

*FRONT PAGE***PROJECT FINAL REPORT**

Grant Agreement number: *317862*

Project acronym: *COMPOSE*

Project title: *Collaborative Open Market to Place Objects at your Service*

Funding Scheme: *Integrating Project*

Period covered: from *Nov. 2012* to *Oct. 2015*

Name of the scientific representative of the project's co-ordinator, Title and Organisation:

Mr. Benny Mandler

IBM Haifa Research Lab

Tel: +972-4-8296378

Fax: +972-4-8281112

E-mail: mandler@il.ibm.com

Project website address: <http://www.compose-project.eu/>

4.1 Final publishable summary report

4.1.1. Executive summary

COMPOSE aims to bring about a new era in services based on the Internet of Things (IoT), making it simple to integrate the physical with the virtual world. COMPOSE achieves this through the provisioning of an open and scalable marketplace infrastructure, in which Smart Objects are associated to applications that can be combined, managed, and integrated to easily and quickly build innovative applications.

The project has developed novel approaches for virtualising Smart Objects into readily usable Service Objects that can be advertised, searched for, and combined with other Service Objects to form novel applications. To this end the project devised solutions for semantically enhancing smart objects and applications, secure and privacy-preserving data aggregation and distribution, dynamic service composition, advertising, discovering, provisioning, and monitoring.

COMPOSE aims to facilitate a new business ecosystem targeted mainly towards SMEs, by democratizing the use of smart objects via an open cloud based platform. At the same time, major ICT players, particularly cloud service providers and telecommunications companies, are able to reposition themselves within a new IoT-enabled value chain, as infrastructure providers.

COMPOSE started out by collecting requirements from all relevant stakeholders which in turn were fed into the architecture that was realized within the project. The main overarching goal of the project, stating the general direction which requirements should follow is to ease the task of a developer creating applications based on internet connected smart objects. The requirements called for the following capabilities to be available through the COMPOSE platform in a consumable manner: (i) Registration of smart objects and data absorption from those objects; (ii) provide access to both historical and real-time data; (iii) provide data processing, manipulation, analytics and notifications services based on incoming data; (iv) actuate smart objects by creating a bi-directional communication channel between the platform and the objects (v) discover and re-use existing building blocks, be them service objects or applications; (vi) support the combination of existing building blocks both at design time vi an assisted composition engine as well as in run time via an orchestration engine; (vii) develop and deploy new applications in a secure cloud environment.

This set of requirements naturally led to the following architecture, depicted in *Figure 1*.

Distinct components and capabilities were developed and are available as separate open source projects. These components are integrated into a whole platform which serves as more than merely the sum of all the individual capabilities. A data management component was developed which forms bi-directional communication mechanisms with smart objects and takes care of the storage and processing of data once it has been ingested into the platform. A

service discovery component was developed, which holds semantic metadata on all COMPOSE entities, and supports a composition and recommendation engines. A developers’ portal was created which serves as the contact point between COMPOSE and the external world. An IoT PaaS was devised to host all capabilities and applications, along with ample security measures. In addition three distinct pilots were used throughout

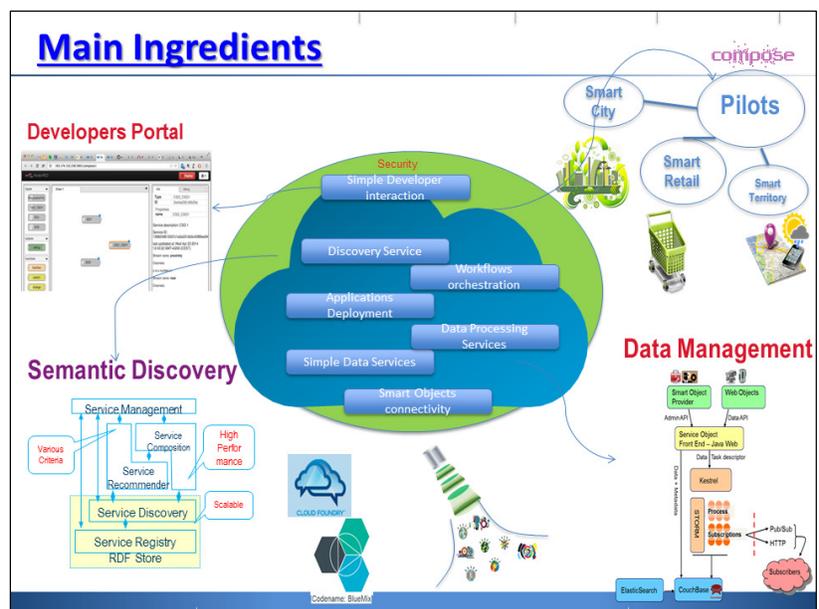


Figure 1: Main Components

the project from the requirements gathering phase, through the architecture definition, all the way to the platform validation.

Accompanying the technical work COMPOSE contributed in several avenues: (i) Standardisation – ample work has been carried out to initiate IoT related standardisation work mainly through W3C (ii) business modelling and exploitation, to take care of the business aspects of the COMPOSE platform and potentially the IoT in general (iii) dissemination work has been performed throughout the project lifetime to spread the word to the world about the existence and the capabilities provided by the COMPOSE platform (iv) attracting a developer's community via numerous events, following a training roadmap.

The integrated COMPOSE platform has been released to be used by external developers.

4.1.2. Summary description of project context and objectives

COMPOSE was conceived not long after the IoT has become a reality. Most technological pieces required were there, the potential was there, the realization that this is potentially a game changer was there, but still not a lot of impact was felt in the real world, and the assertion was that there's a huge potential which is not being materialized. Naturally, the question arose as to why the potential was not fulfilled, and the realization was that it's plainly too difficult. There's too much multi-dimensional heterogeneity, in protocols, data formats, metadata formats, etc. Each prospective developer of an IoT application had to master various disparate technological areas, and moreover he had to start always from scratch as there was no real starting point he could make use of. A lot of the issues which an IoT developer faces are complicated, namely big data, scalable and elastic deployment, etc. So, the prevalent applications were very narrow and focused on a specific vertical market more often than not using a small pre-determined set of smart objects.

Looking at such a reality the COMPOSE project proposed to change it, levelling the playing field so that small players, SMEs, and entrepreneurs can bring their innovation to the market with a reasonable effort, in a reasonable amount of time, with a reasonable investment, The idea was to alleviate the individual developers from a lot of the systems aspects of an IoT application, thus enabling the developer simply to master the domain of their expertise rather than have to deal with the entire stack. COMPOSE proposed to create a platform in which data from smart objects would be ingested and made available to applications; where existing building blocks can be found and used in new applications; security is well established throughout the stack; and all within a scalable cloud based deployment.

Project Context

The increasing diffusion of electronic devices with identification, computing, and communication capabilities, such as mobile phones, digital cameras, and music players, is laying ground for the emergence of a highly distributed service and networking environment. This environment, referred to as the Internet of Things (IoT), is strongly entangled with the physical realm and with the local context. Internet-connected smart objects, integrating RFIDs, sensors, and/or actuators, and deployed in the environment or in human artefacts (e.g., cars), will become the connectors between the “virtual realm” of Information and Communication Technology (ICT) and the “physical realm”.

From a service and application perspective, the ability to seamlessly integrate the virtual and physical realms is expected to foster a major change in ICT systems. This will come from a profound and ubiquitous integration between two elements. On the one hand data will be coming from the environment and on the other hand, virtual ICT entities will be acting on the environment. Such developments enable the development of a novel generation of smart systems and processes enabling: physical goods to be developed, manufactured, bought and sold more effectively; novel services to be delivered. We are now facing a tipping point, whereby, for the first time in history, almost anything can become digitally aware and interconnected. Smart grids, smart cities, smart financial systems, smart transportation—with so much technology available at reasonable cost, the list of possibilities is endless.

The COMPOSE project builds on this potential, as well as on two other trends taking place in the ICT field.

First, mobile personal devices are becoming more and more powerful in terms of processing and storage capabilities. Further, they are being equipped with a variety of sensors, such as accelerometers; sensors to

measure temperature, humidity, and light; and video cameras. Such mobile personal devices will be seamlessly interconnected by means of a variety of wireless communication technologies, giving rise to complex and dynamic networks and also interacting with servers and applications residing in the cloud.

The second trend relates to the architecture of participation and the Internet of Services (IoS) approaches to service delivery. Architecture of participation, initially embodied by the so-called Web 2.0, encourages users to take an active role in the provision of content and services, as prosumers. Community-based solutions are emerging, whereby users share data coming from their smart devices, mostly sensors. The concept of Everything-As-A-Service enabled the introduction of new approaches for the reliable sharing of ICT infrastructural resources (e.g., computing, storage, and communication capabilities) and application functionalities. The Everything-As-A-Service approaches have so far, found only limited application in the IoT field.

When big data coming from smart objects meets the cloud and everything as a service approach, within a participatory environment, the full potential of IoT will start bearing fruits.

The COMPOSE project proposal builds upon an innovative combination of the opportunities offered by such trends to design, implement, and deploy an open, scalable platform, in which objects can be seamlessly turned into standardized services, combined, managed, and integrated into innovative applications.

The concept behind the COMPOSE project is illustrated in Figure 2.

IoT devices will be virtualised and published as Service Objects in the COMPOSE platform, thus becoming part of the Internet of Services. Service objects can be aggregated Applications, which can then be accessed by users (by downloading and/or installing on their devices). Developers will be provided with an open interface and developers' service development kits for building and deploying innovative applications, leveraging the management functionalities provided by the platform.

COMPOSE will combine state-of-the-art solutions from a number of fields (data processing, scalable communications, semantic web, event processing, security, and service-oriented technologies) to give birth to an innovative integrated platform, enabling people and organisations to leverage IoT technologies through dynamic value-added services. The COMPOSE project will provide the necessary framework for tackling the essential challenges of an IoT platform.

The COMPOSE project aims to achieve this by (i) involving large communities of developers in the creation of applications, leveraging the availability of Internet-connected objects and stimulate IoT innovation similarly to what happened with the mobile applications development world, (ii) providing a comprehensive framework for harnessing the complexity inherently present in the management, sharing and delivery of IoT service objects (iii) enabling the channel of innovative consumable contents based on the data and knowledge originating from service objects, and (iv) allowing for flexible security mechanisms that are able to adapt to different security contexts and requirements induced by the data processed. This will include addressing issues related to identity and to security and privacy management of data coming from service objects, their adaptive and dynamic discovery, and the access to them through a homogeneous semantically-enriched, service-oriented interface.

The resulting platform is unique by virtue of basing its activities on data originating from a plethora of sensors, which are transformed into service objects, which can in turn be used to build applications. The

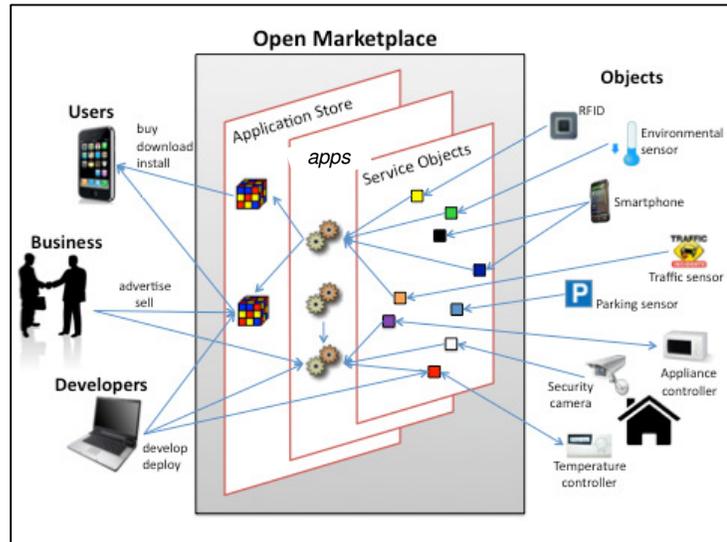


Figure 2: Conceptual View

Marketplace is envisioned as a location in which applications may be made available for developers to build upon as well as applications made ready for end users to consume. This is expected to foster a major change in the IoT landscape, paving the way for innovators, such as inventors, research groups, SMEs, and large enterprises to easily introduce new IoT-enabled applications, in much the same way they currently do for mobile applications.

Technical Approach

The development of such a platform raises a number of technical challenges, requiring network and service infrastructures able to master heterogeneity, complexity, and dependability. They must also ensure behavioural stability in the absence of a centralised control over IoT objects. Further, to be effective, the marketplace shall include a number of technical enablers supporting the secure and rapid development and deployment of innovative IoT services. This includes (i) mechanisms for the creation and maintenance of an ecosystem of service objects; (ii) ways to easily enable shared access to service objects information and resources, while preserving their normal operation and the privacy of the provided data; and (iii) enrich service objects and applications with semantic information that will help turn the base raw data into useful information that can be easily found by additional parties.

The vast majority of current systems using IoT technology are characterised by being vertically-integrated, with the whole system (from the device level to the application level) customarily being within one single administrative domain. The vision of the COMPOSE project is to advance the state of the art in this domain by creating a horizontal platform in which different domains can share the same infrastructure, in which data from Internet-connected objects can be easily published, shared, and integrated into applications. The marketplace will provide all the necessary technological enablers, organized into a coherent and robust, yet lightweight, framework, covering both delivery and management aspects of objects, applications, and their integration.

The COMPOSE project tackles the aforementioned shortcomings by providing a set of technological enablers, organized into a coherent framework covering both delivery and management aspects. These enablers will be identified and designed with the primary goal of simplifying the interactions between applications, objects and the developer.

The following main functional areas are covered by the marketplace technological enablers:

- **Object virtualisation:** The enablers will provide means for publishing objects in the marketplace and creating virtualised counterparts—service objects. This will simplify the interactions with the objects through the abstraction of the offered capabilities. Object virtualisation will additionally allow flexible security mechanisms suitable for the object, the interacting applications, and the respective security context. Virtualisation will represent the way to elicit objects to service objects. In addition, semantic enrichments may be applied to objects at this level.
- **Interaction virtualisation:** These enablers will provide a set of primitives for supporting various types of interactions, including those between objects and applications and those among applications. They will abstract the heterogeneity of devices and network protocols, whilst offering several interaction paradigms, including request-response, publish/subscribe, and event-driven approaches. Finally, they will provide feasible security primitives that will enable secure communication between service objects and applications, as well as between applications.
- **Knowledge aggregation and dispatching:** These enablers provide mechanisms to flexibly aggregate data provided by several service objects to create information or knowledge at the suitable level of abstraction and according to some semantic descriptions. Context awareness can further be added in order to enable higher level constructs to make better use of the raw knowledge flowing in. Such knowledge will then be provided to applications in push or pull mode at the required time and location. The marketplace will offer flexible mechanisms to access/search/mash-up/subscribe the aggregated knowledge. These enablers will include mechanisms to ensure a secure, controlled, and privacy-preserving access to and distribution of data/information from service objects.
- **Discovery and advertisement of Service objects and Applications:** These enablers are in charge of searching for functionalities, applications, and information offered by service objects and applications (or

their aggregation) and inform about their availability. Services location may be further facilitated by adding semantic tagging to the applications.

- **Ad hoc creation, composition, and maintenance of service objects and applications:** These enablers provide support to dynamically assemble and orchestrate the functionalities, applications, and information offered by available service objects and applications. In particular, the latter must ensure the security requirements of the information processed and may use context to implement the appropriate enforcement.
- **Data Management:** COMPOSE will structure object data in two different repositories. The object registry will store metadata associated with the objects, such as characteristics, location, and composition properties. The object data store will be used to store historic and aggregated data generated from the objects, depending on the needs of each object.

Project Objectives

The main expected outcomes of the project are the design, implementation, deployment, and validation of an open platform for service objects, IoT applications, and workflows. This will include the development and release of a set of open APIs for accessing the platform services, as well as of a software development toolkit (SDK). The platform will be based on a Web2.0 approach, whereby any user will be allowed to deploy his own service object and make it available to external applications/users for use. The management of service objects and IoT applications is based on secure and privacy-preserved aspects.

The availability of the platform and of the SDK is expected to lead to a significant reduction in the costs associated with the implementation of IoT applications and to provide a strong push for IoT technologies to enter the mass market.

COMPOSE presents the following main expected outcomes:

- An implementation of the proposed open platform: This will consist of a server component residing on a cloud infrastructure and of a middleware platform enabling the integration of service objects into applications running in the platform. Portions of the developed middleware will be released under an open-source license. A set of open APIs for accessing the platform functionalities will be made freely-available.
- A set of APIs that can be used by developers to implement COMPOSE applications intended for deployment over the open platform. The corresponding libraries will be available in the form of an SDK capable of abstracting most of the complexity inherent in the implementation of applications based on smart objects.
- A quantitative assessment of the performance of the proposed platform, of the design methodology used, and of the SDK developed: This will work by means of a set of experiments held in close collaboration with a set of relevant stakeholders in a number of real-world use cases.
- A security analysis framework guided by information security policies: This will support developers in the design of secure applications by suggesting appropriate security services that match the information security requirements of the data processed by the service objects. Additionally, it will support dynamic reconfigurations of compositions to account for new security requirements emerging during execution.
- An assessment of the time-to-market and cost reductions: The COMPOSE approach will enable this assessment, related to the implementation of Internet of Things and Services (IoTS) systems, with respect to state-of-the-art solutions, together with a study of the potential impact on the development of the IoT applications/services market.

The main goal of the COMPOSE project is to fully unleash the potential of the IoT through the provisioning of an open platform, in which objects are exposed and treated as services. The platform will be a community engagement instrument to build applications related to Internet-connected objects that can be easily developed, deployed, published, and shared, as well as integrated into added-value applications. The platform will include the definition of a service design methodology and a set of tools (including a freely-available open-source SDK and a set of open APIs for the community), enabling the rapid and low-cost creation and deployment of applications leveraging the presence of Internet-connected objects. In this way, a new service ecosystem will arise, enabling small-to-medium enterprises and inventors to introduce new IoT-enabled applications to the market in a short time and at limited cost. This will allow major players in the ICT industry, particularly cloud service providers and telecommunication companies, to develop new, favourable

business models, able to leverage their assets and value proposition to enter new markets and/or to reposition themselves within new value chains.

The project aimed at consistently achieving the following objectives, and did manage to fulfil them all:

- To architect an open platform for the creation and management of applications deployed on/interacting with pervasive pools of Internet-connected objects, leveraging the availability of cloud-based services and infrastructures.
- To design and engineer a set of methods and mechanisms enabling the open platform to natively support:
 - Objects and interaction virtualization, including their associated data streams
 - Semantic representation of services based on smart objects
 - Object capabilities' advertisement and discovery
 - Knowledge derivation, aggregation, and secure distribution
 - Objects augmentation into services
 - Service composition
 - Service delivery
 - Service security through analytical and monitoring methods
- To implement and validate the proposed architecture and methods by means of a prototypical deployment, focused on the representative use cases selected for the project, whereby a number of field trials will be performed involving real actors.
- To develop and release an open-source software development toolkit that will be used by developers of the community to easily interact with the marketplace for the creation, deployment, and management of innovative applications based on service objects
- To study and identify new business models and value chains enabled by the creation of an open marketplace for the IoT service ecosystem, characterized by low-market entry barriers and enabled by the provisioning of standardized APIs and freely available application-development toolkits
- To promote the standardization and adoption by the industry at large of the proposed architecture and interfaces

4.1.3.A description of the main S&T results/foregrounds

The main scientific and technological results have been the articulation and development of a complete software stack for large scale deployment of IoT applications and its successful application to three use cases in the areas of smart retail (enhanced shopping experience and enhanced store management), smart city (ride sharing for high occupancy vehicles) and smart territory (sports and tourism). The complete integrated platform consists of several independent components which were architected to work together in harmony and produce a whole that is bigger than the sum of its individual parts.

