OGC Earth Observation Applications Pilot

Call for Participation (CFP)

Version 1.2, 09/12/2019

Call for Participation (CFP)

Please see for latest changes!

Chapter 1. Introduction

The Open Geospatial Consortium (OGC®) is releasing this **Call for Participation** ("CFP") to solicit proposals for the **OGC Earth Observation Applications Pilot** (also called "Initiative" or just "Pilot"). The goal of the pilot is to evaluate the maturity of the Earth Observation Applications-to-the-Data specifications that has been developed over the last two years as part of various OGC Innovation Program (IP) initiatives in a real world environment. 'Real world' includes integration of the architecture in an environment requiring authenticated user identity, access controls and billing for resources consumed.

The pilot consists of two phases:

Phase One (P1)

Invites application developers that work with Earth observation satellite data to join a requirements definition workshop. The objective of this workshop is to understand the exact requirements of application developers in terms of data discovery, data loading, data processing, and result delivery. The results from this requirements gathering will define the implementation evaluation and performance criteria.

The Goal of Phase 1 is understanding all the requirements that need to be implemented by the OGC Earth Observation Applications Pilot to support efficient app developments.

This phase includes:

1. App developers to join workshop and explain their approach to allow functionality checks and identification of optimization options for the current architecture. The approach description should include aspects such as:

    a. finding data

    b. working with data

    c. making processes available

    d. result handling

2. App developers get informed what is available from platform providers:

    a. data and existing processes and corresponding interfaces

    b. processing environments

    c. user identity and authentication, including the possibility to use an existing identity (depending on Identity Provider)

    d. quoting options

    e. billing mechanisms

3. App developers to describe/document the use-cases for the analysis/applications that they wish to develop and deploy to the platforms. This should consider/agree the data to be made available in order to support the use cases of the developed apps.

Phase Two (P2)

Invites earth observation platform operators to implement the OGC Earth Observation Applications Pilot architecture as it has been defined in previous IP initiatives. The implementation shall take P1 results into account. In essence, this requires implementing two Web API endpoints to allow the registration, deployment and execution of new processes on the platform. Meanwhile, application developers implement their cases (defined in Phase 1) to provide test cases for the architecture.

The Goal of Phase 2 is for Platform Providers to develop the necessary user infrastructure and ADES/EMS services to allow app developers to deploy their apps, and to allow consumers to obtain quotes and execute processes, with associated billing. Simultaneously, app developers implement and deliver their Earth observation applications.

This phase includes:

1. development and deployment of EMS/ADES on platforms

2. development and deployment of apps

3. registration of apps on enterprise platforms

4. consumers using the apps

5. consumers execute workflows across platform boundaries

6. evaluation of architecture design and elements

7. recording of evaluation results

8. definition of future work items

Overall Goal

The overall goal of both phases is to achieve a solid understanding of the suitability of the current architecture. This includes better understanding which elements of the envisioned architecture work, and which parts require what type of modification. In this context, the questions shall be answered regarding what works and what does not, including:

1. discovery and loading of available data

2. discovery and invocation of existing services

3. development and deployment of user defined services

4. usage of external data

5. making results available

6. advertisement of new processes, discovery for consumers

7. enforcement of access controls across platform boundaries (federated/delegated access)

It is expected that the pilot will identify several missing/sub-optimal elements. As is currently hard to predict what form, functionality, or architectural design will require modifications, it is expected that participants have some flexibility during the course of the project in terms of evaluation and analysis. Aspects that cannot be addressed or need further review shall be recorded in a future work definition. Some expected potential issues have been captured in section Architecture Evaluation Aspects.

1.1. Context

The availability of free and open data, such as that provided by the Copernicus Sentinel fleet, together with the availability of affordable computing resources, create an opportunity for the wide adoption and use of Earth Observation (EO) data in all fields of our society.

ESA's "EO Exploitation Platforms" initiative aims at achieving a paradigm shift from "bring the data to the user" (i.e. user downloads data locally) to "bring the user to the data" (i.e. move user exploitation to hosted environments with collocated computing and storage). This leads to a platform-based ecosystem that provides infrastructure, data, computing and software as a service. The resulting Exploitation Platform is where scientific and value-adding activities are conducted, to generate targeted outputs for end-users.

In order to unite the existing and future resources into an 'Network of EO Resources' there is a motivation to define standard interfaces that facilitate the federation and interoperation of these scattered resources – allowing the user to efficiently access and consume the disparate services of different providers seamlessly.

The approach can be applied to different domains (other than Earth Observation) and ESA is thus working closely with OGC in the definition of these standard interfaces and related solutions.

1.2. Background

OGC Testbed activities in Testbed-13, Testbed-14, and the ongoing Testbed-15 have developed an architecture that allows the ad-hoc deployment and execution of applications close to the physical location of the source data. The goal is to minimize data transfer between data repositories and application processes. The following Engineering Reports describe the work accomplished in the Testbed-13/14:

- OGC Testbed-14: Application Package Engineering Report (18-049r1)

- OGC Testbed-14: ADES & EMS Results and Best Practices Engineering Report (18-050r1)

- OGC Testbed-14: Authorisation, Authentication, & Billing Engineering Report (18-057)

- OGC Testbed-14: Next Generation Web APIs - WFS 3.0 Engineering Report (18-045)

- OGC Testbed-13: EP Application Package Engineering Report (17-023)

- OGC Testbed-13: Application Deployment and Execution Service Engineering Report (17-024)

- OGC Testbed-13: Cloud Engineering Report (17-035)

Testbed-13 reports are referenced as they allow to understand the history of this work and design decisions in context, but are mostly superseded by Testbed-14 reports.

At the same time, significant progress towards more Web-oriented interfaces has been made in OGC with the emerging OGC APIs -Core, -Features, -Coverages, and -Processes. All of these APIs are using OpenAPI. These changes have not fully explored in the current architecture yet, which provides additional ground for experimentation as part of the pilot.

1.2.1. OGC Innovation Program Initiative

This Initiative is being conducted under the OGC Innovation Program. The OGC Innovation Program provides a collaborative agile process for solving geospatial challenges. Organizations (sponsors and technology implementers) come together to solve problems, produce prototypes, develop demonstrations, provide best practices, and advance the future of standards. Since 1999 more than 100 initiatives have been taking place.

1.2.2. Benefits of Participation

This Initiative provides a unique opportunity towards implementing the "applications to the data" paradigm in the context of Earth observation satellite data processing. It allows to create new business opportunities, explores the market readiness of the own platform, and close interaction with application developers; while the provided cost sharing boosts your R&D budget. With the European Space Agency (ESA) as main funding organization, new contacts and relationships with ESA and other space agencies can be established.

The outcomes are expected to shape the landscape of cloud-based platforms developed to facilitate and standardize the access to Earth observation data and information. The sponsorship supports this vision with cost-sharing funds to partially offset the costs associated with development, engineering, and demonstration activities that are part of this pilot. The cost-sharing offers selected participants a unique opportunity to recoup a portion of their initiative expenses.

1.3. Acronyms and Abbreviations

The following acronyms and abbreviations have been used in this report.

Term	Definition
ADES	Application Deployment and Execution Service
CFP	Call for Participation
CR	Change Request
DER	Draft Engineering Report (OGC Document)
DWG	Domain Working Group
ER	Engineering Report (OGC Document)
ESA	European Space Agency
FDER	Final Developer Engineering Report
FPPER	Final Platform Provider Engineering Report
GPKG	GeoPackage
IDER	Initial Developer Engineering Report
IP	Innovation Program
IPPER	Initial Platform Provider Engineering Report
OGC	Open Geospatial Consortium
ORM	OGC Reference Model
OWS	OGC Web Services
PA	Participation Agreement
POC	Point of Contact
Q&A	Questions and Answers
RM-ODP	Reference Model for Open Distributed Processing
SOW	Statement of Work
SWG	Standards Working Group
TBD	To Be Determined
TC	OGC Technical Committee
TEM	Technical Evaluation Meeting
TIE	Technology Integration/Interoperability Experiments
URL	Uniform Resource Locator
WFS	Web Feature Service
WPS	Web Processing Service
WG	Working Group

Table 1. Acronyms and Abbreviations

1.4. Applicable and Reference Documents

The following is a list of Reference Documents with a direct bearing on the content of this document.

