous and standard information medium of the Web. Natural Resources Canada believes that better integration of these two global information management resources will result in optimal spatial information use and re-use, and consequent improved return on investment for global SDIs.

The OGC Testbed-13: MapML Engineering Report [http://docs.opengeospatial.org/per/17-045.html] discusses the approach of Map Markup Language (MapML) and Map for HyperText Markup Language (Map4HTML) described in: https://github.com/Maps4HTML and supported by the community in https://www.w3.org/community/maps4html/. The objective of this Engineering Report is to explore how MapML can be harmonized with the OGC standards mainstream and contribute to the progress of the specification avoiding unnecessary duplication. In particular, the ER proposes Web Map Service (WMS) or Web Map Tile Service (WMTS) as services that can be used to deliver MapML documents with small modifications. Another consideration on the ER is the inclusion of the time dimension and directions operation in MapML.

Testbed-14 shall address the following topics:

1. Specify and implement adaptations or integration of OWS services (WMS, WFS,

WMTS) serving MapML.

2. Collect OGC community feedback in the form of github pull requests to the appropriate github:Maps4HTML/repository. Topics to be addressed by such pull requests include, but are not limited to new normative and non-normative sections or updates to existing sections about:
    a. Tiled Coordinate Reference System (TCRS) definitions
    b. Image georeferencing markup
    c. TCRS/projection negotiation
    d. Language negotiation
    e. Security considerations
    f. Hyperlinking within and between map services
    g. Accessibility considerations for map feature markup
    h. Microdata / microformat semantic markup recommendations
    i. Feature creation / update / deletion sections via PUT, POST, DELETE considerations
    j. Caching discussion
    k. Extent processing discussion / algorithm
    l. Feature styling with Cascading Style Sheets
3. Define conceptual and security models, including "panes" (for example for non-spatial content, such as legends, notes, graphics and other standard map meta-information) and JavaScript API for map documents which are accessed as the primary resource in the manner of SVG documents so accessed. i.e. a MapMLDocument
4. Implement an extension to one open source Web browser engine, such as Blink or Gecko, which implements the new <map> element, supporting the text/mapml media type.
5. Define hypertext forms of common map idioms such as those supported by the Open Geospatial Web Services (OWS) service APIs (features, map images, map tiles, dimensions, styles, etc.). Such extensions would increase the availability of MapML by practically specifying (backwards-compatibly) how future OWS could serve MapML, so that a greater proportion of existing Web map / spatial content is available for use by the HTML <map> element to the Web author community.
6. Define and implement the HTML <map> element API and event model as a browser extension, to make programming maps on the Web an exercise in the application of browser-supported standards.
7. Create a cloud-based service to proxy existing CGDI OWS content services [20] as

MapML, potentially hosted at https://maps4html.org/cgdi/

8. Evaluate support for vector tiles in the client HTML <map> element
9. Evaluate support for ‘offline’ MapML maps, using the ServiceWorker [https://developer.mozilla.org/en-US/docs/Web/API/Service_Worker_API] browser API + OGC GeoPackage [http://www.geopackage.org] MapML extension.
10. Define layer grouping / styling and animation in MapML
11. Investigate and implement semantic markup integration, parsing and display using HTML/Microdata + schema.org + OGC Simple Features.

All aspects shall be explored in a **MapML Extension Study** that shall be documented in the MapML Engineering Report (D012).

On the implementation side,

- a **Browser Extension** shall be define and implement the HTML <map> element API and event model as a browser extension, to make programming maps on the Web an exercise in the application of browser-supported standards.
- **Cloud Based Proxy** shall be created to proxy existing CGDI OWS content services [http://www.nrcan.gc.ca/earth-sciences/geomatics/canadas-spatial-data-infrastructure/19359] as MapML, potentially hosted at https://maps4html.org/cgdi/. The proxy shall support vector tiles in the client HTML <map> element and ‘offline’ MapML maps, using the ServiceWorker browser API [https://developer.mozilla.org/en-US/docs/Web/API/Service_Worker_API] together with the OGC GeoPackage [https://en.wikipedia.org/wiki/GeoPackage] MapML extension.

B.14.1. Work Items Overview

The following figure illustrates the *MapML* work items.

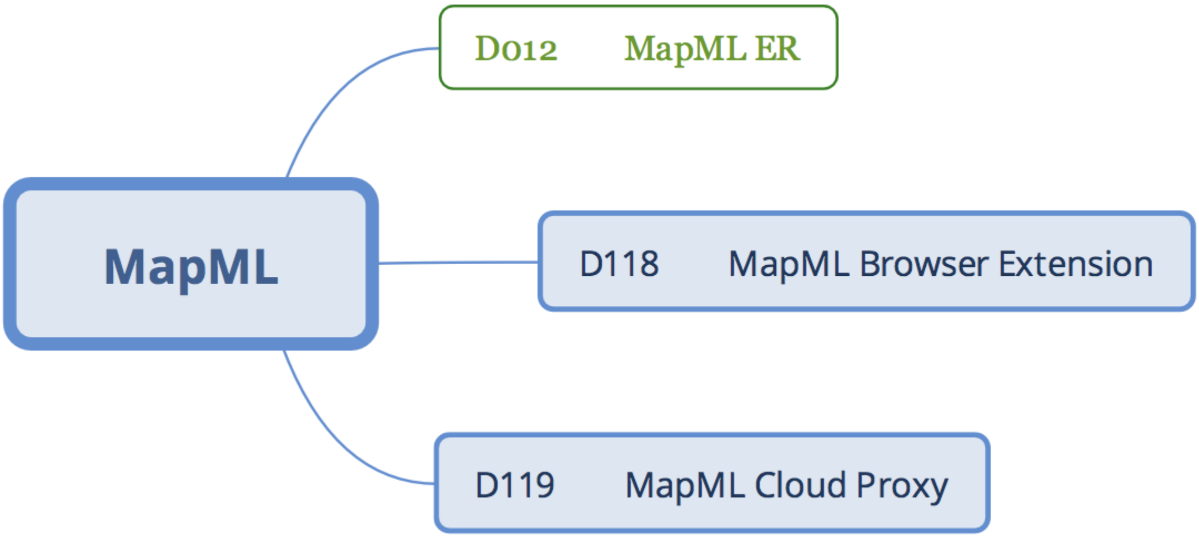


Figure 28. MapML work items

B.14.2. Deliverables

The following list identifies all deliverables that are part of this task. Additional requirements are stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D012: MapML ER** - This Engineering Report captures all results of this task. It shall include the use cases, implementations, best practices, experiences and results. It covers all solutions and implementation scenarios of this task and shall reference change requests as necessary. Particular emphasis shall be on the results of the *MapML Extension Study*.
- **D118: MapML Browser Extension** - Define and implement the browser extension as described above.
- **D119: MapML Cloud Proxy** - Define and implement the proxy service as described above.

B.15. Quality of Service & Experience

Natural Resources Canada's responsibility is to foster an environment for the development and use of Spatial Data Infrastructures (SDI) both within Canada and internationally. This task is specifically concerned with improving SDI Web Mapping Services.

