EO DATA PROVENANCE WITH KSI® BLOCKCHAIN

Issue Brief

February 2020

The European Space Agency's (ESA) role is to catalyse European innovation within the Earth Observation sector by introducing new ideas, featuring new methods and systems and enabling bold and innovative solutions. In 2019 ESA has released a White Paper on "Blockchain and Earth Observation" to define the key focus areas for the EO community to explore in the context of the Space 4.0 and future digital engineering for space missions. This publication is an "Issue Brief" by the ESA Blockchain / Distributed Ledgers and EO Community of Practice (CoP) and providing in-depth analysis of one of the priority actions identified by the White Paper for implementation as a part of the DLT technology research, development and proofing.

How to cite this report: Anna Burzykowska (ESA), Michele Iapaolo (Randstad), Priit Anton (Guardtime), Andreas Sisask (Guardtime), EO Data Provenance with KSI Blockchain, February 2020.

TABLE OF CONTENT

Introduction **3**

Background and Rationale **3**

Earth Observation use cases **5**

User requirements 7

The role of Data Provenance and Traceability 12

Implementation options 13

EO product provenance with KSI Blockchain **15**

KSI Blockchain introduction 15

Requirements analysis 16

Overall concept **19**

Provenance system functional breakdown 19

Provenance model 21

EO products hashing 21

Processing algorithms hashing 22

Provenance Graph Management 23

Verification of the EO Products 25

System components dynamic interaction **28**

Summary and Conclusions **30**

INTRODUCTION

The Blockchain for Space Activities (BC4SA) project is one of the early proofs-of-concept that provided deeper insights into the impact of the blockchain technologies on digital engineering for space missions at ESA[1]. It was implemented the frame of the ESA General Support Technology Programme (GSTP) through which the European space industry develops leading edge space technologies, foster innovation by creating new products and facilitate spin-in from outside the space sector.

Conceived at the ESA PhiLab, BC4SA project aimed to develop and prototype a set of new technologies to enable secured and traceable exploitation of data from space missions. The service demonstration was developed taking into account a compendium of data exploitation activities at the Directorate of Earth Observation Programmes and focused on four high level objectives:

- The use of blockchain technology for verification of integrity and time of EO data products and their provenance throughout the supply chain,
- Demonstration of the software compatibility with variety of EO missions and data formats,
- Demonstration of interoperability allow cryptographic proofs to be extracted and transported for data provenance verification and amending,
- Demonstration of deployment based on Copernicus data architecture.

The technical implementation described in this document consisted of several steps: the state-of-the-art DLT (Distributed Ledger Technology) technology survey and analysis of potential requirements; design and development of trusted data sharing process in supply chain including necessary software components; integration of the developed software onto the selected data acquisition and distribution infrastructures; and validation and demonstration on identified use cases.

BACKGROUND AND RATIONALE

The European Earth Observation sector is currently facing important developments concerning managing of large volumes of EO data coming from Copernicus as well as national European missions. These are already reaching the Petabyte scale and growing along with the needs of the user community to have a real-time and unobstructed access to existing satellite data archives and future acquisitions. This trend is likely to increase even more, in particular regarding global change monitoring requirements which is driving users to request time-series of data spanning 20 years and more.

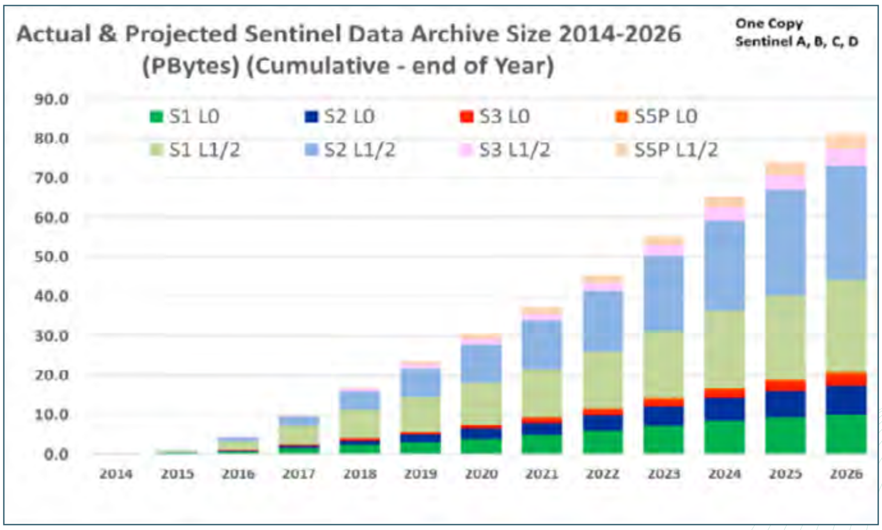


Figure 1. Sentinel Data Archive Growth

To date, more than 12 million digital Copernicus products are available for download through the Copernicus Open Access Hub, equivalent to a total volume of over 120 Petabytes. For EO data providers, such as ESA, the need for advanced EO data management solutions implies not only ensuring and facilitating their accessibility and usability, but also implementation of solutions for **cloud auditing and the maintenance of data provenance information**. This is because maintaining, managing and processing of such large volumes requires hosting of data and processing chains in multiple facilities and cloud infrastructures. Such computing environment introduces the risks of **accidental data corruption, processing errors, vulnerabilities such as security violation, data tampering or malicious interference** in the databases. There is thus an interest in data management workflows that can **provide a secure, definitive reference for data provenance, distribution and workflows tracking defined as a "traceability" or "the record of processing steps"** which can ensure continuous monitoring of the quality of EO products and services (quality assurance) especially given that much EO product generation and processing is increasingly taking place European DIAS (Data and Information Access Services) platforms or Collaborative Ground Segment facilities (CollGS).

Improving and redefining the way EO data and services are processed and distributed to the users, with a guarantee that processing chains deliver the same (high) quality of data products, is a key driver for innovation in the ground segment capabilities. Take the Sentinel Product Life Cycle, for example, which requires the labelling of EO data to make it more searchable and interoperable. Protecting data from tampering, and software and production chains from infringement or errors, is increasingly important vis-a-vis processing chains taking place on-demand. **The capacity to certify the integrity of a product by providing traceability and proof of EO value chain immutability and its authenticity has therefore a high utility value.**

To explore such "certification" functions for EO data, data processing steps need to be attributable, and may require the development of schemes to manage the digital identities (signatures) of contracted/trusted partners, allowing them to, among other things, 'certify' the validity of individual products; 'certify' the validity of lists of products; invalidate individual products and entire product baselines; record and document reasons of invalidity; identify product replacements, and so on.

Finally, from the point of view of scientific community and value adding industry, tracking of provenance of data and providing traceability of workflows became much more relevant with the increasing emphasis on the need for quality information services, explainable machine learning as well as attribution of inputs and results. Quality Assessment has been traditionally focused on scientific community methodologies that can ensure reproducibility of scientific analyses and processes. The dedicated "traceability chain" can additionally help a user in understanding the data production and the assumptions that are made during implementation. It also helps producers identify and understand potential sources of discrepancies between two similar data products produced by different methods or algorithms[2].

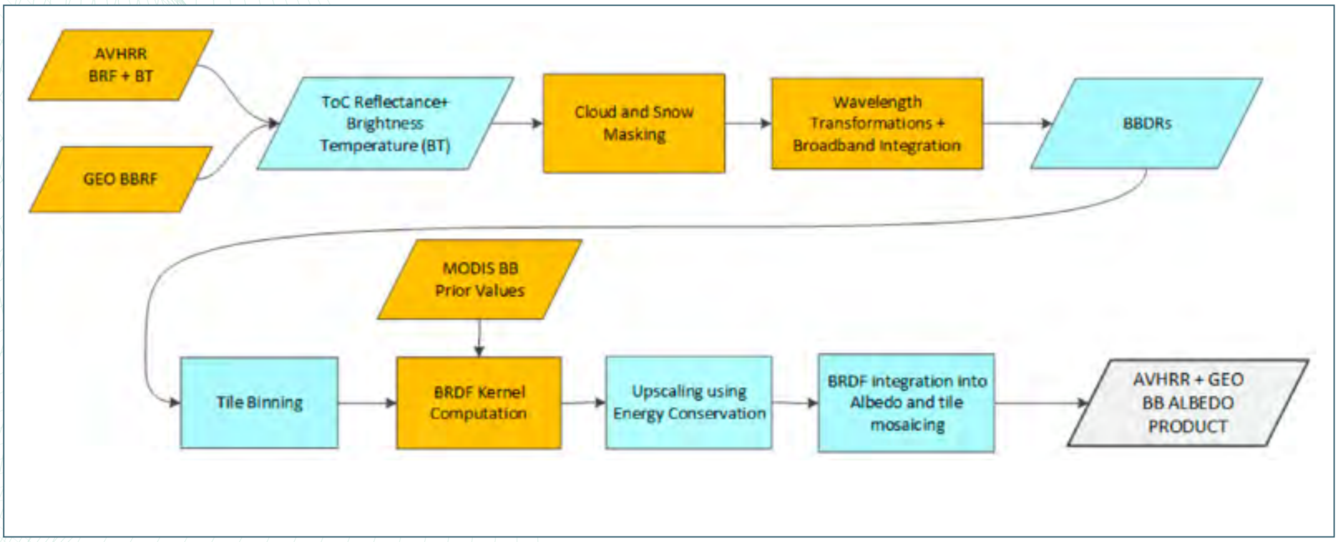

AVHRR BRF + BT
GEO BBRF
ToC Reflectance+ Brightness Temperature (BT)
Cloud and Snow Masking
Wavelength Transformations + Broadband Integration
BBDRs
MODIS BB Prior Values
Tile Binning
BRDF Kernel Computation
Upscaling using Energy Conservation
BRDF integration into Albedo and tile mosaicing
AVHRR + GEO BB ALBEDO PRODUCT

Figure 2. An example of traceability chain implemented as a part of Quality Assessment for the ESA Essential Climate Variables. Source: Nightingale J. et al.