Reference	Document Details
[IPPP]	OGC Innovation Program Policies & Procedures http://www.opengeospatial.org/ogc/policies/ippp

Table 2. Applicable Documents

Reference	Document Details
[TB13-ER]	OGC Testbed-13 EOC Thread Engineering Reports: • 17-035 - OGC Testbed-13: Cloud ER, http://docs.opengeospatial.org/per/17-035.html • 17-023 - OGC Testbed-13: EP Application Package ER, http://docs.opengeospatial.org/per/17-023.html • 17-024 - OGC Testbed-13: Application Deployment and Execution Service ER, http://docs.opengeospatial.org/per/17-024.html
[TB14-ER]	OGC Testbed-14 EOC Thread Engineering Reports: • 18-050r1 - OGC Testbed-14: ADES & EMS Results and Best Practices Engineering Report, https://portal.opengeospatial.org/files/?artifact_id=81794&version=1 • 18-049r1 - OGC Testbed-14: Application Package ER, https://portal.opengeospatial.org/files/?artifact_id=81790&version=1 • 18-057 - OGC Testbed-14: Authorisation Authentication and Billing ER, https://portal.opengeospatial.org/files/?artifact_id=81795&version=1 • 18-045 - OGC Testbed-14: Next Generation Web APIs – WFS 3.0 Engineering Report, http://docs.opengeospatial.org/per/18-045.html
[OGC-ER-PROC]	OGC ER Process https://portal.opengeospatial.org/files/?artifact_id=73351
[OGC-ER-TPL]	OGC ER Template https://github.com/opengeospatial/er_template

Reference	Document Details
[OGC-ABS-SPEC]	OGC Abstract Specifications http://www.opengeospatial.org/docs/as
[OGC-STD]	OGC Standards http://www.opengeospatial.org/docs/is
[OGC-PROF]	OGC Profiles http://www.opengeospatial.org/docs/profile
[OGC-COM-STD]	OGC Community Standards http://www.opengeospatial.org/standards/community
[OGC-STD-BASE]	OGC Standards Baseline https://portal.opengeospatial.org/files/?artifact_id=83926
[OPENEO]	Open EO https://openeo.org/
[OPENEO-UDF]	OpenEO UDF Framework https://open-eo.github.io/openeo-udf/
[ECMWF-CDS]	ECMWF Dataset Delivery: CDS and Toolbox https://confluence.ecmwf.int/display/COPSRV/Dataset+delivery%3A+CDS+and+Toolbox
[SEED]	SEED https://github.com/ngageoint/seed
[DOCKER-OBJ]	Docker object labels https://docs.docker.com/config/labels-custom-metadata/
[OPENAPI]	OpenAPI v3.0.2 https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.2.md

Table 3. Reference Documents

1.5. Roles and Responsibilities

Within this document and activity, the following roles are defined.

Term	Company / Entity	Role
Agency	The European Space Agency	Sponsor of OGC Pilot described by this CFP
EO Exploitation Platform Prime Contractor	Telespazio VEGA UK Ltd	Prime Contractor to ESA and supervisor for this pilot
Subcontractor (Participant)	The Supplier	The organization that will take part in the OGC pilot task and whose activities are defined in this CFP. Subcontractors will have subcontract with OGC.
OGC	Open Geospatial Consortium	Overall responsibility for the Pilot activities and contractor to participants.

Table 4. Roles and Responsibilities

Chapter 2. Procurement Context

2.1. Project Organisation and Responsibilities

This pilot is conducted under the OGC Innovation Program and implements the OGC Innovation Program "Pilot" initiative policies and procedures. In this specific context, additional policies and procedures apply: The pilot is open to all organizations. Cost sharing funds can only be requested by entities from ESA member states, Canada, and Slovenia.

Within the context of this procurement ESA are acting as a sponsor to the OGC Earth Observation Applications Pilot. Participants for ESA activities within the pilot will be selected jointly by ESA/TVUK and the OGC from responses to this Call for Participation, and will become Subcontractors to OGC. Subcontractors will be referred to as Participants.

Once initiated, the pilot activities will be managed by OGC. The Participants will interact directly with OGC for technical, management, and contractual matters. ESA and TVUK should be copied on correspondence and invited to meetings.

Cost sharing payments to Participants selected by this call are funded by ESA. During and on completion of the pilot activities, OGC evaluates achievement of each milestone in order to trigger payments due under this contract. Payments will be made directly from OGC to the Participant.

The contractual conditions, at project level, come from the prime contract between ESA and TVUK, incorporate OGC pilot terms & conditions and then down to the Participant.

2.2. Financial Model for the Pilot

The pilot initiative works on a model of Participant in-kind contributions of engineering resources, technology, and/or facilities and cost-share funding provided by the Sponsors such as ESA.

TVUK, ESA or OGC will not cover the costs of equipment or software required for the normal development of participant software products. Nor will TVUK, ESA or OGC fund the purchase of additional hardware or software. OGC on behalf of TVUK and ESA will support the unique costs associated with engineering labour and travel (limited to kick-off and demonstration events) - noting that travel is often provided as an in-kind contribution for OGC activities.

This would include, but not necessarily be limited to: labour to develop engineering requirements, engineering specification reports, prototypical software component development exercising OGC specifications, instantiating new services based on existing software that exercises relevant OGC specifications, documentation of specifications, demonstrations of prototypical or existing software components that exercise OGC specifications or draft specifications and travel to demonstration events.

Proposals submitted under this call shall include a full breakdown of in-kind contributions by participants and required cost sharing contributions from OGC on behalf of TVUK/ESA.

Chapter 3. Technical Architecture

This section provides the technical architecture and identifies all requirements and corresponding work items. It references the OGC standards baseline, i.e. the complete set of member approved Abstract Specifications ([OGC-ABS-SPEC]), Standards ([OGC-STD]) including Profiles ([OGC-PROF]) and Extensions, and Community Standards ([OGC-COMM-STD]) where necessary. Further information on the OGC standards baseline ([OGC-STD-BASE]) can be found online.

The overall goal is to implement the "application to the data" paradigm for satellite data that is stored and distributed on independent platforms. The basic idea is that each platform provides a standardized interface that allows the deployment and parameterized execution of applications that are packaged in Docker containers. A second platform, called Exploitation Platform, allows chaining applications into workflows with full support for quoting and billing. The architecture is described in full detail in the Engineering Reports for Testbeds 13/14 referenced in [TB13-ER] and [TB14-ER].

3.1. Architecture Components

The OGC Earth Observation Applications Pilot defines a set of interface specifications and data models working on the HTTP layer. The architecture allows application developers and consumers to interact with service endpoints that abstract from the underlying complexity of data handling, scheduling, resource allocation, or infrastructure management.

Figure 1. Earth observation cloud application architecture components

It consists of the following logical components:

- **Application Developers** that develop Earth observation data processing applications

- **Application Consumers** requesting the execution of these applications on remote data and processing platforms

- One (or more) **Docker Hubs** that allow storing containerized applications. The hub(s) need to be accessible for the data and processing platform(s)

- One (or more) **Exploitation Platform** to register applications, to chain these into workflows, and to request the deployment and execution on data and processing platforms

- One (or more) **Data and Processing Platform**, where applications are executed on local data

Partly, these logical components can be implemented jointly. For example, it is likely that a data and processing platform provides exploitation platform elements, in particular as the various APIs are based on the same underlying OGC specifications.

Together, the architecture components allow the execution of the following steps:

1. *Application developers* can develop applications in their local environment and make the application available as a Docker container on a Docker Hub. A corresponding metadata file called *Application Package* describes the application and defines all necessary parameters, such as required input data, start-up parameterization, etc. The Application Package gets registered with the *Execution Management Service* (EMS), a RESTful Web service endpoint. The API of that Web service is defined by an OpenAPI document. It implements the currently emerging standard *OGC API Process*. The EMS backend contains the control logic as well as additional components such as a catalog to register all Application Packages, a catalog client to find appropriate data etc.