Natural Resources Canada leads a Government of Canada initiative called the Federal Geospatial Platform (FGP). The FGP is the online environment for Government of Canada employees to find, combine and view Canada's geospatial data on a map, to support evidence-based decision making. The FGP provides proactive, whole-of-government leadership in geomatics and Earth observations to better support government priorities and presents a collaborative partnership of 21 federal government departments. The FGP is part of Canada’s Open Government Action Plan; as such the public-facing version of FGP is the Open Government Portal Open Maps web site.

The ability to combine and visualize location-based data using web mapping services is a key value proposition of the Federal Geospatial Platform. The FGP includes the capability of selecting datasets with OGC web mapping services to view individually or combined with other services in the FGP viewer. This functionality is provided to allow users to immediately visualize and analyze geospatial data. Unfortunately, user feedback has proven that these geospatial web services are not always easy or intuitive to navigate, combine or understand. Because the FGP’s primary end users are government policy analysts who are not always highly familiar with mapping and web mapping technologies, it is important to make web mapping services, and the content they make available, as user-friendly as possible.

In 2016, to help alleviate this issue, the FGP developed a web map service quality assessment methodology, ran an assessment and developed recommendations and best practices to support a user friendly experience for all employees and citizens using web map services. Assessments to date have shown that key considerations are often very simple, but very impactful on quality of experience. The results of this study were used as the primary input into an OGC Discussion Paper created by the Quality of Service Experience Domain Working Group (QoSE-DWG).

The OGC QoSE-DWG [http://external.opengeospatial.org/twiki_public/QualityOfService/WebHome] have developed a discussion paper (OGC 17-049) entitled "Ensuring Quality of User Experience with OGC Web Mapping Services" [https://portal.opengeospatial.org/files/?artifact_id=74403&version=1] that identifies and describes issues often encountered by users of OGC web mapping services that affect the quality of their experience, and also provides an assessment framework for identifying issues and measuring quality, along with potential solutions and guidance to improve the usability of services.

The assessment framework for measuring quality of experience and the associated recommendations for improving service quality are intended to benefit human end-users who need to rapidly assimilate and use web mapping visualizations to answer questions or input into analysis. In other words, they need to be able to make sense of the information OGC web mapping service provides them.

In addition, Testbed-13 addressed Quality of Service aspects in the aviation domain. Though specific to a particular domain, The Testbed13: Data Quality Specification Engineering Report [http://docs.opengeospatial.org/per/17-018.html] addressed a number of general aspects that apply to this task nevertheless.

Testbed-14 shall address the following Web Mapping Services Usability aspects:

1. **Develop a revision of the OGC Discussion Paper 17-049**: Develop a revision of the OGC Discussion Paper 17-049 [https://portal.opengeospatial.org/files/?artifact_id=74403&version=1] as per the review and assessment of the current OGC Discussion Paper 17-049; the OGC Arctic Spatial Data Pilot: Phase 2 Report [http://www.opengeospatial.org/pub/ArcticSDP/r2/er.html] (OGC document 17-068); the OGC – NRCan Cartography Concept Development Study report that will be concluded at the end of March 31, 2018. Other relevant OGC working groups and web mapping service quality reports may also need to be consulted. This revision shall be performed in coordination with OGC QoSE DWG to address potential new requirement such as quality of Service metadata (Capabilities Extensions) server implementation and monitoring client implementation that allows improved service quality and measurement. The revision shall be part of D011.

2. **Quality of Service Experience Assessment Framework**: The fourteen assessment criteria described in Discussion Paper 17-049 are all aimed at assessing the quality of a Web service in terms of the degree to which it conveys clearly understood information to the user. The user is assumed to be a non-expert in geospatial Web services, but in most cases the criteria are equally valid for all classes of users. A selected number of services in the Testbed will be assessed against the QoSE assessment criteria, and retested once recommendations have been applied. Comparison of results will help to validate the effectiveness of the assessment criteria and the corresponding recommendations to improve usability, and allow for feedback, improvement or correction. Services are provided as D115, a dedicated client as D116. Bidders for D115 shall be prepared to setup multiple WMS instances. Results shall be captured in D011.

3. **Quality of Service Experience Practices to Alleviate Usability Issues**: The fourteen assessment criteria all have corresponding recommendations that describe practices that, once implemented, should help to alleviate user confusion and improve the usability of web mapping services as a means of visualizing and simple visual or query-based analysis of geospatial data. Once selected services have been assessed against assessment criteria, all or as many QoSE recommendations will be applied, then services will be subject to retesting.

4. **Quality of Service Experience Implementing Best Practices: Man or Machine?**: Analysis required here on the making of a geospatial Web mapping service and the opportunities of human operator or programmatic response to make decisions that impact service quality of experience. Assessment of which usability issues are determined/caused by human input or what determined/caused by default via the programmatically generated aspects of the service (not due to direct human input or decisions) via the standard specification implementation. Results shall be captured in D011.

5. **Quality of Service Experience Test Suite**: Develop test suite to programmatically test and validate that best practices have been successfully implemented. The test suite will automatically assess the quality of service according to the assessment criteria and validate or flag services that do not comply to best practices/recommendations. The

results of the test suite (D117) shall be captured in D011.

An OGC Concept Development Study is about to be kicked off December 2017 with results expected in March 2017. The study will address concerns raised that cartography suffers in distributed Web services environments and the poor presentation of data creates a negative perception of the data quality, spatial data infrastructures and Web standards.

The current domain of Web based map driven display often ignores the fundamental concepts of basic cartography. The developers of the Web based map tools are often programmers with little cartographic experience, training or guidance. A major problem exists: A haphazard use of colors, symbols, patterns, shapes and sizes are used to represent similar features. One attempt to help resolve the "cartography gap" was the development of an Open Geospatial Consortium standard that developers could use, called Styled Layered Descriptor (SLD) with a library of styles that web developers could use. The OGC Architecture Board (OAB) recently identified the renewal of SLD as a key priority. The OAB took an action to promote a Portrayal ad-hoc meeting during the June 2017 Technical Committee meetings. The agenda included a range of use cases simple to complex; community efforts. The need for a conceptual model was identified as a method to move forward.

The proposed work for this Concept Development Study is in three major phases:

1. The first phase is to do an in-depth review of the good work that has already occurred with web based symbology to assess what has and what has not worked.
2. The second phase is to consult with implementers and report on two case studies
   - The Arctic Spatial Data Infrastructure combines data from 8 Arctic Countries and their respective cartographies and systems.
   - The Canadian Federal Geospatial Platform combines data from Canadian Federal Government Departments and is now working to include provincial data. Federal portals tend to occupy a hybrid SDI space between data production environments, internal user requirements, Government Open Data Portals and international SDIs, each with unique cartographic issues.
3. The third phase is to provide a report and presentations to make a series of recommendations in a report.

Testbed-14 shall consider this report as an additional resource in the context of *Quality of Service & Experience* of Web mapping.

B.15.1. Work Items Overview

The following figure illustrates the *Quality of Service & Experience (QoSE)* work items.