Provenance and traceability are therefore not new concepts however digital traceability chain defined as a representation of processing steps taken to produce a final data product which can be implemented and retrieved in a completely automated, immutable and machine readable way is a long awaited innovation.

The BC4SA project focused on the use cases where innovative blockchain solutions can be used for Sentinel data provenance tracking, as well as for open dissemination of data provenance and integrity proofs along with products descriptions.

EARTH OBSERVATION USE CASES

One of the BC4SA objectives was to map the current potential user groups for the ESA Copernicus data that require an independent verification of data and workflows provenance (defined as verification of integrity and time), and co-design specific key use cases for each user group. When selecting the use cases, the focus has been on those users that have a direct need for integrity assurance and long-term provenance of EO data products and their value chain. Four major user groups were identified:

- Satellite EO mission operators, including primarily operator of the **Copernicus Core Ground Segment** (ESA) which is providing near real-time EO data processing at the Core Ground Stations and continuous data processing at the Processing and Archiving Centres (PAC) as well as long-term data archiving service.
- Copernicus **Collaborative Hubs** and their nodes (national data hubs in Europe and Canada), and other dissemination hubs (e.g. Copernicus Services Hubs, International Hubs).
- Operators of the five **DIAS platforms** (CREODIAS, MUNDI, ONDA, SOBLOO and WEKEO).
- **Downstream application service providers** in need of a verifiable and trusted data value chain in particular:
    - Insurance companies needing proof of the claim-triggering event and its time (e.g. hazardous event by natural disaster, fire) to activate the clauses of contracts;
    - Services in agribusiness providing financing services in the value chain in relation to specific performance or provenance (e.g. agriculture subsidy payouts, farm-to-fork tracking, credit worthiness);
    - Law enforcement needing objective, error-free and traceable data for dispute resolution.

Within all these user groups the following aspects were considered:

- the responsibilities of the particular user who operates the data hub, platform or a service,
- the identification of main use-case and scenario for a particular user regarding the "data integrity & traceability" requirement,
- analysis of the general architecture and data handling/storage process.

The following table summarizes the high-level use cases that are relevant to each of the user groups. The value in the cell indicates the priority of the use case.

Use Case / User Group	Copernicus Core Ground Segment	CollGS, IntHub	DIAS Platforms	Service Industry
Long-term integrity assurance of own EO archive	High	Medium	Medium	n/a
Disseminating data provenance and integrity proofs along with products	High	Medium	Medium	n/a
Automated verification of imported EO products	Medium	High	High	Medium
Demonstrating the integrity of derived EO products and services to 3rd parties on on-demand basis	Low	Low	Medium	High

User requirements

EO data hubs operators are considered a potential primary user community for the EO data provenance services. The examples of the target segment can in fact range from commercial operators (such as Maxar, Planet) and public operators (ESA, EUMETSAT), all interested in:

- securing their long-term data archives,
- reducing of the storage capacity of EO products by applying the on-demand product processing,
- providing proof and evidence of time, provenance chain and processes used for development of EO downstream services.

Take the Copernicus Core Ground Segment operated by ESA as an example. It allows all Sentinel data to be acquired systematically, processed and distributed including via the Sentinel Open Data Hub (DHuS) which is a Java web based system designed to manage the on-line dissemination of ESA Copernicus Sentinels data access. Managed by the Payload Data Ground Segment (PDGS) the Core Ground Segment includes the facilities responsible for mission control (mission planning, production planning), quality control (calibration, validation, quality monitoring, instrument performance assessment), precise orbit determination, user services interface and acquisition, processing and archiving. It is consisting of the following elements:

- The **Flight Operations Segment (FOS)** – responsible for all aspects of Sentinel flight operations, including monitoring and control, the execution of all platform activities and command of payload schedules.
- The **Core Ground Stations** – where the Sentinel data are downlinked and products are generated in near-real time.
- The **Processing and Archiving Centres (PACs)** – where systematic non-time-critical data processing is performed and all data products are processed and archived for online access by users.
- The **Mission Performance Centres (MPCs)** – responsible for calibration, validation, quality control and end-to-end system performance assessment.

The PDGS operationally acquires satellite images via downlink stations, generates the raw data products and distributes them at the Level-0, processed Level-1 and derived Level-2 products.

Product Level	Description
Level 0	Reconstructed, unprocessed instrument and payload data at full resolution, with any and all communications artefacts (e.g., synchronization frames, communications headers, duplicate data) removed.
Level 1A	Reconstructed, unprocessed instrument data at full resolution, time-referenced, and annotated with ancillary information, including radiometric and geometric calibration coefficients and georeferencing parameters (e.g., platform ephemeris) computed and appended but not applied to Level 0 data.
Level 1B	Level 1A data that have been processed to sensor units. Not all instruments have Level 1B source data, Sentinel 2 has however level 1C an orthoimage product, i.e. a map projection of the acquired image using a system DEM to correct ground geometric distortions. Pixel radiometric measurements are provided in Top-Of-Atmosphere (TOA) reflectances (coded in 12 bits) with all parameters to transform them into radiances.
Level 2	Derived geophysical variables at the same resolution and location as Level 1 source data.

Figure 3 below describes basic architecture of the **PDGS Core Ground Segment**. The Sentinel data acquisition and product generation in Near Real Time is carried out at the Core Ground Stations that are located in Italy Matera (eGeos), Norway Svalbard (K-Sat), Spain Maspalomas (Inta) and USA Alaska (K-Sat). Local stations can provide a regional (within the station coverage) quasi-real-time (10-15 min from sensing) data service via Sentinel collaborative (local) stations. Local/regional stations complementing the core X-band and Ka-band station network with the following potential activities: (NRT) data processing and distribution for Sentinel-1 and/or Sentinel-2 and elaboration of (NRT) products tailored to particular coverage/region, particular services, etc.

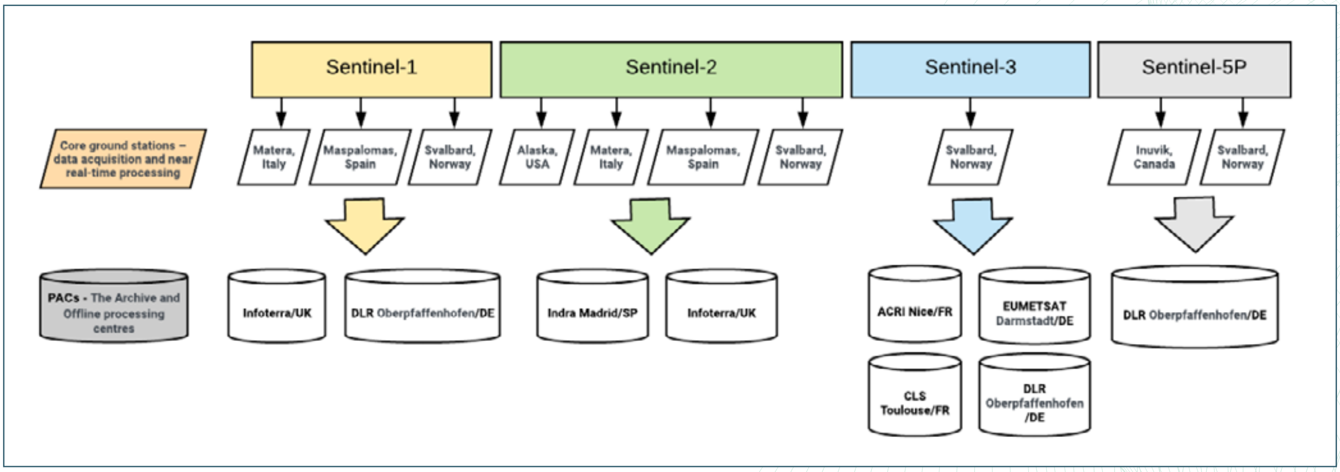


Figure 3 Sentinel core ground stations and processing and archiving centers.

Sentinel **Processing and Archiving Centres (PAC)** are located in various distributed locations:

- Sentinel-1 (Astrium/UK, DLR/Germany),
- Sentinel-2 (Astrium/UK, Indra/Spain),
- Sentinel-3 (OLCI Land DLR/Germany, SRAL CLS/France, SLSTR-SYN ACRI/France), Sentinel-3 (OLCI Marine EUMETSAT/Germany),
- Sentinel-5 (DLR/Germany).

These processing centers are responsible for archiving of the pre-processed data received from the Copernicus acquisition stations, systematically refining them into data products, and making them available worldwide[3].

