

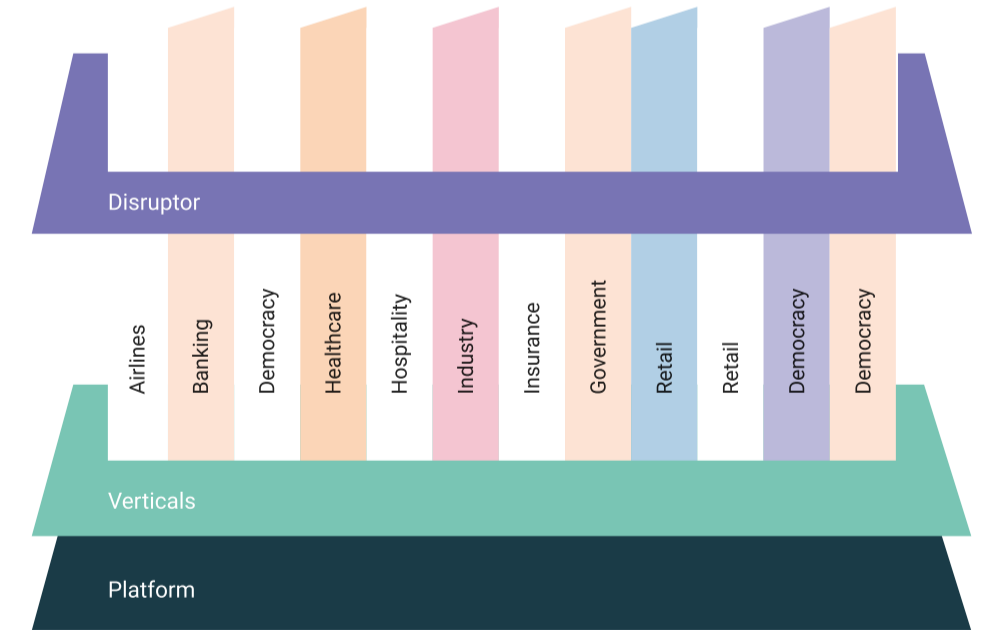


**Criteria for open platform in AHA and AAL domains which onsait platform complies with:**

- Open Source
- Open Standards Based
- Federatable
- Shared Common Information Models
- Vendor and Technology Neutral
- Supports Open Data
- Provides Open APIs



- [Vision](#)
- [Proposal](#)
- [Business Overview](#)
- [Technical Overview](#)
- [Contextual Overview](#)
- [Success Stories](#)



Onesait Platform provides the flexibility so that developers can build their own solutions in a solid and agile way using Open Source technologies, a flexible architecture and an innovative approach.

Based on Open Source components, Onesait Platform covers the entire life cycle of information (from ingest to visualization through its process and analysis).

It offers a unified web console for the development and operation profiles of the solutions.

Open Usage  
(adoptability)

Open Adaptation

Thanks to different assistants and the encapsulation of best practices, it is possible to develop systems with complex architectures in a simple way, reducing cost and time2market.

### Objectives

Onesait innovation suite for the creation of new businesses allow organisations to compete in new value chains and occupy new spaces.

Onesait vertical suites for a quick and efficient response to operational and transformation challenges in 12 sectors. A specific offering designed for each sector and with the ambition to provide an agile and dynamic digital evolution.

Onesait technological platform provides the flexibility so that organisations can build their own solutions in a solid and agile way using Open Source technologies, a flexible architecture and an innovative approach.

[Learn more about Onesait](#)

## Proposal

- **Integrated Government** of the elements that make up the platform.
- **Flexibility** in the use of components, with the possibility of replacing them without impact on business

- **Ability to balance** computing capacity and storage between the cloud and devices.
- Integration between the physical world and the virtual (**Digital Twin**)
- Advanced **Work Load Consolidation** technology for distributed computing

- **Unified view** of business entities.
- The **model** describes the meaning of **entities, relationships, and data.**
- **Interoperability** and self-discovery

- **Open Source:**
- Innovation more effectively
- Critical mass of users, developers, and partners
- Community guarantee transparency and avoid vendor lock-in
- **CaaS:**
- Portability
- Scaling
- Operational simplicity

- **Robustness and scalability** to enable the development of solutions in a safe manner. “Think Big, Start Small”.
- **Agility in the application of the latest technologies** in a cohesive way to enable the construction of new business solutions digital: IoT, Big Data, API Management, Advanced Analytics, Blockchain, etc.