2. *Application Consumers* discover available applications through the same EMS endpoint. For that reason, the EMS delivers a list of available processes that it can execute. The app consumer identifies a specific application and provides the necessary parameterization (e.g. values for area/time of interest). From this data, the EMS then builds the necessary calls to the *Application Deployment and Execution Service* (ADES). The ADES is basically a lightweight EMS with much reduced functionality. It only allows the (un-)deployment of applications and their parameterized execution.

3.1.1. Exploitation Platform

The Exploitation Platform is responsible for registration and management of application packages and the deployment and execution of applications on data and processing platforms. It further supports workflow creation based on registered applications, and aggregates quoting and billing elements that are part of these workflows. Ideally, the Exploitation Platform selects the best suited Data and Processing Platform based on application consumer's needs.

Figure 2. Components of the exploitation platform

The Exploitation Platform itself consists of the following components:

EMS API

The EMS (*Execution Management Service*) provides a RESTful interface to application developers to register their applications and to build workflows from registered applications.

App Registry

Registry implementation (i.e. application catalog) that allows managing registered applications (with create, read, update, and delete options)

Workflow Builder

This optional component supports the application developer to build workflows form the registered applications.

Workflow Runner

Workflow execution engine to execute workflows and to handle the necessary data transfers from on application to the other.

ADES Client

Client application to interact with the data and processing environments that expose the corresponding ADES (*Application Deployment and Execution Service*) API.

Billing & Quoting

A component that aggregates billing and quoting elements from the data and processing environments that are part of a workflow.

Identity & Access Management

> User management

3.1.2. Data and Processing Platform

These platforms have the raw data the applications work upon available locally. They allow deployment and execution of applications through the *Application Deployment and Execution* API.
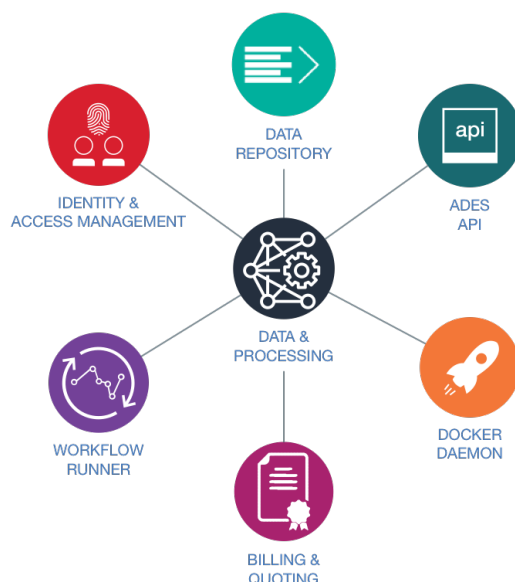
Figure 3. Components of the data and processing environment

The Data and Processing Platform consists of the following components:

Data Repository

> A repository of data that can be made available to Docker containers for local processing.

ADES API

> Application Deployment and Execution Platform_ API to deploy, discovery, and execute applications or to perform quoting requests.

Docker Daemon

> Docker environment to instantiate and run Docker containers.

Billing & Quoting

> Component that allows obtaining quotes and final bills.

Workflow Runner

> Workflow runner that can start the Docker container applications.

Identity & Access Management

> User management

3.2. Architecture Interactions

The following figure illustrates all major components and the corresponding interactions.
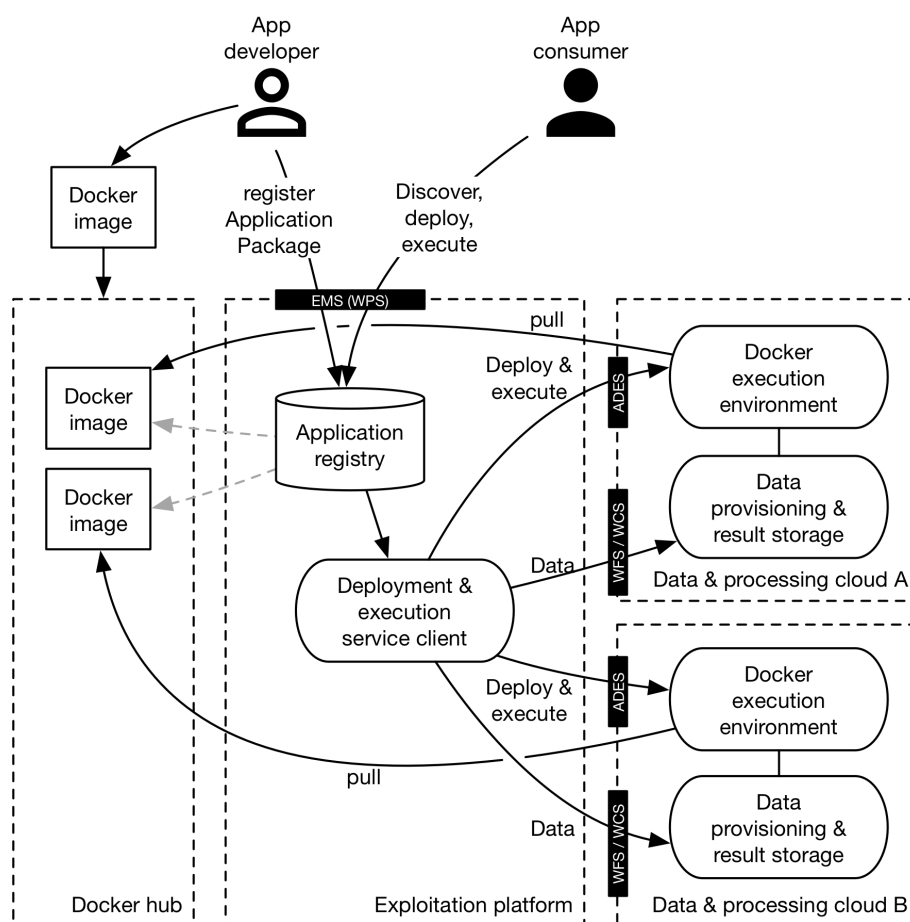
Figure 4. Earth Observation Cloud Application Architecture

The Execution Management Service (EMS) represents the front-end to both application developers and consumers. It makes available an OGC Web Processing Service interface that implements the emerging OGC Web API principles, i.e. provides a Web API that follows REST principles. The API supports the registration of new applications. The applications themselves are made available by reference in the form of containerized Docker images that are uploaded to Docker Hubs. These hubs may be operated centrally by Docker itself, by the cloud providers, or as private instances that only serve a very limited set of applications. Details mostly depend on security considerations. Initially developed to deploy applications only, the EMS has been developed into a workflow environment that allows application developers to re-use existing applications and orchestrate them into sequential workflows that can be made available as new applications again. This process is transparent to the application consumer.

The Application Package (AP) serves as the application meta data container that describes all essential elements of an application, such as its functionality, required satellite data, other auxiliary data, and input parameters to be set at execution time. The application package describes the output data and defines mount points to allow the execution environment to serve data to an application that is actually executed in a secure memory space; and to allow for persistent storage of results before a container is terminated.

The execution platform, which offers EMS functionality to application developers and consumers, acts itself as a client to the Application Deployment and Execution Services (ADES) offered by the data storing cloud platforms. The cloud platforms support the ad-hoc deployment and execution of Docker images that are pulled from the Docker hubs using the references made available in the deployment request.

Once application consumers request the execution of an app, the exploitation platform forwards the execution request to the processing clouds and makes final results available at standardized interfaces again, e.g. at Web Feature Service (WFS) or Web Coverage Service (WCS) instances. In the case of workflows that execute a number of applications sequentially, the exploitation platform realizes the transport of data from one process to the other. Upon completion, the application consumer is provided a data access service endpoint to retrieve the final results. All communication is established in a web-friendly way implementing the emerging next generation of OGC services that provides the various OGC APIs -Features, -Coverages, or -Processes.