Moreover, the Sentinel Core ground segment is complemented by several important data hubs for more targeted Sentinel data access and distribution. The so called **Collaborative Ground Segment (CollGS)** makes complementary access to Sentinel data and/or to specific data products or distribution channels, and are hosted by selected ESA member states. These “collaborative data products” include specific tailoring for regional coverage or specific applications, generation of local/regional data sets with correction, projection, calibration, merging etc., different to the standardised data set offered by the Core Ground Segment[4]. **Copernicus Services Data Hub** is providing dedicated access to pre-processed Sentinel data to entities providing Copernicus services.

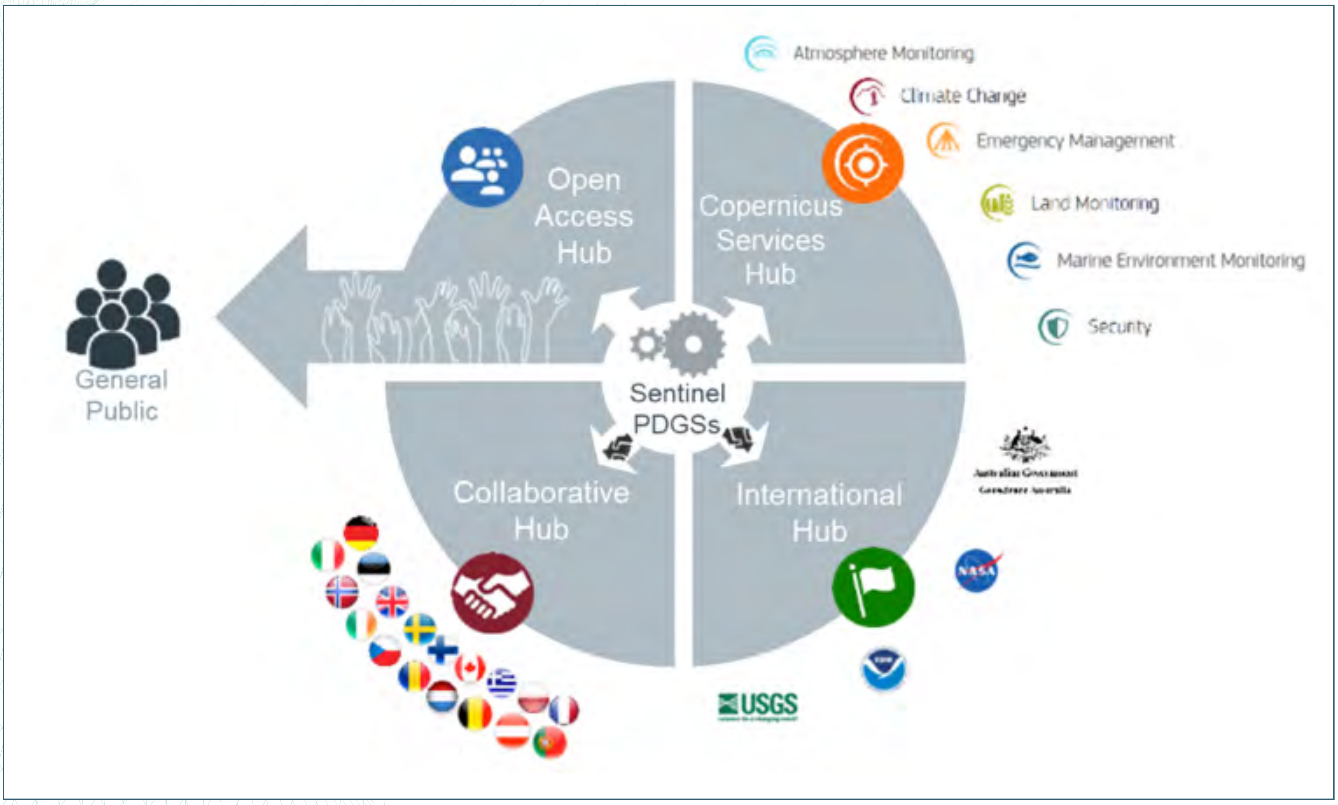


Figure 4. The Sentinel Data Access System Configuration at the end of Y2018

There are also **International Sentinel Hubs** open to international partners established following signature of a cooperation agreement with the European Commission and technical operating agreements with ESA. They are currently hosted by Geoscience Australia (GA), the National Oceanic and Atmospheric Administration (NOAA – US), the National Aeronautics and Space Administration (NASA – US), and the US Geological Survey (USGS – US).

Finally, thanks to **Copernicus Open Access Hub** any user or organization can collect Sentinel data and share them through their own cloud storage. Examples of such implementations include Google Cloud and Earth Engine hosting Sentinel-2 Data, Amazon Web Services (Sentinels on AWS) or DIAS (Copernicus Data and Information Access Service).

The Figures 3,4 and 5 taken together visualise a complex distribution of original Copernicus data from PDGS through the entire architecture where data dissemination and pre-processing takes place at various stages of the acquisitions.

Figure 5 shows, the data flow between the key data centers. The data from the PDGS are primarily hosted within a large data centre from T-Systems in Frankfurt, part of the overall Copernicus Wide Area Network connecting all major centres involved in the acquisition, processing and archival of the Sentinels data. Complementary Centers based at OVH and GRNET provide on-line redundancy. The Figure 5, on the left, illustrates how the Sentinel PDGSs and Auxiliary Centres provide data products to the Data Access System 'Back End' through which the system is run. The data flow continues to the right, the 'Front End' Data Access Hubs through which the data is exposed to end users including via DIAS, CollGS, International Hubs and other Scientific Hubs.

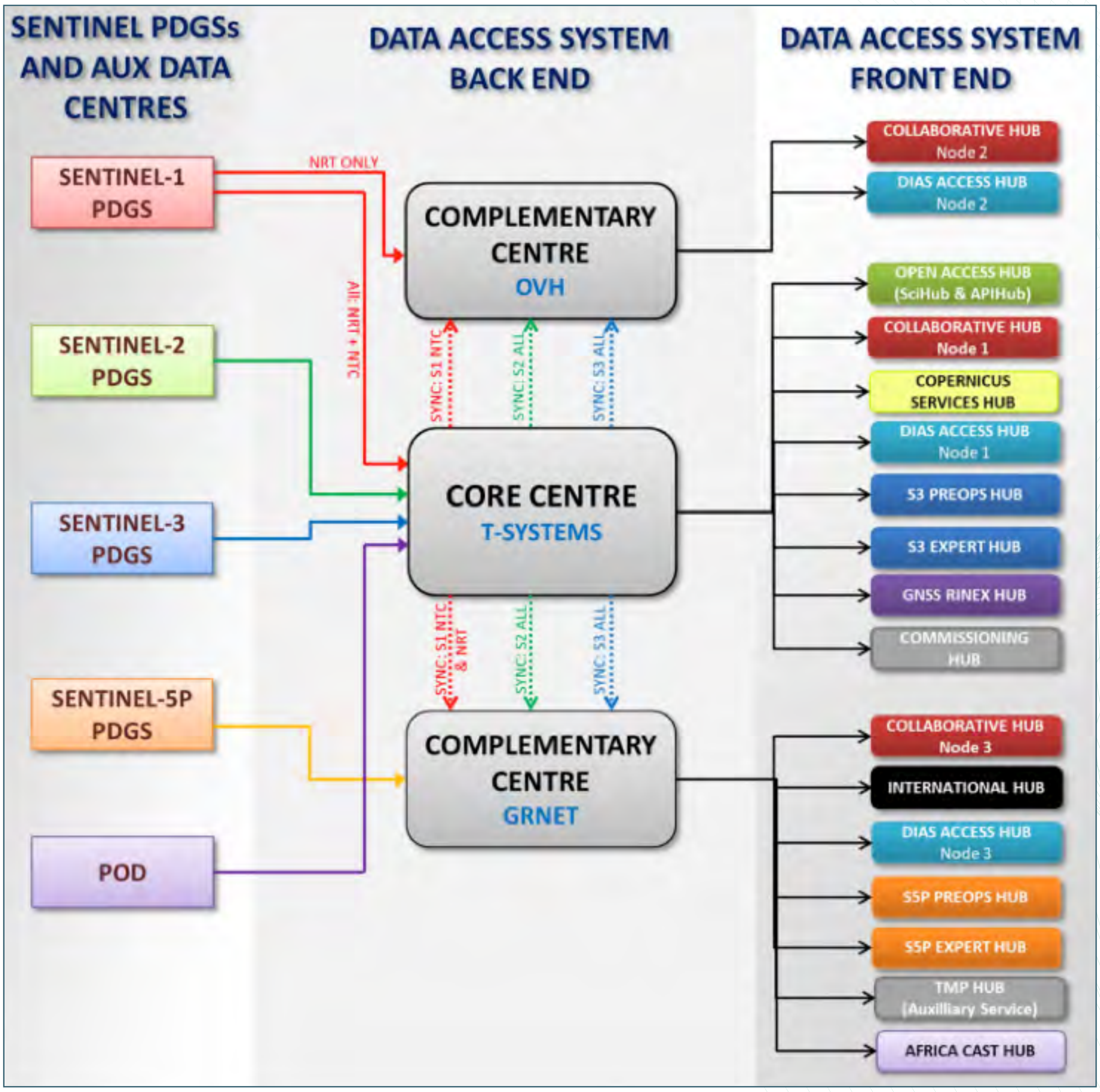


Figure 5. Data Access System Physical Architecture Overview

Going forward the evolution of the ESA Ground Segment will move products older than, for example, 18 months, from the online Sentinel Data Hubs to a Long Term Archive (LTA) services, already present as part of the Sentinels' Processing and Archiving Centres (PACs). The strategy is also to progressively phase out centralised (batch) L2 level processing and storage of Copernicus L2 products. Instead, the objective is to enable higher level products processing on-demand using the cloud environment and taking advantage of the Long Term Data Archiving (LTA) of L0 and L1 in different PACs via LTA broker service.