Source: [onesaitplatform.atlassian.net](https://onesaitplatform.atlassian.net)

[Learn more about onesait](#)

## Business Overview



Onesait Platform is owned by Indra Company and provides the flexibility so that developers can build their own solutions in a solid and agile way using Open Source technologies, a flexible architecture and an innovative approach.

The value proposition of the platform evolves around the open source components covering the entire life cycle of information (from ingest to visualization through its process and analysis). It offers a unified web console for the development and operation profiles of the solutions. Thanks to different assistants and the encapsulation of best practices, it is possible to develop systems with complex architectures in a simple way, reducing cost and time to market.

Onesite offers a data centric architecture which is the main and permanent asset. There is a data model, a semantic data model, and each application functionality reads and writes through the shared model. Further on, the platform capabilities such as publishing, viewing and messaging are based on a microservices architecture. This ensures concept's isolation between different parts and offers flexibility for the customization of different capabilities.

Moreover, the platform is designed to speed up complex systems development based on unified version, business centric development, software lifecycle management and flexible development. Further, the platform's services can be deployed where preferred, on any public or private cloud or own CPD (Physical or VM). Onesait is built on a large number of widely tested and standard open-source components in its field. This reduces the curve of use of these technologies for new platform users, where it also offers tools that make it easier and more productive to use them. The platform is also completely open-source and is released on GitHub.

Core end-users of the platform are developers who have access to the systems, information, tools and resources of Onesait. More than 100,000 companies in around 90 countries worldwide are connected with the Onesait ecosystem.

Onesait Platform has two editions, one is Community Edition (Open Source) and the other one is Enterprise Edition (commercial support). Onesait Platform Community edition is a free. Onesait Platform Enterprise edition represents the premium paid version of the platform.

[Learn more about onesait](#)

Show less

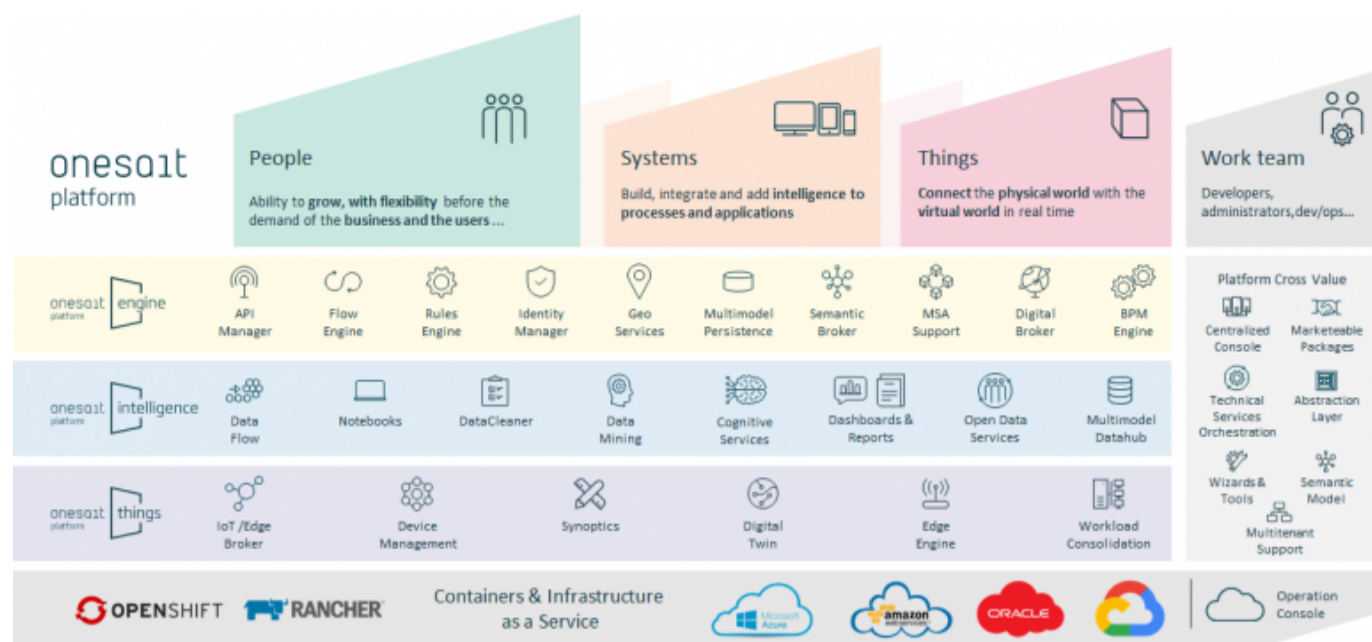
## Technical Overview



The Onesait Platform provides flexibility so that developers can build their own solutions. Based on Open Source components, Onesait Platform covers the entire life cycle of information.