BILLING AND QUOTING

Currently, satellite image processing still happens to a good extent on the physical machine of the end-user. This approach allows the end-user to understand all processing costs upfront. The hardware is purchased, prices per satellite product are known in advance, and actual processing costs are defined by the user's time required to supervise the process. The approach is even reflected in procurement rules and policies at most organizations that often require a number of quotes before an actual procurement is authorized.

The new approach featured in this pilot requires a complete change of thinking. No hardware other than any machine with a browser (could even be a cell phone) needs to be purchased. Satellite imagery is not purchased or downloaded anymore, but rented just for the time of processing, and the final processing costs are set by the computational resource requirements of the process. Thus, most of the cost factors are hidden from the end-user, who does not necessarily know if his request results in a single satellite image process that can run on a tiny virtual machine, or a massive amount of satellite images that are processed in parallel on a 1000+ machines cluster.

The currently ongoing efforts to store Earth Observation data in Datacubes adds to the complexity to estimate the actual data consumption. The reason is that the old unit "satellite image" is blurred when data is stored in multidimensional structures not made transparent to the user. Often, it is even difficult for the cloud operator to calculate exact costs prior to the completed execution of a process. This leads to the challenging situation for both, cloud operators that have to calculate costs upfront, and end-users that do not want to be negatively surprised by the final invoice for their processing request.

To address this challenge, OGC has started the integration of quoting and billing features into the cloud processing architecture. Specific service endpoints support quote requests that are executed prior to the actual execution requests. These allow a user to understand what costs will occur for a given service call, and they allow execution platforms to identify the most cost-effective cloud platform for any given application execution request.

Quoting and Billing information has been added to the Execution Management Service (EMS) and

the Application Deployment and Execution Service (ADES). Both services are implemented in a web-friendly way as a Resource Oriented Architecture (ROA) Web API that resembles the behavior of the current transactional OGC Web Processing Service v2.0 (this version, v.3.0, is not published yet by OGC). The API is described by an OpenAPI 3.0 specification (see [OPENAPI]) that allows deploying and executing new processes by sending HTTP POST request to the `DeployProcess` operation or `Execute` operation endpoints. Following the same pattern, it allows posting similar requests against the *Quota* endpoint, which causes a JSON response with all quote related data. The sequence diagram in Figure 5 illustrates the workflow.
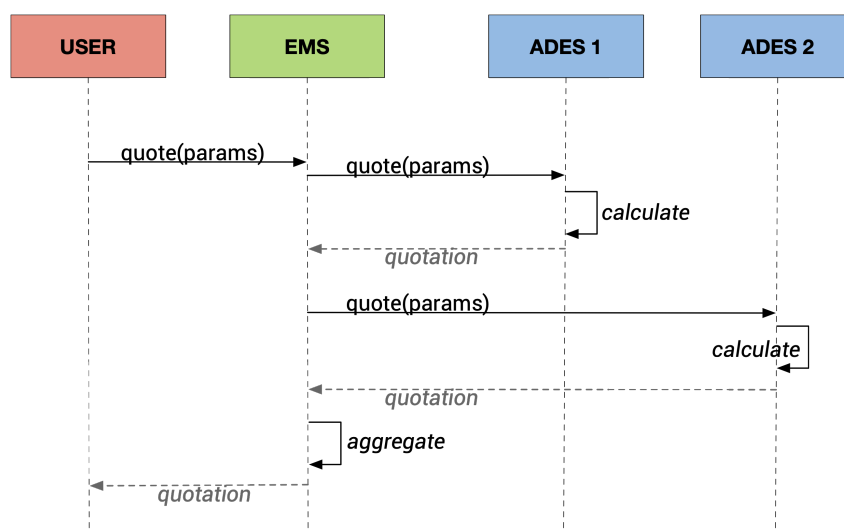

Figure 5. Quoting for applications and workflows

A user sends an HTTP POST request as a quasi-execution request to the EMS /quotation endpoint. The EMS now uses the same mechanism to obtain quotes from all cloud platforms that offer deployment and execution for the requested application. In case of a single application that is deployed and executed on a single cloud only, the EMS uses the approach to identify the most cost-efficient platform. In case of a workflow that includes multiple applications being executed in sequence, the EMS aggregates involved cloud platforms to generate a quote for the full request. Identification of the most cost-efficient execution is not straight forward in this case, as cost-efficiency can be considered a function of processing time and monetary costs involved. In all cases, a quote is returned to the user.

The quote model is intentionally simple. In addition to some identification and description details, it only contains information about its creation and expiration date, currency and price tag, and an optional processing time element. It further repeats all user-defined parameters for reference and optionally includes quotations for alternatives, e.g. at higher costs but reduced processing time or vice versa. These can for example include longer estimated processing times at reduced costs. Quotation requests resemble execution requests, i.e. contain the same elements and values as if an execution would have been requested. It is then up to the execution platform to obtain realistic quotes. It is emphasized that it is fully up to the platform to implement a reasonable cost calculation function. Kubernetes full metrics pipelines or metrics server might be an option.

Platforms can learn over time what costs are caused by specific requests and can generate better quotes by learning. The generation of a quote may not be based on calculations and experiences

from prior requests exclusively, but may take business considerations into account. As cloud platforms are competing with each other, a platform might be motivated to advertise its performance by providing specifically low quotes for a limited period of time.

USER IDENTIY AND ACCESS MANAGEMENT

In order to support the concerns of billing described above, it is necessary to establish the authenticated identity of the user to define the execution context of all requests and accesses to resources. Moreover, in the case of cross-platform workflows, the user's request context must extend across platform boundaries to ensure proper authorisation and associated billing of access to resources.

3.3. Pilot Scenario

The pilot shall implement a scenario that contains several operational data and processing platforms, such as e.g. DIAS or Thematic Exploitation Platforms. Each platform can participate as a Exploitation Platform, as a Data and Processing Platform, or in both roles. To participate as one or the other, the functionality and capabilities described in sections Architecture Components and Architecture Interactions shall be provided.

Application developers are invited to build applications to process data that is made available by the various platform providers. Application developers shall develop at least one application with its corresponding application package.

The original architecture had identified two different users only, Alice, the application developer, and Bob, who discovers Alice's app and executes it, and creates a workflow from registered applications. This pilot shall extend the set of users according to Figure 6.

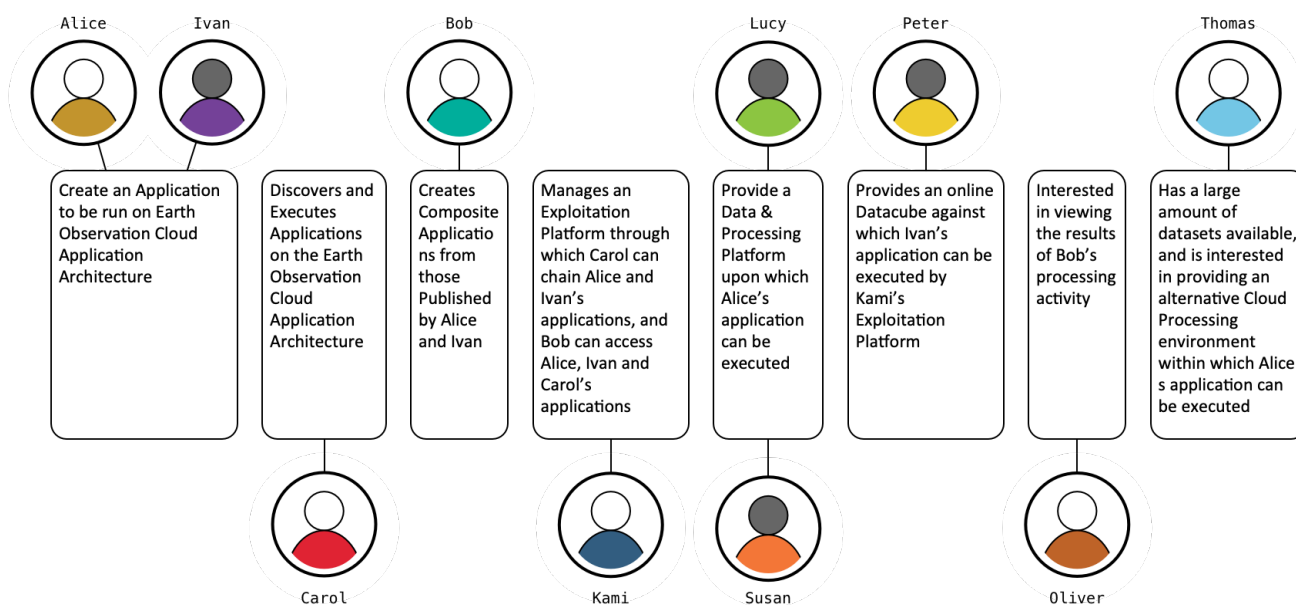| Alice	Ivan	Bob		Lucy	Peter		Thomas
Create an Application to be run on Earth Observation Cloud Application Architecture	Discovers and Executes Applications on the Earth Observation Cloud Application Architecture	Creates Composite Applications from those Published by Alice and Ivan	Manages an Exploitation Platform through which Carol can chain Alice and Ivan's applications, and Bob can access Alice, Ivan and Carol's applications	Provide a Data & Processing Platform upon which Alice's application can be executed	Provides an online Datacube against which Ivan's application can be executed by Kami's Exploitation Platform	Interested in viewing the results of Bob's processing activity	Has a large amount of datasets available, and is interested in providing an alternative Cloud Processing environment within which Alice s application can be executed