The software stack consists of five components. (i) A data management component which is in charge of bi-directional communication between the platform and the physical world, and incoming data storage, processing, and providing data access in a consumable manner. (ii) Service discovery, a design phase component, which holds metadata of all COMPOSE entities and supports semantic queries. This component provides also recommendation services as well as assisted composition support. (iii) Developers' portal which serves as a one stop shop for external users when interacting with the COMPOSE platform. It presents COMPOSE capabilities in a consumable manner to developers such that the task of creating and deploying new service objects and application is made easy. (iv) All-encompassing security services at all platform levels. This component provides access control, as well as static and dynamic analysis of applications, and data flow and provenance support. (v) Finally the ultimate deployment vehicle for such a platform is the IoT PaaS, which is a customization of an open cloud based PaaS, to fit best the IoT needs. This is a scalable and elastic cloud based environment hosting and connecting all required applications and services.

Below we describe the main results for each of the components mentioned above, followed by a description of the main results for the three pilots which validated the COMPOSE platform.

An important asset created is solid reference architecture for creating an IoT platform.

4.1.3.1. ServioTicy

ServioTicy is a state-of-the-art platform for hosting Internet of Things (IoT) workloads in the Cloud. It provides multi-tenant data stream processing capabilities, a REST API, data analytics, advanced queries and multi-protocol support in a combination of advanced data-centric services. ServioTicy aims to provide a technological platform for easily creating services based on the Internet of Things (IoT), thus unleashing the full potential of an Internet of Services (IoS) based on the IoT. The main focus of ServioTicy is to provide a rich set of features to store and process data through its REST API, allowing objects, services and humans to access the information produced by the devices connected to the platform. ServioTicy allows for a real time processing of device-generated data, and enables for simple creation of data transformation pipelines using user-generated logic. Unlike traditional service composition approaches, usually focused on addressing the problems of functional composition of existing services, one of the goals of the ServioTicy is to focus on data processing scalability.

The architecture of ServioTicy is composed of different elements. The Front-End of platform is Web Tier that implements the REST API that sits at the core of ServioTicy. The API contains parts of the logic of the Service Objects and Data Processing Pipelines, related to authentication, data storage and data retrieval actions. The Stream Processing Topology is responsible for the execution of the code associated to Data Processing pipes as well as the forwarding of data across Service Objects and to external entities (e.g. external subscribers that want data forwarded on real-time using a push model on top of MQTT or STOMP). Finally, the data Back-End includes the Data Store that provides scalable, distributed and fault-tolerant properties to ServioTicy, and the Indexing Engine that provides search capabilities across sensors data using different criteria, like timestamps, string patterns or geo-location.

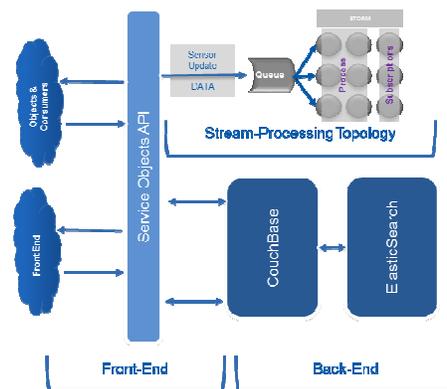


Figure 3: COMPOSE platform architecture – the data processing plane

Abstractions and Nomenclature

Several abstractions are used in ServioTicy to embrace the different entities involved in the existence of IoT ecosystems.

- **Web Object:** Web Objects are physical objects sitting on the edge of the ServioTicy and capable of keeping for example HTTP-based bi-directional communications, such that the object will be able to both send data to the platform and receive activation requests and notifications. Not all such objects will support the same set of operations, but a minimum subset will have to be guaranteed to make them usable ServioTicy.
- **Service Object:** Service Objects are standard internal ServioTicy representations of Web Objects. This entity serves mainly for data management purposes and has a well-defined and closed API that provides search capabilities across sensors data using different criteria, like timestamps, string patterns or geo-location. ServioTicy, in an effort to embrace as many IoT transports as possible, allows Web Objects to interact with their representatives in the Platform (the Service Objects) using a set of well-known protocols: HTTP, STOMP over TCP, STOMP over WebSockets, and MQTT over TCP.

- **Data Processing Pipeline:** A Data Processing Pipeline is a data service and aggregation mechanism, which relies on the data processing and management back-end component to provide complex computations resulting from subscriptions to different Service Objects as data sources. This construct can support pseudo-real time data stream transformations, combined with queries concerning historical data. Data analytics code defined by the user may be provided as well. Just like a Service Object, this entity serves mainly for data management purposes and has a well-defined and closed API.
- **Subscription:** Data subscriptions are a mechanism in ServioTicy that allow Service Objects, Data Processing Pipelines and external data consumers to get data updates automatically and asynchronously forwarded for further processing.
- **Sensor Update:** Sensor Updates are the unit of data sent by a Web Object to its Service Object. It contains the different synchronously sensed values and a timestamp that is maintained all over the pipelines. A subscription or a query to a Service Object will get the data in this format.

API

Service Objects are exposed through a RESTful API that uses HTTP as a transport and that acts as the SO front-end. This basically implies that SOs can be identified unambiguously using unique URIs. The API provides resource actuations through the four main HTTP operations: GET (retrieve), POST (create), PUT (update) and DELETE. The following table contains a summary of the operations implemented in the REST API. SOs are created by POSTing a JSON document to the API. The document is a basic description of the main properties of the Service Object about to be created.

Table 1: REST API methods (not including actuations)

operation	Target URI	Role
Create	POST /	Create a new SO posting a JSON document.
Retrieve	GET /	Retrieve the list of all the SOs created.
Retrieve	GET /(sold)	Retrieve attributes from the <SO_Id> Service Object.
Update	PUT /(sold)	Modify the <SO_Id> ServiceObject.
Delete	DEL /(sold)	Delete the <SO_Id> Service Object.
Retrieve	GET /(sold)/streams	Retrieve the list of all the SO streams.
Create	POST /(sold)/streams/(streamId)/subscriptions	Subscribe the Service Object <soId> to a service .
Update	PUT /(sold)/streams/(streamId)	Store <soId> data putting a JSON document.
Retrieve	GET /(sold)/streams/(streamId)	Retrieve the list of all the data of <soId> Service Object.
lastUpdate	GET /(sold)/streams/(streamId)/lastUpdate	Retrieve the last piece of data generated by a <soId> Service Object.

The API Web Services are developed using a Java programming language API, **JAX-RS** (Java API for RESTful Web Services). JAX-RS provides some annotations to aid in mapping a resource class as a web resource. In addition, it provides further annotations to method parameters to pull information out of the request.

Jersey is the REST framework providing the JAX-RS Reference Implementation and more. Jersey provides its own APIs that extend the JAX-RS toolkit with additional features and utilities to further simplify RESTful service and client development.

As a JSON processor Java library we use the **Jackson** Project (<http://jackson.codehaus.org/>). Jackson is a High-performance suite of data-processing tools for Java, including the flagship JSON parsing and generation library, as well as additional modules.

The Jackson Project also has handlers to add data format support for JAX-RS implementations like Jersey.

As a HTTP Web Server a Java Servlet container we use **Jetty**. Jetty is a pure Java-based HTTP server and Java Servlet container. Jetty is often used for machine-to-machine communications, usually within larger software frameworks. Jetty is developed as a free and open source project.

The API data backed is the **CouchBase** Server. CouchBase Server is an open source, distributed NoSQL document-oriented database that is optimized for interactive applications. Unlike traditional relational databases, CouchBase stores information in documents rather than table rows.

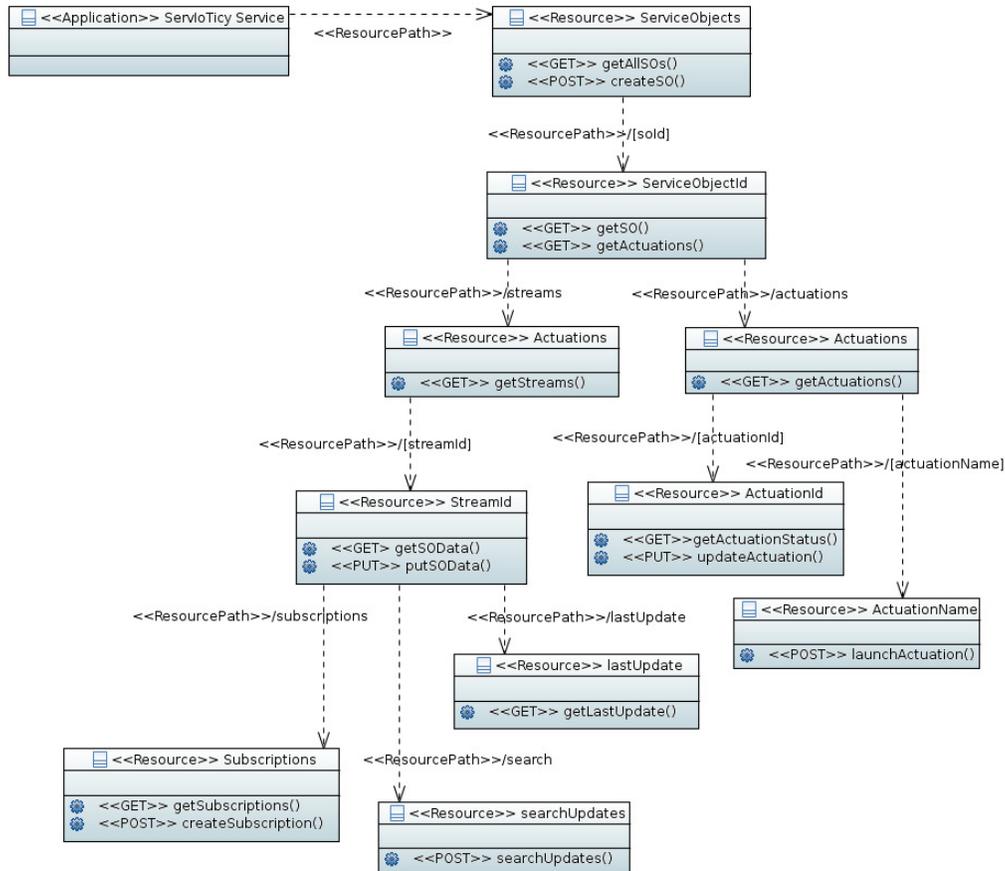


Figure 4: RESTful servIoTicy web service model

CouchBase Server has native support for JSON documents. A document is a JSON object consisting of a number of key-value pairs that you define. There is no schema in CouchBase; every JSON document can have its own individual set of keys. Each JSON document can have a different structure, and multiple documents with different structures can be stored in the same CouchBase bucket. Document structure can be changed at any time, without changing other documents in the database.

The API Web Services uses **ElasticSearch** (<http://www.elasticsearch.org/>) as search engine. ElasticSearch is a distributed search engine built for the cloud. It provides a distributed, multitenant-capable full-text search engine with a RESTful web interface and schema-free JSON documents. ElasticSearch is developed in Java and is released as open source. ElasticSearch can be used to search all kinds of documents. It provides scalable search, has near real-time search, and supports multi-tenancy.

ElasticSearch is connected with CouchBase to provide search capabilities. These searches are implemented in different parts of the API, for example, the Map Demo uses spatial queries.

Data Processing Pipelines runtime

The main components used in the architecture (shown in **Error! Reference source not found.**) for the SO composition system are Kestrel, Storm and Rhino. The next figure shows the connection between these three components.

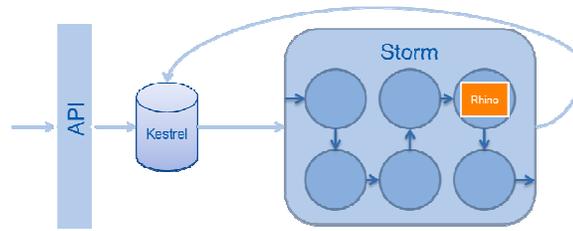


Figure 5: Stream composition high level architecture

- **Kestrel** is a scalable distributed message queue system. Each Kestrel node handles a set of reliable, ordered message queues. When you put a cluster of these servers together, with no cross communication, and pick a server at random whenever a message is sent or received, you end up with a reliable, loosely ordered message queue. It is used by the API as the entry point of Sensor Updates to Storm in order to perform the composition.
- **Storm** is a distributed real-time computation system. Similar to the manner in which Hadoop provides a set of general primitives for doing batch processing, Storm provides a set of general primitives for performing real-time computation. To define a workflow in Storm, a topology of “Bolts” must be defined. A “Bolt” is a computational node that can run in parallel with all the other Bolts. The nodes are sharing messages between them, working as a network topology. This topology is static, and once it is running, the only way to modify it is by stopping it and deploying a new modified one. In our case, the topology used divides the processing of one SO Stream in several generic stages. One of those stages is the actual operation to generate or filter a new one, and it is done by executing dynamically JavaScript code.
- **Rhino** is an open-source implementation of a JavaScript engine written entirely in Java. It is typically embedded into Java applications to provide scripting to end users. It is embedded in J2SE 6 as the default Java scripting engine. It is used to execute the JavaScript code for each composite stream in a SO, every time it receives a related Sensor Update. The execution is performed inside one of the Storm topology stages.

Storm computation workflows are static, and so cannot be modified or extended. This is a major problem for the case of composing Service Object streams in COMPOSE, because the goal is to generate pipelines of previous service objects that are already in use, when it is needed.

When defining a service object composite stream, several stages shared by all composite stream computations were identified, which were easily turned into a static Storm topology. One of these stages is the actual computation, which can be arbitrary operations, in this case JavaScript code.

The following figure shows the Storm topology with each Bolt (computational node) and its first Spout (a node in charge of getting updates from the outside).

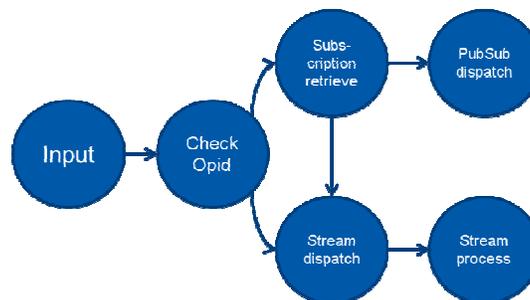


Figure 6: Static Storm topology

- *Input*: The only Spout of the topology. It gets SUs in JSON format from the Kestrel queue.
- *Check opid*: It verifies that the SU is already stored in the database and so it begins the computation.
- *Subscription retrieve*: It retrieves all the subscribers to the received SU. The subscribers can be either SOs (not streams) or external subscriptions through PubSub.

- *Stream dispatch*: Given the input SU and a SO, it gets the streams from the SO that need to make use of the SU. If the input comes directly from the *Check Opid Bolt*, then the SO is the same the SU comes from, and it will look for inner stream references. If it comes from the *Subscription retrieve Bolt*, then the SO is subscribed to this SU and it will look for streams using it.
- *Stream process*: This Bolt has the input SU, and the stream that has to process it. It will query the ServioTicy API for all the other needed updates to generate/filter the new SU. Also it will check that the timestamp of the new update (the biggest from the used inputs) is bigger than the last one stored.
- *PubSub dispatch*: It is a publish-subscribe dispatcher Bolt that receives the SU and the topic to where it will send it to.

Rhino is used inside the *Stream process Bolt*. The description of a composite stream contains the output type of its current value. This way Rhino executes the evaluation and performs the proper type cast to the outcome. It is also used for the filters which are always cast to Boolean.

4.1.3.2. iServe

A fundamental step towards seamlessly supporting users in creating advanced IoT applications is the provisioning of powerful yet scalable supporting infrastructure for the location of potentially relevant services/things, and for helping to combine and use the found services and things into novel applications. The service management work package (WP3.1) addresses those demanding needs through three main components:

- An advanced linked services discovery engine, whose job is to discover distributed and heterogeneous COMPOSE services. The service discovery engine is layered on top of a service registry, which exploits information retrieval and semantic search and storage technologies.
- An advanced service recommender system, which is in charge of suggesting new relevant services based on users' previous interactions, similarity between services, and other non-functional properties such as performance, trust, etc.
- An assisted service composition engine, which is meant to help users create new applications and workflows by (semi) automatically combining existing services to obtain the desired data.

At the core of this infrastructure lies iServe. iServe, is an open source platform that unifies the publication, discovery, and use of Web Services (e.g., WSDL services), Web APIs (e.g., Twitter API, Flickr API, etc.), and Things available on the Web. iServe is, to the best of our knowledge, the first platform to provide this level of support homogeneously across types of devices and software interfaces, providing a convenient one-stop-shop for the location and use of distributed software building blocks as necessary for creating novel Web and IoT applications.

iServe exploits and expands state of the art research and development from a number of fields. From a data acquisition perspective, the platform includes several import plugins providing the platform with the ability to ingest, process, and index interface descriptions in several formalisms including WSDL, SAWSDL, Swagger, OWL-S and WSMO-Lite to name a few. Additionally, Machine Learning techniques are used to support the automated discovery of Web APIs over the Web by automatically identifying Web pages that provide technical documentation of these APIs. Subsequently specific Web Mining techniques have been devised to automatically extract important features from these descriptions such as the functionality provided, the endpoints. The aforementioned processes are fed by a Web crawl, namely CommonCrawl, which is processed using a combination of Hadoop and Mahout jobs to support their efficient processing.

Once imported, all descriptions are homogenised into a common representation expressed in terms of an ontology called the Minimal Service Model (MSM). This ontology essentially provides a common ground for describing i) the overall structure of the interface exposed by a Web API or a *Thing* including aspects like the operations or resources exposed, ii) human-oriented information such as a text-based descriptions about the component at hand, and iii) machine-oriented descriptions of the components including semantic annotations about the type of component, the functionality offered, or the semantics of the data manipulated.

Every Web Service, Web API, and Thing are homogeneously described as *services* expressed in terms of a simple RDF(S) vocabulary called Minimal Service Model (MSM). iServe exposes services publicly following

the Linked Data principles, which essentially dictate that every piece of data should be given an HTTP URI which, when looked up, should offer useful information using standards like RDF and SPARQL. Additionally, data should be linked to other relevant resources therefore allowing humans and computers to discover additional information. In a nutshell, adopting Linked Data principles enables both humans and machines to seamlessly retrieve and process the descriptions of the software components indexed by iServe in a convenient manner. After import, iServe tries to enrich the data with external information. Currently, for instance, iServe attempts to identify the provider of the service, and tries to interlink it with two large Linked Data datasets, i.e., DBPedia and Yago, so that we can leverage external information about the provider to assess, for instance, how trust worthy or stable a service might be. Similarly, we enrich the descriptions with annotations to Schema.org Actions, providing in this manner a coarse-grained classification for the registered services in terms of a widely used schema. Finally, in order to better support the annotation of sensors, iServe empowers a novel sensor annotation facility which is able to exploit textual descriptions, sensor data and their combination in order to figure out the kind of sensor (e.g., temperature sensor) and type(s) of data produced (e.g., Celsius degrees).

The descriptions gathered and enriched by iServe represent the common grounds upon which the platform offers advanced support for the discovery and use of the registered components, allowing users to find relevant services or things that can help them fulfil their business logic needs. In particular, iServe implements a range of search and retrieval facilities allowing users to filter and rank all registered services according to several criteria. First and foremost the platform relies on Lucene for supporting text-based search over the entities registered. Additionally, the platform offers state-of-the-art and semantic discovery by reasoning over semantic annotations whenever they exist may they be general classifications (e.g., with Schema.org) or semantic annotations of inputs and outputs. These discovery features are enhanced with further facilities ranking the results by taking into account other aspects such as the popularity of a given component or how active the community behind it is. See Figure 7 for the overall architecture of iServe.