From a general point of view the platform can be seen as consisting of three main layers:

- Interaction layer: layer that allows for communication and collaborative interaction between people and machines, based on a unified representation of information. It interacts with different channels, keeping a common semantic language.
- Action Layer: it includes business rules and execution of delocalised, scalable and flexible processes. It executes parametrized commercial actions and processes to agilely adapt to business requirements.
- Intelligence Layer: It extracts analytical information and stores it according to its type, latency, size and security level.



Source: <https://onesaitplatform.atlassian.net/>

With reference to the purely technical aspect, the platform is in turn divided into five layers: Acquisition Layer, Knowledge Layer, Publication Layer, Management Layer and Support Layer.

The *Physical layer* links to the Acquisition Layer and is composed of five main elements:

- IoT Broker: this Broker allows devices, systems, applications, websites and mobile applications to communicate with the platform through one of the supported protocols. It also offers APIs in different languages.
- Kafka Server: the platform integrates a Kafka cluster that allows for communication with systems using this exchange protocol, usually because they handle a large volume of information and need a low latency.
- Dataflow: this component allows you to configure streaming data streams from a web interface. These flows are made up of one origin (which can be files, databases, TCP services, HTTP, queues, ... or the IoT Broker platform ), one or more transformations (processors in Python, Groovy, JavaScript, ...) and one or more destinations (the same options as the origin).
- Digital Twin Broker: this Broker allows communication between the Digital Twins and the platform, and with each other. It supports REST and Web Sockets as protocols
- Video Broker: allows to connect to cameras via WebRTC protocol, and to process the video stream associating it to an algorithm (people detection, OCR, etc.).

The core of the *Service layer* is the Data-Centric approach, which refers to an architecture where data is the main, permanent asset, and applications come and go. In the Data-centric architecture, the data model precedes any given application's implementation and will be valid long after these are replaced: there is a data model, a semantic data model and each application functionality reads and writes through the shared model. Data-Centric architecture

is supported through the Ontology, and all the functionality of the Platform looks around this concept. The Ontologies are the Entities that the system manages.

The solution's information flows will navigate through the Platform, (ingestion and processing, storage, analytics and publication) from the data producers to the information consumers, following the paradigm of "listen, analyse, act". This allows for the ingestion of information from real-time data sources of virtually any nature type, from devices to management systems. This real-time information from devices and systems accesses the platform through the most appropriate gateways (multiprotocol interfaces) for each system, is then processed, reacting in real time to the configured rules, and finally remains persisted in the storage module's Real Time Database (RealTimeDB).

On the other hand, the rest of the information coming from more generic sources ("Batch Flow") that is obtained by means of processes of extraction, transformation and loading in batch mode (not real time) accesses the solution through the mass loading module of information (ETL).

The *Application layer* links to Publication, Management and Support layers that are described as follows.

Publication Layer:

- API Manager: this module allows you to visually create APIs on the ontologies managed by the platform. It also offers an API Portal for the consumption of the APIs, and a Gateway API to invoke the APIS.
- Dashboard Engine: this engine allows to create, visually and without programming, complete dashboards on the information (ontologies stored in the platform), and then make them available for consumption outside or inside the platform.

Management Layer:

- Control Panel: the platform offers a complete web console allowing for a visual management of the platform's elements using a web-based interface. This entire configuration is stored in a configuration database. It offers a REST API to manage all these concepts and a monitoring console to show each module's status.
- Access Manager: allows to define how to authenticate and authorize users defining Realms with roles, users directory (LDAP, ...), protocols (OAuth2, ...)
- Caas Console: allows to manage from a web console all the modules deployed (as Docker containers orchestrated by Kubernetes), including version updates and rollback, number of containers, scalability rules,...

Support Layer:

- Market Place: allows to define assets generated into the platform (APIs, dashboards, algorithms, models,

- rules, ...) and publish in order other users can use it, free or paid)
- GIS Viewers: from the console you can create GIS layers (from ontologies, WMS services, KML, images) and GIS viewers (currently under Cesium technology) from these layers managed by the platform
  - Open Data Portal: platform includes a CKAN Portal connected with the platform so that the ontologies can be published as datasets or datasets can be exported to ontologies to be processed with the rest of the pieces of the platform
  - Files Manager: this utility allows you to upload and manage files from web console or REST API. These files are then managed with the platform's security.
  - WebApps Server: the platform allows you to serve Web applications (HTML + JS) uploaded through the web console of the platform.
  - Config Management: this utility allows you to manage configurations (in YAML format) of the applications of the platform by environments