Carol · Kami · Susan · Oliver

Figure 6. Extended set of users/interested parties

Each of these actors has their own needs from the Architecture, which aren't addressed in the standard scenario described above. For example, Carol is interested in making sure that their

composite (chained) application is robust against later updates to Alice and Ivan's applications (perhaps through some versioning control); Kami is interested in making sure that when they present Alice's application/service to Bob that it will execute successfully on Lucy and Susan's platforms – and that they know it won't necessarily execute correctly on Peter's platform; Thomas is interested in knowing precisely what is required from him to ensure that Alice's application runs against his datasets, however they may be internally organized within his environment, and that relevant constraints are in place to avoid Alice's application having an unfortunate impact on the remainder of his services.

The *extended set of users* may cause new operational issues, such as chains of responsibility or intellectual property protection. For example, how is a problem resolved, which the non-expert user Bob experiences executing Carol's chained application (drawing on the work of Alice and Ivan) on Kami, Susan, Lucy and Peter's platforms ?

3.4. Architecture Evaluation Aspects

The pilot shall help understanding which areas - though they worked successfully in the laboratory like environments of Testbeds - need further specification work and testing in order to support real world setups and operational platforms. In addition, this pilot shall help defining future pilot initiatives until a final and mature set of specifications can be released as OGC Standards.

Therefore, the following paragraphs discuss evaluation aspects that are of particular interest. It is not necessary that each participant/setup addresses all aspects, but experiences that match any of these aspects shall be captured as part of the scenario implementation summary report.

3.4.1. Discovery

Discovery includes several aspects that shall be addressed, such as discoverability of data, suitability of data, and suitability of services, including:

- Discovery of available applications to solve a specific problem

- Discovery of available data sets and Earth observation products

- Discovery of already deployed processes that can be used "as is"

- Discovery of Analysis Ready Data that can be used as part of a workflow

- Discovery and publication of results (particularly where access to these should be limited to subset of just Bob and Oliver, and not Alice, Kami, Lucy etc.).

3.4.2. Application Execution: Provision of Data

Earth Observation processes typically consume raster data from different kind of missions (e.g. Sentinel, Landsat, Proba) delivered as tiles units (e.g. 110x110 km² for Sentinel-2). Most of the EO applications developed in OGC Testbeds implied restrictions to a particular raster or satellite format, and accepted as input a set of atomic image tiles directly provided by a Catalogue search

response. Some applications required preserving the original filename, the extension, or miming type of the source files as reported in the OGC 18-050r1, (ref. [TB14-ER]).

Unfortunately, each Cloud/Data Environment exposes data in a different way (sometimes even the same data, in a different format). The Application Package provides no means to identify supported formats or encodings for a given input/output; and therefore ends up containing significant complexity which would more correctly be moved to the ADES instances.

WPS terms do not contain sufficient detail to enable this (wps:complexdata can be anything, and a MIME type of "application/zip" is unhelpful), see Section 10.8.2 of OGC 18-50r1. It may be nice for all EO applications to simply request data provisioned via OGC services, but this may not yet plausible for many legacy applications and so some intermediate layer provided by either the ADES or Application Package would be required for the foreseeable future. In this context, see the envisioned Data Provisioning Extension to overcome data source format issues.

For the time being, one option might be that a cloud platform may need to be associated with a "profile" (e.g. a guarantee that it supports these specific ways of making data available to an application package) and an application package qualified against those profiles.

User Defined Functions ([OPENEO-UDF]) as defined by OpenEO ([OPENEO]) shall be considered as a partial answer to the question of the interface between the Application Package and the Data Layer on supported platforms within a Docker container.

3.4.3. Application Execution: ADES & AP Constraints

What constraints are applicable to the interface between the ADES and Application Package, which are not limited to the question of how a command line request can be constructed to execute a Docker command, including:

- Whether data translation / formats need to be specified, and therefore converted by the Cloud Processing Platform's ADES instance into a format expected by the packaged Application;

- Whether a Data & Processing Platform needs to support all possible Applications, or whether there is some kind of qualification/capability profiling taking place;

- What constraints are placed upon an application running within Data & Processing Platform (e.g. in terms of user, in terms of network access, in terms of resource usage);

- What technical/conduct constraints are placed upon where an application can be deployed in order to protect intellectual property (publication to a public DockerHub precludes many of these);

- Given possible intellectual property issues, how can a result set of image be 'pulled' if there are access controls to be satisfied;

- How these interfaces/profiles are modified to support use cases such as interactive applications, simple REST based applications, API-based access via notebooks, or WCPS implementations? It is noted that the WPS/container based approach discussed in the

reference documents is one solution to this problem; and that alternatives such as Google Earth Engine's programmatic interface, the ECMWF's CDS Toolbox ([ECMWF-CDS]), and the Horizon 2020 project OpenEO ([OPENEO]) exist and may be more appropriate for many use cases.

3.4.4. Application Execution: Workflow Language

The current architecture only uses a subset of CWL as a workflow language. Therefore the level of support required from an EMS or ADES (which may have to support only those parts required for a command line interface specification) is unclear. The complete CWL language does not map neatly to `wps:ProcessDescription`, and therefore it is unclear whether there's a real advantage at the ADES in requiring the implementation of a CWL parser, as opposed to defining conventions against which the Application Package is developed (which, is not simply the command line interface).

In addition, there is a the lack of support in the existing tooling or CWL specification for aspects important when running in a mainstream IT environment and still requires significant implementation by the ADES in order to convert inputs during stage-in/stage-out to the application package's expected format and location.

The usage of CWL shall be revisited in general. In particular, it shall be evaluated if the CWL-runner does meet the needs in the EMS in terms of the ability to remotely deploy/execute steps by interaction with the remote ADES.

3.4.5. Application Execution: Multiple Outputs

Section 10.9 of OGC 18-050r1 ([TB14-ER]) raises some concern with respect to multiple outputs, where the constraints of WPS make operations such as "identify all objects of type <a>" in data, "map processing over all identified objects of type <a>" (i.e. `cwl:Scatter`) difficult to express or implement.

3.4.6. Trust, Deployment and Security

The architecture foresees that a package can be deployed, on demand, by Bob on any cloud within the Network of Platforms. This requires a significant simultaneous three-way trust relationship between Alice, Bob and the Data & Processing Platform Providers. This trust relationship becomes more complicated when the deployed service doesn't work as expected – does Bob (the non-expert user) discuss the problem with Alice, with the Cloud Provider, with the Exploitation Platform?