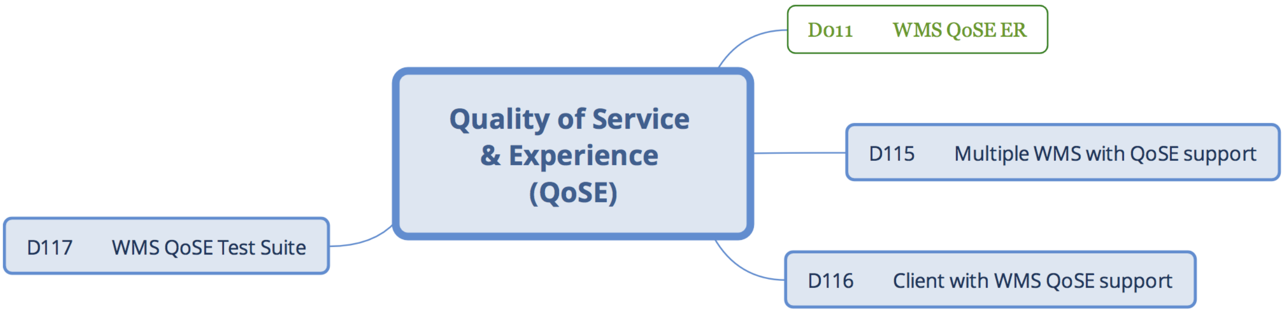

Quality of Service & Experience (QoSE)
D011 WMS QoSE ER
D115 Multiple WMS with QoSE support
D116 Client with WMS QoSE support
D117 WMS QoSE Test Suite

Figure 29. Quality of Service & Experience (QoSE) work items

B.15.2. Deliverables

The following list identifies all deliverables that are part of this task. Additional requirements are stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D011 WMS QoSE ER** - This Engineering Report captures all results of this task.
- **D115 Multiple WMS with QoSE support** - At least two WMS instances to test the assessment criteria.
- **D116 Client with WMS QoSE support** - Client implementation dedicated to test the assessment criteria at WMS instances.
- **D117 WMS QoSE Test Suite** - Implementation of the Test Suite as described above.

B.16. Compliance and Interoperability Testing

The OGC Compliance Program [http://cite.opengeospatial.org/welcome] provides the resources, procedures, and policies for improving software implementations' compliance with OGC standards. The Compliance Program provides an online free testing facility, a process for certification and branding of compliant products, and coordination of a vibrant community that develops and supports test scripts.

To further enhance the CITE tests and compliance analysis engine, Testbed-14 shall address topics illustrated in the figure below.

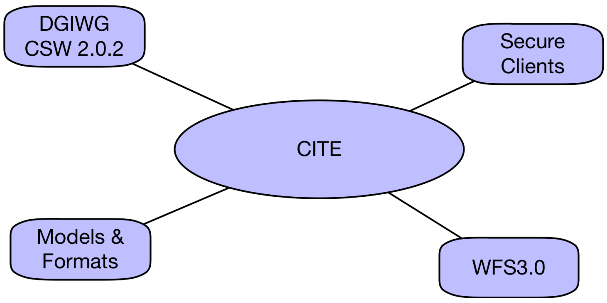


Figure 30. Machine Learning, Deep Learning & Artificial Intelligence deliverables

B.16.1. DGIWG CSW 2.0.2

Participant shall review and analyze the latest versions of DGIWG/NSG Profiles of CSW 2.0.2 and develop both compliance tests and a reference implementation.

B.16.2. Secure Clients

Testbed-14 shall support the implementation of the new Web Services Security Standard [https://portal.opengeospatial.org/files/?artifact_id=76516&version=1]. The standard specifies how conformable OWS Services shall advertise their Information Assurance (IA) Controls, describes the governance process for the IA Control registers, examples of register contents, and descriptions on how this information should be used. Next, this standard defines conformance classes and requirements classes to be used for reaching compliance and their validation via conformance tests. Finally, the standard defines client behaviour to ensure interoperable processing of advertised security controls.

The Web Services Security Standard defines an annotation mechanism for Capabilities documents or responses to the *GetCapabilities* request, ensuring interoperability between a secured OGC Web Service instance and a client application. It further overrides existing HTTP protocol limitations and exception handling from existing OGC Web Services standards for the purpose to achieve interoperability with main stream IT security standards and their implementations.

Testbed-14 shall develop a client validation engine (client validator). This validator shall test compliance of software products (client software) with Web Services Security, chapter 8, for two use cases:

- Test whether a client can operate in compliance with the standard according to use case I of the standard (section 5.2). A server that only does server side authentication using HTTPS but no client authentication or authorization. It can be tested for one OGC service (preferably WMS or WFS)
- Test whether a client can operate in compliance with the standard according to use case II of the standard (section 5.3). A server that has an authentication method in place for client authentication. It can be tested for one OGC service (preferably WMS or WFS) with authentication according to requirement class OpenID Connect (section 7.11.2)

Results shall be captured in a Testbed-14 Engineering Report (ER). The ER shall make recommendations to the Web Services Security Standard based on the experiences made while developing the validator. Testbed-14 participants shall put special emphasis on how to extend the Web Services Security Standard to allow proper specification of different authentication methods. These recommendations shall provide a solid foundation that can be used in future initiatives to develop software clients that test support for most of modern security/authentication methods.

B.16.3. WFS3.0 Compliance

All OGC standards require a description of the conformance test approach and an Abstract Test Suite. The existing WFS version 2.x conformance test approach is unlikely to be sufficient for the new WFS version 3.0. Therefore, the Next Generation OGC Web Services, Federated Clouds, Security & Workflows WFS3.0 developers together with the *D021 Secure Resource Oriented Geospatial Data Handling Engineering Report* editor shall work in cooperation with the OGC CITE development Team and the Compliance and Interoperability Testing task participants on an appropriate conformance test approach for Resource Oriented services. Though services like WMS and WCS are addressed in D021 on a rather theoretical level, robust WFS3.0 tests shall be developed in Testbed-14.

B.16.4. Models & Formats

OGC has begun to embrace JSON and JSON-LD encoding of data. JSON does not have rigorous validation tools such as XML Schema and Schematron. Testbed-14 shall perform an analysis of alternatives and recommend options for tools and techniques to validate JSON and JSON-LD encodings to the same level as is possible with XML.

In the context of linked data, OGC Testbed-12 addressed the conversion of Geospatial-Intelligence (NSG) Application Schema (NAS) to an ontology (using RDF(S)/OWL/SKOS). This work shall be continued in Testbed-14 by an analysis of RDF-based data validation options

---

In detail, Testbed-14 shall address and implement the following aspects:

1. Develop or update compliance tests and a reference implementation for the DGIWG Profile *DGIWG CAT 2.0.2*.
2. OGC has begun to embrace JSON and JSON-LD encoding of data. JSON does not have rigorous validation tools such as XML Schema and Schematron. Perform an analysis of alternatives and recommend options for tools and techniques to validate JSON and JSON-LD encodings to the same level as is possible with XML.
3. Perform an analysis of alternatives and recommend options for tools and techniques to validate NEO-conformant RDF-based geospatial data. The NEO is the U.S. National System for Geospatial Intelligence (NSG) Enterprise Ontology [http://www.dsp.dla.mil/Portals/26/Documents/Publications/Journal/161001-DSPJ-04.pdf]. The work shall take results of the Testbed-12 ShapeChange Engineering Report [http://docs.opengeospatial.org/per/16-020.html] into account, which addressed the conversion of the Geospatial-Intelligence (NSG) Application Schema (NAS) to an ontology (using RDF(S)/OWL/SKOS).