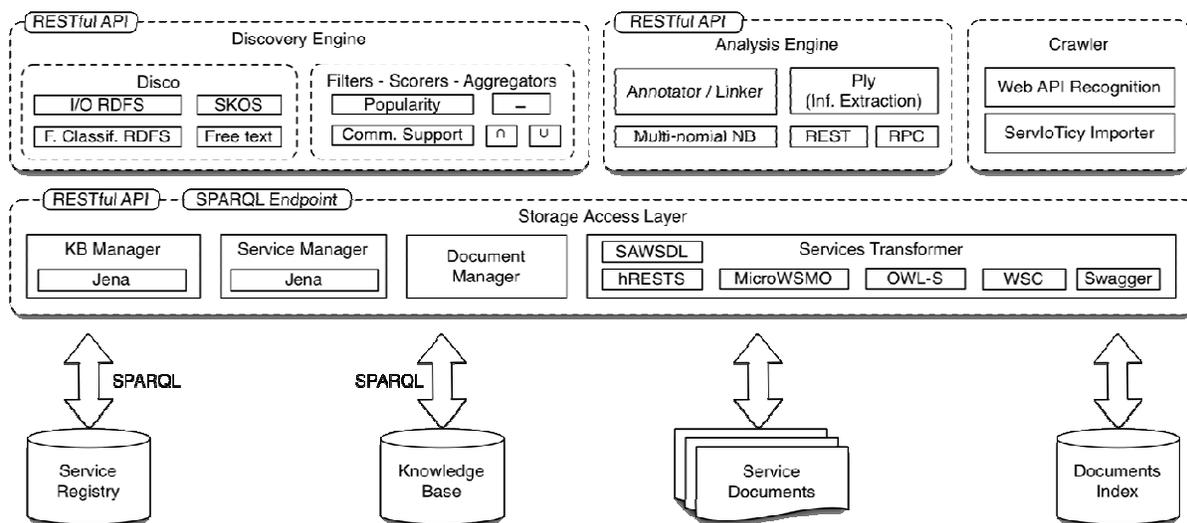


Figure 7: iServe Architecture

Besides providing an unprecedented integrated solution for the aforementioned features, iServe embodies scientific contributions in several aspects:

1. **Distributed fault-tolerant RDF Store.** The service discovery engine is layered on top of a service registry, which provides basic CRUD (Create, Read, Update, and Delete) operations of RDF service descriptions. The backend used for storing these descriptions is implemented using a novel distributed and fault-tolerant RDF store, connected to the upper-layers using the standard SPARQL query, update, and graph store protocols for compatibility and transparency. **Error! Reference source not**

found. The RDF storing solution devised is based on Apache Fuseki (an open source RDF store) and the Paxos replication infrastructure. The Paxos replication algorithm is a quorum based algorithm where a write succeeds once a majority of the replicas accept it. This style of replication can withstand F failures Given $2F+1$ replicas. The advantage is that it can tolerate slow replicas, and can gain progress even if some replicas are engaged in long running queries. With careful design, read performance can scale linearly with the number of replicas. *This replication style fits the case where there is need for highly consistent replication, with support for scalable read performance, and support for long queries on some of the replicas.* The Paxos variant that we use supports 'reconfiguration' - the set of replicas can be dynamically changed without stopping the cluster - a tricky issue that is often glossed over in competing approaches and implementations. This feature facilitates dynamic cloud deployments and geospatially dispersed deployments. Our implementation also provides a framework for snapshot management & automatic state transfer after failure and network partition.

2. **Highly-scalable discovery.** Service discovery has traditionally been approached as a one-of activity to be sporadically carried out by humans when looking for services. As a consequence the interface exposed by discovery engines assumes that requests are fully specified in terms of a well-defined interface and categorisation. The need for providing convenient finer-grained interfaces that could, for instance, help discover services able to consume or produce (a subset of) certain types of data as usually required during composition has hardly been identified, let alone addressed. Similarly, discovery being understood as a task triggered by humans, response time has usually not been much of a concern. Thus, response times of discovery engines are orders of magnitude above what would be acceptable for a composition engine should it delegate the thousands of fine-grained discovery requests it needs to issue at composition time. As part of this research we have carried out theoretical and empirical analysis showing that with the adequate interface granularity and indexing, discovery engines can support high performance advanced service discovery with minimal response times. These indexing mechanism and discovery algorithms have been implemented within iServe.
3. **Automated sensor annotation.** Fundamental towards supporting advanced service discovery is the existence of rich annotations to the sensors allowing the infrastructure to know the type of sensor and output data produced. As part of the infrastructure we have devised an advanced annotation solution that is able to exploit existing textual descriptions and data produced by the sensors to figure out the right annotations with high accuracy. Experiments dealing with over 15 different types of sensors showed an accuracy of over 74% in the annotation process.
4. **Advanced Recommendation and Filtering.** The discovery engine implements a configurable pipeline allowing the seamless integration and use of diverse results filters and rankers based on the metadata available. Most notably the discovery engine includes advanced Trust, Security and Popularity based filters and rankers allowing us to exploit metadata about users, services/things, and providers to better refine discovery results.
5. **High-performance Service Composition.** An analysis of the service composition literature highlights that, regardless of the approach, a central task that needs to be frequently performed throughout the composition activity, is the actual discovery of suitable services to use. Yet, despite the strong dependency between both activities, research and development in both areas has evolved for the most part independently. As a result composition engines do not benefit from the latest advances in discovery which prevents their evolution. Additionally, and most importantly, this approach relies on the unnecessary and often unrealistic assumption that the entire set of services should be locally available to the composition engine. This assumption requires pre-importing all services locally which is only viable for those registries providing entire public dumps of the service descriptions they hold. However, this leads to an unnecessary replication of data with potential synchronisation issues, and indeed it may well go against the interests of registry providers. Furthermore, relying on pre-importing service descriptions prevents benefiting from newly registered services as they appear, and significantly hampers the use of distributed third party registries as sources of service descriptions. We have carried out a theoretical analysis of service composition in terms of its dependency with service discovery, and have provided a practical implementation showing how a scalable and high-performance composition engine can effectively be devised through a seamless integration with third

party, possibly remote registries. As part of this research we have provided a number of contributions including i) a formal framework that defines service composition in terms of core service discovery tasks; ii) a reference implementation of this formal framework based on the adaptation of two independently developed components, namely ComposIT and iServe; iii) a detailed performance analysis of the integrated system, highlighting both the unacceptable performance achieved when using the typical out-of-the-box discovery implementations, as well as the fact that top performance is achievable with the adequate discovery granularity and corresponding indexing optimisations. All these results are implemented within COMPOSE's service management infrastructure

4.1.3.3. glue.things

The developers' portal glue.things is the first access point into the COMPOSE platform for end-users and developers. It integrates various front-end components for IoT application development as well as back-end components which enable the COMPOSE core features. The interworking between all components is the integrated COMPOSE platform. Data flows from bottom-up through the data management and the service layer to the developers' portal. Integrated components such as data management, service discovery, security and cloud deployment interact with the developers' portal. The developers' portal via the GUI provides direct access to particular features of the back-end components.

The developers' portal is a multi-tenant one stop shop for developers who wish to create IoT based applications in COMPOSE. This portal serves as the entry point for external users of the platform, whether the users are application developers, smart objects providers, or end-users who consume COMPOSE applications. The developers' portal provides a user-friendly GUI based interaction mode which helps developers whatever applications they might come up with.

Architecture Overview

The developers' portal contains three parts that guide the developers through the process of application creation. The smart objects manager is used to create and manage service objects, which are the COMPOSE internal digital counter-parts of real world physical smart objects. It provides features for smart object virtualization, management and policies for authorization and authentication. Once a service object is successfully created end-users are directed to the smart object composer for creating applications of their choice, using data coming from the smart objects, or directing commands to these smart objects. The smart object composer integrates capabilities for applications' discovery, security and deployment. Finally, the automations component allows for sharing of created compositions with other registered users. Developers can choose to share their creations publicly which are persisted in storage. These persisted automations are available for other users for instantiation and deployment.

At its back-end the developers' portal interacts with the cloud infrastructure in order to deploy, run, and manage created applications. The main interactions of the developers' portal are with:

1. Security – to identify and authenticate users; obtain proper tokens for smart object interaction, and access to applications; identify applications with security flaws and recommend corrections.
2. Cloud deployment – to make the actual deployment of the created application.
3. Service discovery – to locate existing building blocks that developers can use for creating a new application.
4. Service recommendation – present the developers with recommendations concerning the choice between different services that can serve the same functionality.
5. Service composition – provide a front-end for the assisted composition services to help guide the developer to the COMPOSE application that will fulfil his requirements.
6. SDKs – to integrate and access various Smart Objects from the developer portal.
7. Service composition – to connect existing building blocks to workflow applications and make them accessible via RESTful APIs.

8. Data management (ServIoTicy) for registering service objects and data processing pipes, and displaying their data streams.
9. Reputation: present to the developers the reputation score associated to COMPOSE entities of interest to him.

Smart Object Manager

The smart object manager is responsible for the registration of devices. The developer uses the platform for creating a user account and getting an API token for development. The smart object manager provides three easy steps for connecting devices with the platform. The manager is fully integrated with the COMPOSE platform for initiation and registration of service objects. Developers can easily add devices to the platform via choosing device templates.

glue.things also fully supports the creation and usage of various combined Service Object streams, so called Data Processing Pipelines (DPP). It integrates with the APIs and functionalities that are provided by ServIoTicy in regards to DPPs.

Creation of Data Processing Pipelines: In order to create DPPs, glue.things offers a web form to users which is easily comprehensible. The form is similar to the form that is used to create Service Objects, in order to provide a familiar interface and that caters to a friendly user experience.

Smart Object Composer

The smart object composer was built on top of the browser-based flow editor Node-RED. The composer is a graphical tool, to easily create workflows, that integrate the various services and service objects delivered by the COMPOSE platform.

The smart object composer combines textual and graphical models as appropriate. An example of the former is a syntax coloured scripting language editor. An example of the latter is Yahoo Pipes¹. While this might be a simple wizard like process to construct causality chains, like IFTTT², the smart object composer provides a more complex tool, to create applications and workflows. It is best compared to the Yahoo Pipes editor, though providing more standardized building blocks and staying more extensible. The smart object composer provides building blocks that enable the creation of workflows that rely on clearly defined control flow constructs. The smart object composer prohibits the linking of building blocks with incompatible inputs and outputs and thus already provides a basic validation capability.

Use of Data Processing Pipelines: DPPs can be used from inside the Composer that is part of glue.things. The Composer which provides a developer-friendly interface to quickly create IoT applications in a graphical manner makes it easy to create logic flow graphs. Users simply select desired nodes from a palette and connect these nodes by wires to create a logic flow in which a message is passed along the graph and whereby nodes act upon received messages to send out their own messages to further connected nodes.

The composer provides nodes for Service Objects as well as Data Processing Pipelines. The interface for both of them is the same, so there is no unnecessary distinction between them. The same nodes that support Service Objects also support Data Processing pipelines.

4.1.3.4. Security

COMPOSE generates multitude degrees of freedom. It enables the interaction with numerous devices and applications, the ability to process a new magnitude of data in completely novel ways. Combined with the simplification of the development process COMPOSE requires a security framework that supports the openness of the IoT, ensures the governance over data, and supports non-security-experts in the development of secure applications.

In contrast to the assumptions of existing security frameworks we face unpredictable contexts in which data is processed and applications are executed. Data becomes easily reusable and may be processed by various types

¹ <http://pipes.yahoo.com/pipes/>

² IFTTT <https://ifttt.com/> is a service that enables customers to connect channels (e.g., Facebook, Evernote, Weather, Dropbox, etc.) with personally created or publicly shared profiles known as "recipes".

of applications with different functionalities and properties. Further, applications emerge from the combination of other services or applications. Their internal complexity and impact on data may be unknown or very hard to determine. Thus, static security perimeters and their respective policies defined on applications or devices are infeasible.

COMPOSE addresses these issues by shrinking the security perimeter to the granularity of data. We apply techniques inspired by the decentralized label model (DLM) to highly dynamic architectures in which enforcement must be applied in an ad-hoc fashion and on various computational entities. More importantly, we consider user-defined security policies. Users specify how, by whom, and in which context their data should be processed and which data should gain which kind of access. To define and enforce this control, COMPOSE combines formalisms and techniques of the important DLM and the Usage Control (UCON) model.

COMPOSE also faces devices which may run legacy, vulnerable, or manipulated code changing their intentional functionality. Devices bound to our platform may be physically manipulated to generate malformed data. Further, services which process and consume IoT data may contain malicious or vulnerable code. As a consequence, our security architecture detects and allows the tracking of misbehaving entities which do not allow direct security enforcement. Hence, our architecture provides a reputation system, monitoring entities in COMPOSE. As reputation values are another dimension of our policy framework, we enable architecture configurations which only involve entities with particular properties. In combination with the generation of provenance information for individual data items and its integration in the policy framework, users will be able to define security policies which can also prevent complex data harvesting attacks.

Architecture Overview

The centre of the security architecture is the security core. It hosts essential components such as an attribute based IDM and the global policy decision (PDP) as well as policy information points (PIP). For scalability we support local instances of these components. Yet, the centralised servers guarantee the consistency of security critical data. The same holds for additional components that extend the functionality of the security core: The reputation- and provenance manager, the static analysis, and the instrumentation component.

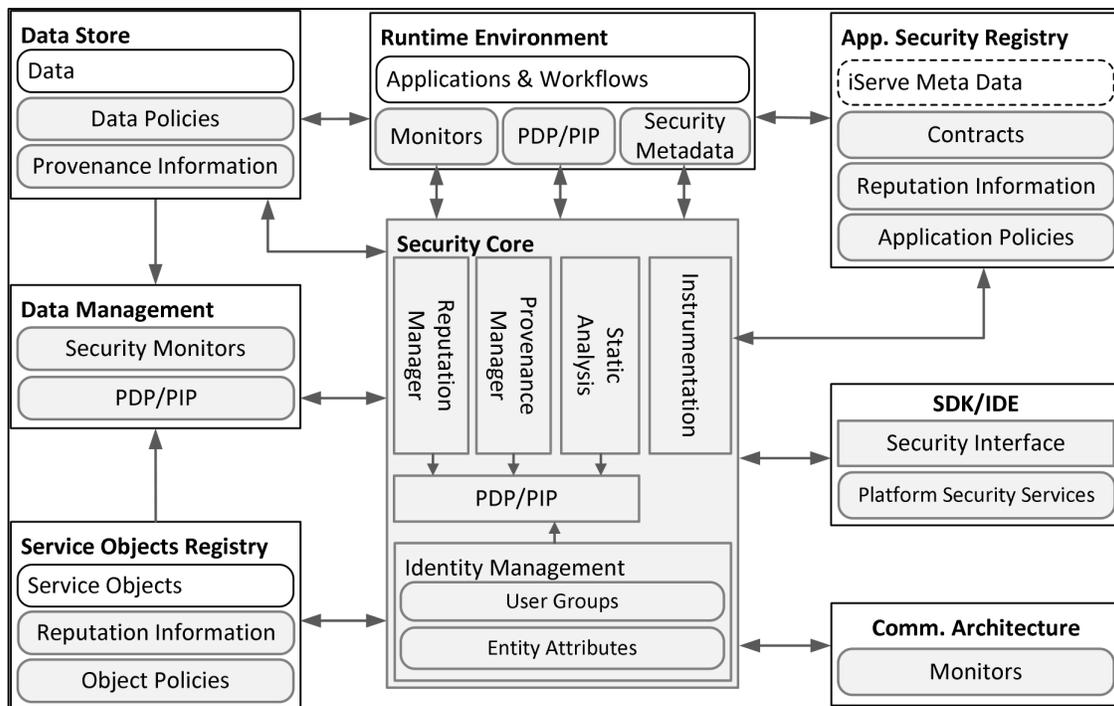


Figure 8: Security Architecture

To abstract from the complexity of ServioTicy (see above) we distinguish three main components: Data Store, Data Management, and Service Object Registry. The latter administrates virtual representations of devices, together with their security policies and reputation information. In particular, reputation information is updated frequently according to the use of the associated devices. As soon as devices generate data, it is stored in the Data Store together with its appropriate flow policies derived from device policies. Every time data is used to generate a new data item, provenance data is updated based on the operations performed.

The data management deploys a local PDP and PIP. In this way, appropriate security monitors control access within the central data processing component of the overall architecture. The authorised usage of devices and data can be enforced and provenance and reputation information can be generated. Additionally, local security monitors can control data-centric flow-control policies.

At the application layer, the execution is secured by dynamic security monitors which can be either holistic, i.e., all components of an application are monitored, or selected in-lined reference monitors control execution. This enables the fine-granular flow tracking and enforcement of data and the monitoring of specific application properties. To better support scalability, we also deploy local PDPs, PIPs, and stores which allow the local accumulation of security metadata. System-wide security meta data about an application, i.e. reputation information, ownership, or access rules, are stored in the security registry.

The simplification of the creation and deployment of IoT applications is an essential functionality offered by COMPOSE. Hence, we complement this functionality with mechanisms that allow non-experts to assess the compliance of applications with their security requirements and policies. For this purpose, we integrated features for editing policy settings, validating and re-configuring data flows within applications, for checking provenance, and for assessing the reputation of system entities into glue.things.

Identity Management

Our platform offers an attribute-based approach. Every user can tag himself or his entities with attribute values. Once entities are tagged with attributes, e.g. the brand of the device, they can be used to specify security policies, e.g. accept data only from devices from brand X. The main problem with this approach is to ensure trust in this information without creating a centralised authority.

Our platform solves this by providing a generic attribute-based IDM framework. This framework allows users to approve attribute information depending on the group to which they belong.

Flow Enforcement

Inputs and outputs of any COMPOSE entity which can process data are dynamically labelled with a data-centric flow policy. Apart from specifying the entities allowed to access, execute, or alter a component, flow policies also describe the security requirements of data entering a component and the security properties of data leaving it. Thus, each data item is annotated with security meta-data.

Therefore, COMPOSE developed a policy framework which covers all these aspects to avoid additional evaluation overhead. The language used in COMPOSE is inspired by ParaLocks. *parameterized locks* specify when a possibly polymorphic actor can retrieve information about a data item. Open locks represent fulfilled conditions under which information can flow. A set of such locks is interpreted as a conjunction of conditions. Combining those conjunctions by disjunctions yield a policy. We adopt this security specification as it is also based on the DLM, it is simple, evaluation is efficient, and it can be used to map against classical access control schema. To obtain a specification language feasible for the IoT domain, we merge the Usage Control approach UCON with ParaLocks and obtain the so called UsageLocks. It introduces typed actor and item locks which allow the checking of actor and item attributes, defined by IDM. UsageLocks allows for flexible flow policies for all inputs and outputs of virtualized IoT objects and applications. They are complemented with classical access policies described in a unified way, using the same specification language.

To maintain scalability and efficiency COMPOSE generates contracts which are also modelled using the lock-rule system. A static analysis generates an over-approximation of the behaviour of an entity and stores it in a JSON format which uses locks to specify how the programming logic of an entity impacts the flow of

information, e.g. by indicating under which conditions (lock status) a flow from an input parameter to an output takes place.

Dynamic Flow Enforcement: glue.things runs on top of node.js and the interaction and flow of data between single nodes is managed by Node-RED. In order to support the enforcement of data-centric security policies even when facing user-defined JavaScript code in applications, we modified the execution environment for single nodes by integrating JSFlow. It is a security-enhanced JavaScript interpreter which allows dynamic flow tracking. It covers the full non-strict JavaScript as specified by the ECMA-262 standard and allows the annotation of values with basic security labels. We extended these labels and the JSFlow infrastructure to support UsageLocks and their policies. Further, we extended JSFlow to support most language features and libraries of node.js and Node-RED.

To also track the information flow between nodes, we modified the basic node template of Node-RED. The primitives used for sending and receiving messages between nodes have been extended. The message exchange now ensures that security information for data, i.e. security policies and lock states, is available in the nodes processing them but protected from the user. To maintain scalability and decrease the performance impact, all nodes, except for function node which run JavaScript code, we only symbolically execute the contract of a node to propagate the security labels for data. This removes the need to apply JSFlow to the complete execution of a node.

Static Flow Enforcement: Evidently, contracts are an important pillar for the scalability of our dynamic flow control. However, they also have an important impact on the static flow enforcement. Contracts are generated by an over-approximating static analysis of the flows generated by an application. The analysis for basic nodes, i.e. nodes which do not represent another composed application, is based on TAJs. Comparable to JSFlow, we extended this static analyser by important language features required to analyse Node-RED applications and to process policies based on UsageLocks. Results from this version of TAJs can be transformed into a contract indicating the flow-relevant behaviour of a COMPOSE entity.