A simpler relationship would be for Alice to deploy and test her application on Cloud Platforms, and then publish her combined service at the Exploitation Platform's EMS (or a central repository accessible by the EMS instances) as having been tested in a particular version and demonstrated to work on Platforms A/B/C3 with datasets D/E/F. This is equivalent to Cloud Native Computing Foundation or Linux ecosystems – packages/applications are qualified against particular distributions or conformance classes of such distributions: not an arbitrary "any" distribution. In

such a case, Bob as an application consumer, or Carol as a chained workflow developer, has much more chance of getting reasonable data out. Bob's first point of contact is the Exploitation Platform. The major concern is the 'maintenance overhead' for Alice to keep her application available and current in a growing/evolving set of Enterprise Platforms.

This requirement for Alice to deploy also enables the Data & Processing Platform to perform additional security checks if necessary – including ensuring that it has a local copy of the image in its repository (to reduce latency) if it wishes, and the opportunity to control/block execution of a service if it starts to misbehave. The Data & Processing Platform's ADES instance could determine whether Bob's execute request involved execute and deploy, or just execute.

3.4.7. Developer Support

Creating chained workflows, even within one platform, requires appropriate tooling to debug/analyze when failures occur. In the idealized architecture, WPS provides the connective tissue, and Section 10.3 of OGC 18-050r1 ([TB14-ER]) suggests the use of HTTP error codes. This is unlikely to be sufficient to support the development/debugging or support of a workflow of arbitrary processes running on arbitrary cloud platform environments, in which failures can arise from:

- Networking errors (internally within a platform, between platforms, between platforms and users);

- Data errors (availability, authorization, delay, corruption);

- Platform errors;

- Application errors;

- Chaining errors (e.g. failure to support error handling in a workflow);

- or combinations of any of the above.

It is obvious that the problem is broader than "what is the protocol by which error messages are propagated". There are many places from which logs may need to be made available to discover the root cause of an issue, and it's unclear how these would be sanitized and made available across the architecture to Bob, or Alice (looking at the output of Bob's problem), or the maintainer of the EMS/Exploitation Platform.

3.5. Possible Future Extensions or Architecture Modifications

The architecture developed in the last two years has been analyzed by a set of experts prior to this pilot. The analysis has identified several possible starting points for modifications. These modifications shall be explored in this pilot as much as possible.

3.5.1. Data Provisioning

Portability and flexibility would be achieved if the EO data could be retrieved using a standard mechanism and format. This is the purpose of the Web Coverage Service (WCS) and Web Coverage Processing Service (WCPS). These standards define Web-based retrieval of coverages, that is, digital geospatial information representing space/time-varying phenomena. A Coverage server provides access to coverage data in forms that are useful for client-side rendering: a spatio-temporal portion of the (possibility multidimensional) data and format supported by the application.
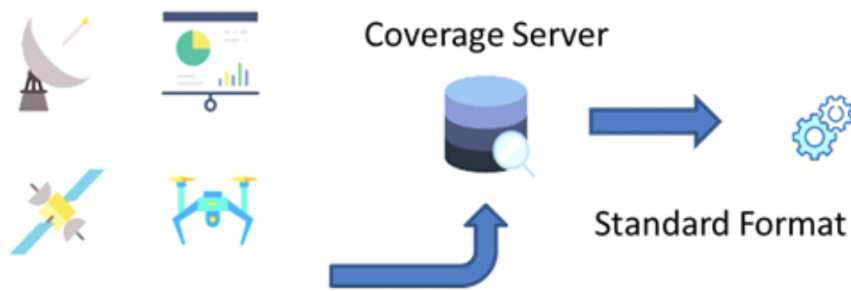

Figure 7. WCS ensures standardized access and format to and for coverage data

This pilot should explore the integration of a Web Coverage Service in the Exploitation and Data & Processing Platforms. Instead of focusing on the selection of individual product tiles in a particular format, the data selection would rely more intuitively on the area and time of interest aspects, and the applications and processes could be data format agnostic. In other words, the EO application should not have dependency anymore on the source data.

Also, the *Web Coverage Processing Service* may provide support for multi-dimensional array known as data cubes (time-series multi-dimensional stack of spatially aligned pixels) and provide analysis ready data to reduce data preparation burden and enable data interoperability.

3.5.2. Application Package

SEED ([SEED]), an alternative language of CWL, provides the application metadata embedded directly within the container image using a label mechanism. Therefore, the application metadata can be directly discoverable from the registry. SEED is used together with SCALE. SCALE is an open source system that provides management and scheduling of automated processing on a cluster of machines. It uses the SEED specification to aid in the discovery and consumption of processes packaged in Docker containers.

The underlying approach using Docker Labels ([DOCKER-OBJ]) could be applied to the current WPS application package.

3.5.3. Billing and Quoting Model

Extensions to the Execution/Billing/Quotation model, in particular with consideration of how Bob may specify preferred environments in which to execute Alice/Carol's application (e.g. those in which Bob has an existing relationship, preferential credit, or persists storage).

3.5.4. Communication Message Exchange

Today's developers much prefer working with JSON than XML. This affects community adoption, particularly for developers looking to construct browser-based applications. Therefore, the pilot shall explore usage of JSON where XML has been used for message exchange before.

3.5.5. Asynchronous Communication

Instead of synchronous communication, in particular between application consumers and services that execute applications, the pilot shall explore asynchronous communication patterns. The polling method of determining when the job is done is not very popular. Instead callback mechanism such as web hooks shall be explored.

3.5.6. Docker or Kubernetes

Despite Exploitation Platforms are provisioned in a Cloud infrastructure, the OGC Testbeds have currently limited the execution on a single machine environment known as a Docker host. This strongly limits the load escalation and computing resources management.

Kubernetes has become the de facto standard for deploying containerized applications at scale in private and public cloud environments. The most popular cloud platforms such as AWS, Google Cloud, Microsoft Azure, and Oracle Cloud now provide managed services for Kubernetes. Also, RedHat replaced a few years ago their OpenShift implementation with Kubernetes. Finally, Docker also integrated its direct competitor instead of its own Docker Swarm solution.

The pilot recommends prototyping and testing the execution and deployment of the processing chains onto a Kubernetes cluster. In particular, the automated deployment in a scaled pod (CPU, memory, disk space) should be studied.

Additionally, the topic of Kubernetes clusters federation should be investigated to federate the data and computing resources in case of processing chains (workflows) involving multiple exploitation platforms. This needs to take account of affinity of processing close to the data, and also to ensure that the billing responsibilities are clear.

Multi-cloud Kubernetes deployments (e.g. Google Anthos) are an obvious next step to be explored to allow for combining private cloud resources with public cloud investments in a transparent way. *Anthos* relies on GKE On-Prem, a way to run Kubernetes clusters.

3.6. Pilot Demonstration

The Pilot expects a face-to-face demonstration at ESRIN in Feb/Mar 2020.

The pilot demonstration shall include the following:

- Multiple applications

- Multiple platform providers

- Each application deployed and executed on more than one platform

- Platform user identity, access management and billing

Nice to have:

- Execution of workflows that span multiple platforms

- User identity, access management and billing for multi-platform workflows

- Alternative data access methods for applications accessing platform data as processing input

3.7. Work Packages and Deliverables

Dedicated work packages have been defined to distinguish the obligations of the Application Developers from the Platform Providers. Detailed requirements are stated above. All participants are required to participate in all technical discussions.

Multiple 'instances' of each work package can be funded, each producing their own set of deliverables.

3.7.1. Work Package WP1 – Application Developer

3.7.1.1. WP1 Tasks:

- Attend Requirements Definition Workshop, describing their processing approach, data requirements, etc.

- Understand offering made available by Platform Providers

- Document use case(s) for their analysis/applications, taking into account the Platform Provider offerings. Use case descriptions to be included in a first draft of the Engineering Report

- Development of Application Packages in accordance with the architecture

- Deployment, integration and test of apps in provided platforms, see Work Package WP2 – Platform Provider. Each Application Package shall be deployed and executed on all available Platforms.

- Demonstration of the application functionality as described in the use case

- Make their application available to other participants in the form of an Application Package file and docker image, as described in OGC Testbed-14: ADES & EMS Results and Best Practices Engineering Report, OGC 18-050r1, [TB14-ER]

- Production of Engineering Report describing and evaluating the activity from the Application Developer's perspective

- Production of 'screen capture' video that showcases the live demonstration

3.7.1.2. WP1 Deliverables:

- **D101 – Application Developer Engineering Report**
One summary report per app developer covering all of their apps, their use cases, and their experiences interfacing with each platform.