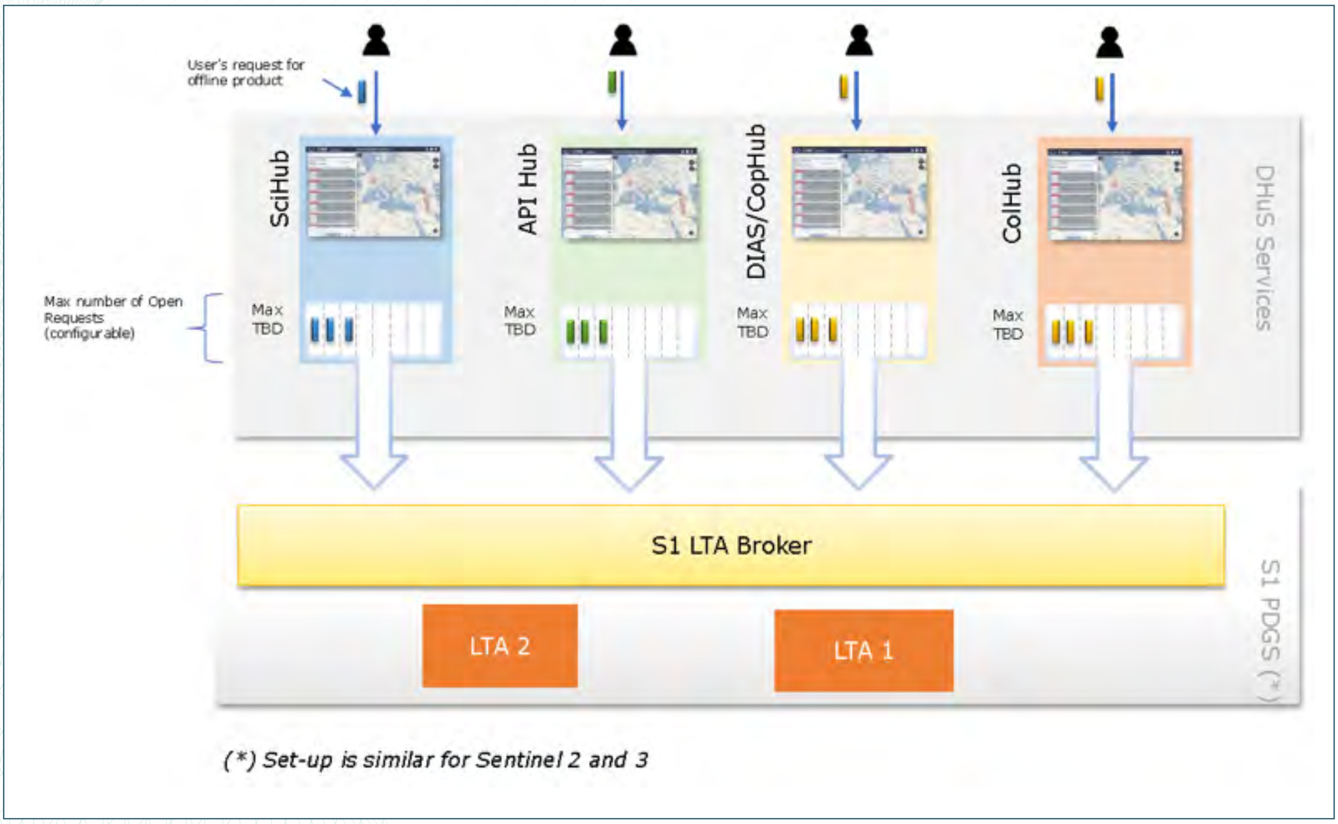
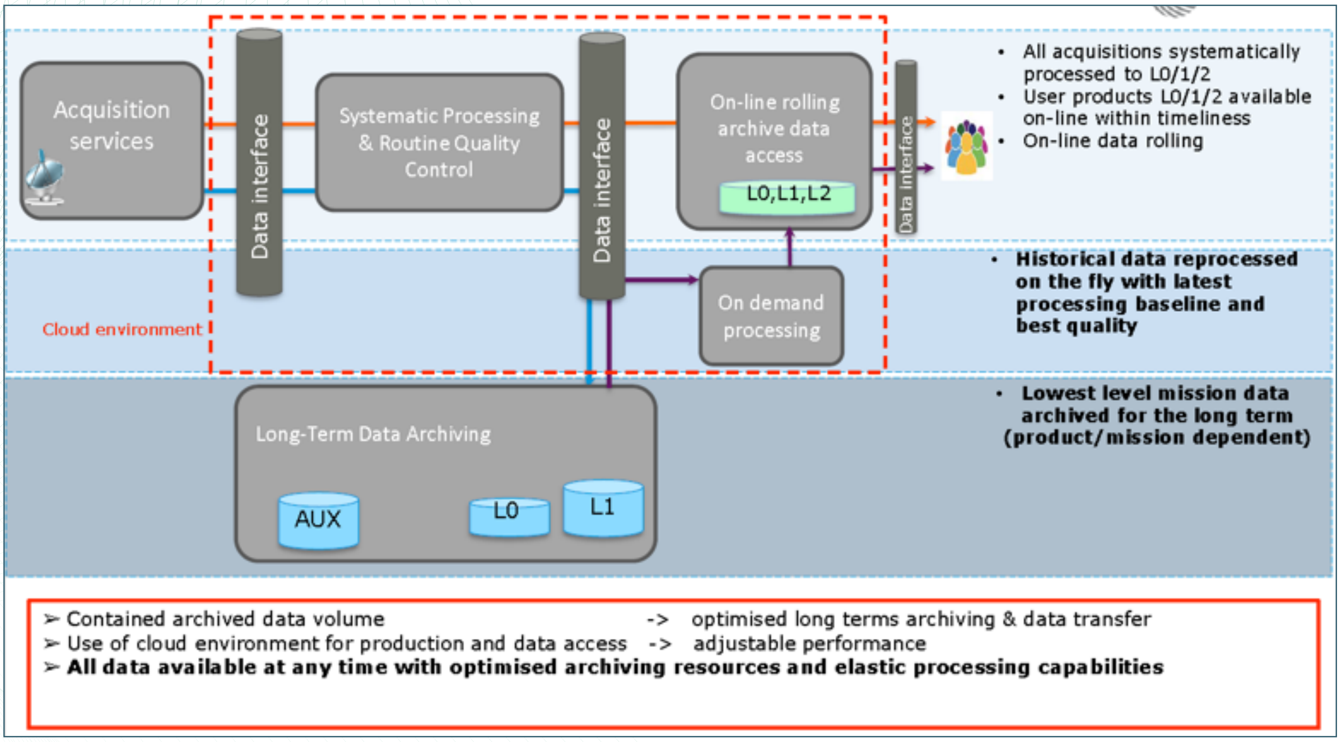

User's request for offline product
SciHub
API Hub
DIAS/CopHub
ColHub
DHuS Services
Max number of Open Requests (configurable)
Max TBD
Max TBD
Max TBD
Max TBD
S1 LTA Broker
LTA 2
LTA 1
S1 PDGS (*)
(*) Set-up is similar for Sentinel 2 and 3

Figure 6. Long Term Data Archive (LTA) functionality as exemplified for the Sentinel 1 data.



- All acquisitions systematically processed to L0/1/2
- User products L0/1/2 available on-line within timeliness
- On-line data rolling

- **Historical data reprocessed on the fly with latest processing baseline and best quality**

- **Lowest level mission data archived for the long term (product/mission dependent)**

➢ Contained archived data volume -> optimised long terms archiving & data transfer
➢ Use of cloud environment for production and data access -> adjustable performance
➢ **All data available at any time with optimised archiving resources and elastic processing capabilities**

The role of Data Provenance and Traceability

Data lineage and provenance is of critical importance for Copernicus data flows. However, currently, there are no resilient means for the systematic verification of the EO products integrity or time-stamp which would for example, ensure the identification of the source of the original Sentinel data or protect data against different potential forms of malicious tampering. The current baseline approach is to compute an MD5 hash of the SAFE (Sentinel Standard Archive Format for Europe) package (as zipped) however, this is only suitable for detection of short-term unintentional data alterations (e.g. corruption if copying data over network). Currently the SAFE packages of the Sentinel products do include the limited information of the provenance - e.g. the description of the processing resources used, however, better verification of that information is needed.

The Copernicus Sentinels Product Traceability Service is currently being planned to address some of these issues[7]. The service, which details were revealed in 2019, has several objectives: to record product lifecycle events, to allow the retrieval of product history, to verify if a product copy is genuine and to enhance trust in processing chains[8]. The service is planned to operate to address the exponential growth of available products and its technology stack includes:

- traces,

- hash functions,

- the certification process and

- digital signatures.

It will allow, for example, verification of a CollGS data collection as compared to the ESA PDGS data collection and provide an independent record of what is available.

It is therefore a part of an ESA long-term strategy to enable objective verification of the provenance of Copernicus EO products and thus address data security. This is seen as a general "must have" feature from security and auditability point of view, and an important element of the future development of the Sentinel ground segment capabilities, independent of particular data applications.