4. The OGC Next Generation task will develop a new generation of OGC services based on a Resource-Oriented Architecture pattern. This effort is a complete white-sheet re-development of WFS capabilities using modern REST and API practices. This new generation of services will require a new generation of compliance tests. This CITE thread will work with the Next Generation OGC Web Services, Federated Clouds, Security & Workflows task to develop both the test approach and initial compliance tests for WFS 3.0.

5. Testbed-14 shall develop a client validation engine (client validator) and corresponding tests. This validator shall test compliance of software products (client software) with Web Services Security, chapter 8, for the use cases described above.

B.16.5. Work Items Overview

The following figure illustrates the *Compliance and Interoperability Testing* work items.

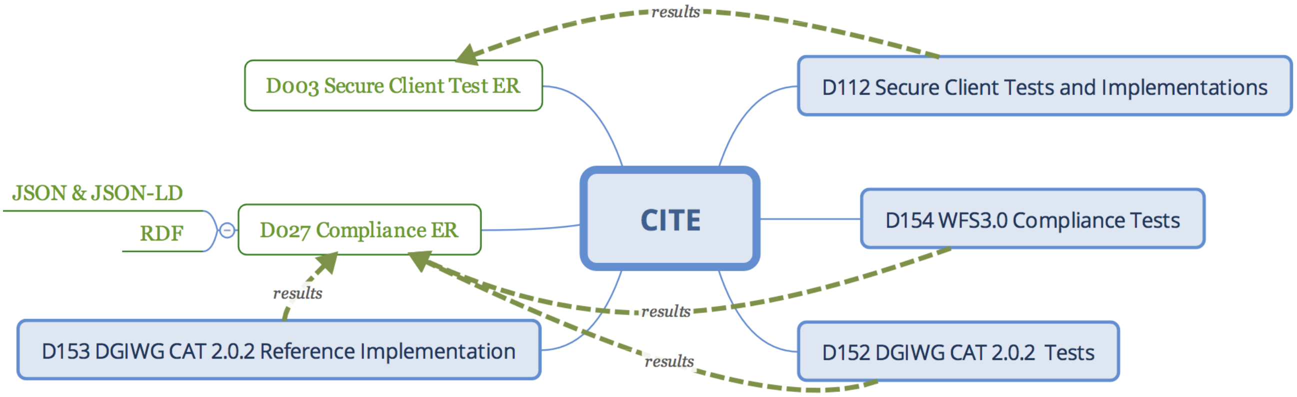

results
D003 Secure Client Test ER
D112 Secure Client Tests and Implementations
JSON & JSON-LD
RDF
D027 Compliance ER
CITE
D154 WFS3.0 Compliance Tests
results
results
results
D153 DGIWG CAT 2.0.2 Reference Implementation
D152 DGIWG CAT 2.0.2 Tests

Figure 31. Machine Learning, Deep Learning & Artificial Intelligence deliverables

B.16.6. Deliverables

The following list identifies all deliverables that are part of this task. Additional requirements may be stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D003 Secure Client Test ER** - This Engineering Report captures all results and experiences of the implementation of D112.

- **D112 Secure Client Tests and Implementations** - Client validation engine and tests to address requirements stated above.

- **D154 WFS3.0 Compliance Tests** - Test approach and initial compliance tests for WFS 3.0.

- **D152 DGIWG CAT 2.0.2 Tests** - Test suite for CSW 2.0.2 compliant to the DGIWG profile.

- **D153 DGIWG CAT 2.0.2 Reference Implementation** - Reference implementation of CSW 2.0.2 compliant to the DGIWG profile. Shall be tested with D152.
- **D027 Compliance ER** - Engineering Report capturing all experiences and results of that task that are not documented in D003. Provides a short summary and references to D003 sections nevertheless.

IMPORTANT The following task Exploitation Platform is included in this CFP for the sake of completeness! Special conditions apply for proposals for this task! Please review the Main Body carefully!

B.17. OGC Testbed-14 – ESA Sponsored Threads – Exploitation Platform

B.17.1. Technical Architecture

Making arbitrary applications available on cloud infrastructures or exploitation platforms in a standardised way is a key technology for Big Data in general and particularly true for Earth Observation satellite data processing. When the transport of large amounts of data is not feasible or simply not cost efficient, processes need to be shipped and executed as closely as possible to the actual data.

ESA has developed the concept of Earth Observation (EO) Exploitation Platforms [https://tep.eo.esa.int/about-tep]. "In short, an EO exploitation platform is a collaborative, virtual work environment providing access to EO data and the tools, processors, and Information and Communication Technology resources required to work with them, through one coherent interface. As such the Exploitation Platform may be seen as a new ground segments operations approach, complementary to the traditional operations concept." (ESA [https://tep.eo.esa.int/about-tep]) The concept has been implemented in 7 concrete Thematic Exploitation Platforms [https://tep.eo.esa.int], TEPs.

Testbed-13 experimented with different approaches that allow packing any type of application or multi-application based workflow into a container that can be dynamically deployed on any type of exploitation platform (which can, but does not need to be hosted in a cloud). Consumers can discover these containers, provide the necessary parameterisation and execute them even easier than on their local machines, because no software installation, data download, or complex configuration is necessary. Instead, these containers will be executed on exploitation platforms such as the ESA TEPs.

→ TEP COASTAL
→ TEP FORESTRY
→ TEP GEOHAZARDS
→ TEP FOOD SECURITY
→ TEP HYDROLOGY
→ TEP POLAR
→ TEP URBAN
Wouldn't it be nice if … I didn't need to be an ICT wizard or instrument expert to integrate different data into my research or application?

Figure 32. ESA Thematic Exploitation Platforms (source: ESA [https://tep.eo.esa.int/about-tep])

An application package descriptor has been developed in Testbed-13 that allows the application developer to provide all necessary information to register an application at standardised Web services; in this case OGC Web Processing Services (WPS). The application itself is containerised in a Docker container and provided on a Docker Hub. On demand, the container gets dynamically deployed and executed on the platform that hosts the data. Once processing is completed, results are accessible at standardised data access interfaces.

Testbed-14 shall continue the work of Testbed-13, which is documented in the Engineering Reports OGC Testbed-13: EP Application Package ER [http://docs.opengeospatial.org/per/17-023.html], OGC Testbed-13: Application Deployment and Execution Service ER [http://docs.opengeospatial.org/per/16-024.html], and OGC Testbed-13: Cloud ER [http://docs.opengeospatial.org/per/17-035.html]. These Engineering Reports are publicly available once the electronic voting period is completed (towards the end of December 2017). Preliminary copies (be aware that the final versions might deviate slightly, thus bidders are advised to check the URLs given above regularly!) are available at the following URLs:

- OGC Testbed-13: EP Application Package ER (preliminary version [https://portal.opengeospatial.org/files/?artifact_id=76996])
- OGC Testbed-13: Application Deployment and Execution Service ER (preliminary version [https://portal.opengeospatial.org/files/?artifact_id=76997])
- OGC Testbed-13: Cloud ER (preliminary version [https://portal.opengeospatial.org/files/?artifact_id=76998])

As described in the Engineering Reports, Testbed-13 experimented with different options to describe for example the mounting of external data to Docker containers, discussed different elements to be included in the Application Package itself etc. During the first third of the project, Testbed-14 participants shall develop a clear recommendation on the various aspects left open by Testbed-13. These recommendations are the base for the workflows and implementations described in the scenarios below. It is emphasised that this is a consensus process that needs to be supported by all participants.