- **D102 – Application Developer Demonstration video**
A short video that can be used in outreach activities on a royalty-free basis.
See section Demonstration Videos below for a description of the video contents.

3.7.2. Work Package WP2 – Platform Provider

3.7.2.1. WP2 Tasks:

- Describe the platform offering in terms of data and services – to aid the Application Developers in defining their use cases. The platform offering shall be described in a first draft of the Engineering Report

- Understand the OGC Earth Observation Applications Pilot architecture as it has been defined in previous IP initiatives

- Implementation of required EMS/ADES platform functionality, including billing, quoting and user handling

- Develop and deploy EMS/ADES on their platform

- Make EMS/ADES endpoints available to all pilot participants to allow experimentation with deployment and execution of Application Packages provided by Application Developers.

- Make platform data available to deployed apps running in ADES

- Collaborate with the Application Developers to support the integration of the developed apps into their platform

- Demonstration of the EMS/ADES as an integrated component of their platform offering

- Production of Engineering Report describing and evaluating the architecture from the perspective of a Platform Provider

- Production of 'screen capture' video that showcases the live demonstration

3.7.2.2. WP2 Deliverables:

- **D201 – Platform Provider Engineering Report**
One summary report per platform provider, describing the platform offering , and covering all aspects of app integration, including IAM/billing.

- **D202 – Platform Provider Demonstration video**
A short video that can be used in outreach activities on a royalty-free basis.
See section Demonstration Videos below for a description of the video contents.

3.7.3. Demonstration Videos

Participants shall produce a short video (ref. D103/D202) that can be used in outreach activities on a royalty-free basis.

The video should illustrate the initial challenge(s) and developed solutions. The video can be done using screen-capturing of clients or slides with voice over. Good examples for videos are available from previous initiatives, such as Arctic Spatial Data Pilot (video 1, video 2), Vector Tiles Pilot (video), or Testbed-13 (video 1, video 2).

Chapter 4. Schedule

A master schedule showing deliverable due dates and other major milestones appears in the Major Milestones Table below. Deliverables meeting the requirements described in this document will contribute to the authorization of invoicing. It should be noted that dates are subject to change.

Milestone	Date	Event
M01	04 November 2019	Release of Call for Participation (CFP)
M02	13 December 2019	Proposals due
M03	18 December 2019	Participant selection and agreements
M04	January 21-22, 2020	Kick-off meeting & Application developer workshop, at ESA/ESRIN, Frascati (2 days)
M05	05 April 2020	• Application packages available • Platform services completed and available • Architecture component integration and start of technical interoperability experiments (TIEs)
M06	19 May 2020	• End of TIEs • Demonstration meeting at ESA/ESRIN, Frascati
M07	31 May 2020	Summary Reports due
M08	June 2020	Presentation of final results at OGC TC meeting

Table 5. Master Schedule

Chapter 5. Requirements for Management Deliverables

Participants shall agree to share the Technical-part only of their proposal with other participants of the OGC Earth Observation Applications Pilot.

5.1. Participant Points of Contact

Participant agrees to designate a primary Technical Point of Contact (POC) who shall remain available throughout the Pilot execution for communications regarding status.

Participant agrees to also identify at least one secondary Technical POC to support the Primary Technical POC as needed.

Contact information for the OGC and TVUK will be provided in the subcontract.

5.2. Meeting Attendance Requirements

5.2.1. In-Person Attendance

Most meetings for the pilot shall be held remotely via virtual meetings and teleconferences. Participants are encouraged to send a technical representative to attend in-person for the kick-off and demonstration event that will be held at ESA – ESRIN, Largo Galileo Galilei 1, Frascati, 00044 Rome, Italy.

5.2.2. Remote Attendance

Participant agrees to provide the services of at least one Technical POC to attend both regularly scheduled (weekly) and ad hoc web meetings and teleconferences.

5.3. Reporting Requirements

Initiative participant representatives are required to report the progress and status of the participant's work regularly.

Appendix A: Pilot Organisation and Execution

A.1. Initiative Policies and Procedures

This initiative will be conducted under the following OGC Policies and Procedures:

- This Initiative will be conducted in accordance with OGC Innovation Program Policies and Procedures ([IPPP]).

- OGC Principles of Conduct will govern all personal and public Initiative interactions.

- Participants drafting documents for the Initiative are required to allow OGC to copyright and publish documents following the OGC Intellectual Property Rights Policy.

A.2. Initiative Roles

The roles generally played in any OGC Innovation Program initiative include Sponsors, Bidders, Participants, Observers, and the Innovation Program Team ("IP Team").

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

The Initiative Architect will work with Participants and Sponsors to ensure that Initiative activities and deliverables are properly assigned and performed. They are responsible for scope and schedule control, and will provide timely escalation to the Initiative Director regarding any severe issues or risks that happen to arise.

A.3. Types of Deliverables

All activities in this pilot will result in a Deliverable. These Deliverables can take the form of Documents or Implementations.

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 (members-only) OGC Pending directory 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 for consideration by way of Standards Working Groups and Domain Working Groups.

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 WPS instance) are delivered by deployment of the service or component for use in the Initiative via an accessible URL. A Client software application or component may be used during the Initiative 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 for integration and interoperability testing in support of the agreed pilot scenario(s) and technical architecture.

A.4. Proposals & Proposal Evaluation

Proposals are expected to be short and precisely addressing the work items a bidder is interested in. Details on the proposal submission process are provided in Proposal Submission Guidelines; including links to proposal templates. The proposal evaluation process and criteria are described below.

A.4.1. Evaluation Process

Proposals will be evaluated according to criteria based on three areas: Technical, management, and cost. Each review will commence by analyzing the proposed deliverables in the context of the Sponsor priorities, examining viability in light of the requirements and assessing feasibility against the use cases.

The review team will then create a draft Initiative System Architecture from tentatively selected proposals. This architecture will include the proposed components and relate them to requirements in the Technical Architecture. Any candidate interface and protocol specification received from a Bidder will be included.

At the Technical Evaluation Meeting (TEM), the IP Team will present Sponsor(s) with draft versions of the initiative system architecture and program management approach. The team will also present draft recommendations regarding which parts of which proposals should be offered cost-sharing funding (and at what level). Sponsors will decide whether and how draft recommendations in all these areas should be modified.

Immediately following TEM, the IP Team will begin to notify Bidders of their selection to enter negotiations for potentially becoming initiative Participants. The IP Team will develop for each selected bidder a **Participant Agreement** (PA) and a **Statement of Work** (SOW).

A.4.2. Management Criteria

- Adequate, concise descriptions of all proposed activities, including how each activity contributes to achievement of particular requirements and deliverables. To the extent possible, it is recommended that Bidders utilize the language from the CFP itself to help trace these descriptions back to requirements and deliverables.

- Willingness to share information and work in a collaborative environment

- Contribution toward Sponsor goals of enhancing availability of standards-based offerings in

the marketplace

A.4.3. Technical Criteria

- How well applicable requirements in this CFP are addressed by the proposed solution

- Proposed solutions can be executed within available resources

- Proposed solutions support and promote the initiative system architecture and demonstration concept

- Where applicable, proposed solutions are OGC-compliant

A.4.4. Cost Criteria

- Cost-share compensation request is reasonable for proposed effort

- All Participants are invited to provide at least some level of in-kind contribution (i.e., activities or deliverables offered that do not request cost-share compensation). Participation may be fully in-kind.

- Cost-share funding is offered exclusively to organizations from participating states to ESA's EOEP-5 program (includes AT+ BE+ CA+ CH+ CZ+ DE+ DK+ EE+ ES+ FI+ FR+ GB+ GR+ IE+ IT+ LU+ NL+ NO+ PL+ PT+ RO+ SE+ SI)

A.5. Proposal Submission Guidelines

A.5.1. General Requirements

The following requirements apply to the proposal development process and activities.