The management can be done through the Platform Control Panel, a complete web console that allows a visual management of the elements of the platform through a web-based interface. All this configuration is stored in a configuration database (ConfigDB). Within its functionality it adds:

- Development control panel: integrates all the tools in the platform that the developer will use when creating applications, including creation ontologies, rules, panels, assigning security, etc.
- DevOps & Deploy: this console allows you to configure the tools for the continuous integration of the platform, and also to implement the platform instances and the additional components that a solution may require.
- Security: allows you to configure all security aspects of the solution, such as the user repository (LDAP, the platform itself), and define and manage users and roles, etc.
- Device management: allows managing and operating the devices of the IoT solutions.
- Monitoring: helps monitoring the platform and solutions through KPIs, alerts, etc.

The *Semantic layer* is represented by the Knowledge Layer and these are its main components:

- Semantic Information Broker: once the information is acquired, it reaches this module, that: validates whether the Broker client has permissions to perform that operation (insert, query, ...) or not, and gives semantic content to the received information, validating whether the sent information corresponds with this semantics (ontology) or not.
- Semantic Data Hub: this module acts as a persistence hub. Through the Query Engine, it allows to persist and

consult on the underlying database where the ontology is stored, where this component supports MongoDB, Elasticsearch, relational databases, Graph databases,...

- Streaming Engines: supported by:
  - Flow Engine: this engine allows creating process flows, both visually and easily. It is built on Node-network. A separate instance is created for each user.
  - Digital Twin Orchestrator: the platform allows for the communication between Digital Twins to be orchestrated visually through the same FlowEngine engine. This orchestration creates a bidirectional communication with the Digital Twins.
  - Rules Engine: allows you to define business rules from a web interface that can be applied to data entry or scheduled
  - SQL Streaming Engine: allows to define complex streams as data arrives in a SQL-like language
- Data Grid: this internal component acts as a distributed cache as well as an internal communication queue between modules.
- Notebooks: this module offers a multi-language web interface so that the Data Scientist team can easily create models and algorithms with their favourite languages (Spark, Python, R, SQL, Tensorflow ...).

The *Interoperability layer* is closely related to the *semantic layer*, as previously described the Platform uses a Data-Centric Architecture and the Ontology represents this data model that can be shared between applications and systems: an Ontology is the definition of an Entity in the simplest case or of a Domain Model in the most complex case, that is finally represented as a JSON-Schema. Besides, the platform offers a set of Templates (or Data Models) that allow ontologies to be created following the best recommendations and standards in this regard.

A good example of an attempt to standardize in the Smart Cities field is the [FIWARE Data Model](#) (supported by the platform).

Of the many advantages related to the use of ontologies the following can be mentioned:

- Storage independence: the platform uses Mongo by default to store the ontologies but, depending on the use case, it allows to store the ontologies on other repositories. The platform will manage the creation on the DB (indexes, partitions, ...)
- SQL query engine: whichever be the underlying database of the ontology, the platform allows you to query the ontologies in SQL. This, coupled with the independence of the database, allows, where appropriate, to migrate from the chosen database



technology should the scenario change. It will be simple:

- Security associated with the ontology: working with ontologies has other advantages. For example, the platform automatically manages security. By default, the user who creates the ontology is the only one who can access it, and she can give read, write or complete management permissions to the users that she considers. Besides, if you use the project concept, you can create shared entities for the users that make up the project
- Syntactic validation: as said, the ontology is represented as a JSON-Schema, and ontology instances (in a relational model, that is the records) are JSON's. Whichever the repository where the ontology is stored, its interface is a JSON (you can see it by doing a query). Well, the platform automatically validates that the information sent to the platform complies with the defined JSON-Schema. In a JSON-Schema, you can define simple (mandatory), numerical (> 10) and complex semantic validations.
- Rules and visual flows associated with ontologies: the platform offers several engines that can be executed upon arrival of an ontology instance. It is very common to have a control ontology that triggers an action in another system (invocation to a REST Service) or simply sends an e-mail. The platform allows you to define this without any programming:
- Transparent protocol management: the platform allows you to manage the ontologies in different protocols, including REST, MQTT, Kafka, WebSockets, ... and the client can choose the protocol according to the use case.
- Multilanguage APIS: in addition to the independence of the protocols, the platform offers APIs in several languages.

[Learn more about onesait](#)

Show less

## Contextual Overview

The Onesait platform is developed based on the outcomes of previous projects:

- January 2009 – March 2012: Sofia Artemis Project (<https://artemis-ia.eu/project/4-sofia.html>)

- March 2012 – June 2019: Sofia2 by Indra (discontinued, replaced by Onesait Platform)
- June 2019 – currently: Onesait Platform (<https://onesaitplatform.atlassian.net/>)

In the Platform, data is the main and permanent asset.