Other Testbed-13 Engineering Reports provide further insights into relevant aspects, such as e.g. the Security Engineering Report [http://docs.opengeospatial.org/per/17-021.html]. All Engineering Reports are publicly available at http://docs.opengeospatial.org/per/.

B.17.2. Testbed-14 Focus

Focus of Testbed-14 is on client-platform interaction patterns and mechanisms to allow for secure workflows that can be dynamically deployed and executed.

Testbed-14 shall implement a set of scenarios defined further below. These scenarios mimic a possible real world situation with several external Exploitation Platforms being involved. The scenarios are described using the Mission Exploitation Platforms (MEP) for Sentinel-2, Deimos, and Proba-V as well as the Thematic Exploitation Platform (TEP) "Agriculture". As key conditions, Testbed-14 shall demonstrate the usage of

- different Mission and Thematic Exploitation Platforms
- different types of users, one being an application developer, the other a consumer
- different applications where one application uses the results of the other
- standardised error handling and user messaging

The following paragraphs outline a context for possible implementation scenarios. It is emphasised that the final implementation scenarios will be defined at the Testbed-14 kick-off meeting. Alternative suggestions from participants are appreciated as long as the key conditions stated above are fulfilled.

B.17.3. Context

Testbed-14 shall implement two different types of users. One is the developer of the process, the other the consumer. The *Authorisation for platform usage* includes the rights to deploy and execute applications and use of data hosted on platform or resulting from application processes. User rights differ and both user types shall have the minimum set of rights necessary to either deploy and test applications or to consume them.

User	Role	Authorisation for platform usage
Alice	Application Developer	Agriculture TEP

User	Role	Authorisation for platform usage
Bob	Application Consumer	Agriculture TEP, Sentinel-2, Deimos, Proba-V MEP, and Pacific Forestry Center (PFC) Boreal Cloud

Table 3. User Roles

Testbed-14 shall implement two different demonstration applications. Testbed-14 participants shall provide these applications. The applications described in this document are exemplars and can be modified by the participants if the general flow of information survives, including in particular the access to various cloud platforms and chaining of applications.

The first application shall calculate specific values using satellite imagery from different platforms/satellites. The second application shall make use of the results from the first application and integrate the first application's results.

Exemplarily, the first application is described here as a *MultiSensorNDVI application* that allows to calculate the *Normalized Difference Vegetation Index* (NDVI) using satellite imagery from various platforms. The second application integrates the results of the first one.

Application	Satellites
MultiSensorNDVI for the generation of NDVI from Earth Observation data generated by different satellites. The app allows to execute the NDVI calculations for an area and time of interest across different satellites.	* Sentinel-2 * Proba-V * Deimos
NDVIStacker which creates a pyramid of NDVI images obtained from different satellites. It requires a number of input files to produce a single output file.	* any
MultiSensorNDVIStackGenerator chains the two other applications together. It allows executing the MultiSensorNDVI app and ensures that all output is provided to the NDVIStacker as input. It does not add any other processing to the data.	* any

Table 4. Applications

The following platforms are available for Testbed-14. All ADES shall be able to deploy resources on these clouds. The platform-side provisioning of APIs is subject to the Testbed.

Platform	Host
TEP which addresses users of a specific community: Agriculture	Uses different cloud providers, e.g. AWS
Sentinel-2 MEP	IPT Poland
Proba-V MEP	Private cloud
Deimos MEP	Private cloud
AWS	Amazon cloud as test environment

Platform	Host
PFC	Pacific Forestry Centre Boreal Cloud

Table 5. Platforms

Security and Privacy Settings

For Testbed-14, all platforms belong to a single security domain, which is a simplification of the real world implementation in which the platforms belong to a federation of multiple security domains.

- Users can use the same credentials to access all the above mentioned platforms
- Each platform accepts requests (processing, deployment) on behalf of a user from any other associated platform
- There is a central Identity Provider (IDP) to manage user basic profile
- There is a central catalogue of all publicly visible data available (Data Central Catalogue)

Finally the goal when it comes to Identity Management is not to be integrated with ESA SSO [https://eo-sso-idp.eo.esa.int/idp/umsso20/admin] but to be able to integrate with and leverage on the eduGain federation [https://technical.edugain.org] (of federations) and any similar existing SAML2 federations. Ideally, participants address this aspect in their proposals.

Participants have to add authorisation capabilities to their EMS and ADMS implementations (i.e. policy enforcement points)

Each application has the following privacy settings and allowed values. Each *private* setting refers to the user owning the application:

- Visibility: private, public
- Execution: private, public
- Update/remove: private

The output of an application run is visible and accessible only to the user that launched the application.

Implementation Scenarios

Within the TEP it is possible to define an application as chaining of other applications. In this case the output of one application becomes the input of the next application in the chain. In the second scenario, Alice will define the MultiSensorNDVIStackGenerator application as a chain of the MultiSensorNDVI and NDVIStacker applications. The

MultiSensorNDVI automatically executes parallel runs on different platforms. Each run uses input data from a different satellite in the same Area of Interest (AOI) and Time of Interest (TOI). Each of the three runs produces a single file as result. The results of the three runs will then be used as input (passed by reference) to the NDVIStacker to create the pyramid.

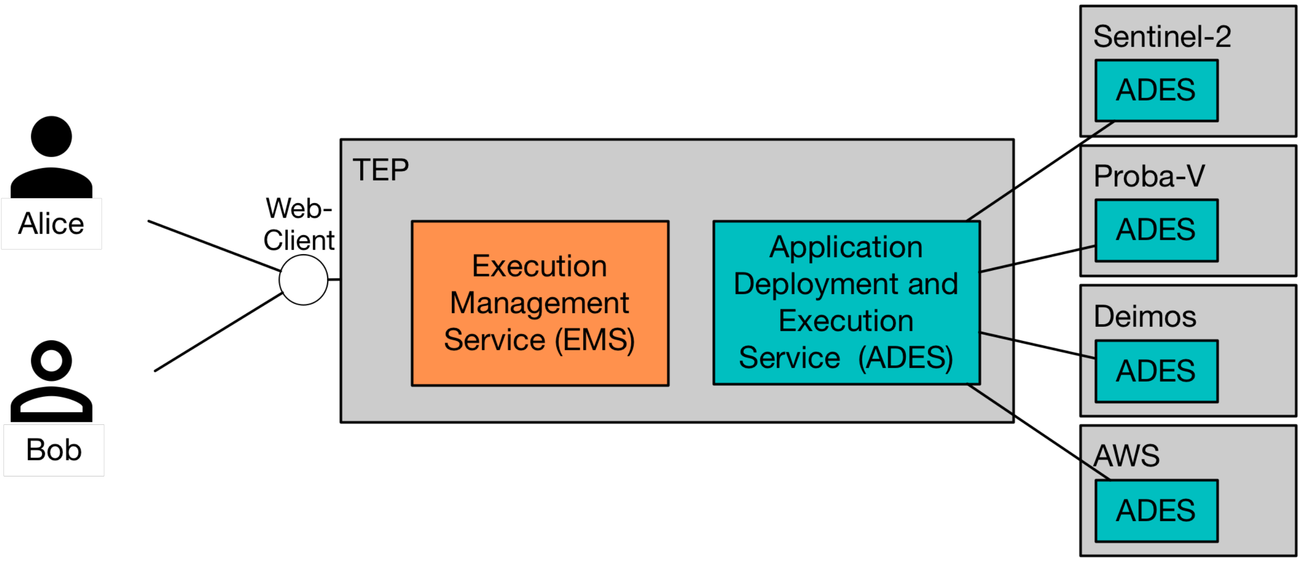

Alice
Bob
Web-Client
TEP
Execution Management Service (EMS)
Application Deployment and Execution Service (ADES)
Sentinel-2
ADES
Proba-V
ADES
Deimos
ADES
AWS
ADES