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

- OGC welcomes proposals from members and non-member organizations. Membership will be offered to non-members participating for the first time for the duration of the initiative

- Proposals may address selected portions of the initiative requirements as long as the solution ultimately fits into the overall initiative architecture. A single proposal may address multiple requirements and deliverables. To ensure that Sponsor priorities are met, the OGC may negotiate with individual Bidders to drop, add, or change some of the proposed work.

- Participants selected to implement component deliverables will be expected to participate in the full course of interface and component development, Technology Interoperability Experiments (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 Initiative 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 attend with at least one technical representative to the Kick-off meeting. Participants are also encouraged to attend at least with one technical representative the Demonstration Event.

- 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 Initiative stakeholders.

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

- Bidders will be selected to receive cost sharing funds on the basis of adherence to the requirements (as stated in the Technical Architecture) and the overall quality of their proposal. Bidders not selected for cost sharing funds may still be able to participate by addressing the stated CFP requirements on a purely in-kind basis.

- Bidders are advised to avoid attempts to use the Initiative as a platform for introducing new requirements not included in the 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 Initiative scope or not. Items deemed out-of-scope might still be appropriate for inclusion in a later OGC Innovation Program initiative.

- Each Participant (including pure in-kind Participants) that is assigned to make a deliverable 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. Each PA will include a statement of work ("SOW") identifying Participant roles and responsibilities.

A.5.2. What to Submit

The two documents that shall be submitted, with their respective templates are as follows:

1. Technical Proposal: https://portal.opengeospatial.org/files/?artifact_id=82493

2. Cost Proposal: https://portal.opengeospatial.org/files/?artifact_id=82494

A **Technical Proposal** should be based on the Response Template and must include the following:

- Cover page

- Overview (Not to exceed one page)

- Proposed contribution (Basis for Technical Evaluation; not to exceed 3 pages per work item)

- Understanding of interoperability issues, understanding of technical requirements and architecture, and potential enhancements to OGC and related industry architectures and standards

- Recommendations to enhance Information Interoperability through industry-proven best practices, or modifications to the software architecture defined in the Technical Architecture

- If applicable, knowledge of and access to geospatial data sets by providing references to data sets or data services

The Cost Proposal should be based on the two worksheets contained in the Cost Proposal Template and must include the following:

- Completed Pilot Cost-Sharing Funds Request Form

- Completed Pilot In-Kind Contribution Declaration Form

Additional instructions are contained in the templates themselves.

A.5.3. How to Transmit the Response

Guidelines:

- Proposals shall be submitted to the OGC Technology Desk (techdesk@opengeospatial.org).

- The format of the technical proposal shall be Microsoft Word or Portable Document Format (PDF).

- The format of the cost proposal is a Microsoft Excel Spreadsheet.

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

A.5.4. Questions and Clarifications

Once the original CFP has been published, ongoing authoritative updates and answers to questions can be tracked by monitoring this CFP.

Bidders may submit questions via timely submission of email(s) to the OGC Technology Desk. Question submitters will remain anonymous, and answers will be regularly compiled and published in the CFP clarifications table.

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

Update to this CFP including questions and clarifications will be posted to the original URL of this CFP.

Appendix B: Management Requirements (OGC)

B.1. Technical Interoperability Experiment (TIE) Planning

Participants assigned implementation deliverables agree to provide the services of a technical representative to participate in Technical Interoperability Experiment (TIE) planning. This planning will begin during Kick-off and continue throughout Testbed execution.

Multiple TIEs and multiple iterations of a particular TIE will be conducted during the testbed.

B.2. Component Interface Testing

Participants assigned implementation deliverables agree to provide the services of a technical representative to conduct TIEs that exercise server and/or client component's ability to properly implement the interfaces, operations, encodings, and messages to be integrated during the initiative.

Participant agrees to maintain sufficient version control to ensure that other participants can rely on partner TIE components that will not be modified unilaterally and without notice. OGC may conduct multiple TIEs and multiple iterations of a particular TIE during the course of the initiative.

B.3. Test Result Analysis

Participants assigned implementation deliverables agree to provide the services of a technical representative to report the outcome and relevant software reporting messages from TIEs in which the Participant participates. Participants agree to submit these TIE reports to the relevant thread-level email list(s) and attached to Participant monthly technical status reports.

B.4. Performance Testing

Participant may conduct testing to demonstrate that a prototype component performs well enough to encourage use as a public, Web-based application.

Appendix C: Tips for new bidders

Bidders who are new to OGC initiatives are encouraged to review the following tips:

- In general, the term "activity" is used as a verb describing work to be performed in an initiative, and the term "deliverable" is used as a noun describing artifacts to be developed and delivered for inspection and use.

- The roles generally played in any OGC Innovation Program initiative are defined in the OGC Innovation Program Policies and Procedures, from which the following definitions are derived and extended:

    ◦ Sponsors are OGC member organizations that contribute financial resources to steer Initiative 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 Initiative 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 might be receiving cost-share funding, but they can also make purely in-kind contributions. Participants assign business and technical representatives to represent their interests throughout Initiative execution.

    ◦ Observers are individuals from OGC member organizations that have agreed to OGC intellectual property requirements in exchange for the privilege to access Initiative 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 Initiative 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 Initiative execution, provides Initiative scope and schedule control, and assists stakeholders in understanding OGC policies and procedures.

    ◦ The term Stakeholders is a generic label that encompasses all Initiative actors, including representatives of Sponsors, Participants, and Observers, as well as the IP Team. Initiative-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.

- Any individual wishing to gain access to the Initiative's intermediate work products in the restricted area of the Portal (or attend private working meetings / telecons) must be a member-approved user of the OGC Portal system. Intermediate work products that are intended to be shared publicly will be made available as draft ER content in a public GitHub repository.

- Individuals from any OGC member organization that does not become an Initiative Sponsor or Participant may still (as a benefit of membership) quietly observe all Initiative activities by registering as an Observer.

- Prior initiative 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.

- The Sponsor(s) 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).

- A Bidder may propose against any or all deliverables. Participants in past initiatives have often been assigned to make only a single deliverable. At the other extreme, it's theoretically possible that a single organization could be selected to make all available deliverables.

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

  ◦ What is delivered instead is the behavior of the component installed on the Participant's machine, and the corresponding documentation of findings, recommendations, and technical artifacts as contributions to the initiative's Engineering Report(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 Initiative as a by-product of implementation.

- Results of other recent OGC initiatives can be found in the OGC Public Engineering Report Repository.

- A Bidders Q&A Webinar will likely be conducted soon after CFP issuance. The webinar will be open to the public, but prior registration will be required.

Appendix D: Corrigenda & Clarifications

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
A.5.2 What to Submit	"Proposed contribution (Basis for Technical Evaluation; not to exceed 1 page per work item)" has been replaced by "Proposed contribution (Basis for Technical Evaluation; not to exceed 3 pages per work item)"
Schedule	Proposal submission deadline extended to Friday, December 13, 23:59 Pacific Time

The following table identifies all clarifications that have been provided in response to questions received from organizations interested in this CFP.

Question	Clarification
The PDF version of the CFP section A.5.2 (bottom page 33) requests the "Technical Proposal" be based on the Response Template. This template includes "Overview" (A1) (1 page max) and "Proposed deliverables" (A2) sections (1 page per deliverable) with page limitations. The CFP page 34 requests, besides "overview" (B1) and "proposed contribution" (B2), also to include "understanding of interoperability issues" (B3), "recommendations to enhance…" (B4) and "if applicable, knowledge of and access …" (B5). Could you clarify whether the elements (B3)(B4)(B5) are to be appended/added to the provided MS-Word template (where they are missing) or are they assumed to be included in the "A2 – proposed deliverables" section which is said to be maximum 1 page per deliverable ? Are there page limitations applicable to the additional sections B3, B4, B5 in case they are not part of A2 of the provided Word template ?	Please add these sections B3, B4, and B5 to A2. The page limit has been modified to 3 pages per work item.
Is it allowed to submit Financial Proposals in EUR currency if the bidder is from an ESA member state, or are Financial Proposals to be submitted in USD ?	Please submit your proposal in your favorite currency and make sure the currency is clearly identified

<< End of Document >>