To analyse and generate contracts for complete workflows, the project developed a model checker for Node-RED flows. The flow specification is translated into an abstract model and enhances it with the security policies and contracts for basic nodes or applications. The checker validates the correctness of this model and identifies non-compliant data flows, i.e. it searches for traces which perform access to data items although locks specified in the security requirements for these items have not been opened. Results of this analysis are also used in glue.things. Developers with little or no security expertise are able to validate the compliance of applications with the security requirements of their data or the data of other COMPOSE users. This prevents the deployment of insecure applications which will then be subject to dynamic enforcement. In case of solvable security conflicts, the platform offers them so called reconfigurations which will insert additional security services to solve detected problems.

Hybrid Flow Enforcement: The use of contracts can tremendously simplify the analysis of complex software and support dynamic enforcement. We further exploit the precise information generated during the static analysis to allow code instrumentation. This does not only involve the instrumentation of JavaScript code in user-defined function or customised Node-RED nodes, but it also includes the instrumentation of workflows. Thus, we are able to further reduce the performance impact of the dynamic flow control and can integrate logging or enforcement mechanisms.

Reputation

The reputation manager, also called PopularIoTy, calculates reputation for entities based on three aspects: popularity, activity, and feedback.

Popularity reflects how often a certain Service Object or application is used, i.e. invoked by other entities. In case of data management, whenever data is generated, COMPOSE stores notifications in the data storage. Likewise, monitors placed in the runtime store notifications when an application is called. This information is processed to calculate a popularity score.

The *activity* score attempts to reflect whether an entity is behaving properly. In specific cases, data items, i.e. sensor updates, within the data management can be discarded due to several reasons: lack of security policy compliance, developer-provided source code interpretation errors, or developer-intended filtering. When

sensor updates are discarded due to the previous reasons, a notification is stored in the data store. Afterwards, this information is used by PopularIoTy to decrease the activity score for Service Objects that drop sensor updates, especially in the case of policy compliance and source code problems.

PopularIoTy also encourages users to contribute to the reputation calculation through *feedback*. It is comprised of text, and a numerical value reflecting the user's perception about the entity. Furthermore, to allow the system to filter biased or useless feedback items, users can rate feedback elements specifying whether the comment is helpful or not. Afterwards, the feedback score calculation uses this meta-feedback as a basis for excluding or including a particular feedback comment in the overall score.

Data Provenance

COMPOSE combines data sources to provide innovative functionality. This generates additional privacy concerns when sharing data among entities, and it can complicate the process of debugging applications; particularly, because data sources are commonly provided by third parties.

We provide a provenance manager. It tracks origins of data, the operations performed on it, and the time when operations took place. These tracking mechanism have been integrated into ServioTicy and in the application runtime environment. It empowers users to define policies based on data provenance, e.g. to allow a Service Object to receive data only if it has been processed by a particular application. Further, visualising the provenance of data can help users to detect when certain errors occur; for example, if several data sources are combined, but one of them is malfunctioning, the developer could examine the sources of correct values, and compare them with wrong values to isolate the malfunctioning device. Also, provenance information has an interesting potential to help to protect the user's privacy. For instance, it could eventually help to detect when particular applications harvest and correlate information from specific entities, hinting to the possibility of user profiling.

4.1.3.5. IoT PaaS

The COMPOSE IoT PaaS is manifested as a customization of the Cloud Foundry (CF) environment to better serve the IoT domain. This IoT PaaS provides the basic mechanisms for deploying, hosting, and managing the COMPOSE platform internal components, as well as the service objects, and applications provided by COMPOSE developers. The starting point was an open web oriented PaaS to which necessary components and services to better suit the IoT domain were added.

All the pieces of technology detailed above together with the CF infrastructure and several additional internal components constitute the IoT PaaS. Main added external services are the data management component, which communicates with the real world to ingest data, store and process it; service discovery which holds searchable information on all COMPOSE entities; Developers' portal which provides all COMPOSE services in a consumable manner; along with ample security measures. The main internal components which were added are the COMPOSE controller, which communicates with the developers' portal and takes care of life cycle management, along with the mediator which interacts with the underlying cloud infrastructure on behalf of COMPOSE.

All COMPOSE components, such as new applications and service objects, are hosted or exposed via the platform, either as services or as applications. In addition, external developers applications are hosted as well on the COMPOSE platform, and can bind to COMPOSE provided services.

In order to ease the introduction of the various COMPOSE customized services into the platform we have developed a Universal Service Broker, which eases the task of the new COMPOSE customized service provider, as well as easing and mainstreaming the life of the platform provider.

The COMPOSE cloud lies at the heart of the COMPOSE platform, and all other COMPOSE components are integrated and interact with it.

An integral part of the COMPOSE PaaS is the service deployment component, which is realized within the COMPOSE controller. This capability is integrated within the platform cloud and interacts on the one hand internally with the cloud controller in order to deploy, monitor, and manage COMPOSE applications, and on the other hand interacts with the developers' portal in order to provide the automatic applications deployment capabilities in an easy to consume manner for the developer.

From the developers' portal a click of a button initiates an elaborated application deployment on the underlying cloud. Capabilities include instantiation, binding, running, and managing the application within the cloud. This component interacts with the security component to ensure that the application is checked and validated. Following this the application will be bound to the services it requires, it shall be instantiated on a cloud VM, and the platform management will be made aware of the newly run application, making it accessible to the external world.

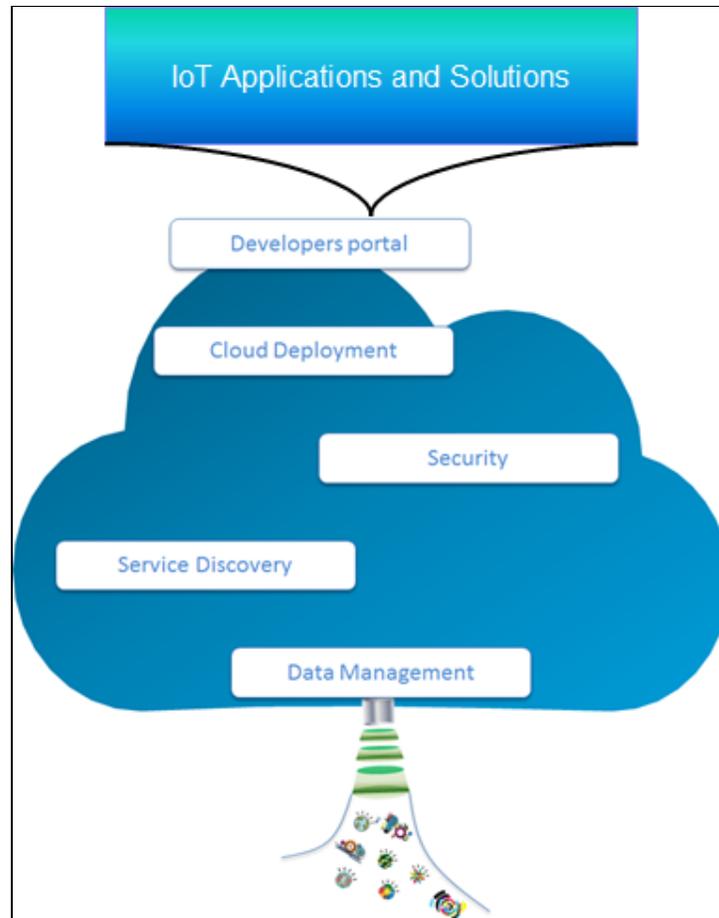


Figure 9: IoT PaaS

A central part of the COMPOSE controller is the COMPOSE mediator. The mediator is an internal multi-tenant component implemented in java which runs as a Cloud Foundry application. The mediator is responsible for pushing the applications to COMPOSE's Cloud environment. The COMPOSE mediator is a central piece of the COMPOSE controller. Both COMPOSE applications and COMPOSE workflows are handled by the mediator. For COMPOSE Workflows the mediator controls the correct order of execution of the different components.

The COMPOSE IoT PaaS comes with two kinds of services: Generic services, such as databases, and a variety of additional custom made middleware that are unique to COMPOSE. The most prominent customized services which are incorporated into COMPOSE are:

Data management – Serves for the ingestion of data from smart objects as well as dispatching commands to smart objects. In addition a data storage layer provides access to historical and real-time data, along with the possibility for applications to subscribe to data related states and conditions which are of interest to them.

Service discovery component which helps developers find existing building blocks to embed within their new application. The service discovery component serves as the backbone for the recommendation engine as well as the assisted composition engine.

Security – security related services include among others an Identity management component, to manage users along with their access rights. In addition static analysis and dynamic checking of security related policies is supported, as well as data flow, provenance, and users and applications reputation.

The COMPOSE communication service is a highly scalable, light-weight, fully-distributed, network-aware communication middleware that has self-organizing structured overlay properties. It provides publish/subscribe messaging to applications, for inter communication services between applications and service objects. In addition it serves as the backbone for the monitoring infrastructure.

4.1.3.6. Pilots

Smart Retail

The Smart Spaces pilot consisted of an *in-store analytics* solution for monitoring the behaviour of customers in a grocery store. By means of a high-precision localisation technology (Quuppa), the position of all shopping carts and baskets in the store was tracked over time. The data was stored in COMPOSE and then processed in order to produce meaningful KPIs for store managers to understand what people do in their store.

The store where the pilot was executed is part of the COOP Trentino group, and is located in Ravina (Trento, Italy). It is approximately 1000 square meters large, and organized in 8 aisles and 4 macro areas. The overall deployment consisted of 20 antennas and 100 TAGs. The initial deployment was completed in April 2014, with the pilot lasting for 18 months. During the entire duration of the pilot, we had bi-weekly meetings with store executives in order to discuss the data being collected and potential future developments. Various store units were involved in (i) the evaluation of the pilot (ii) discussions about the possible utilization of IoT technologies for improving store operations. This included executives responsible for store operations, sales and marketing. The overall feedback of the store executives was extremely positive, especially for the role that such technologies can play in modern retail.

The solution is implemented as a web application augmented through the COMPOSE platform. Shopping carts and baskets are modelled as Service Objects, and provide a shopping cart/basket location update every 0,5 seconds. The data is stored in COMPOSE and consumed by a COMPOSE application (3 NodeRed flows) in charge of providing some basic data processing such as, geo-fencing, data smoothing. The COMPOSE application was then exposing both a REST interface and a web socket towards external applications.

We have implemented 2 external applications: an analytics application, which was processing such data and providing a web dashboard to store managers. Through the dashboard it is possible to have a real-time view of what is happening in the store (e.g., where customers are in real-time, how many customers are at the service desk, etc.), as well as an aggregated view of what happened during the previous week (e.g., average time spent by customers in each aisle or department).

COMPOSE semantic service discovery was used to e.g., easily group all the shopping carts and baskets in the store into a Data Processing Pipe (DPP). As an example, a DPP was used to reveal the presence of queues forming at the checkout counters.

Finally, security was used to control the access to data. In the specific pilot setting this feature was not used, but considered as extremely relevant by store executives for providing different data access to different applications.

The Smart Retail pilot was awarded of the following prizes:

- Lamarck price 2013: this is a prestigious price awarded during SMAU 2013, the leading Italian exhibition on ICT technologies.
- Innov@Retail price: on March 26th, 2014, RetailerIN was awarded the prestigious Innov@Retail prize by Accenture/Sole24Ore for the best analytics application
<http://www.ilsole24ore.com/art/impresa-e-territori/2014-03-27/sono-dodici-innovatori-retail-063703.shtml>

Smart City

The Smart City pilot has been built to provide a proof of concept of the integration of urban infrastructures and IoT devices into the COMPOSE platform; and enabling citizen sensor scenarios by using personal smartphones as Smart Objects for location sensing.

- UC1: SmartVAO - Sustainable mobility by promoting the use of High Occupancy Vehicles

Main prototype of the Smart City pilot is based on a ride sharing application. Car owner users can create groups for sharing the car with other users that should follow the same route. The goal is to reduce the number of cars by promoting the high occupancy of vehicles (HOV). The shared car can access a restricted parking area if it accomplishes the HOV conditions.

UC1 prototype is based on a back-end system and two types of user applications: a web application for the manager of the restricted parking area that is also the promoter of the car sharing service; and a mobile application for end-users of the car sharing service.

In addition, a service, which is running in the COMPOSE run-time, has also been designed and deployed. The SmartVAO system interacts with this service for checking the HOV conditions of a car-sharing group.

Furthermore, the prototype interacts with the COMPOSE SDK for registering, deploying and managing Service Objects under security premises.

The pilot is being carried out with the University of Tarragona, and could be easily replicated in other environments.

- UC2: Contextual information based on location

In this use case, the prototype also interacts with the COMPOSE SDK for registering, creating and managing Service Objects under security premises.

UC2 prototype is based on a backend system and a web application. The backend supports several M2M gateways of urban infrastructure (Urbiotica, WorldSensing, Sentilo, etc.) which are connected to the SDK for registering, creating and updating data of the urban infrastructure located in the SmartZone, as well Barcelona city.

The web application is connected to the COMPOSE platform using several of its components (COMPOSE SDK, IDM, ServioTicy and iServe) to show and manage the status of the urban infrastructure.

In addition, a Sentilo M2M gateway has been included in the Glue.Things composer in order to be used by any other developers that aim to manage a Sentilo Urban infrastructure.

Through the Smart City pilot, main benefits of COMPOSE platform for different stakeholders have been highlighted:

Smart City managers	COMPOSE allows them to manage efficiently, in a homogeneous way, different urban infrastructures, overcoming interoperability issues.
Actuators & sensor providers	COMPOSE allows them to virtualize their physical devices becoming more accessible, by breaking the “silos” solutions for smart cities, in order to be easily integrated in new smart services.
Smart services developers	By using COMPOSE tools, they will have homogeneous access to different Smart Objects and services, in order to create quickly and easily new Smart Services.
Citizens	They will benefit from the rise of new smart services that improve their daily quality of life.
SmartVAO pilot parking manager	The manager of the HOV pilot parking interacts with COMPOSE to provide the parking spots, manage and monitor the overall car sharing service through a web application also connected to COMPOSE.

SmartVAO pilot users They use a mobile application to create car-sharing groups and request a parking spot of the reserved parking area.

Architecture Overview:

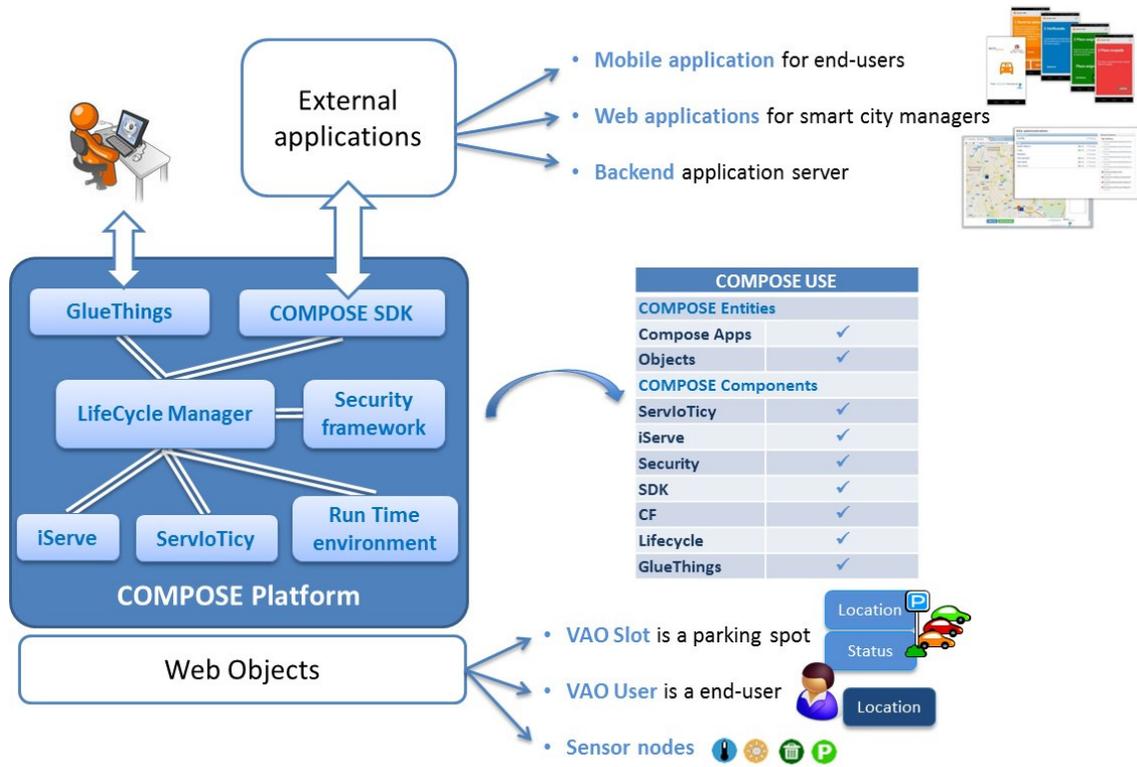


Figure 10 Smart City pilot & COMPOSE components integration

Web Objects of the Smart City use cases involve smartphones and heterogeneous urban sensors. Whereas smartphones provide user location data, urban sensors provide contextual information related to environmental measures, parking status and location systems through the Cellnex M2M gateway, which has the capability to communicate over HTTP with the COMPOSE platform for storing application data and supporting data processing.

COMPOSE Service Objects (SOs) are deployed in the Smart City pilot in order to serve as endpoints for connecting Web Objects to the COMPOSE core platform. SOs will be needed for interacting with Web Objects, for pushing updated data to the platform (i.e. location, environmental measures and parking status), and for subscribing to notifications (i.e., when the slot parking sensor status is updated).

High level applications will be needed for monitoring both the information that is gathered from the sensor platform and the information that is provided by users of Smart City pilot use cases. Related to the SmartVAO pilot, COMPOSE applications are used to check the HOV conditions when a car-sharing group requests a parking spot (number of users, distance between users, distance trip). For the contextual information COMPOSE applications are being used as a Location Based Service (LBS).

The Smart City pilot has an external layer of mobile and web applications. This external layer is formed by a front-end, which provides the User Interface (UI) for end-users of the Smart City pilot; and a back-end system that supports the applications. External applications enable the interaction of end-users with COMPOSE internal components through the COMPOSE SDK in order to perform the automation of Service Objects registration, authentication, deployment and management.

The COMPOSE SDK is connected to the Life Cycle Management module (LCM), which provides a single and unified external access end-point to the COMPOSE platform. Therefore, LCM simplifies the complexity of the whole COMPOSE platform, because it isolates external users and applications from all the internal COMPOSE components and their interrelationships.

Therefore, the list of COMPOSE components that are used by the Smart City pilot is:

- **COMPOSE SDK:** This component provides an API for the automatic creation of Smart Objects, which is used by the applications and M2M gateways for the automatic creation and registration of Smart Objects (<http://docs.composesdk.apiary.io/>)
- **Security framework:** Security mechanisms are applied by the Smart City pilot in order to ensure that Service Objects are only managed by authorized users; as well as that only authorized services can access to their information channels.
- **iServe:** When a Service Object is created, this SO entity is also registered into a semantic repository to support contextual searches, allowing third parties to re-use Smart City pilot SOs and applications.
- **ServioTicy:** This component is used to manage data channels of Smart Objects.
- **Run time environment:** The use of Cloud Foundry and the run time are embedded in the overall use of the COMPOSE platform by the Smart City Pilot. They contain the resources to support the management of applications.
- **Life Cycle Management module:** Its use is also embedded in the overall use of the COMPOSE platform by the Smart City Pilot for creating and registering Smart Objects. It is present when there is an interaction with the developer portal through Glue.Things, and when an external application of the Smart City pilot interacts with the COMPOSE SDK.

In addition, the COMPOSE developer portal is used by the provider of the sensor infrastructure for the manual creation and registration of Smart Objects. Besides, a M2M gateway for Sentilo³ platforms has been designed, deployed and integrated in the Glue.Things composer in order to be used by any other developer.

Smart Territory

In the case of the Smart Territory, the pilot targets the development of a location-based service providing user recommendations for outdoor activities in the Trentino region. It combines several data sources into a smart and personalized service enhanced with context-aware capabilities to provide users with real-time recommendations and notifications on events and conditions of outdoor facilities, such as Ski resorts.