Figure 33. Overview Scenario. The TEP has an Execution Management Service (EMS) and an Application Deployment and Execution Service (ADES), which communicates with other ADES on Mission Exploitation Platforms

Both users, Alice and Bob, interact with the TEP through a Web client. The Web client shall support all necessary interaction steps as described below. The TEP itself contains two main components, the Execution Management Service (EMS) and the Application Deployment and Execution Service (ADES). The EMS includes most of the control logic, such as deployment and execution of applications to different MEPs (or other TEPs), and the orchestration/chaining of applications executed on MEPs/TEPs. The ADES is responsible for application deployment and execution on a specific platform. Both EMS and ADES shall support authentication and authorisation mechanisms to verify user credentials for any given request. Both EMS and ADES shall support standardised interfaces.

Scenario 1: Alice loads the application package for the MultiSensorNDVI and NDVIStacker in the TEP

In Scenario 1, Alice deploys the application packages for both applications. Alice has developed both application packages before. Both applications are available in Docker containers and uploaded to Docker hubs. The following figure illustrates the main steps of the scenario.

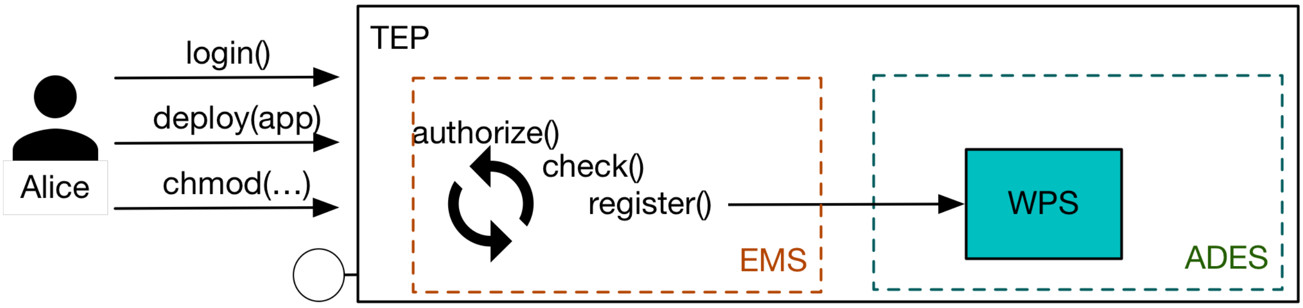


Figure 34. Scenario 1: Alice deploys applications on the TEP

a. Alice logs-in the TEP

b. Alice submits an application deploy request for the MultiSensorNDVI to the TEP via the TEP web client

c. The TEP checks that Alice is authorised to deploy applications on the platform

d. The TEP registers the application on the platform. The application default privacy settings are:

   - Visibility: private
   - Execution: private

e. The TEP links the application only to Alice's profile (as default).

f. Alice changes the privacy settings to:

   - Visibility: public
   - Execution: public

g. Optionally, the TEP registers the application at the Catalog service instance D150 as described in OGC Testbed-14 CFP section "B.7. Next Generation OGC Web Services, Federated Clouds, Security & Workflows", subsection "Workflows".

h. Alice repeats steps from b. to f. for the NDVIStacker

Scenario 2: Alice registers a basic chaining of applications on the TEP

After registering both applications on the TEP, Alice registers a third application on the TEP in Scenario 2. This third application makes use of the two previously registered applications.

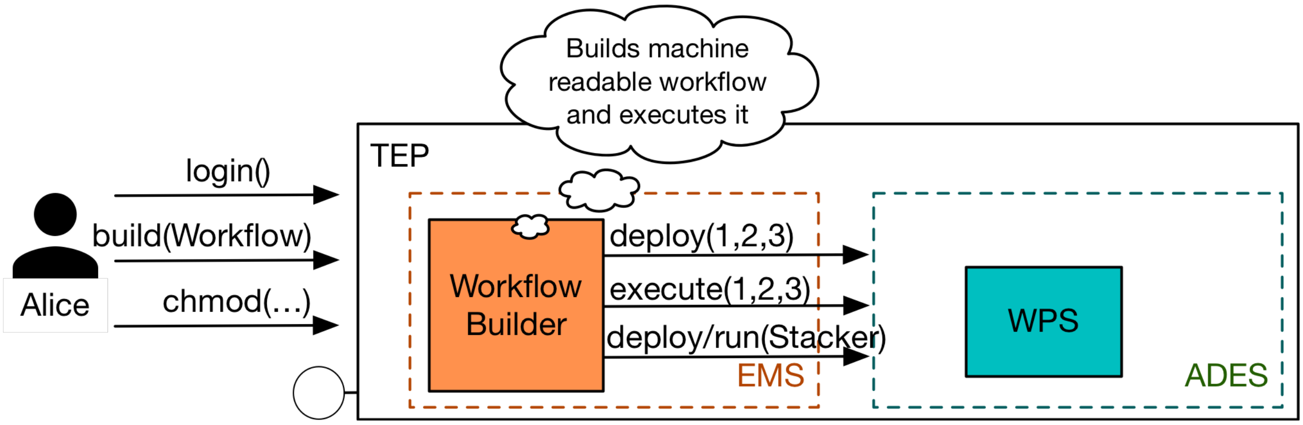


Figure 35. Scenario 2

a. Alice logs-in the TEP

b. The TEP checks that Alice is authorised to deploy applications in the platform

c. Alice defines a new application named "MultiSensorNDVIStackGenerator" as a simple chaining of the two previously deployed applications using a Web based tool provided by the TEP. This new application defines that

    1. MultiSensorNDVI shall run once on the three platforms:

        - Sentinel-2

        - Proba-V

        - Deimos

    2. NDVIStacker collects the output of the MultiSensorNDVI runs and produces the required stacked output. The Execution Management Service (EMS) is able to automatically link the output of the first application with the input of the other taking the relevant information from their application package. It creates a machine readable description of the application chaining which can be used to correctly execute the chain by the TEP.

d. The TEP registers the application on the platform. The application default privacy settings are:

    - Visibility: private

    - Execution: private

e. The TEP links the application only to Alice's profile (as default).

f. Alice changes the privacy settings to:

    - Visibility: public

    - Execution: public

g. Optionally, the TEP registers the application at the Catalog service instance D150 as described in OGC Testbed-14 CFP section "B.7. Next Generation OGC Web Services, Federated Clouds, Security & Workflows", subsection "Workflows".