The requirement to address a security (immutability) of a provenance trail of the EO products supply chain is also seen as an important feature of the future traceability service, however provenance analytics and visualization for mining and extracting knowledge from provenance data, is still largely unexplored. This, on the other hand, becomes particularly relevant for all of the on-demand processing scenarios. For example, one of the requirement could be to provide the user with the provenance trail of an on-demand product (down to the L-0 product or products), one that captures the details of the inputs used for processing (data, processor) as well as details of the processing itself and the product created as a result. Moreover, as the ESA Core Ground Segment intends to involve cloud services and external parties in various pre-processing tasks (e.g. DIAS), being able to grow and verify the provenance trail across different infrastructures becomes paramount.

Implementation options

There are several possible implementations for the provenance chain which rely on hashes calculated from EO products and processors.

Public Key Infrastructure (PKI) is a technology traditionally used for authenticating users and devices participating to the data value chain. The PKI is typically implemented by a trust authority which provides s a set of roles, policies, hardware, software and procedures needed to create, manage, distribute, use, store and revoke digital certificates and manage public-key encryption[9].

The KSI keyless signatures, on the other hand, is a technology designed for automated verification of digital signatures based on record of data integrity and time. The main difference between two approaches is that while PKI relies on the continuous secrecy of private keys, which is necessary for the identification of the origin, keyless signatures only relies on cryptographic properties of hash functions and the availability of widely published verification codes[10].

Public Key Infrastructure (PKI) and keyless signatures both intend to make electronic data more reliable by providing mechanisms for identifying the origin of data and to create irrefutable proofs of applied data processing steps.

The established approach based on the PKI approach is to generate Traces (hashes) and store them in Traceability Service (e.g. a Cloud database) which is can be verified using two cryptographically connected keys: a public key that is made widely available and acts as authentication anchor, and a private key that is used to produce digital signatures. Such digital signature allows for verifying authenticity of digital traces: recipients of digitally signed messages can verify the origin and integrity of a received message by checking that the attached signature is valid under the public key of the expected sender. The management of such signatures is bestowed to the Certifying Authority (CA) tasked with delivery of digital certificates, ownership of a public and private keys, verification of the key and the identity of its owner, as well as the certificate's content[11].

The EO data provenance service implemented using blockchain technology as a proof-of-concept (PoC) can add additional complementary features to these functions. The cryptography behind the KSI(®) Blockchain signatures ensures that they never expire and remain quantum-immune i.e. secure even after the realization of quantum computation. In this sense, KSI Blockchain can, for example, “indemnify” PKI against the cryptographic threat of practical quantum computers[12]. Moreover the signatures (hashes) are stored in the immutable time-stamped blockchain to ensure immutability of the provenance chain information. A simple comparison of the two concepts is presented in Figure 7 highlighting the added value and complementarities of both solutions.

KSI Blockchain EO data provenance concept	Sentinels Product Traceability Service concept
Solution / Vision	
Rely on hashes calculated from EO products and processors.	
Signatures (hashes) are stored in an immutable and time-stamped blockchain.	Traces (hashes) are stored in Traceability Service (presumably a Cloud database). Need to clarify the storing, protecting and accessing of the hashes.
Storing signatures to KSI blockchain authorized with user/ password, additional metadata can be added (e.g. PKI signature)	Storing signed data to database verified using traditional PKI
Each operation leaves a time-stamped record in blockchain.	Each operation generates a Trace (json) to the database.
Quantum threat - not vulnerable	Quantum threat - vulnerable (PKI)
Offline verification possible (publication code from widely witnessed event)	Verification relies on online services (CA, Cloud infrastructure availability)
Challenges	
Processors need to be able to reproduce products byte-by-byte.	
Handling the unique ID of different EO products needs to be addressed.	
Scalability addressed at the system architecture level and based on industrial use of KSI blockchain in other sectors	The complexities and cost of key-management make it hard to use PKI for managing integrity at scale.

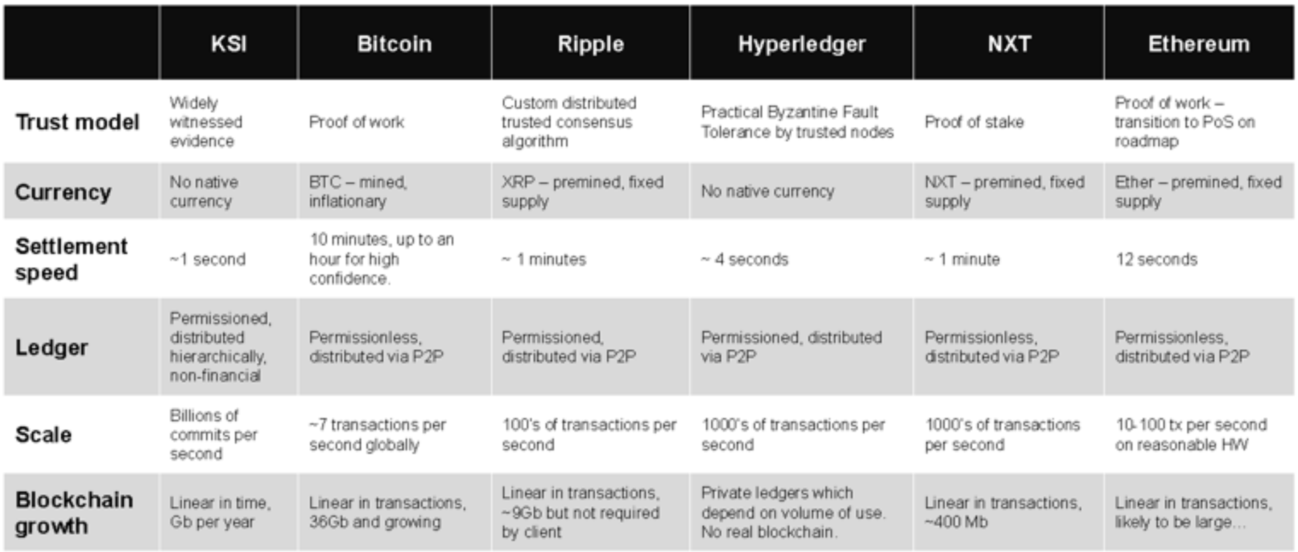
EO PRODUCT PROVENANCE WITH KSI BLOCKCHAIN

The implementation of the KSI Blockchain for the EO data provenance is a general technology demonstrator prototyping a set of new technologies to enable secured and traceable exploitation of data.

In the past two years the blockchain technology has emerged as candidate for addressing the automated audit trail and the so called trusted data sharing. The implementation scenario presented in this chapter is an example of the permissioned blockchain which is taking advantage of the mature solutions which is an example of an "industrial scale" deployment.

KSI Blockchain introduction

There is a number of blockchain infrastructures currently at focus of different pilot projects addressing cybersecurity or digital assets sharing. KSI Blockchain functionality delivered in the context of the ESA pilot focused on demonstration of the APIs for cryptographic proof of data integrity, data provenance, and asset transfer. It provided a demonstration on an "enterprise solution" which is a permissioned DLT platform, designed for use in operational contexts, that delivers some key differentiating capabilities over other popular platforms: tagging system for electronic data designed for ingestion of data at a very large scale, a signature response in seconds (as opposed to minutes) and independent verification by third parties.



	KSI	Bitcoin	Ripple	Hyperledger	NXT	Ethereum
Trust model	Widely witnessed evidence	Proof of work	Custom distributed trusted consensus algorithm	Practical Byzantine Fault Tolerance by trusted nodes	Proof of stake	Proof of work – transition to PoS on roadmap
Currency	No native currency	BTC – mined, inflationary	XRP – premined, fixed supply	No native currency	NXT – premined, fixed supply	Ether – premined, fixed supply
Settlement speed	~1 second	10 minutes, up to an hour for high confidence.	~ 1 minutes	~ 4 seconds	~ 1 minute	12 seconds
Ledger	Permissioned, distributed hierarchically, non-financial	Permissionless, distributed via P2P	Permissioned, distributed via P2P	Permissioned, distributed via P2P	Permissionless, distributed via P2P	Permissionless, distributed via P2P
Scale	Billions of commits per second	~7 transactions per second globally	100's of transactions per second	1000's of transactions per second	1000's of transactions per second	10-100 tx per second on reasonable HW
Blockchain growth	Linear in time, Gb per year	Linear in transactions, 36Gb and growing	Linear in transactions, ~9Gb but not required by client	Private ledgers which depend on volume of use. No real blockchain.	Linear in transactions, ~400 Mb	Linear in transactions, likely to be large…

Figure 8. Comparison of blockchain / distributed ledger technologies.

The Figure 8 exemplifies the key features of KSI Blockchain platform as compared to other popular blockchain platforms highlighting its usability at industrial scale[13]:

Scalability: One of the most significant challenges with traditional blockchain approaches is scalability – they scale at O(n) complexity i.e. they grow linearly with the number of transactions. In contrast the KSI blockchain scales at O(t) complexity – it grows linearly with time and independently from the number of transactions.

Settlement time: In contrast to the widely distributed crypto-currency approach, the number of participants in KSI Blockchain distributed consensus protocol is limited. By limiting the number of participants it becomes possible to achieve consensus synchronously, eliminating the need for Proof of Work and ensuring settlement can occur within one second.