The main stakeholders of the Smart Territory pilot are the following:

- **Go2Ski App users:** the end users of the developed mobile app who can benefit from the aggregated and presented contextual data. In addition, they generate data through slope ranking and social media uploading.
- **MeteoTrentino data provider:** The main provider of weather and snow-related sensors (current temperature and rain sensors, weather forecast data, snow quality sensors, etc.).
- **OpenData providers in the area of Trentino:** Providers of static data like OpenStreetMap data and information about slope geography, types, points of interest etc.
- **Ski resort managers and Trentino province:** The province of Trentino along with ski resort managers receive information on the utilization of the app and have a real time view of users (number of users and their location) on slopes.

³ Sentilo platform is an open source sensor and actuator platform promoted by the city of Barcelona, and replicated in other Spanish cities (www.sentilo.io)

- External developers (potential stakeholders): Developers who wish to create similar apps (e.g., for different types of sport or service delivery, like booking facilities) can use the services and SOs developed and provided by the pilot for location-based weather information, user review collection, etc.

From a high-level approach, the system is architected as a web application backend delivered through a REST service, and consumed from a mobile client responsible for UI and context data collection.

Integration with the COMPOSE platform has been fully realized by registering each of the involved Service Objects into the general system infrastructure. The pilot application provides an integration, on which the client mobile app communicates with COMPOSE platform for object and data management.

The final version of the Smart Territory pilot is built as a Smartphone application built using the Appcelerator Titanium Platform for enabling the cross-platform deployment of the code (Android, iOS and Windows Mobile are available). In terms of functionality, the current version is capable of collecting the user location and the user preferences and of presenting aggregated information. As mentioned, a very simplistic back-end environment has also been already implemented, providing mock-up COMPOSE services for Web Objects provision (weather forecasts, status, Open Data sources...).

The smartphone application consists of the following components: The UI: The main graphical interface that the user is using to interact with the application, the sensor services that run in the background and collect information about the user context, and the communication mechanism as implemented by the MobileSDK. Both REST-based and WebSockets-based communication is used for pushing sensor data to the COMPOSE platform but also for receiving important notifications from the back-end.

In terms of physical device requirements, Smartphones running the Smart Territory application need to be equipped with location and activity sensors, and of course Internet connection needs to be available. An additional aspect to be remarked is the creation of support tools and components for development, which has also already been addressed. Specifically, the COMPOSE MobileSDK has been created, containing a native Android COMPOSE library, the JavaScript COMPOSE library for Titanium Appcelerator (with MQTT bindings), and the JavaScript library for browser support. The availability of this SDK is intended to simplify the development of client applications exploiting the COMPOSE platform.

In the scope of the final version of the system, 276 Service Objects have been generated with pilot data (206 weather stations, 18 forecast stations and 52 snow stations).

These data sources enable the full set of system functionalities, as described in D7.1.1 through requirements:

- Set-up of user profile parameters, including the preferred types of ski slopes and the notification frequency
- Provision of recommendations on demand or on detection of a specific location
- Provision of push recommendations when changes of weather are detected
- Social networking functionalities (management of friends, groups...)
- Collection of crowd-sourced data about context.
- The initial version of the system provides a prototype implementation of the core application features, such as enabling the collection of geolocation information, querying the system for nearby POIs, real-time delivery of location-based information and real-time delivery of information about friends.

During the development of the pilot and the Go2Ski app components there were no SOs or applications available for service discovery. However in the context of the pilot (as mentioned in previous sections) a great number of weather and open data sensors has been integrated into COMPOSE and is available to external developers. For instance, a developer can search for slope-related service objects within the developer's portal and retrieve a list with all available SOs that contain such related information.

A total of 140 users have downloaded and actively used the app during the ski season of 2015 in the area of Trentino and provided positive feedback regarding the features and usability of the platform.

4.1.4. The potential impact (including the socio-economic impact and the wider societal implications of the project so far) and exploitation of results

COMPOSE is expected to have a big impact on industry, academia, and in society. The impact on industry is due to the ease in which diverse IoT applications can be built in multiple domains. This capability is made possible by the combined capabilities within the platform, namely the data management, discovery, scalable and elastic deployment, all in a secure environment. New IoT applications in various domains can be conceived; investments in existing applications will become lower.

Academic impact is attributed to the wide publications in many domains made by consortium partners, both academic and industrial, in addition to the establishment and management of new conferences. This is expected not only to lead to future citations, but also to the formation of new alliances and collaborations. Current advances in IoT, mainly scale and data processing are expected to lead to fresh ingestion of ideas from the academic to the industrial arena.

The impact on society is expected thanks to the ease in which new applications can be introduced to the market as well as the reduced costs for existing applications. IoT applications are mainly geared towards end consumers, whether directly or indirectly, so this is a field which is expected to have a noticeable favourable effect on society as a whole.

Due to the consortium's belief in the potential impact we agreed to maintain a live version of the platform for one more year, to attract and support external developers to interact with the platform, hoping to foster a larger developers' community around the platform, to help the probability of its long term sustainability.

IBM

IBM's contributions to the project, besides acting as the coordinator and technical leader, are: (i) Architecture (ii) IoT PaaS (iii) Scalable graph store (iv) scalable communication (v) monitoring. This work has resulted in the publication of 5 papers in top conferences, in addition to internal exposure in IBM conferences and forums.

For dissemination, in addition to the conference papers, the project has been showcased in various conferences and workshops, two of them chaired by IBM. One keynote has been given as well in PDGC 2012. In addition various blog posts were published, and interviews were published in digital media.

The IoT PaaS potential impact on industry is high, as explained above. Seven patent applications are in different stages of processing, 4 were filed and 3 are being drafted by the IP lawyers, after IBM has decided to file them.

The main exploitation path for IBM is related to promoting within the company parts of the technologies developed as part of COMPOSE. This internal promotion process started with the patent applications mentioned above, and is being pursued with the new IBM Business Unit promoting the IoT. The final aim is to push some of the technologies developed within the COMPOSE project into new and existing (e.g. Bluemix / IoT Foundation) IBM market products.

Two related pieces of technology developed by IBM are currently being integrated into IBM products:

1. Advanced Scalable communication infrastructure elements
2. Scalable consistency services.

As for impact on society, that is a corner stone of the IBM work in this field as can be seen by the long lived predecessor work pioneered by IBM on Smarter Planet. This area has been identified as one that can have a strong favourable impact on society as a whole due to the new possibilities this technological advancement open in the near and far horizon.

CREATE-NET

As outcomes of the COMPOSE project CREATE-NET has identified two different potential impact areas that will be explored and supported beyond project lifecycle, and specifically:

1. Customizations of COMPOSE platform components for specific application domain needs deriving from potential customers requirements.
2. Evolution of the Trentino Smart Territory use case into a platform for business analytics tool for marketing and mobile e-commerce in the ski areas.

CREATE-NET identified these two related main exploitation paths:

1. Regarding the COMPOSE platform, CREATE-NET started to engage with organisations that could be potentially interested in IoT platforms. The aim is to extend the current IaaS services of CREATE-NET with a **PaaS solution based on COMPOSE**. The main target customers are currently companies providing advisory services in the IoT domain (B2B target market segment). As a concrete example, CREATE-NET is developing on behalf of **IoTango** (<http://www.iotango.com>, a U.S. based company) a back-end platform for managing IoT devices and data. Specifically, they need a prototyping environment for mock up solutions. ServioTicy will be the main component from COMPOSE utilised in such a platform. Following this first positive contact and ongoing collaboration, CREATE-NET will exploit its existing contacts to identify other potential early adopters.
2. Regarding the developed **Trentino use case**, the plan is to maintain the developed app for another ski season (Winter 2015-2016) and evolve it into a service to be offered to ski areas integrating additional requirements that were collected, validating the solution with different ski areas in Trentino. Currently (Oct. 2015) there is already an agreement to co-invest in the evolution of the current solution with 2 ski areas (22 potential ski areas in Trentino could be involved within the end of this year). Specifically, ski areas are interested in using the mobile ski app as a means for collecting analytics on user behaviour in the ski areas and in using the “push” notification channel offered by the mobile app in order to deliver safety and mobility messages in real-time to skiers. After the 2015-2016 winter season where the mobile ski app will be maintained and promoted again, ski areas will discuss the evolution and consolidation of the service in a business analytics tool for marketing and mobile e-commerce in the ski areas. The actual and future engagement and interest of local stakeholders (e.g. ski slope management and service providers, local promotion and tourism organisations, sport associations, etc.) will drive in any case the final decision for the following seasons.

CREATE-NET has already defined plans to support the above mentioned exploitation paths. For the Trentino use case CREATE-NET foresees the evolution of current software solution into a new service that will leverage on COMPOSE backend features. A dedicated backend infrastructure will be put in place, based on the COMPOSE solution. The mobile app will be revised and evolved in order to include additional features for skiers (mainly, performance tracking and social network capabilities). The backend infrastructure will be expanded in order to integrate a big data analysis and a business analytics tool. A web interface will be created for ski areas to allow ski areas to perform marketing analysis of collected data and to use the “push” messaging channel according to their needs. Maintenance and support activities will be also provided in order to guarantee for the 2015-2016 winter season availability and requested level of services to skiers and ski areas.

Ski areas will at the same time co-invest with own resources in marketing the ski app in order to maximize adoption by skiers.

For the ongoing collaboration with IoTango the following activities are in the plans of CREATE-NET:

- Created a first demo version of the platform;
- Platform exposed to external clients of IoTango company, for evaluation;
- Collection of feedback on specific requirements deriving from customers working in specific market segments (collaboration with IoTango);
- Evaluation of a possible commercial agreement with IoTango for platform customization and adaptation to the specific contexts and needs;

Evaluation of additional potential partnerships with other companies in order to evaluate market potential of the solution that will derive from the previously mentioned activity line.

FOKUS

FOKUS aims to exploit glue.things as the main interface of COMPOSE solution. In particular, FOKUS will leverage its wide industrial network (including several IT and manufacturing big players, as well as several innovative SMEs and start-ups) to create a relevant set of potential early adopters. The list includes:

- **Consumer Electronic Manufacturer:** FOKUS is in discussion with a Consumer Electronic manufacturer to adapt glue.things as a potential platform for managing and connecting different CE devices and to offer to end-users an easy-to-use UI to create automations on-the-fly without the need of programming experience. The Consumer Electronic Manufacturer asked about supporting offline capability of glue.things in addition to the online capability which requires to be always connected to a central server. **Energy Provider:** FOKUS also discussed with an energy provider to use glue.things as a potential Automation Tool for end-users to define automations and rules that let users monitor, control and save energy. The energy provider asked to offer a subset of the features in a native mobile application that allow users to create simple tasks directly from their Smartphones. Another important aspect discussed is about reducing the complexity of connecting devices from different manufacturers and using different protocols. This is a general issue due to the fragmentation of the IoT market and to the different technology silos. Since FOKUS is actively participating and contributing to different W3C groups especially W3C WoT IG (Web of Things Interest Group), a proposal for a Thing API that defines an abstraction layer between applications and underlying technologies was introduced to the group.

The main idea is to create a cooperation/business partnership with the engaged companies and develop customised/vertical IoT platforms, based on the open source COMPOSE technologies. Specifically:

- **Add Offline support for glue.things:** FOKUS plans to extend glue.things to add offline support in addition to the online mode. Users will be able to decide if an automation should run locally in the home network or in the cloud as currently supported in COMPOSE. Using this feature, users will be able to define automations and control their devices at home without the need to be connected to the Internet. From a privacy point of view, users should be able to decide and monitor, if data can leave the network or not. Furthermore, there are many devices especially those using BLE (Bluetooth Low Energy) which are only accessible from devices nearby like a tablet or Smartphone.
- **glue.things mobile app:** FOKUS also plans to provide a mobile application for glue.things for different mobile platforms with reduced features that enables users to create, deploy and monitor simple automations directly from the mobile app. The mobile App may also include a runtime environment to run automations in an offline mode as described previously. The smartphone or tablet will act in this case as a bridge between the home devices.

Thing API proposal: FOKUS proposed to the W3C WoT IG (Web of Things Interest Group) an API called “Thing API” (name of the API may be changed in the future). The main goal of the API is to bridge between different technology silos for discovery, provisioning and communication between applications and things from different manufactures. An application can be a web page running in the browser or an application running in a JavaScript Runtime Environment like Node.js. Members of the WoT IG expressed interest in the API proposal and discussed potential extensions to support additional features. The Thing API proposal was designed from the experience gained during the COMPOSE project in terms of connecting objects from different manufacturers and using different technologies together. The concept of a “Thing” is equivalent to a Service Object in COMPOSE. It is used in the W3C context because it is more suitable for the Web of **Things**. On the other side, the API will have an impact on the COMPOSE platform in the future. Both the COMPOSE Dashboard and Composer may use the Thing API as a generic interface to connect to different underlying technologies. New technologies and protocols need to be mapped to the API, but the existing applications and services will not be modified. This will motivate developers and providers to implement applications and services that run on different runtimes or user agents without any modification. FOKUS will continue working on the proposal and will discuss with industrial partners to create a W3C Working Group.

The Open University

OU decided to further invest on the know-how gained in the COMPOSE project and create a dedicated IoT Lab. The lab has been launched by the KMi (which is the department directly involved in COMPOSE) and

currently involves distinct research groups within the KMi but also in other departments of the OU. Specifically, the following groups are currently involved:

- The Data and Services group directly involved in COMPOSE has deployed a subset of the COMPOSE infrastructure locally to support in-house smart sensing. Currently these activities are targeted at the automated indoors location using passive devices, and at the automated identification and approximation of relevant events (e.g., ongoing meeting) through the analysis of distributed sensor data and their correlation.
- The Social Mining group is contributing to the indoors deployment within KMi with facilities for tracking and helping reduce energy consumption automatically. This deployment, being performed in conjunction with the Decarbonet project (<http://www.decarbonet.eu/>), leverages the aforementioned infrastructure and ties it to commercial energy trackers as well as to existing research in Social Science in order to trace, understand, and promote engagement and behaviour change in citizens.
- The IoT Lab also helped the Institute of Educational Technology (IET) part of The Open University develop the SALSA application in the context of the OU led MK:Smart project. SALSA, which is currently live across Milton Keynes, uses beacons technology to help non-native English speakers get contextual help and learn English while they live the city. Although in this case COMPOSE technology is not directly involved, the know-how gained during the project certainly contributed to the success of SALSA.

In general, the future plan is to support internal but also external (e.g. via collaborative R&D projects) researchers in developing IoT applications, by exploiting the COMPOSE solutions (in particular the iServe and the ServioTicy components). The OU shall thus continue with its ongoing R&D cycles incorporating new solutions improved and developed in the course of COMPOSE as well as exploiting the know-how gained in the course of the project.

The IoT Lab launched in the course of COMPOSE will be established as a main catalyser in this regard whereby existing deployments and collaborations shall continue. Notably, the energy consumption initiative is being contemplated as one possible exploitation opportunity both in terms of future research activities as well as internally in an attempt to reduce the university's carbon footprint and the derives costs.

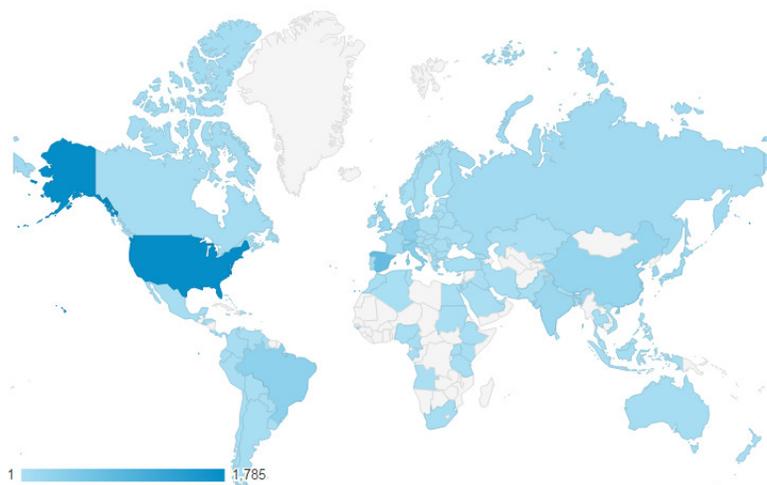
Further related activities within the department include a Quantified Self initiative focussed on health aspects which is currently planned together with the CARRE project (www.carre-project.eu) and we are also contemplating further opportunities in the context of the MK:Smart smart cities project beyond the previously mentioned SALSA application.

Finally, it is worth mentioning that iServe has been adopted to support Uruguay's eGovernment services portal (<http://www.linked.uy>). Although The Open University is not directly involved in the project, the OU have provided help and support during the development of this project which is now approaching the end of the prototyping phase.

Barcelona Supercomputing Center

BSC has been the main contributor for the design and implementation of the ServioTicy component, which implements one of the core capabilities of COMPOSE: the data processing plane. But the work has not been limited to this component, but has reached the integration with many other aspects of the COMPOSE platform. In particular, the integration of all the security components, restrictions, requirements and policies has been significant. The work performed by BSC has resulted in several papers published in international conferences and workshops, a paper submitted to a peer-reviewed journal, several participations on events and invited talks, and more publications to come possibly after the end of the project.

The potential impact of the technology developed in the ServioTicy component in the industry is high. Proof of this is the fact that only in the last 12 months of the COMPOSE project, the ServioTicy portal has received visits from 118 different countries in the world, with a significant presence of North American and Asian countries, besides that all countries in Europe can be found among the visitors for the period.



The main exploitation path for BSC is related to an internal project that has been recently approved by the BSC direction body. The idea is to collect, analyse and share several data streams coming from the BSC super-computing centre. Data will include both sensor data (e.g. temperature, energy consumption) and computer-processing data (e.g. workload). ServioTicy will be the main component of the devised platform. The platform will be accessible from several research group in Europe (e.g. cooperating with BSC and/or using the BSC super-computing) to carry-on their specific experiments by using real-time data from the available data streams and creating/re-using data analytics services. An initial step of this effort has been the integration of millions of simulation results for dust in suspension all over Europe and North Africa. The simulations are run periodically by the Earth Sciences department at BSC, and are provided to the national meteo agency in Spain. The Earth Sciences department is seeking for means to integrate simulations like this with data coming from sensors carried on by people on the streets. Dust in suspension is only one of the metrics, but others like air pollution are very sensitive for citizens living in large European cities.

The initial pilot of the project has started and is available in its early stages at servioticy.com/demo-earth/. As for impact on society, the results of the integration of a sophisticated and scalable data plane like ServioTicy with a set of advanced security technologies like those developed in WP5 is very valuable in terms of privacy and traceability of data. There are multiple technological solutions that in some way overlap with ServioTicy capabilities, but security-wise they lack real mechanisms to ensure privacy to IoT users. The outcome of the COMPOSE project includes a set of techniques that allow for user-customizable policies on data associated to IoT sensors, which is a very relevant topic for the years to come.

INNOVA

The COMPOSE project allowed INN to gain a lot of know-how in the IoT field, from both the technology and reference markets perspectives. This, together with its technology transfer and valorisation mission and main role in the project (exploitation manager), enabled INN to investigate and develop a specific exploitation path.

INN started to identify and engage with potential early adopters, which could be interested in the COMPOSE technologies (specific components, whole platform and vertical applications). The aim is to create awareness of the COMPOSE platform, work with possible customers to understand the limits and the key benefits of the solution and progressively move towards a commercial COMPOSE platform (see also Section **Error! Reference source not found.**). In other words, INN is committed to exploit the whole COMPOSE platform by first creating partnerships and vertical applications to fund the necessary commercial development (including both business/marketing and technology development activities). The work to achieve this has already started, but it will necessarily continue after the end of the project.

Concretely, the following collaborations have been already launched:

- INAVYA (<http://inavya.com/>). COMPOSE platform could be the technology enabler for several ideas/business ventures. INAVYA could couple its main business of start-up accelerator in UK with a

technology service to quickly mock-up applications in the area of IoT and IoS and mobile computing. The resulting demo can be used to attract investors.