Scenario 3: Bob runs MultiSensorNDVIStackGenerator

Scenario 3 follows the perspective of a typical user, Bob. Bob explores all registered and public applications, selects the chaining application MultiSensorNDVIStackGenerator, identifies relevant data using catalogs, runs the application, and eventually accesses the final product.

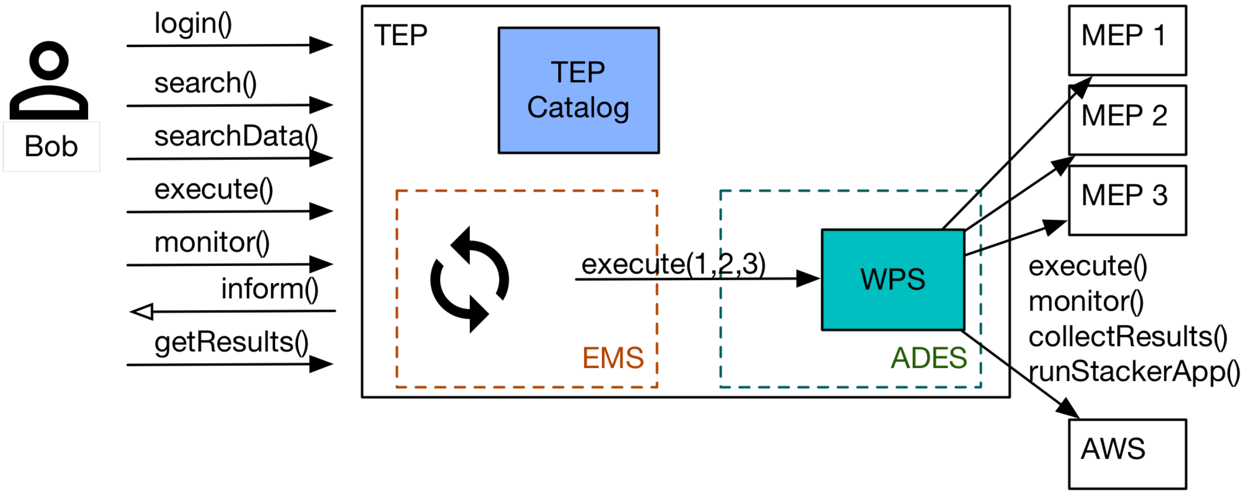


Figure 36. Scenario 3

a. Bob logs-in the TEP

b. Bob explores applications that are available on the TEP

c. Bob selects the MultiSensorNDVIStackGenerator application

d. Bob uses the TEP catalogue for Sentinel-2, Proba-V and Deimos data and selects those to provide as input to the application

e. Bob requests the execution of the MultiSensorNDVIStackGenerator application

f. The TEP checks that Bob is authorised to execute the MultiSensorNDVIStackGenerator application

g. Bob monitors the MultiSensorNDVIStackGenerator application execution via a Web interface provided by the TEP

h. The TEP performs an application deployment request in each of the Sentinel-2, Proba-V and Deimos MEPs for the MultiSensorNDVI application on behalf of Bob.

i. Each MEP checks that Bob is authorised to its usage

j. The TEP performs an application execution request in the Sentinel-2, Proba-V and Deimos MEPs for the MultiSensorNDVI application on behalf of Bob.

k. Each MEP checks that Bob is authorised to execute the application and use the needed data

l. The MultiSensorNDVI application execution starts on each TEP.

m. The TEP monitors the execution of the applications on behalf of Bob.

n. When all the executions of the MultiSensorNDVI application are completed, the TEP collects the output reference of each of the runs.

o. The TEP performs an application deployment request in PFC of the NDVIStacker application on behalf of Bob

p. TEP components deployed in AWS checks that Bob is authorised to its usage.

q. The TEP performs an application execution request in AWS for the NDVIStacker application on behalf of Bob, passing as input the output of the 3 runs of MultiSensorNDVI application.

r. AWS checks that Bob is authorised to execute the NDVIStacker application and use the needed data

s. The NDVIStacker application execution starts

t. The output of the NDVIStacker is produced and made accessible only to Bob

NOTE	This scenario could be possibly extended verifying that a user missing the authorisation to use a platform is blocked in running the application on it. AWS shall be used as an additional platform in this context.

Architectural Constraints

1. Each of the platforms involved in the scenarios (Sentinel-2, Proba-V, Deimos, AWS, PFC) shall have an Application Deployment and Execution Service (ADES) which is in charge of:
    - Application registration/deployment in the platform
    - Application Execution
    - Authorising users to register, execute and modify an application
2. Each participant providing a Execution Management Service (EMS) or Application Deployment and Execution Service (ADES) shall embed this service in a TEP that shall be provided by the participant. Both services can be part of the same TEP.
3. At TEP level there is an Execution Management Service (EMS) which is in charge of
    - Dispatching the application execution to the relevant MEP/Execution Platform
    - Orchestrate a basic chaining of an application
4. The application package shall have two facets:
    - Facet 1 has to target application developers who are not IT experts (e.g. scientists).

It has to have a simple encoding with the bare minimum information required to deploy application in an automated way and understand the inputs and outputs. The exploration of the use of YAML or JSON for such encoding is encouraged.

  - Facet 2 has to be used for M2M application package exchange. It has to contain all the information necessary to ensure that the application will behave in the same expected way in any platform supporting it. For instance it should contain information about privacy settings.

5. The application package shall support 2 types of applications:
   - Interactive application (e.g. those with a GUI)
   - Not-interactive applications (e.g. data batch processing)
6. All M2M interfaces shall use RESTful architecture with JSON encoding
7. All implementations shall be released as Free and Open Source Software (FOSS)
8. Authorisation shall be based only on the attributes in the user profile (i.e. those passed from the IDP to the Service Provider during user authentication). A possible minimum set of user attributes, including those mandatory by eduGAIN, is reported in the following table.

Attribute	Description
Surname (sn)	User last name
Given Name	User first name
Common name (cn)	Common name composed by first name and last
mail	User email
eduPersonTargetedID	Unique, persistent, opaque and targeted identifier of the user. (Serialised) Example: https://aai-logon.switch.ch/idp/shibboleth!https://filesender.funet.fi!u82n3lsslw=
eduPersonScopedAffiliation	user's organisational affiliation Example: staff@example.org [mailto:staff@example.org];member@example.org [mailto:member@example.org]
eduPersonPrincipalName	The eduPersonPrincipalName looks like an e-mail address with a user part that must be unique followed by an @ sign and the domain of the organisation to which the user belongs. Example: jdoe@example.org [mailto:jdoe@example.org]

Table 6. Possible minimum set of user attributes

Billing Services

In the future, billing will become an essential component in the relationships between service and data providers and consumers. Currently, there is no option to add billing information to OGC Web services other than using extensions to Capabilities information that result in limited interoperability. In addition, consumers cannot request quotas for specific jobs nor authorise job requests that a liable to pay.

Testbed-14 shall support the implementation of billing capabilities added to OGC Web service environments running on cloud infrastructures. These services shall allow service providers to receive remuneration for executed service requests. Focus of Testbed-14 is on providing billing information for individual jobs and on intercepting job requests for verification of billing/accounting acceptance on the provider side, and on billing request and authorisation functionality on the consumer side. Testbed-14 shall enable

- service providers to expose billing information to their services and for specific job requests, and
- service consumers to request quotas for specific requests and to execute specific job requests accepting a given quota.

