Proceedings of the 2019 conference on

Big Data from Space (BiDS'19)

---Turning Data into Insights---

19-21 February 2019
Munich (Germany)

Edited by P. Soille, S. Loekken, and S. Albani

EUROPEAN UNION SATELLITE CENTRE
SatCen
Analysis for decision making

European Commission

esa
European Space Agency

EUR 29660 EN

Joint Research Centre

QUOTING AND BILLING: COMMERCIALIZATION OF BIG DATA ANALYTICS

Ingo Simonis

Open Geospatial Consortium (OGC), London, UK

ABSTRACT

Big Data processing chains working on geospatial data such satellite imagery have been subject of research for many years and are now being operationalized in standardized ways. Well-defined interfaces and standardized data exchange formats have been developed that make working with satellite imagery end-user friendly. Recent advances in the standardization domain even support the 'applications to the data' paradigm that allows executing arbitrary applications close to the physical location of the data. What was missing so far was a smooth integration of commercial principles, such as solid authentication and authorization, as well as quotation and billing mechanisms. This paper outlines how the integration of those commercial key elements can happen.

Index Terms— spatial data infrastructures, markets, standards, big data, billing, security

1. INTRODUCTION

Historically, space and more precisely earth observing satellites has been an industry only space administrations could handle. The picture is changing with more private industry now operating their own satellites, though most of earth-observing data is still produced by space administrations. Nevertheless, given that prices to build and launch a satellite into orbit have dropped from USD 200M a decade ago to just around 200k, space becomes more accessible and with the growing amount of data, new exploitation avenues are opening. The data itself is becoming much more useful for businesses and governments. In particular the recent advances in machine learning and artificial intelligence allow developing applications that provide previously inaccessible insights on global-scale economic, ecological, social and industrial processes.

At the same time, the availability of cloud based storage and processing environments and the widespread acceptance of container technologies open new exploitation pathways. Data does not need to be sold anymore. Computing cycles, input/output transactions, and on-demand storage capacities form new markets. An essential requirement for these emerging markets are robust billing and quotation, authentication and authorization,and auditing services. These services combined with new data usage principles are the cornerstones towards a new era of Big Earth Observation data commercialization. Data is not purchased anymore, but rented just for the time of analysis. This approach avoids high overhead costs for data that might be used only once before being replaced by newer data.

The private sector with companies such as Digital Globe has made their petabyte sized satellite data archives available on a ”pay-per-use basis” for quite some time now. Lately, space agencies such as ESA follow these developments by outsourcing satellite product storage, access, and processing to commercial cloud operators. Data that has been formally available for purchase is now offered through a number of commercial partners for rent. End-users now enter in contracts and service level agreements with cloud operators, but instead of downloading the data, it is rented for the time of processing. Rental fees depend on provider and nature of data, with more and more data offered royalty-free. In such a case, payments are being made afterwards based on used computing cycles or other parameters typical to cloud environment, such as input/output transactions or temporary storage volume, but less often on a ”per-product” basis. In such an environment where data is offered by a multitude of providers, where applications can be developed and offered by third parties that barely get in contact with the end-users, and consumers that are interested in the results produced by the applications rather than the original satellite data, standards play an essential role. In the following, this paper will layout a standards-based software architecture that allows such a market to grow and succeed. It will identify the various decoupled players and their respective roles, and identify important standards and design principles that allow a geospatial Big Data market to be successfully established. It introduces a billing and quoting model that has been implemented and tested in the recent OGC Innovation Program initiative Testbed-14.

2. STANDARDIZED BIG DATA PROCESSING

Standardization efforts are under way that allow the ad-hoc deployment and execution of arbitrary applications close to the physical storage location of the data. The Open Geospatial Consortium (OGC) has worked for the last two years on a set of standards and software design principles that allow a vendor and platform neutral secure Big Data processing ar-

doi:10.2760/848593

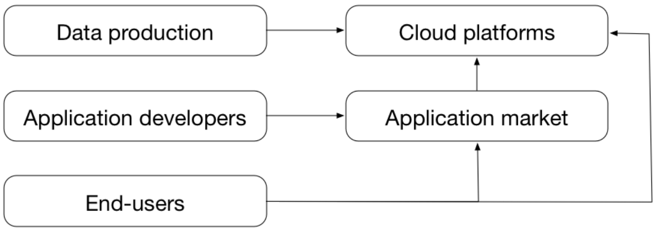

Fig. 1. High level software architecture

chitecture. Driven by requirements set by European Space Agency and Natural Resources Canada, OGC has developed a software architecture that decouples the data and cloud operators from earth observation data application developers and end-consumers.