Status: different ideas coming from INVAYA customers have been drafted and waiting for public/private funding.

One of them (**DATI**) received a grant of about £100k for a feasibility study of a novel C2B solution where personal data (coming from wearable sensors and/or mobile devices) can be really exploited (i.e. sold) by data owners to get benefits from a wide range of interested companies (health, restaurants, shops, etc.). The project started in October 2015 and will last 1 year. At the current stage, the following companies have already agreed to cooperate as possible test-bed of the DATI solution (in particular: enhancing their current digital solutions with COMPOSE capabilities, such as integrating real-time data stream, matchmaking among services, deriving insights about collected data and trigger actions accordingly):

- Blippar (www.blippar.com), in the context of mobile Augmented Reality solutions.
- IbexLab (www.IbexLab.com), in the context of mobile solution for Point-of-Sale (POS) Services.
- DueDil (<https://www.duedil.com/>), in the context of brand management.
- GrubClub (<http://grubclub.com/>), in the context of restaurants.
- TBL Connector (<http://www.tblgroup.co.uk/>). COMPOSE platform (in particular ServioTicy and iServe) can be used as backend platform for applications in the energy and environmental reporting areas. In the next years the EC, but also other big countries like India and China, will issue strong regulations and policies on reporting environmental and fuel consumption aspects (in particular for ships and trucks). The data consumption capabilities of COMPOSE are a perfect enabler for those applications.

Status: a business plan applying the idea illustrated above in the field of shipping companies (which have strong energy efficiency regulations already in place) has been elaborated and different activities for public and private funding carried out. Currently they are waiting for feedback (end of November 2015)

University of Passau

As a university, UNI PASSAU plans to exploit the gained know-how in the IoT domain mainly for training purposes. Specifically, UNI PASSAU is planning to create training courses and research opportunities (e.g. for new PhD students) in the area of security analysis for high level IoT applications, by starting the investigations and concrete outcomes (solutions) of the COMPOSE project. In this context, UNI PASSAU is also participating in the collaboration with the SECOR (Secure Service Composition for the Internet of Things) project which runs until August 2017. The goal of the project is to create infrastructures which help to design secure composed applications for the IoT. This infrastructure should be used for teaching and training as well as for development at the University of Passau and also within the SECOR project. For this purpose, particular interest has been shown in the extensions of University of Passau which allow the analysis of software composed by non-security experts. Thus, in the near future, our effort in this project will try to elaborate the first collaboration which started during the runtime of the project. Two students helped to elaborate a user-friendly interface to define flow policies developed for the project and integrate them into glue.things.

In addition, UNI PASSAU plans to exploit (also commercially) the library use case. The responsible for the library at the university expressed his interest to carry on the activities also after the end of the project. He is particularly interested to obtain useful insight (e.g. student behaviour, peak hours, etc.) out of collected data analysis. To this end, the university committed to buy additional hardware to extend the experiment. In addition, initial contacts have been started to extend the experimentation (and possibly sell a concrete solution) to other universities in Bavaria.

In order to develop its exploitation path, UNI PASSAU has planned the following activities:

- The COMPOSE architecture has already been focus of several Master and Bachelor thesis and seminar topics at the University of Passau. We introduced several students to the security internals of the platform. The fact, that COMPOSE is a functional piece of software which can be extended easily is very inspiring

for students and will help to advance the COMPOSE marketplace to an interesting playground for sophisticated security analysis and enforcement.

- We have been discussing possible COMPOSE contributions to the kick-off meeting of the SECOR project in Tunis in November. It will discuss and plan the activities of our University in the SECOR project. We plan to contribute a short tutorial video to show some of the features we can offer to SECOR. Further, the first steps for the design of the training infrastructure have been discussed. The glue.things interface extended with the respective security features is one of the key elements, COMPOSE will contribute.

The Smart Library use case is currently in an experimental pilot phase. The technology and user acceptance is tested in a busy area of the library. Thus, fast adaptations to new requirements and evaluation of deployed technology are possible and simplified. If this pilot phase is successful the library administration of the University are thinking to extend the test area. We have also discussed a possible deployment of our application in the whole University. In the end, the potential application of the system in the set of all Bavarian libraries has been mentioned

U-Hopper

The main exploitation path for UH is related to the technology and expertise developed through the pilot in the Smart Spaces domain. This allowed UH to enter the retail analytics market, with a solution that is able to address a problem affecting all retail store managers: "What do customers do inside my store?" Indeed, while most retailers employ business intelligence solutions to analyse receipts and understand purchase trends and data on the number of customers entering the store, very little (if anything) is known on the in-store conversion funnel (also called the 'purchase funnel'). Retailers and other brick-and-mortar businesses are actively looking for tools and technologies able to analyse how shoppers behave in the store in order to improve sales, operations and security. A study from Brickstream of 124 retail executives across operations, marketing and merchandising concluded that "developing an understanding of which promotions attract customers, where customers go in the store and what products they choose are key requirements in today's market"⁴.

Within COMPOSE, UH developed a solution to unleash the full potential of in-store analytics for the retail sector. In particular, it provides a comprehensive in-store analytics platform capable of allowing store and marketing managers to make informed decisions on the many aspects related to the daily management of their stores. This enables them to reduce operational expenditures, while enabling more effective, yet profitable, shopping experiences for customers.

UH is already marketing a solution that is partially based on the technology and expertise developed within the COMPOSE project. This allowed UH to acquire new customers both in EU and US. In particular:

- In US U-Hopper deployed 3 in-store analytics solutions to Adagio Teas Inc., namely (i) in 3 stores in Chicago, and (ii) in the warehouse in New Jersey.
- In Italy, the grocery store SAIT which piloted the solution will be converted into a paying customer in 2016.
- We are currently piloting the solution in a large Italian grocery chain. The pilot started in autumn 2015 and will end by the end of 2015.

Finally, U-Hopper recently spinned off such solutions into a separated company name ThinkINside (thinkinsi.de), focusing on the area of services and analytics for indoor spaces. The company will operate initially in the retail sector and then expand into other sectors such as, e.g., healthcare and Oil&Gas. The company has been established in the US (S. Jose), with an EU branch based in Trento (Italy). This will allow us to properly target the US market and scale the company operations internationally.

ThinkINside is currently targeting an Angel investment of 1,5 Mil EUR in order to further accelerate the company activities, especially in the US. We are currently in advanced discussions with various Seed Investors.

⁴ Brickstream, "Retail Analytics: What's In-Store?", <http://brickstream.com/resources/research/retail-analytics-whats-in-store/>

GEIE ERCIM/W3C

In terms of impact, COMPOSE has enabled the launch of a credible standardisation effort on the “Web of Things” within the W3C. The IoT remains at early stage of maturity and is currently being held back by fragmentation and data silos. Companies are still exploring different approaches and have yet to strongly emphasise the need for convergence to realise the full potential of the IoT. There is opportunity to apply ideas learned within the Compose project to bridge different IoT platforms via Web technology standards based upon sharing semantics and using rich metadata to decouple application logic from the underlying platforms and protocols. Standardisation in the IoT area has the potential to overcome the current “silo” solutions, thereby increasing the market for solution providers and leading to overall growth of the market for IoT solutions in a way similar to the effect of standardized Web technology had for the exponential growth of the Internet.

W3C is seeking to exploit COMPOSE project results in its work on standardisation. We launched an Interest Group in early 2015 and plan to launch a Working Group in early 2016.

The Interest Group will continue its work after the end of COMPOSE and is focusing on pre-standardisation activities including a survey of use cases and requirements across a broad range of application domains and business sectors, a study of the architecture and technical landscape, including discovery and provisioning, security and privacy, and liaisons with external IoT organisations.

We now plan to clarify the details, charter a standardization working group, and to standardise the Framework in this future Working Group. The aim is to enable distinct IoT platforms to interoperate through the World Wide Web via a new class of Web servers that range in scale from microcontrollers to cloud-based server farms (e.g. the Compose run-time platform), and which expose virtual objects as proxies for physical and abstract entities. These entities are modelled in terms of metadata, events, properties and actions, and described with W3C’s formalism for Linked Data, along with standard bindings to scripting APIs and common protocols. This builds upon the experience gained within Compose on the use of JSON for describing services, and the use of Linked Data for service semantics.

Eurecat

The impact of the Eurecat participation in COMPOSE can be found through some of the core activities the organization has been involved lately, and is also bound to some strategic approaches already in the company roadmap.

As a research and technology transfer organization, Eurecat’s technical background acquired and grown within the framework of COMPOSE provide a highly valuable asset for future research activities. The Data Analytics group within Eurecat has taken the main responsibility on the contribution to Compose, and as a result a significant part of the provided efforts have been put into activities around Data Analytics. Specifically, works on stream processing and data quality in the specific domain of sensor data offer a significant potential for Eurecat. As an example, developments on Anomaly detection on sensor data carried out within the WP3.1 of COMPOSE have already been incorporated as technical background by Eurecat researchers working in the e-Health scenario.

In addition, functionality provided by COMPOSE as a data publication platform is also being evaluated to check how feasible is to use such platform as the basis for further R&D projects in the Eurecat roadmap. Scenarios around patient monitoring in smart homes and energy management in buildings provide an optimal scenario for using COMPOSE as the underlying infrastructure for sensor management.

Beyond the scientific and technical aspects, participation in COMPOSE has also provided a relevant impact in terms of visibility. As an example, the Barcelona Hackathon that took place in early 2015 brought a significant and highly relevant impact amongst the local developer community.

In terms of exploitation potential, background acquired in COMPOSE will contribute to the formulation of new proposals for H2020 projects. Eurecat is also addressing the market dimension of such potential. The organization manages the Catalan IT cluster, gathering many of the local technological companies. COMPOSE is being disseminated to companies in the Cluster, and an estimation of its potential impact as a foundation for future commercial exploitation activities is in scope.

RETEVISION

RETE will follow two main exploitation paths:

- 1) The experience of the Smart City application will be replicated in other cities or scopes, and possibly extended with additional smart services for giving added-value to the urban infrastructure. Currently, the solution is a working prototype (under validation in a concrete setting at the University of Tarragona). To become a market product additional steps need to be performed, also according to the validation outcomes. Thanks to the cloud approach of COMPOSE, the scalability and replication of the solution will not be an issue, and RETE will mainly focus on customising and adapting the application.
- 2) Some of the COMPOSE components (mainly ServioTicy, the Security framework, the Cloud Foundry service runtime environment, the Lifecycle Management module and the COMPOSE SDK will be integrated into our existing smart city platform: the SmartBrain platform. The latter is designed in order to cover the needs in the cities, allowing the homogenization of the data collected from different sensor networks or other platforms, enabling simultaneous use by different potential users (citizens, public services, providers or developers). Another important aspect of SmartBrain is the isolation of the urban infrastructure from software developments.

In order to develop its exploitation paths, RETE has planned the following activities:

1. Continue, during at less 2016, the Operation of the current developed components: LCM and COMPOSE SDK.
2. Integration of some of the COMPOSE components in the product roadmap during 2016 in Italian and Spanish market.
3. Include the selected COMPOSE components as a basic piece of the international smart cities strategy
4. Push the selected COMPOSE components as interoperability reference model, at least in Spain and Italy.
5. Replicate the Smart City pilot in other areas
6. Promote the Smart City pilot and the selected COMPOSE components in several international shows like the Smart Cities World Congress and the Mobile World Congress.

EVERYTHING

EVT exploitation paths are mainly related to assimilate lessons learned in COMPOSE in the creation of IoT platforms and integrating part of the models and solutions provided by COMPOSE into the existing company's products and services.

EVT main product and business is a cloud platform that can benefit from the advances provided in COMPOSE. For EVT, the most interesting aspects that have the potential to be adapted to our products and clients are:

- Data pipelines: many platforms, including EVT's product, deal with data updates from different sources in the cloud, but the ability to combine this information in real time and create easily aggregations is challenging. COMPOSE's data pipelines adoption could enable new functionalities to be offered to clients that are looking for solutions where different flows of data need to be combined in real time.
- Semantics: the use of semantics in the data model to enable discovery and composition is something that has been always in the to-do list of EVT. COMPOSE's iServe component provides enough flexibility to potentially integrate with our platform.
- Scalability for IoT: one of the biggest challenges of IoT platforms is the scalability. Even though EVT's platform relies on a different architecture, strategies and evaluation methods carried out in COMPOSE will influence the way we evolve in this aspect.

Standard models for the Web of Things: EVT has always been involved in fostering the Web of Things community, promoting the adoption of simple yet powerful data models for the Web of Things. The work of COMPOSE in this point, which led to the submission to the W3C standardisation organization of a proposal for a "Web Thing Model" sets the beginning of the activities that will take this recommendation to the next level of acceptance and usage in the Web of Things community.

Planned activities for EVT are divided in terms of following up the standardisation efforts started under the umbrella of COMPOSE, as well as having a deeper analysis of the possible integration strategy of COMPOSE in our product. More concretely we are planning to exploit COMPOSE at the following levels:

- Active participation and involvement in the W3C Web of Things Interest Group and Community Group, following up the member submission originated in COMPOSE. We have internal plans of adopting this model and we have identified already some clients that could benefit from it.
 - Timeline for participation: ongoing
 - Timeline for adoption of model: Mid 2016
- EVT has currently no capacity of integrating the full stack of COMPOSE technology as is, due to the nature of our processes and status of our business and clients. However, the aspects identified below are starting to appear in conversations with clients and are therefore of high interest for us. We will continue to explore them both internally as well as in an upcoming Horizon 2020 project in which we are participating.
 - Timeline for upcoming project: 2016-2019
- EVT systematically takes part as well in several international IoT events where key adopters and future clients attend and present an opportunity to talk about the platform and future features. Upcoming planned events: CES 2015 (Consumer Electronics Show), DIGIT.EMEA WebIT 2015, IFA 2016 (Consumer Electronics Unlimited).

4.1.5. Main dissemination activities

Dissemination activities performed during COMPOSE project lifecycle have been structured and executed adapting and harmonizing dissemination means and type of dissemination activities to the progress of technical and exploitation work packages, thus adapting dissemination strategy accordingly to the different outcomes made available within project lifespan.

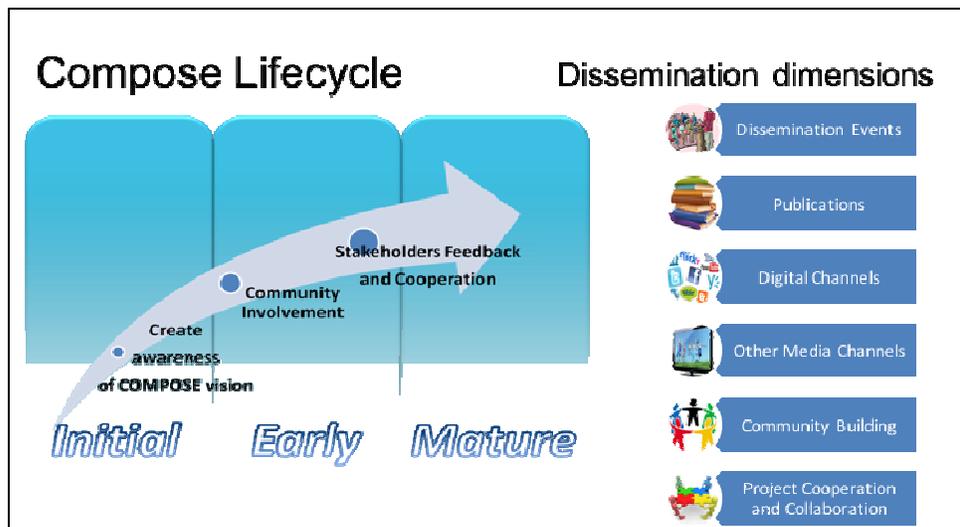


Figure 11: Overall COMPOSE Dissemination Strategy

During first year of the project, dissemination has been directed to communicate the COMPOSE vision and to create awareness of COMPOSE objectives, technical approach and expected results. In the second project period, when the first components of the platform were made available (more specifically, ServioTicy, iServe and Glue.Things portal) a first level of community involvement has been started and put in place, both leveraging online channels (project website, social channels, components dedicated websites and open software repositories for source code sharing) and physical events organization

(hackathons, training sessions and events) involving early adopters in the process of evaluating and testing the delivered solution. Finally, the last project period has been devoted to promote platform adoption towards possible interested stakeholders thus getting in contact through relevant events with potential adopters of the platforms, demonstrating platform capabilities and potential also thanks to the availability of the use cases outcomes that have been widely demonstrated and exhibited. The last period has also been dedicated to reinforce involvement of external developers in the usage of the platform thanks to the availability of the fully final integrated version of the COMPOSE platform; dedicated training material has been prepared and delivered in order to allow developers to autonomously test and use the publicly available version of the COMPOSE platform, while platform availability has been promoted on all available dissemination channels mainly focusing on the online channels (social channels, dedicated videos and the organization of a virtual meetup) in order to reach the widest number of target users possible. The dissemination plan has been structured from the early project beginning in this perspective.

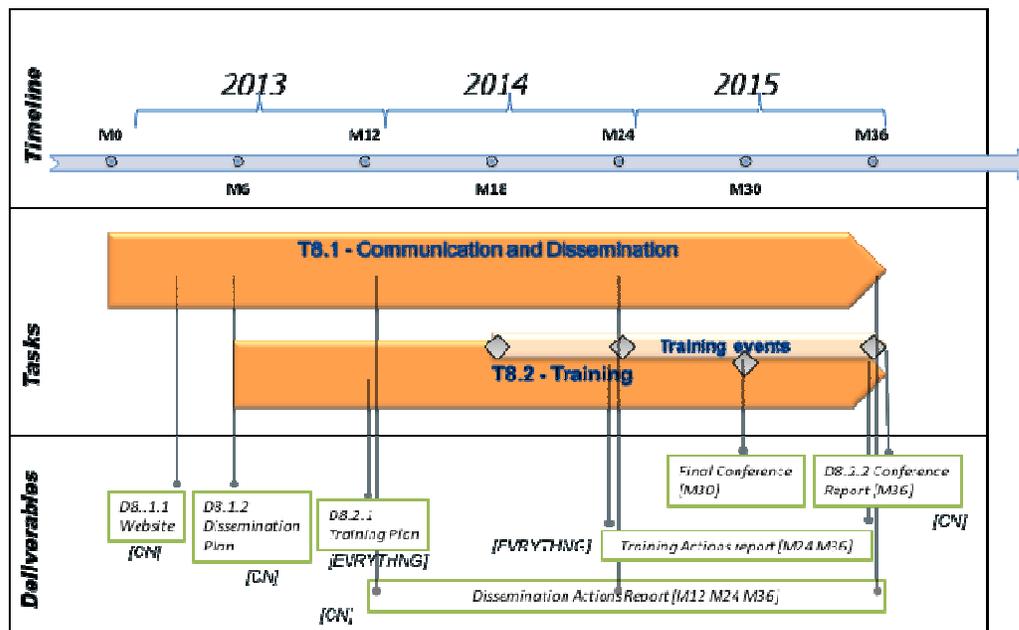


Figure 12: COMPOSE Dissemination Plan

One of the important pillars of COMPOSE dissemination has been the realization of relevant scientific publications in order to disseminate research outcomes towards the proper research communities and exploitation potential of project results.

All details about peer reviewed publications and other dissemination activities performed during the whole project lifecycle are listed in Section A (Tables A1 and A2) of the current document. Enclosed is a summary of the main types of activities that have been put in place during the whole project period.

Online Channels

The **COMPOSE Website** has been created from early in the project lifetime (<http://www.compose-project.eu/>) and it has represented the main channel in the project social channels' communication strategy, being the first tool for disseminating material and news, and the reference point for all the other social channels' dissemination contents.