The actual billing settlement shall not be topic of Testbed-14. No financial transactions are executed. Instead, Testbed-14 shall concentrate on the information exchange process between a consumer and the provider of a chargeable resource. This resource shall be accessible through an OGC Web service interface.

Participants shall propose and develop solutions on top of the third scenario, where Bob requests first a quota for his job request, receives different offers (e.g. shorter processing times result in higher costs), selects the appropriate option and then executes the request.

Error Handling and User Messaging

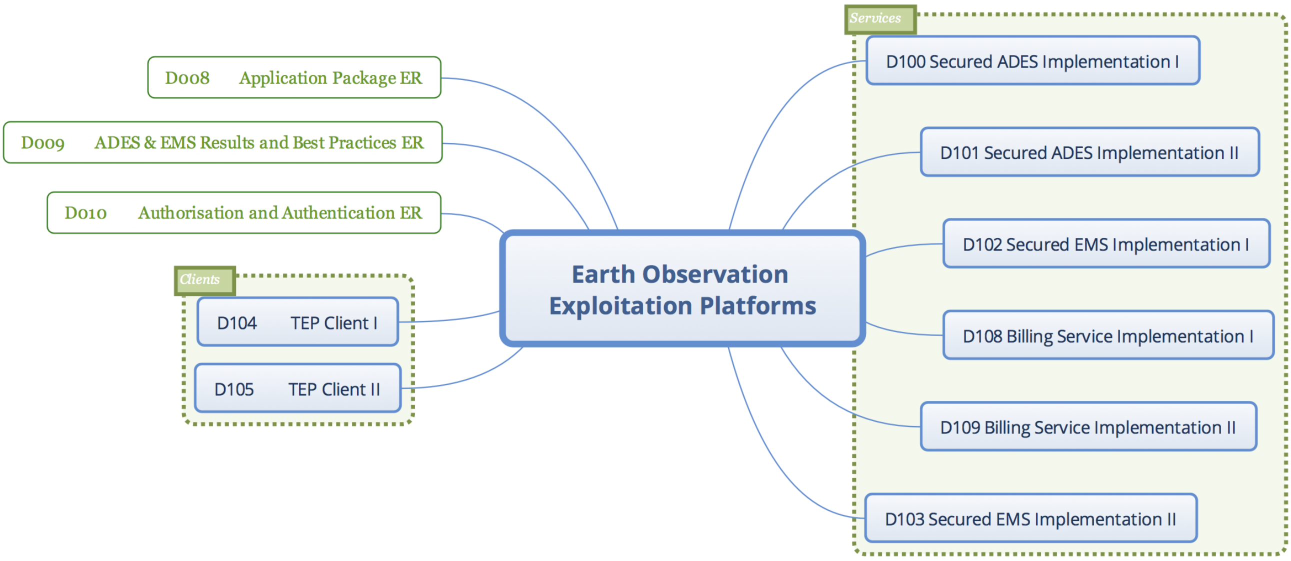
Errors and problems appear at all levels of the cloud processing stack - from underlying issues relating to the cloud infrastructure, to errors generated by Web services and applications, standardised error handling needs to be addressed in order to provide a service to end users.

Error codes are generated by the client, server or middleware. Common client error codes can be caused by something that the client did (e.g. providing an invalid parameter, executing an action out of sequence or even due to an access control restriction). Other errors could be caused by the server side infrastructure. These errors can be related to storage, virtual machine health or even networking problems. Middleware issues could be related to memory usage by threads, file permission issues, HTTP error codes, allocation of resources, credits etc.

Testbed-14 shall propose a common mechanism for error handling and messaging for the Earth Observation Cloud services developed as part of the Client, EMS and ADES above, to provide a more intuitive interface to end users. Participants shall propose and develop solutions on top of the third scenario, demonstrating how errors are encoded, propagated and reported.

B.17.4. Work Items

The following figure illustrates the *Application and Workflow Management on Cloud Platforms* work deliverables.



D008 Application Package ER
D009 ADES & EMS Results and Best Practices ER
D010 Authorisation and Authentication ER
Clients
D104 TEP Client I
D105 TEP Client II
Earth Observation Exploitation Platforms
Services
D100 Secured ADES Implementation I
D101 Secured ADES Implementation II
D102 Secured EMS Implementation I
D108 Billing Service Implementation I
D109 Billing Service Implementation II
D103 Secured EMS Implementation II

Figure 37. Application and Workflow Management on Cloud Platforms deliverables

B.18. Deliverables

The following list identifies all deliverables that are part of this work item. Additional requirements may be stated above. Thread assignment and funding status are defined in section Summary of Testbed Deliverables.

- **D008 Application Package ER** - Engineering Report capturing all elements relevant to the Application Package
- **D009 ADES & EMS Results and Best Practices ER** - Engineering Report capturing all experiences and lessons learned for the ADES and EMS implementations, interface designs, and workflows. These include the error handling & messaging recommendations.
- **D010 Authorisation, Authentication & Billing ER** - Engineering Report capturing all information relevant to the security setup and billing extensions
- **D100 Secured ADES Implementation I** - FOSS implementation of the ADES as described above, including authorisation handling using SAML2 assertions.
- **D101 Secured ADES Implementation II** - FOSS implementation of the ADES as described above, including authorisation handling using SAML2 assertions.
- **D102 Secured EMS Implementation I** - FOSS implementation of the EMS as described above, including authorisation handling using SAML2 assertions.
- **D103 Secured EMS Implementation II** - FOSS implementation of the EMS as

described above, including authorisation handling using SAML2 assertions.

- **D104 TEP Client I** - FOSS implementation of the TEP client and GUI as described above. The client shall support all user interaction, including support for application chaining.
- **D105 TEP Client II** - FOSS implementation of the TEP client and GUI as described above. The client shall support all user interaction, including support for application chaining.
- **D106 Application Package and App Development I** - Application packages and applications that support the information and data workflow described above.
- **D107 Application Package and App Development II** - Application packages and applications that support the information and data workflow described above.
- **D108 Billing Service Implementation I** - Add on for ADES and EMS that supports billing and quota workflows as described above.
- **D109 Billing Service Implementation II** - Add on for ADES and EMS that supports billing and quota workflows as described above.

Appendix C: Bibliography

- J. Becedas, R. Pérez and G. González, "Testing and validation of cloud infrastructures for Earth observation services with satellite constellations", International Journal of Remote Sensing, vol. 36, nº 19-20, pp. 5289-5307, 2015.
- [Becedas et al, 2015b] J. Becedas, R. Pérez, G. González, J. Álvarez, F. García, F. Maldonado, A. Sucari and J. García, "Evaluation of Future Internet Technologies for Processing and Distribution of Satellite Imagery", The 36th International Symposium on Remote Sensing of Environment, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., vol. XL-7/W3, pp. 605-611, doi:10.5194/isprsarchives-XL-7-W3-605-2015, 2015.
- [Ramos, 2016]J. J. Ramos and J. Becedas, "Deimos' gs4EO over ENTICE: A cost-effective cloud-based solution to deploy and operate flexible big EO data systems with optimized performance", Procedings of the 2016 conference on Big Data from Space (BiDS'16), pp. 107-110, Santa Cruz de Tenerife, Spain, 2016.