2.1. Decoupled Roles for Successful Markets

As shown in figure 1, data generated by a data production organization is hosted and made available for access by a second organization that operates a cloud platform. Application developers make their products available on application markets that are used by end-users for application discovery. An application market can be organized as an independent entity, though in its current setup, it is rather tightly coupled to the cloud platforms. A connection that needs to be relaxed in future. To ensure maximal interoperability between all components, the envisioned architecture illustrated in figure 2 is leveraging OGC standards and has been complemented lately with security and billing and quotation elements to further qualify for competitive market situations.

Data producers nowadays often host their data on cloud platforms that are physically and organizationally decoupled, i.e. the actual hosting of the data and the provisioning of additional cloud features is outsourced to third parties. In our working example, ESA data is provided on cloud platforms operated by Cloudferro, Vito, or Amazon. These cloud operators generate revenue by selling cloud services on top of the actual data. Often, download capabilities are limited and cloud services such as deployment and execution of containerized applications are offered instead. This approach allows independent application developers to generate applications that operate on the data. These applications follow emerging OGC standards to ensure interoperability across the various cloud platforms. In future, end-users will discover the applications on application markets similar to the Apple's App store or Google's Play market. Thanks to standards, these applications can be deployed and executed on the various cloud platforms seamlessly. Results are made available through standardized data access services.

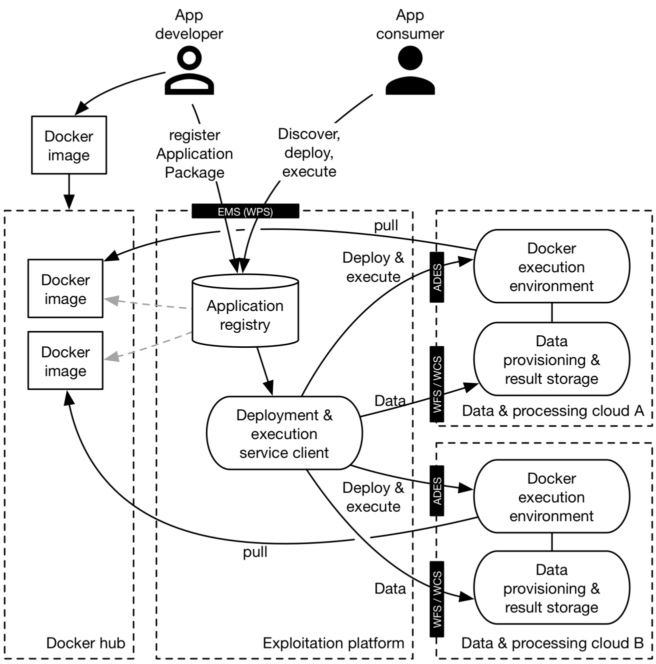

Fig. 2. Detailed Architecture with ADES, EMS, and AP

2.2. Big Data Processing Architecture

The architecture, described in full detail in [1] and outlined in figure 2, builds primarily on three emerging standards: The application deployment and execution service (ADES), the execution management service (EMS), and the application package (AP) standard. All three standards have been initially developed in OGC Innovation Program initiative Testbed-13 and are available online [2, 3]. They are currently revised in Testbed-14, with final results expected in early 2019.

As illustrated in figure 2, 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 new resource oriented paradigm, i.e. provides a Web API that follows REST principles (WPS v3.0). 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. Initially developed to deploy applications only, the EMS is now emerging into a workflow environment that allows application developers to re-use existing applications and orchestrate them into sequential work-flows 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,

doi:10.2760/848593

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 known as WPS, WFS, and WCS 3.0.

3. 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 outlined in this paper requires a complete change of thinking. No hardware other than any machine with a browser, which 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 100+ machines cluster. The currently ongoing efforts to store Earth Observation data in Datacubes adds to the complexity to estimate the actual data consumption, because the old unit "satellite image" is blurred with data is stored in multi-dimensional 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 difficult 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.