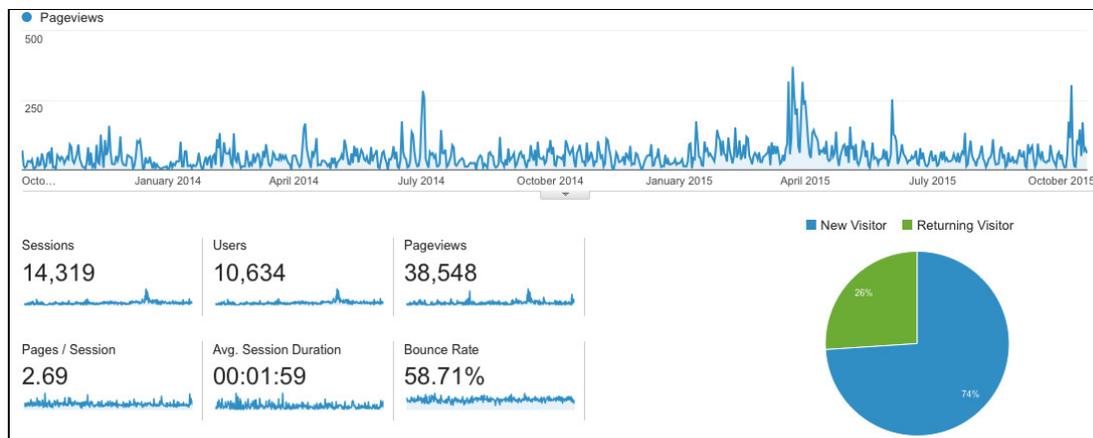


Figure 13: COMPOSE Website: Audience

COMPOSE project received a good visibility worldwide outside Europe, especially in the US and also in Asia and South America.

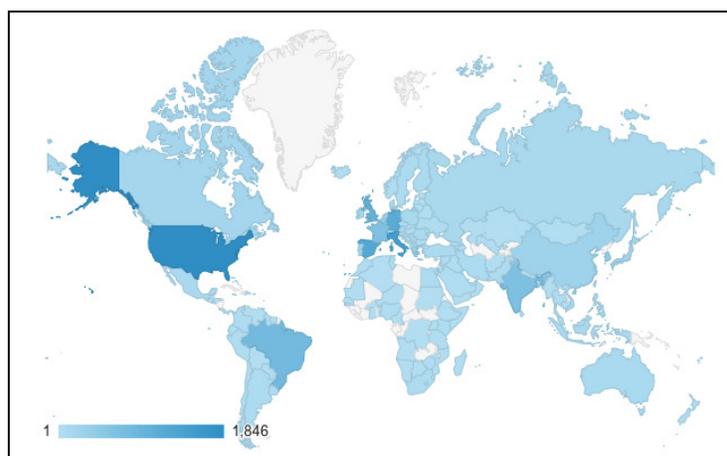


Figure 14: COMPOSE Website: Geography

A periodic newsletter has been issued by COMPOSE, more frequent (monthly, bi-monthly) in the first period of the project, while later substituted by the online social channels presence.

Social Channels most used for COMPOSE dissemination have been the following:

- Twitter: https://twitter.com/COMPOSE_Project
- Facebook: <https://www.facebook.com/compose.eu>
- YouTube: <https://www.youtube.com/user/ProjectCompose> (an official project video, 10 tutorials/demo videos)

Dissemination material has been created, used and customized in order to present the COMPOSE project in many relevant events.

- Project Posters and Flyers
http://www.compose-project.eu/sites/default/files/publications/compose_poster.pdf
http://www.compose-project.eu/sites/default/files/publications/COMPOSE_Poster.pdf
- Project Factsheet:
http://www.compose-project.eu/sites/default/files/publications/COMPOSE_v2_factsheet.pdf

Project Hackathons

During project lifecycle several hackathons have been organized in order to involve external developers in the COMPOSE platform usage

2013		
2013, Q3	Web of Things Hackathon at UbiComp 2013 in Zurich (8 September)	EVERYTHING CREATE-NET
2014		
2014, Q2	COMPOSE Hackathon at IoTWeek 2014 Hackfest in London (16-20 June)	ALL
2014, Q3	1st Co-located Hackathon of COMPOSE – IoTogether (9 September)	CREATE-NET, BSC
2014, Q3	Luna Hackathon (Night of researchers)- Bolzano, TIS Innovation Park	CREATE-NET

Open Software Repositories

As part of the dissemination towards external developers, a set of open software repositories have been made publicly available, where platform components source code, libraries, tutorials and examples have been published. The main project software repository is accessible openly on GitHub through the compose-eu account: <https://github.com/compose-eu>.

In there all the components software and some development tools (e.g., libraries and SDK for mobile development) are at visitor disposal.

COMPOSE components dedicated sites

As soon as the different components of the COMPOSE platform have reached a good level of stability and maturity they have been made available through dedicated sites in order to allow developers to benefit of the full features and documentation offered by the various components. This has allowed to target training activities of COMPOSE in this first project stage offering to participants of hackathons and to external developers software components and services as a set of “tools” offering different sets of capabilities. Now that the integrated platform of COMPOSE has been made available, all the content (APIs, documentation, tutorials, etc.) published for each component are combined into a unique and integrated view.

Specifically, the following COMPOSE components web sites have been setup:

- **ServioTicy:** <http://www.servioticy.com/>
- **iServe:** <http://iserve.kmi.open.ac.uk/>
- **Glue.things:** <http://www.gluethings.com/>

Besides documentation APIs and SDK, these websites provide also information about current development/implementation status, available features, current and upcoming software releases.

COMPOSE Developers Portal

Now that the COMPOSE platform has been made available, a dedicated and unified portal for external developers has been setup. The portal offers a single entry point to the platform, giving an high level description of the COMPOSE platform, the entry point to the dashboard of COMPOSE, the publicly online version of the COMPOSE platform, including an integrated view of all available documentation and a set of training material created ad-hoc in order to allow external developers to autonomously access and test the COMPOSE platform. This portal is available at:

- <http://platform.compose-project.eu>

Prizes/Recognitions

- The COMPOSE Use Case – *Augmented Mobile Retailing Experience* was presented for the first time at SMAU and supported through the COMPOSE project. U-Hopper, the project partner in charge of this use case development, was awarded the Lamarck Prize at SMAU Milano by Confindustria as “*as the most innovative start-up company for its innovative retailer analytics platform, RetailerIN*”.
- COMPOSE has been nominated for the 2014-2015 Internet of Things Award, special category - Technical Enabler: Application Enablement (<http://postscapes.com/internet-of-things-award/2014/iot-application-enabler.html>).
- In 2014, for the second edition of Innov@Retail Award, prize promoted by Accenture in collaboration with Gruppo Sole 24 Ore in innovation in the retail sector, COMPOSE smart retail Use Case was awarded as Best Analytics Innovation”.

Scientific Publications, Presentations

During the three year’s project, COMPOSE partners have produced a meaningful number of relevant scientific publications, in the form of journal papers, conference papers, and books.

A substantial number of invited talks and presentations have been performed in different venues and contexts.

Dedicated workshops and two specific conferences (“IoT As a Service”, Rome, Italy, October 2014, October 2015) have been organized in the context of COMPOSE project.

A detailed list of scientific publications and of other relevant dissemination actions are reported in Section A of this document.

4.1.6. Project public website, and contact details

Contact Details

Mr. Benny Mandler
IBM Haifa Research Lab
Tel: +972-4-8296378
Fax: +972-4-8281112
E-mail: mandler@il.ibm.com

Additional Information

Project website address: <http://www.compose-project.eu/>

Project Logo: http://www.compose-project.eu/sites/all/themes/omega_compose/logo.png

Project Factsheet:

http://www.compose-project.eu/sites/default/files/publications/COMPOSE_v2_factsheet.pdf

Project Posters and Flyers

http://www.compose-project.eu/sites/default/files/publications/compose_poster.pdf

http://www.compose-project.eu/sites/default/files/publications/COMPOSE_Poster.pdf

Project Videos: <https://www.youtube.com/user/ProjectCompose>

COMPOSE Developers Portal: <http://platform.compose-project.eu>

4.2 Use and dissemination of foreground

TEMPLATE A1: LIST OF SCIENTIFIC (PEER REVIEWED) PUBLICATIONS, STARTING WITH THE MOST IMPORTANT ONES										
NO	Title	Main author	Title of the periodical or the series	Number, date or frequency	Publisher	Place of publication	Year of publication	Relevant pages	Permanent identifiers (if available)	open access?
1	<i>COMPOSE: An Open Source Cloud-Based Scalable IoT Services Platform</i>	<i>Raggett, Dave</i>	<i>ERCIM News 2015</i>	<i>no. 101 (2015)</i>	<i>ERCIM EEIG, B.P. 93, F-06902</i>	<i>Sophia-Antipolis Cedex, France</i>	<i>2015</i>		<i>http://ercim-news.ercim.eu/en101/special/compose-an-open-source-cloud-based-scalable-iot-services-platform</i>	<i>yes</i>
2	<i>ACQUA: Automated Community-based Question Answering through the Discretisation of Shallow Linguistic Features</i>	<i>Gkotsis, George, Maria Liakata, Carlos Pedrinaci, Karen Stepanyan, and John Domingue</i>	<i>The Journal of Web Science</i>	<i>no. 1 (2015)</i>	<i>now publishers inc.</i>	<i>Boston - Delft</i>	<i>2015</i>	<i>1-15</i>	<i>http://www.nowpublishers.com/article/Download/JWS-0001</i>	<i>yes</i>
3	<i>An Integrated Semantic Web Service Discovery and Composition Framework</i>	<i>Rodriguez-Mier P., Pedrinaci C., Lama M., and Mucientes M.</i>	<i>IEEE Transactions on Services Computing</i>	<i>no.99 (2015)</i>	<i>IEEE</i>		<i>2015</i>		<i>http://arxiv.org/pdf/1502.02840v1.pdf</i>	<i>Yes</i>
4	<i>ServioTicy and iServe: A Scalable Platform for Mining the IoT</i>	<i>Villalba, Álvaro, Juan Luis Pérez, David Carrera, Carlos Pedrinaci, and Luca</i>	<i>Procedia Computer Science</i>	<i>52 (2015)</i>	<i>ELSEVIER</i>		<i>2015</i>	<i>1022-1027</i>	<i>http://www.sciencedirect.com/science/article/pii/S1877050915008972</i>	<i>Yes</i>

		<i>Panziera</i>							
5	<i>The Web of Things: Challenges and Opportunities</i>	<i>Raggett, Dave</i>	<i>IEEE Computer</i>	<i>vol.48, no.5</i>			2015	<i>pp.26-32</i>	<i>http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7111885</i>
6	<i>W3C plans for developing standards for open markets of services for the IoT</i>	<i>Raggett, Dave</i>	<i>ACM Ubiquity</i>						
7	<i>Why Web Servers Should Fear their Clients:Abusing Websockets in Browsers for DoS</i>	<i>Juan D. Parra Rodriguez and Joachim Posegga</i>	<i>(accepted at) 11th EAI International Conference on Security and Privacy in Communication Networks</i>	<i>Oct.</i>	<i>EAI</i>	<i>Dallas,USA</i>	2015		
8	<i>Data-Centric Security for the IoT</i>	<i>Daniel Schreckling, Juan David Parra, Charalampos Doukas, and Joachim Posegga</i>	<i>(accepted at) 2nd EAI International Conference on IoT as a Service</i>	<i>Oct.</i>	<i>EAI</i>	<i>Rome, Italy</i>	2015		
9	<i>Design of Routing Protocols and Overlay Topologies for Topic-based Publish/Subscribe on Small-World Networks</i>	<i>Chen Chen, Yoav Tock</i>	<i>(accepted at) Middleware 2015 conference</i>	<i>Dec.</i>		<i>Vancouver, Canada</i>	2015		

10	<i>A BASILar Approach for Building Web APIs on top of SPARQL Endpoints.</i>	<i>Daga, Enrico, Luca Panziera, and Carlos Pedrinaci</i>	<i>European Semantic Web Conference (ESWC), Workshop: Services and Applications over Linked APIs and Data (SALAD)</i>	<i>Jun.</i>		<i>Portoroz, Slovenia</i>	<i>2015</i>			
11	<i>Linked USDL agreement: effectively sharing semantic service level agreements on the Web</i>	<i>Garcia, Jose Maria, Carlos Pedrinaci, Manuel Resinas, Jorge Cardoso, Pablo Fernández, and Antonio Ruiz-Cortés</i>	<i>In Web Services (ICWS), 2015 IEEE International Conference</i>	<i>Jun.</i>	<i>IEEE</i>		<i>2015</i>	<i>pp. 137-144</i>		
12	<i>Weighted Overlay Design for Topic-based Publish/Subscribe Systems on Geo-Distributed Data Centers</i>	<i>Chen, Chen, Yoav Tock, Hans-Arno Jacobsen, and Roman Vitenberg</i>	<i>In Distributed Computer Systems, IEEE 35th International</i>		<i>IEEE</i>		<i>2015</i>			

			Confere nce on (ICDCS 2015)						
13	<i>Developing and deploying end-to-end interoperable & discoverable IoT applications</i>	<i>Doukas, Charalampos, and Fabio Antonelli</i>	<i>Communi cation s (ICC), 2015 IEEE Internati onal Confere nce</i>		<i>IEEE</i>		2015	<i>pp. 673- 678</i>	
14	<i>What's around me? Location analytics over Software-Defined WLANs</i>	<i>Riggio, Roberto, Iacopo Carreras, and Erinda Jaupaj</i>	<i>Integrat ed Network Manage ment (IM), 2015 IFIP/IEE E Internati onal Symposi um</i>		<i>IEEE</i>		2015	<i>pp. 1133- 1134</i>	
15	<i>A Data-Centric Security Framework for the IoT</i>	<i>Schreckling, D., Parra, J.D., Posegga, J., Doukas, C.</i>	<i>(Invited Paper), EuCNC 2015: Europea n Confere nce on Network s and Communi cation</i>				2015		

			s						
16	<i>Linked USDL agreement: effectively sharing semantic service level agreements on the Web</i>	<i>Garcia, Jose Maria, Carlos Pedrinaci, Manuel Resinas, Jorge Cardoso, Pablo Fernández, and Antonio Ruiz-Cortés</i>	<i>Web Services (ICWS), 2015 IEEE International Conference</i>		<i>IEEE</i>		2015	<i>pp. 137-144</i>	
17	<i>Enhancing the Retail Experience through BLE Sensors and the COMPOSE IoT Framework</i>	<i>Doukas, C., Antonelli, F., Miorandi, D.</i>	<i>IEEE Sensors Application Symposium 2015</i>		<i>IEEE</i>		2015		
18	<i>Performance Characterization of the Servioticity API: An IoT-as-a-Service Data Management Platform</i>	<i>Perez, Juan Luis, and David Carrera</i>	<i>Big Data Computing Service and Applications (BigDataService), 2015 IEEE First International Conference</i>		<i>IEEE</i>		2015	<i>pp. 62-71</i>	
19	<i>Providing generic support for IoT and M2M for mobile devices</i>	<i>Doukas, Charalampos, Luca Capra, Fabio Antonelli, Erinda Jaupaj,</i>	<i>Computing & Communication Technol</i>		<i>IEEE</i>		2015	<i>pp. 192-197</i>	

		Andrei Tamilin, and Jacopo Carreras	ogies- Researc h, Innovati on, and Vision for the Future (RIVF), 2015 IEEE RIVF Internati onal Confere nce						
20	<i>Linked USDL: a vocabulary for web-scale service trading</i>	Pedrinaci, Carlos, Jorge Cardoso, and Torsten Leidig	11th Extende d Semanti c Web Confere nce (ESWC 2014), In The Semanti c Web: Trends and Challen ges		Springer International Publishing		2014	pp. 68-82	
21	<i>It's all in the content: state of the art best answer prediction based on discretisation of shallow linguistic features</i>	Gkotsis, George, Karen Stepanyan, Carlos Pedrinaci, John Domingue, and Maria Liakata	Proceed ings of the 2014 ACM confere nce on		ACM		2014	pp. 202- 210.	

			Web science						
22	<i>Identity Management in Platforms Offering IoT as a Service</i>	<i>Juan D. Parra Rodriguez, Daniel Schreckling, and Joachim Posegga</i>	<i>IoT As A Service, IoTaaS</i>			<i>Rome, Italy</i>	2014		
23	<i>A Full End-to-End Platform as a Service for Smart City Applications</i>	<i>Charalampos Doukas, Fabio Antonelli</i>	<i>in Proc. of The 10th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob 2014</i>			<i>Larnaca Cyprus</i>	2014	<i>Pp 8-10</i>	
24	<i>glue.things – a Mashup Platform for wiring the Internet of Things with the Internet of Service</i>	<i>R. Kleinfeld, L. Radziwonowicz, C. Doukas</i>	<i>5th Workshop on Web of Things</i>	Oct.		<i>Cambridge, MA, USA</i>	2014		
25	<i>Bridging Web Technologies with M2M Protocols</i>	<i>C. Doukas, J.L.Pérez, A. Villalba and D. Carrera</i>	<i>Position paper at W3C workshop on the Web of</i>	Jun.		<i>Berlin, Germany</i>	2014		

			Things							
26	<i>It's all in the Content: State of the art Best Answer Prediction based on Discretisation of Shallow Linguistic Features</i>	Gkotsis, G., Stepanyan, K., Pedrinaci, C., Domingue, J. and Liakata	ACM Web Science 2014 Conference			Indiana University, Bloomington, USA	2014	Pp. 9		
27	<i>Linked USDL: a Vocabulary for Web-scale Service Trading</i>	Pedrinaci, C., Cardoso, J. and Leidig, T.	11th Extended Semantic Web Conference (ESWC 2014)		Springer		2014			
28	<i>On the Use of Social Trajectory-Based Clustering Methods for Public Transport Optimization</i>	Jordi Nin, David Carrera, and Daniel Villatoro	Citizen in Sensor Networks Second International Workshop, CitiSens 2013		Springer	Barcelona, Spain	2013	pp 59-70		
29	<i>COMPOSE: Building Smart & Context-Aware Mobile Applications utilizing IoT Technologies</i>	Charalampos Doukas, Fabio Antonelli	5th IEEE Global Information Infrastructure &	Oct.		Trento, Italy	2013		http://www.compose-project.eu/sites/default/files/publications/COMPOSE-%20Building%20smart%20and%20context%20aware%20mobile%20applicati	

			Network ing Symposi um,						ons%20utilizing%20IoT% 20Technologies.pdf	
30	:Whitepaper, IoT Governance, Privacy and Security Issues	Gianmarco Baldini, Daniel Schreckling, et al.	Whitepa per, IERC - Activity Chains, AC5 - Govern ance, Privacy and Security issues	Jan.	IERC			2015	http://www.internet-of-things-research.eu/pdf/IERC_Position_Paper_IoT_Governance_Privacy_Security_Final.pdf	
31	Design and Implementation of a Scalable Membership Service for Supercomputer Resiliency-Aware Runtime	Yoav Tock, Benjamin Mandler, José Moreira and Terry Jones	Euro- Par 2013	Aug,		Aachen, Germany		2013	http://www.compose-project.eu/sites/default/files/publications/EuroPar13%20Membership%20Tock%20Mandler%20Moreira%20Jones%20-%20preprint.pdf	
32	A self-managed self-optimized publish- subscribe system", at, , 30 June - 2 July 2013., , ACM	Shahar Chen, Liane Lewin- Eytan, Nir Naaman and Yoav Tock	SYSTO R '13, Article No. 7, Proceed ings of the 6th Internati onal Systems and Storage Confere		ACM	New York, NY, USA.		2013	http://dl.acm.org/citation.cfm?id=2485735	

			nce, Haifa, Israel						
33	URANOS: User-Guided Rewriting for Plugin-Enabled Android Application Security	Daniel Schreckling, Stephan Huber, Focke Höhne and Joachim Posegga	at WISTP (Workshop in Information Security Theory and Practice)	May		Hersonissos, Crete, Greece	2013		http://www.compose-project.eu/sites/default/files/publications/WISTP-.pdf
34	COMPOSE - A journey from the IOT to IOS.	Benny Mandler, Fabio Antonelli, Robert Kleinfeld, Carlos Pedrinaci, Alessio Gugliotta, Daniel Schreckling, Iacopo Carreras, Dave Reggett, Marc Pous, Carmen Vicente Villares and Vlad Trifa	27th IEEE International Conference on Advanced Information Networking and Applications (AINA-2013)	Mar		Barcelona, Spain	2013		http://www.compose-project.eu/sites/default/files/publications/CCPI%2713%20-%20A%20journey%20from%20the%20IOT%20to%20IOS.pdf