Trust model - The properties of the signed data can be verified without reliance or need for a trusted authority. The hash is published in widely witnessed ways e.g. on the website, or in a newspaper and other public media. This provides a reliable trust anchor as distribution is wide and numerous and copies are retained long-term.

Data Privacy - KSI Blockchain does not ingest any customer data; data never leaves the customer premises. Instead the system is based on one-way cryptographic hash functions that result in hash values uniquely representing the data, but are irreversible such that one cannot start with the hash value and reconstruct the data – data privacy is guaranteed at all times.

User requirement

ID	Requirement
REQ-U01	Ability to verify the time and integrity of an EO product.
REQ-U02	Ability to verify the time and integrity of the provenance of the EO product - direct and indirect input EO products, data processors together with their configuration and other local inputs, down to the downlinked raw stream.
REQ-U03	Ability to support different type of EO products in different formats (e.g. SAFE, netCDF, TIFF) without changing the overall architecture and only developing only part that is specific to format (e.g normalization + hashing).
REQ-U04	Ability to provide the proofs (for both EO product and its provenance) for an on-demand EO product that is generated on-the fly (tens of thousands per day). The producer may not want to retain the product, its provenance info or proofs for storage efficiency purposes.
REQ-U05	Ability to verify the "identity" of the data processor captured in the provenance chain. Such verification may not necessarily need the non-repudiation (e.g. KSI identity metadata may be sufficient).
REQ-U06	Ability to build the provenance chain by multiple parties without a central service that serves all these parties and maintains all the provenance information and proofs they are interested in retaining.
REQ-U07	Ability to verify the integrity of EO product and visualize its provenance without access to any online services or resources. Such verification and visualization function should have both the user interface and API.
REQ-U08	Ability to support future cloud-based deployments of EO processing facilities.
REQ-U09	Ability to revoke old EO products which have been already distributed because a new (improved, corrected) version of the product has been made available.

Technical requirement	
ID	**Requirement**
REQ-T01	In order to unambiguously identify both the resources (e.g. EO products) as well as processors in the entire EO provenance trail created by many parties, a global identification scheme (similar to URI) is needed.
REQ-T02	The amount of EO products of the Copernicus program to be supported is approximately 20 000 across all missions per 24h. This results in roughly 15 TB of data. This will grow when Sentinel-4,5 are launched.
REQ-T03	The system must contain a common product model which applies to all Earth Observation products (Java interface).
REQ-T04	The system must contain a common processor model which applies to all systems processing EO products.
REQ-T05	The system must contain a DataFormatAdapter interface which can be implemented to support hashing of any of EO products.
REQ-T06	The hashing of EO products must be deterministic.
REQ-T07	The system must contain a library for integration into arbitrary JVM-based software.
REQ-T08	The system must be able to chain together multiple EO products and the baseline processors into a unified container.
REQ-T09	The system must be able to extend the chain in the container each time a new product is generated.
REQ-T10	Each resource in the container can contain additional metadata (e.g. adding additional identity).
REQ-T11	The system must be able to verify the integrity and time of the container files above.

Overall concept

During the first phase of the BC4SA project, the following components have been developed and implemented in response to the user and technical requirements:

- REST API (Java implementation) for operations between KSI blockchain and EO products;
- Java interfaces for generic EO product and product format adapters;
- Implementation of the above interfaces for Copernicus SAFE and GeoTIFF based products;
- Web application (Java based) which simulates the Core Ground Stations and Processing and Archiving Centers;
- Prototype implementation for DHuS supporting blockchain secured products;
- Sandbox testing environment (Docker based) connecting all the modules above;

They are described in details below.

Provenance system functional breakdown

The following diagram illustrates the top-level breakdown of the system into functional logical components. Each component contains a free text description of the functionality and the implementation idea.

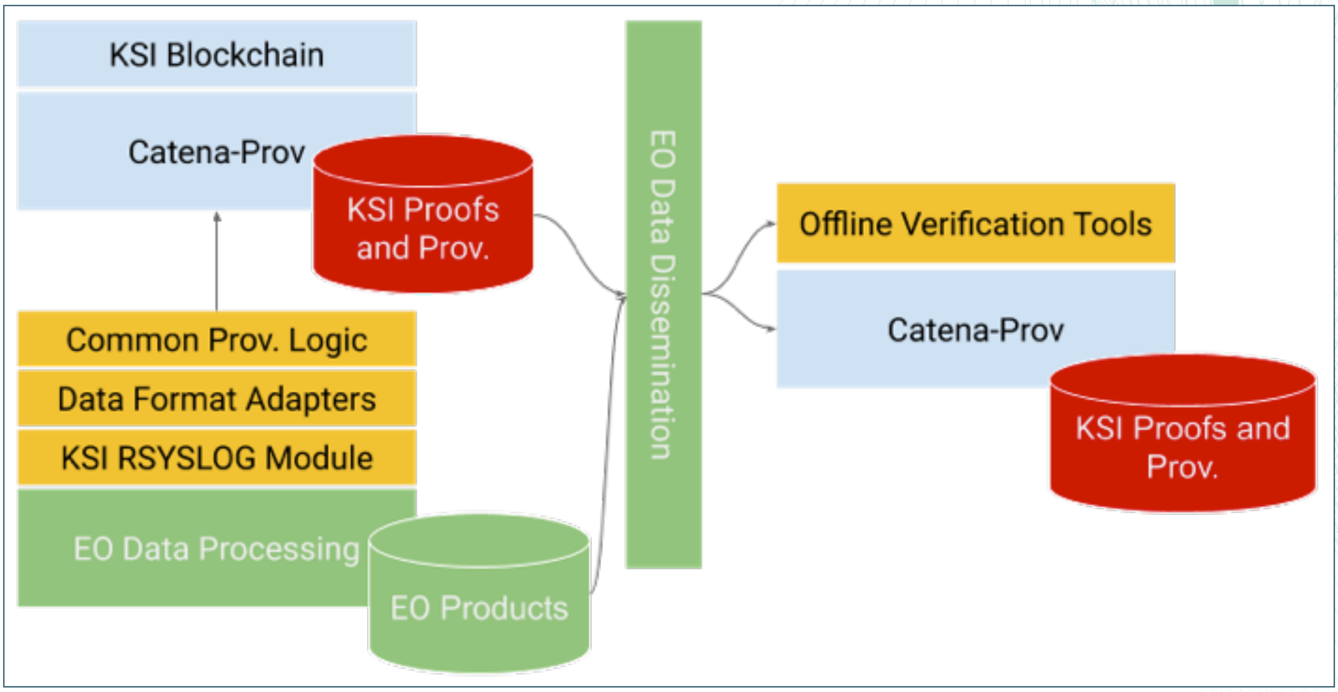


Figure 9 Top-level breakdown of the system

The **KSI Blockchain** is used as a service through the **Catena-Prov middleware application** which provides provenance functionality. This requires each organization (legal entity) to deploy:

a. KSI Gateway (a cluster with 2 or more members depending on the high-availability needed)

b. One or more Catena-Prov middleware instances together with authentication proxy layer as appropriate to the organization (e.g. LDAP).

The resource requirements for both KSI Gateway and Catena-Prov are very small. Only the disk space required by the PostgreSQL database to persist the proofs, is significant, however still it does not require large storage relative to the space required to persist the EO products themselves.

The existing EO data processing, archiving and dissemination systems (in green) are expected to be maintained as they are, however enhancements are required to integrate them for the generation, dissemination and verification of the proofs as necessary. The implementation depends on the technical realization of each such component.

- **Input**: arbitrary hashed data (matching KSI DataHash model)

- O**utput**: signature file for later verification

- Relevant requirement(s): **T02**

The **Common Prov. Logic** layer is implemented as a library that helps integrating the EO systems by realizing the provenance logic that is common to all EO systems and providing interface that is optimized to this functionality (instead of generic Catena-Prov REST API). Also, this component can easily be used for automated periodic verification of the EO products persisting in the archive.

- **Interfaces**

    - **Resource** model - matching a generic EO Product.

        - Id, contentType, algorithmVersion, dataHash

    - **Processor** model - matching a generic Processor.

        - Id, annotationsList, dataHash

- Relevant requirement(s): **T03, T04, T07, T08, T09, T10**

The **Data Format Adapters** are implemented as libraries and handle the specifics of a particular EO product data format, such as computing the hash of that product. Detailed description in the following chapter.

- Relevant requirement(s): **T05, T06**

The **KSI protected RSyslog** module protects the log records generated by the used modules to avoid unintentional or intentional tampering. This module enables efficient log signing using KSI blockchain. It provides long-term proof of integrity and time of log records. The module is coming with the tool - *logksi* - offering commands to integrate, verify, sign, extend and extract log signatures.

- **Input**: log records streamed by RSyslog

- **Output**: efficient periodic signatures matching the log records

The **Offline Verification Tools** are used to independently verify the provenance and proofs by any party. Proofs for that purpose are exported from a Catena instance in KSI Envelope format. The command-line tool for verifying the KSI Envelope has a built-in provenance verification policy which handles the case of externally persisted data (as files on the file system) and checks the links between the provenance entities. However, an EO provenance specific verification policy would be required to implement as it has to make use of the data format adapters to compute hash of EO products.