3.1. Billing and Quoting Standardization Efforts

The OGC has started the integration of quoting and billing services into the cloud processing architecture illustrated in figure 2. The goal is to complement service interfaces and defined resources with billing and quoting information. These allow a user to understand upfront what costs may 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 as an OpenAPI v3.0.0 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 3 illustrates the workflow. A user sends an HTTP POST request to provide an 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 pricetag, 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

doi:10.2760/848593

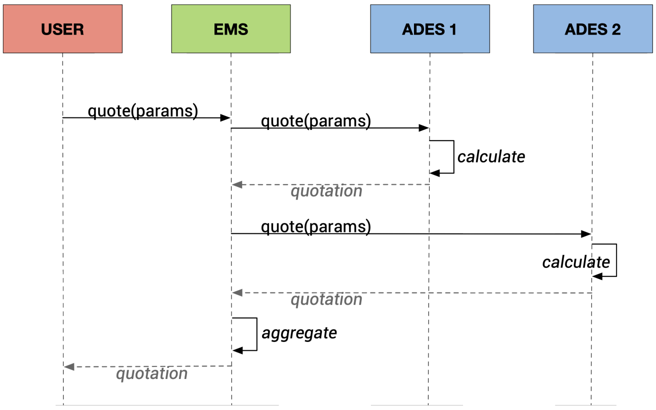

Fig. 3. Sequence diagram for quotes

have been requested. It is then up to the execution platform to obtain realistic quotes. This process is platform specific and can be implemented in multiple ways. We expect machine learning approaches to play an important role in this context. Platforms learn over time what costs are caused by specific requests and can generate better quotes for future requests 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.

4. SECURITY

Reliable communication within business environments requires some level of security. This includes all public interfaces as well as data being secured during transport. As shown in 4, the system uses identity providers to retrieve access tokens that can be used in all future communication between the application consumer, EMS, and ADES. The authentication loop is required to handle multiple protocols to support existing, e.g. eduGAIN [4], as well as emerging identity federations. Once an authentication token has been received, all future communication is handled over HTTPS and handles authorization based on the provided access token. Full details on the security solution are provided in [5].

5. OUTLOOK

The current setup allows requesting quotations for service requests and supports sequential executed processing chains. The quotation and billing model are both intentionally simple to be applicable to a wide range of domains. Both models feature simple extension mechanisms to address specific needs in some communities. Further research is necessary to develop the described approach into a core and profile model.

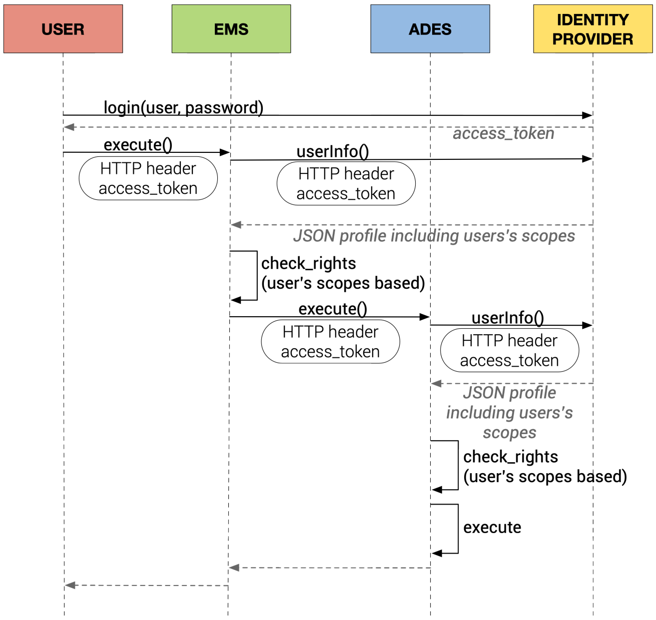

Fig. 4. Security overview

This model would then allow negotiating specific quote and billing profiles at runtime. Similar mechanisms can be used to define Service Level Agreements on the fly. Further research is required to add Linked Data principles that would allow making the billing and quotation model to leverage Semantic Web principles and capabilties. Processing results could receive identity, be linked to the original data and input parameter, and may serve as a valuable resource to others.

REFERENCES

[1] Simonis, I. (2018). Standardized Big Data Processing in Hybrid Clouds. In Proceedings of the 4th International Conference on Geographical Information Systems Theory, Applications and Management - Volume 1: GISTAM (pp. 20510). SciTePress. doi: 10.5220/0006681102050210

[2] OGC (2018). OGC Testbed-13: Application Deployment and Execution Service Engineering Report. OGC Document OGC 17-024. 17-024.html

[3] OGC (2018). OGC Testbed-13: Enterprise Platform Application Package Engineering Report. OGC Document OGC 17-023. 17-023.html

[4] GEANT (2018). eduGAIN Home. Website eduGAIN

[5] OGC (2019). OGC Testbed-14: Authorisation, Authentication, Billing Engineering Report. OGC Document OGC 18-057.