The Data-Centric approach turns data into its core. There is a data model, a semantic data model, and each application functionality reads and writes through the shared model.

The platform has integration with different Cloud services such as Azure, GCP, Amazon and allows these services to be used transparently (eg Kubernetes service in Azure or AWS, Google BigQuery, Azure Intelligence Services, ...).

Regarding external communications, everything is done through TLS and SSL (both incoming and outgoing). All communication between platform's microservices can also be configured to be done via HTTPS, so that the containers specify the Sidecar pattern that is responsible for establishing security.

The platform supports various authentication and authorization mechanisms, using OAuth2 by default. The platform also offers credential lifecycle management.

The platform is built on a large number of widely tested and standard open-source components in its field. The commitment to Open Source is such that the platform is also completely open-source and is released on Github.

The Data Governance Model of Onesait rests on the following principles:

- Completeness of the information: Ensuring that the standardized data as a whole has logic for its exploitation.
- Global business / infrastructure vision: Establishment of the business definition for the data, allowing them to be identified through an end-to-end vision, as well as with an infrastructure vision that allows knowing the physical location of the data.
- Responsibility for data: Assignment of responsibilities and roles regarding information management, reliability, integrity, provisioning and exploitation of data, through a governance model that facilitates the monitoring of its evolution.
- Efficient data: Compliance with the principles of non-duplication, integrity and consistency of data.

#### **KEY ELEMENTS IN QUALITY ASSURANCE**

- Definition of standards: Generation of minimum information requirements to consider the data as correct (verification of lengths, type of data, formats).
- Data validation: Establishment of validation mechanisms that allow the integration of the data in the storage infrastructure according to established standards, minimizing the occurrence of incidents.
- Cross view of the data life cycle: Identify the complete process of the data life cycle, allowing its global

management through the traceability and mapping of information.

- 

[Learn more about onesait](#)

Show less

## Success Stories



### **The Bogotá Digital Health success story**

The Bogotá District Health Secretary simplified citizen access to healthcare services through the interoperability platform, which provides services to 128 healthcare service units, Capital Healthcare EPS, medical appointment call center and medication operators, who will have unified and real-time information to improve user services.

#### **The challenge**

The Bogotá District Health Secretary needed to integrate information on real-time care processes, streamline clinical records, manage appointments for patients and professionals, and promote medical form management. This would enable a unification of information of all the patients cared for in the public hospital network of Bogotá.

#### **The solution**

Integration was made in the Bogotá digital healthcare platform, the 4 healthcare subnetworks in Bogotá (North, Central-east, South and Southwest), all medical centers, Capital Healthcare systems as insurer, Audifarma systems as a medication provider and the information systems of the Bogota Secretary of Health as a regulatory body. The information of the platform includes the clinical and information content of: Emergencies, hospitalization, outpatient consultations, diagnostic aids, oral health, perinatal maternal medical record, vaccination card (PAI), Public Health and Notification event alert (Sivigila district).

#### **The impact**

“Bogotá Digital Health” with the capacity to upload the medical records of over 8 million people, schedule more than 7.5 million appointments a year and store over 2.6

million medical forms.

Source: Onsait website:

<https://www.onsait.com/testimonials>

Show less



### **The Bidafarma success story**

Bidafarma has made it possible for patients to attend dermatology consultations at community pharmacies in order to detect early signs of suspected skin cancer, both melanoma and non-melanoma, thereby reducing diagnosis and travel times.

#### **The challenge**

The challenge was to take advantage of the network of Bidafarma's partner pharmacies, generating new innovative services and business models to make dermatologist consultations available to patients, thus simplifying skin cancer screening in partnership with the Hospital Viamed Santa Ángela de La Cruz and securing a medical report within 48 hours.

#### **The solution**

Bidafarma has provided 9,200 Spanish pharmacies with remote consultation services for the early detection of skin cancer. The service is provided by filling out a questionnaire and obtaining two photographs with a dermatoscope, which are sent securely and anonymously to medical professionals, so that they can issue an assessment that will be given to the patient by his or her pharmacist.

#### **The impact**

In the two years of service provision, 84% of cases have been resolved digitally, of which 56% were benign, 27% lesions proved to be skin cancer or pre-cancer, and 17% were considered doubtful, in which regard patients were referred for a face-to-face consultation with a specialist to make a more accurate diagnosis.

Source: Onsait website:

<https://www.onsait.com/testimonials>

Show less