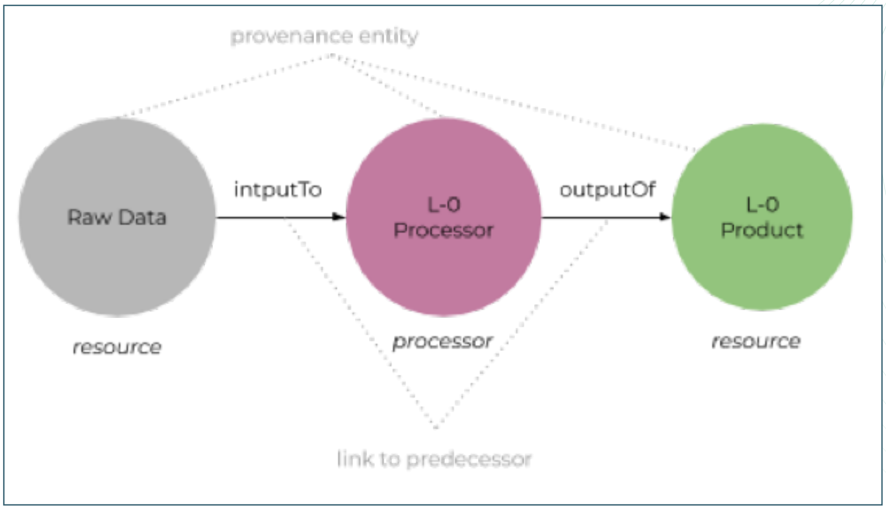
- **Input**: signature file, input data, [publication-code* (optional for fully offline verification)]
- **Output**: verification result
- Relevant requirement(s): **T11**

Provenance model

The provenance function in KSI blockchain has an abstract concept of a provenance entity and links to zero or more preceding entities (called predecessors) to reflect the relationship between the entities in order to form an immutable provenance graph. The mapping of EO product supply chain entails the following elements:

- There are 2 (general) types of provenance entities: *resource and processor.*
- If a *resource* is a *predecessor* of a processor, it has a meaning that the resource was an input to the processor and the association is named "inputTo" in provenance chain.
- If a *processor* is a predecessor of a *resource*, it has a meaning that the resource is the output of the processor and the association is named "outputOf".

The following diagram illustrates:



The number of links between the provenance entities is not limited (there can be even more than one link between the same two entities), thus adding new links with other meanings is always possible, between whatever two entity types.

EO products hashing

In most cases the *resource entity* in Catena corresponds to an EO product (which can be a single image file such TIFF or a complex package such as SAFE). Other types of inputs and outputs (such as a processor's configuration file) can be captured as a *resource* if this is reused between multiple processors.

In Catena each provenance entity captures three mandatory elements - the hash of the corresponding data and its name and content type. In case of the *resource* entity that corresponds to an EO product:

- The **dataHash** is the hash of the EO product which is computed according to the format of the product in a deterministic manner. In case of a TIFF image, this can be as simple as hashing the stream/file of the TIFF "as is" but in case of complex structure such as SAFE a dedicated algorithm is necessary to construct a single hash of all elements in the SAFE package.
- The **name** is a globally unique resource name of the EO product, so that all participants of the supply chain can unambiguously verify that product and use it as input for further processing across system and organizational boundaries.
- The **contentType** reflects the format of the underlying data and how it should be interpreted, e.g. image/tiff or application/vnd.esa.safe.sentinel1-v1. This is also used as a basis to select the right data format adapter (for the normalization of the data).

The responsibility of computing the hash value for the **dataHash** is delegated to the corresponding data format adapter.

The **name** annotation is formed using URN with the following syntax:

urn:eo:<authority>:product:<productID>:<version> where:

- <**authority**> is the domain name of the producer of the product, e.g. esa.int or sentinel.esa.int
- <**productID**> is the unique product identifier as assigned by the authority, e.g. **S1A_EW_RAW__0SDH_20190117T102422_20190117T102530_025516_02D453_B320.SAFE**
- <**version**> is the version of the product, e.g. 1, 2, etc.

Full sample for a Sentinel 1A product issued by ESA would be:

urn:eo:sentinel.esa.int:product:S1A_EW_RAW__0SDH_20190117T102422_20190117T102530_025516_02D453_B320.SAFE:1.

Another annotation **eo.hashing.dataNormAlgVer** will be used to capture the version of the algorithm that was used for the normalization of the data by the corresponding data format adapter. The value of the version is a string with the format "nv1" (first version), "nv2" (second version), etc. and the possible values are "maintained" by the corresponding data format adapter that knows how to interpret each of them.

Processing algorithms hashing

In the context of the EO product provenance the *processor* entity corresponds to a “unit of logic” that produces an EO product using existing EO products and other inputs. At lower level the processor may be implemented by a sequence of processing algorithms each doing a specific type work but these steps as well as the intermediate results do not need to be exposed in the provenance chain.

In case of the *processor* entity:

- The **dataHash** is the hash of the baseline of the processor. The computation of such hash heavily depends on the specifics of the processor but overall the binary code as well as major configuration files make sense to be involved.
- The **name** and **contentType** annotation collectively form a globally unique identifier of the instance of the processor, in order to avoid ambiguity in terms of who and where performed the processing.

For capturing other “local” inputs to the processor, custom annotations of the processor entity are used as follows.

Key	Value	Notes
eo.processor.in.<n>.id	Identifier of the local resource, e.g path to config value.	<n> is an integer used to map the identifier of the resource to its hash.
eo.processor.in.<n>.hash	A hash imprint (includes both hash algorithm identifier as well as value)	It is assumed that such hash is a file which can be hashed “as is” without normalization.

In case the value of the local input is a primitive (e.g. a short string, integer, etc.), its value could be captured directly using simply a custom annotation specific to this processor, e.g. eo.processor.processorX.parameterZ=18

In addition, the following standard annotations are foreseen for the processor entity:

Key	Value	Notes
eo.processor.start	The start time of processing	
eo.processor.stop	The stop time of processing	
eo.processor.org	The organization running the processor, e.g. ESA	
eo.processor.site	The site running the processor, e.g. ESRIN PDGS, CollGS,etc	

It should be noted that the processor should be overall a pure function (always producing the same result byte-by-byte, e.g. dates, floating point calculation non-determinism) so a later verification could be done on demand. In parts where this is not possible, some data normalization should be used.

Provenance Graph Management

The provenance graph is amended each time a new EO product is generated (and potentially archived) by the processor. The processor is responsible for triggering this activity and providing the necessary input information required for the processor and resource entities. The necessary lower level functions (e.g. computation of the hash of the EO product, registering the data in Catena and KSI Blockchain) will be implemented by the data format adapters and other reusable libraries. These are implemented as bc4sa-data-format-adapters and bc4sa-common modules respectively.

If the processor uses existing EO products as input (and these products are known to be captured in provenance chain):

1. The processor entity must use these products as predecessors in the provenance graph.
2. The integrity of these input products should be verified at least asynchronously whenever using them as predecessors (ideally, before using them as input to the processing but this has impact to performance, thus not feasible everywhere).

Exchange of Provenance Graph

As EO products are exchanged between multiple parties, the provenance graph for verification and amending must also be made available between them. This includes all types of parties - the ones producing new products and want to amend the provenance graph as well as parties that are interested only in verification. Since it is not feasible to have one online service that would maintain the global provenance graph for all parties, the provenance graph (the necessary sub-graphs) need to be exchanged between participants as they exchange the EO products.

The base technical capability to export the provenance graph from Catena as a KSI envelope (zip archive file) already exists - it then can be imported to another instance of Catena or verified offline. For verification of KSI envelope, the SDK and command-line tool are available but need to be enhanced to include the specifics of EO products (e.g. how a hash of a SAFE package is computed). These are implemented as modular plugins to the SDK or command-line tool.

Providing the Provenance Graph

The provenance graph is made available by an EO producer over the same technical interface as the corresponding EO product - e.g. if there is a REST API for obtaining the EO product, there will be another endpoint (or improvement to existing endpoint) in this REST API for obtaining the provenance graph.

By default the graph would contain all (both indirect and direct) preceding entities of the given EO product, however, this may be fine-tuned as necessary for each case, e.g. the REST API endpoint may have additional parameters for the desired depth.

For serving the provenance graph requests, the provider can take the implementation approach that best suits the type of the EO product and the interface used for distribution, e.g:

- Query the provenance graph from Catena on the fly when serving the request;

- Export the provenance graph immediately after the creation of the EO product (e.g. when EO product is made available over an FTP site, the provenance graph is also provided as a file on the same FTP site).

In case of on-demand EO products which are generated on the fly, the provider can persist the corresponding provenance graph in his Catena for a limited time, so that users can download it if necessary and the provider does not need to keep it forever (which may not be economically feasible).

Verification of the EO Products

The verification of EO products can be divided into the following use cases:

- Automated (integrated) verification by a system (e.g. before using the EO product as input, after receiving it from a remote system, etc.);
- Manual verification by a human (e.g. before using it for analysis, proving integrity to a 3rd party).

The internal logic of verification (e.g. the checks performed during verification) will be the same for both cases, however, the interface for the functionality will be different. For automated verification by a system, the SDK will be provided whereas for manual verification a command-line tool is used. Also, automated verification may be performed by fetching the provenance graph entities directly from Catena, instead of using a provenance graph in KSI envelope.

When is Verification Performed?

In case of manual verification, the verification is triggered by human using the corresponding tool. In case of automated verification, the EO product should be verified when:

- the EO product is used as an input to create a new product;
- the EO product is imported from an external system;
- the EO product is downloaded by an external system or user;
- the verification (result) of the EO product has expired.

The latter is expected to be performed by a scheduled periodic process that monitors the verification results and triggers the verification if results are expired.

As verification failures are expected to be very seldom, the following approach is used to have as little impact on system performance:

- the verification is performed in a non-blocking manner (e.g. if the verification is triggered during the download of the EO product, the download can proceed without waiting for the verification to complete);
- the verification results are cached (for a configurable period of time) so that cached results can be used instead of performing the verification.

Verification results should be always written to the system (audit) log for two purposes:

- This allows the detection of tampering with the cached verification results as system log can be easily protected using KSI Blockchain (using the Rsyslog module);
- In case of verification failure, the alarm can be raised using standard log monitoring tools (which are necessary for other purposes anyway and thus can be reused).

How is Verification Performed?

The input to the verification process is:

- one or more EO products (as a file);
- the provenance graph (in Catena or as a KSI Envelope) that contains the proofs for these products.

It shall be noted that verification should be possible even if not all EO products of the given provenance graph are available, e.g. an end-user verifying L-2 product may not have the L-0 product. However, when the products are available, they should all be always provided to the verification process together, in order to make sure that they really are part of the same provenance graph.

The verification can be divided into the following major parts:

A. Verifying that the provenance graph as such is valid. This includes:

   a. Verification of the provenance entities (according to KSI Envelope specification), including the KSI signatures that protect this information.

   b. Verification of the links between provenance entities (according to Catena Provenance logic) are correct.

B. Verifying the EO domain specific aspects:

   a. Computing the hash of the given EO products and comparing them to the hash in the corresponding provenance entities (must be equal).

   b. Reading the production time of the EO product and comparing it to the time of the corresponding KSI signature (must not differentiate more than the configured threshold).

The part A is not specific to EO domain and provided by standard KSI SDK-s. The part B is EO domain specific and implemented as part of this project.

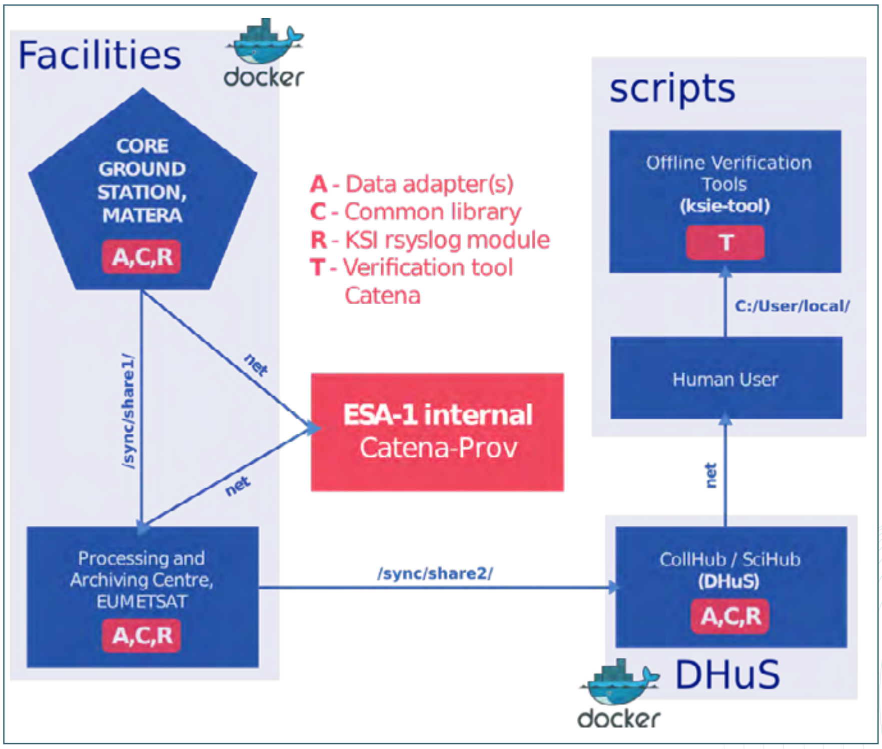
The mapping between the provenance entities and EO products will be done based on the name attribute of the provenance entity (globally unique resource name details described in the section “Resource Entities”). The data format adapter to be used, is derived from the contentType attribute.

The possible verification results are as follows:

Verification Result	Description
RESULT_NOK	- Verification of the provenance entities (according to KSIE specification) returned RESULT_NOK - Verification of the links between provenance entities failed - The computed hash of the EO product did not match the one in the provenance entity - The production time of the EO product differs from the KSI signature more than allowed by threshold.
RESULT_WARN	- Verification of the provenance entities (according to KSIE specification) returned RESULT_WARN - One or more predecessors of the provenance graph were missing - Not all EO products listed in the provenance graph were provided as input to the verification process
PROOF_NA	- The provenance entity for the EO product was not found. - The verification of the KSI signature (as part of the KSIE verification) returned NA
RESULT_OK	- All other cases

System components dynamic interaction

The following diagram illustrates the high level modules which were developed for the sandbox environment (**BC4SA-Docker**). The lines illustrate the API calls and data movement between each component. The names match their real life counterparts and would be integrated at those locations respectively.



The environment is composed of the following modules:

- **facilities** - this module simulates the features of a Core Ground Station (CGS) and a Processing and Archiving Center (PAC).
- **dhus** - this is an empty running DHuS instance with the proof-of-concept KSI implementation.
- **explorer** - this module contains the ksi-provenance-explorer for visualizing the provenance chains.
- **rsyslog** - this module captures the log messages coming from the running dhus instance and secures them with the rsyslog-ksi-ls12 plugin.
- **catena** - a clean installation of Catena for storing the signatures and envelope files.
- **auth** - an LDAP authentication server which mediates the communication between Catena and other parties.

The deployment guide for this environment can be found in BC4SA-startup-guide.

The screencast showing the main use cases of this software stack:

https://drive.google.com/file/d/1fXU1FOfEWCGusXss4yqA-Sp48agD9I1I/view?usp=sharing

SUMMARY AND CONCLUSIONS

The recent OGC Engineering Report of Federated Cloud Provenance provides an overview of the state-of art concepts related to “digital provenance” which refers to collecting and sharing information about production of digital data and processing workflows, which can be used to form assessments about its quality, reliability, or trustworthiness[14]. The OGC survey of the areas for provenance research and development emphasize the following high level elements for consideration:

- Interoperability for different provenance systems and tools to aid in the integration of provenance information.
- Information management infrastructure to manage growing volume of provenance data.
- Provenance analytics and visualization for mining and extracting knowledge from provenance data.
- Data provenance security and inference control.

The next steps for the BS4SA project include:

- testing implementation of the software developed under the prototyping phase (traceability services for PDGS) in particular to benchmark with other solutions
- demonstrate added value of KSI blockchain solution managing provenance information in Federated Clouds in line with the recommendations of the OGC Testbed 15 engineering report
- implementation involving value adding use cases (EO services Quality Assurance Facility) including demonstration of the traceability services.

Information on the project progress available at:

https://eo4society.esa.int/communities/blockchain-distributed-ledgers-and-eo

REFERENCES

[1] http://emits.sso.esa.int/emits-doc/ESTEC/News/GSTPE1-DevelopCompendium2017.pdf

[2] Nightingale J. et al. Quality Assurance Framework Development Based on Six New ECV Data Products to Enhance User Confidence for Climate Applications, Open Access Remote Sens. 2018, 10(8), 1254; https://doi.org/10.3390/rs10081254

[3] The German Remote Sensing Data Center (DFD) at DLR at https://www.dlr.de/eoc/en/desktopdefault.aspx/tabid-10192/17388_read-41807/

[4] Copernicus Data Access Annual Report 2018 available at https://scihub.copernicus.eu/twiki/pub/SciHubWebPortal/AnnualReport2018/COPE-SERCO-RP-19-0389_-_Sentinel_Data_Access_Annual_Report_Y2018_v1.0.pdf

[5] Copernicus Data Access Annual Report 2018

[6] Long Term Archive presentation available at http://emits.sso.esa.int/emits-doc/ESRIN/1-9950_CSC/LTA_Copernicus_Industry_Day_20190712-final.pdf

[7] Collaborative Ground Segment Workshop Report, 14–15 October 2019 available at https://sentinel.esa.int/documents/247904/1935538/CollaborativeGS-executive-summary-2019.pdf

[8] Demanage C. (2019) Traceability Service for EO data Products available at https://sentinel.esa.int/documents/247904/3963136/8.Traceability-CollGS-Oct-2019.pdf

[9] Public Key Infrastructure https://en.wikipedia.org/wiki/Public_key_infrastructure

[10] (PDF) Keyless signature infrastructure and PKI: hash-tree signatures in pre- and post-quantum world. Available from: https://www.researchgate.net/publication/313235634_Keyless_signature_infrastructure_and_PKI_hash-tree_signatures_in_pre-_and_post-quantum_world [accessed Feb 03 2020].

[11] Traceability Service for EO Data Products available at https://sentinel.esa.int/documents/247904/3963136/8.Traceability-CollGS-Oct-2019.pdf

[12] Ibid.

[13] https://www.guardtime-federal.com/ksi

[14] Fellah S., OGC Testbed-15: Federated Cloud Provenance ER available at http://www.opengis.net/doc/PER/t14-ID