TEMPLATE A2: LIST OF DISSEMINATION ACTIVITIES								
NO	Type of activities	Main leader	Title	Date/Period	Place	Type of audience	Size of audience	Countries addressed
1	Workshop		<i>D. Raggett "Semantics and the Web of Things", AOITI workshop, hosted by ETSI</i>	07 Jul 2015	<i>Sophia Antipolis, France</i>	<i>Scientific Community</i>		
2	Workshop		<i>Daniel Schreckling, Juan David Parra, Joachim Posegga, Charalampos Doukas, "A Data Centric Security Framework for the IoT", 'Citizens Engagement to the IoT: Security and Privacy Challenges, European Conference on Networks and Communications, EUCNC 2015</i>	29 June 2015	<i>Paris, France</i>	<i>Scientific Community</i>		
3	Workshop		<i>Elias Tragos, Charalampos Doukas, Peter Langendörfer, 'Citizens Engagement to the IoT: Security and Privacy Challenges', Workshop 6 at EUCNCN 2015, 2015</i>	29 June 2015	<i>Paris, France</i>	<i>Scientific Community</i>		
4	Workshop		<i>Villalba, A., Perez, J., Carrera, D., Pedrinaci, C. and Panziera, L. "ServioTicy and iServe: a Scalable Platform for Mining the IoT", Workshop: Big Data and Data Mining Challenges on IoT and Pervasive Systems (BigD2M) at International Conference on Ambient Systems, Networks and Technologies (ANT), 2015</i>	2015		<i>Scientific Community</i>		

5	Invited Talks		Carlos Pedrinaci, "Developing the Internet of Things: Opportunities and Challenges", SummerSoc - Advanced School on Service Oriented Computing	30 June – 5 July, 2014	Hersonissos, Crete, Greece	Scientific Community		
6	Invited Talks		W3C Workshop on the Web of Things - Enablers and services for an open Web of Devices: Panel session on core technologies (Charalampos Doukas)	25–26 June 2014	Berlin, Germany	Scientific Community, Industry		
7	Invited Talks		Juan David Parra Rodríguez, "Security Framework in COMPOSE", IERC- Trusted Internet of Things	17 June 2014	London, UK	Scientific Community		
8	Invited Talks		COMPOSE project was presented (MAY 13) by Carlos Yubero during the event called "Future Internet opportunities in the smart cities framework" in the discussion panel called "Experimental Platforms of Future Internet"	May 2014		Scientific Community		
9	Invited Talks		Charalampos Doukas (2014) "Internet of Things, How?", TechPeaks Trento accelerator IoT Idea Generation Lab Day	17 April, 2014	Trento, Italy	Industry		
10	Invited Talks		Presentation at the 2014 Internet of Things European Summit, C. Doukas, "Financing the IoT through open source EU-funded Projects",	March 3-4, 2014		Scientific Community, Industry, Policy makers		
11	Invited Talks		Attendance at CeBIT Panel - about the Internet of Things and talked about Compose - The panel was organized by the W3C German Office in Hannover	11 Mar 2014	Hannover, Germany	Industry		
12	Invited Talks		Charalampos Doukas (2014), "Financing the IoT through open source EU-funded Projects", 2014 Internet of Things European Summit	3-4 March, 2014	Brussels, Belgium	Scientific Community, Industry, Policy makers		
13	Invited Talks		Charalampos Doukas (2014), "COMPOSE: Building an IoT Marketplace for Makers & Entrepreneurs", 2013 IoT Forum,	12 December, 2013	Barcelona, Spain	Scientific Community, Industry, Policy		

						makers		
14	Invited Talks		Daniel Schreckling: presentation of COMPOSE to the RERUM consortium	10 Dec 2013	Munuch, Germany	Scientific Community		
15	Invited Talks		Charalampos Doukas (2013), "Enabling an Open Marketplace of Services for the Internet of Things", Web of Things Workshop, the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Com	2013	Zurich, Swiss	Scientific Community		
16	Talk		"Semantics for IoT Development". C. Pedrinaci, STI Summit 2015.	September 2015	Crete, Greece	Scientific Community		
17	Talk		"In-Store analytics: the RetailerIN case", Iacopo Carreras, IT 2015 (http://www.dfki.de/ith2015),	July 2015	St. Wendel, Germany	Scientific Community		
18	Talk		"Introduction to the Web of Things", D. Raggett, Invited talk to the Web Science Institute, University of Southampton	15 Jul 2015	Southampton, UK	Scientific Community		
19	Talk		"Enabling IoT Devices to talk Web: The Community Approach", Charalampos Doukas, 5th Fokus Media Web Symposium,	May 20-21	Berlin, Germany	Scientific Community, Industry, Policy makers		
20	Talk		"Use cases and pilot projects for the COMPOSE platform", Charalampos Doukas, W3C Web of Things Interest Group F2F meeting	20-22 April 2015	Munich, Germany	Scientific Community, Industry		
21	Talk		"IoT platform glue.things – a Mashup Platform for wiring the Internet of Things with the Internet of Services", Robert Kleinfeld, W3C Web of Things Interest Group F2F meeting	20-22 April 2015	Munich, Germany	Scientific Community, Industry		
22	Talk		"Semantics for discovery and interoperability of services", C. Pedrinaci, W3C Web of Things Interest Group F2F meeting	20-22 April 2015	Munich, Germany	Scientific Community, Industry		
23	Talk		"Towards a Lightweight Standard to Make Things talk Web", D. Guinard, W3C Web of Things Interest Group F2F meeting	20-22 April 2015	Munich, Germany	Scientific Community		

						y, Industry		
24	Talk		"Securing the Web of Things - A COMPOSE Perspective", D. Schreckling, W3C Web of Things Interest Group F2F meeting	20-22 April 2015	Munich, Germany	Scientific Community, Industry		
25	Talk		"Open Source IoT Building Blocks for Startups", ThingMonk 2014, Internet of Things Event,	02-03 December 2014	London, UK	Industry		
26	Talk		"Enabling Open Markets for the Web of Things" D. Raggett, Web of Thing Session of "MindTrek 2014"	5 November 2014	Tampere, Finland	Industry		
27	Talk		"Enabling Open Markets for the Web of Things", D. Raggett, IoT 2014,	08 Oct 2014	Cambridge MA, USA	Scientific Community, Industry, Policy makers		
28	Talk		"What is the Web of Things", D. Raggett, Samsung Research,	30 October 2014	San Jose, CA, USA	Industry		
29	Demo/Exhibit		Planned event: The Smart City Pilot deployed with COMPOSE project will be also presented in the future edition of Smart City Expo World Congress (17-19 November 2015, Barcelona) at the Cellnex booth.	17-19 November 2015	Barcelona, Spain	Industry		
30	Demo/Exhibit		A demo of COMPOSE has been presented also during the "IoT as a Service" Conference (IoTaaS	26th October 2015.	Rome, Italy	Scientific Community		
31	Demo/Exhibit		An Exhibit has been organized in IoT 360 Summit, Rome, Italy, 27-29 October 2015, presenting a set of interactive demos supported by the public online COMPOSE platform	27-30 th October 2015	Rome, Italy	Scientific Community, Industry, Policy makers		
32	Demo/Exhibit		COMPOSE Project has been also promoted during last IoT World	16-18 September 2015	Barcelona,	Scientific		

	<i>it</i>		<i>Congress show</i>		<i>Spain</i>	<i>Community, Industry, Policy makers</i>		
33	<i>Demo/Exhibit</i>		<i>Berlin, 5th Media Web Symposium, 20-21 May (about 400 visitors): full event attendance (Lukasz Radziwonowicz and Alexander Futász + Charalampos Doukas) with a COMPOSE dedicated demo as part of the hourly demo tour + IOT tutorial with COMPOSE showcase. Project activities and results promotion via posters, leaflets and presentation.</i>	<i>20-21 May 2015</i>	<i>Berlin, Germany</i>	<i>Industry, Media, Civil Society</i>		
34	<i>Demo/Exhibit</i>		<i>Lisbon IoTWeek2015.: full event attendance (Fabio Antonelli, Antonio Francescon) with a COMPOSE dedicated booth. Project activities and results promotion via posters, leaflets, videos and demos.</i>	<i>16-18 June 2015</i>	<i>Lisbon, Portugal</i>	<i>Scientific Community, Industry, Policy makers</i>		
35	<i>Demo/Exhibit</i>		<i>Demo presentation at NetFutures 2015 (http://netfutures2015.eu/compose/) COMPOSE was exhibited at the NetFutures 2015 event. Guests had the chance to get informed on project status, objectives and main outcomes so far, learn how to use the developer's portal for creating and deploying IoT apps graphically with no-coding and how to connect their embedded devices for managing data and controlling them remotely.</i>	<i>25-26 March 2015</i>	<i>Brussels, Belgium</i>	<i>Scientific Community, Industry, Policy makers</i>		
36	<i>Demo/Exhibit</i>		<i>"COMPOSE: an End-To-End IoT Cloud Platform" is the title of the demo shown by David Carreras at Cloudscape VII</i>	<i>10 March 2015</i>		<i>Scientific Community</i>		
37	<i>Demo/Exhibit</i>		<i>COMPOSE Exhibit at IoT360 Summit</i>	<i>28-29 October 2014</i>	<i>Rome, Italy</i>	<i>Scientific Community, Industry, Policy makers</i>		
38	<i>Demo/Exhibit</i>		<i>Fraunhofer FOKUS at IoT People Berlin Meetup, , Berlin</i>	<i>17/07/2014</i>	<i>Berlin,</i>	<i>Industry,</i>		

	<i>it</i>				Germany	Civil Society		
39	Demo/Exhibit		<i>Charalampos Doukas presented COMPOSE at the 10th International FabLab Conference - Fab10</i>	July 5, 2014	Barcelona, Spain	Industry, Civil Society		
40	Demo/Exhibit		<i>Dominique Guinard and Iker Larizgoitia, "Web of Things: Concepts, technologies and applications for connecting physical objects to the Web", ICWE 2014 - International Conference on Web Engineering,</i>	1-4 July 2014	Toulouse, France	Scientific Community, Industry		
41	Demo/Exhibit		<i>Fraunhofer FOKUS at ETSI IoT Workshop</i>	02-03 July 2014	, Sophia Antipoles, France	Scientific Community, Industry		
42	Demo/Exhibit		<i>W3C Workshop on the Web of Things - Enablers and services for an open Web of Devices: Panel session on core technologies (Charalampos Doukas)</i>	25-26 June 2014	Berlin, Germany	Scientific Community, Industry		
43	Demo/Exhibit		<i>Demo "Bridging HTTP REST with M2M protocols like MQTT" at W3C Workshop on the Web of Things,</i>	25-26 June 2014	Berlin, Germany	Scientific Community, Industry		
44	Demo/Exhibit		<i>"Enablers and services for an open Web of Devices" Workshop on the Web of Things</i>	25-26 June 2014	Berlin, Germany	Scientific Community, Industry		
45	Demo/Exhibit		<i>Fraunhofer FOKUS at IoT Meetup</i>	25 th June 2014	Berlin, Germany	Scientific Community, Industry		
46	Demo/Exhibit		<i>Fraunhofer FOKUS at Re.Work Technology Summit</i>	20 th June 2014	Berlin, Germany	Industry		
47	Demo/Exhibit		<i>COMPOSE Hackathon at IoTWeek 2014 Hackfest</i>	17-19 June 2014	London, UK	Scientific Community, Industry, Policy makers		
48	Demo/Exhibit		<i>Workshop on the Internet of Things and Open Data with the help of COMPOSE, presented at the AngelHack</i>	May 10th 2014	Bolzano, Italy	Industry, Civil Society		

49	Demo/Exhibit		Inner Circle Track - IBM Impact 2014, "Building an "Internet of Things as-a-Service" Using IBM BlueMix" (Yoav Tock, IBM Research)	April 27 - May 1, 2014	Las Vegas, NV, USA	Industry		
50	Demo/Exhibit		PaxDB: a Fault-tolerant and Scalable RDF Store, SYSTOR 2014	10-12 June 2014	Haifa, Israel	Scientific Community		
51	Demo/Exhibit		"COMPOSE: A Cloud Platform for the IoT", SYSTOR 2014	10-12 June 2014	Haifa, Israel	Scientific Community		
52	Demo/Exhibit		Factsheet presented at the Abertis booth in the Mobile World Congress	24-27 Feb 2011	Barcelona, Spain	Industry		
53	Demo/Exhibit		Alessio Gugliotta (2013), "Cloudwatch", ICT Event 2013,	6-8 November 2013	Vilnius, Lithuania	Scientific Community		
54	Demo/Exhibit		Factsheet presented at the Abertis booth in the Barcelona Smart City Expo	18-20th Nov 2013	Barcelona, Spain	Industry		
55	Presentation		The University of Passau organized the seminar titled "Real Life Security", linked to the COMPOSE project (in particular presentation Gromova Evgeniya, "Information Flow in JavaScript"). The Seminar lasted the whole summer 2015	2015	Passau, Germany	Scientific Community		
56	Presentation		Alessio Gugliotta presented the COMPOSE platform to two private investors, as part of a dedicated event organised by INNOVA at its offices in Rome (where multiple projects and start-up products have been presented)	21 May 2015	Rome, Italy	Industry		
57	Presentation		Presentation: Benjamin Mandler, "Introduction to the Internet of Things in IBM", in IBM Israel's Business Connect 2015 (http://www-05.ibm.com/il/businessconnect/index.html)	June 16, 2015	Tel Aviv, Israel	Industry		
58	Presentation		The COMPOSE project has been presented at the "IoT as a Service" Conference (IoTaaS)	26th October 2015	Rome, Italy	Scientific Community		
59	Presentation		Presentation: Benjamin Mandler, "Internet of Things overview" to the Israel patent examiners, Israeli patents office, Justice department	June 4, 2015	Tel Aviv, Israel	Industry		
60	Presentation		The COMPOSE project has been promoted at the abertis telecom booth during Mobile World Congress	2-5 March 2015	Barcelona, Spain	Industry		

61	Presentation		Presentation of COMPOSE PaaS components in the context of the talk "Fostering Connectivity & Interactivity Between all Urban Entities", invited talk given at Re.Work – Future Cities Summit	04-05 December 2014	London, UK	Industry		
62	Presentation		A COMPOSE project presentation has been given by Abertis telecom at Smart City Expo 2014	18-20 November 2014	Barcelona, Spain	Industry		
63	Presentation		The COMPOSE project has also been presented in a dedicated session at the IoT 360 Summit	28th October 2014	Rome, Italy	Scientific Community, Industry, Policy makers		
64	Presentation		The COMPOSE project has been presented by project coordinator (Benjamin Mandler) at the "IoT as a Service" Conference (IoTaaS)	27th October 2014	Rome, Italy	Scientific Community		
65	Presentation		Presentation at the "Night of Researchers", TIS Innovation Park	25-27 September 2014	Bolzano, Italy	Civil Society		
66	Presentation		Participation & Presentation of Compose at the GET-D Conference - GETDECENTRALIZED, Agora	17 -19 September 2014	Berlin, Germany	Scientific Community, Industry		
67	Presentation		Fraunhofer FOKUS at IoT Conference Berlin,	01-03 September 2014	Berlin, Germany	Scientific Community, Industry		
68	Presentation		Participation in the SenZations 2014 Summer School on IoT Applications	31 August-6 September 2014	Biograd, Croatia	Scientific Community		
69	Presentation		Jordà Polo, Yolanda Becerra, David Carrera, Jordi Torres, Eduard Ayguadé, and Malgorzata Steinder, "Adaptive MapReduce Scheduling in Shared Environments", to appear in proceedings of the 14th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2014).	20 May 2014	Chicago (IL), US	Scientific Community		
70	Presentation		Organization of 5th International Workshop on Web APIs and RESTful Design, at 23rd International World Wide Web Conference in Seoul, South Korea	2014	Seoul, South Korea	Scientific Community		

71	Presentation		<i>Fabio Antonelli presented to all the participants the COMPOSE project, IoT Day Trento,</i>	9 Apr 2014	Trento, Italy	Scientific Community, Industry, Civil Society		
72	Presentation		<i>"Connected living: New applications for smart devices" (Benny Mandler, IBM Research News, 2 April)</i>			Scientific Community		
73	Presentation		<i>Benny Mandler, "Cloudwatch",</i>	12-13 March 2014	Brussel, Belgium	Scientific Community		
74	Presentation		<i>U-Hopper participated and presented RetailerIN, the COMPOSE application scenario on Smart Retailing at EIT ICT Labs, Euroshop 2014</i>	16-20 February 2014		Scientific Community, Industry		
75	Presentation		<i>Break out session at W3C Technical Plenary in</i>	13rd November 2013	Shenzhen, China	Scientific Community, Industry		
76	Presentation		<i>Dominique Hazael-Massieux (2013), "W3C and Web of Things", Web Intelligence Summer School on Web of Things. http://www.w3.org/2013/Talks/dhm-wot/#/step-4</i>	2-6 September 2013	Saint- Germain- Au-Mont- d'Or, France	Scientific Community, Industry		
77	Presentation		<i>Charalampos Doukas (2013), "Enabling an Open Marketplace of Services for the Internet of Things", Web of Things Workshop, the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2013). http://www.compose-project.eu/sites/default/files/publications/PDGC-2012.pdf</i>	8-12 September 2013.	Zurich, Switzerland	Scientific Community		
78	Presentation		<i>Dave Raggett (2013), "Expanding the Web beyond desktop and mobile to the Web of Things", Internet of Things Mash-up Day, http://www.compose-project.eu/sites/default/files/publications/webinos-iot-2013-07-23.pdf</i>	23 July 2013.	Oxford, UK	Scientific Community, Industry		

79	Presentation		<i>Charalampos Doukas (2013), "COMPOSE: Best Practices & Semantic Interoperability in the IoT", IoT Week 2013.. http://www.compose-project.eu/sites/default/files/publications/COMPOSE_doukas_IoTWeek-1.pptx</i>	17-20 June 2013	Helsinki, Finland	Scientific Community, Industry, Policy makers		
80	Presentation		<i>Eliezer Dekel (2012), "COMPOSE Collaborative open Market to Place Objects at your Service", Compose – Internet of Things Market Place at the Second IEEE International Conference on Parallel, Distributed and Grid Computing (PDGC 2012).. http://www.compose-project.eu/sites/default/files/publications/PDGC-2012.pdf</i>	Solan, India, 6 – 8 December 2012	Solan, India, 6 – 8 December 2012	Scientific Community		
81	Presentation		<i>Presentation of COMPOSE project - Internet of Things Day 2013. http://www.compose-project.eu/news/create-net-internet-things-day-2013-trento-italy</i>	9 April 2013	Trento, Italy	Scientific Community, Industry, Civil Society		
82	Presentation		<i>Factsheet presented at the Abertis booth - MWC (BCN).</i>	25-28 Feb 2013	Barcelona, Spain	Industry		
83	Presentation		<i>Presentation on Smart Cities at Media Web Symposium: Smart Cities as a web of people, things and services.</i>	14-15 Mar 2013		Industry		
84	Presentation		<i>Poster Session, Position Paper Submitted. All material is available via Cloudscape V Conference website. www.cloudscapeseries.eu/Pages/Home.aspx</i>	27 Feb 2013		Scientific Community		
85	Presentation		<i>Joint City SDK Workshop for potential collaboration between COMPOSE and City SDK project, Future Everything Conference. http://futureeverything.org/</i>	21 Feb 2013		Scientific Community		
86	Presentation		<i>Factsheet presented at the Abertis booth in the Barcelona Smart City Expo.</i>	13 Nov 2012		Industry		
87	Presentation		<i>Semantic requirements in COMPOSE at the IoS collaboration event in Brussels. http://ec.europa.eu/digital-agenda/events/cf/ios12/item-</i>	17 Oct 2012		Scientific Community		

			display.cfm?id=10098			y		
88	Presentation		General overview of COMPOSE at the IoT collaboration event in Brussels. http://ec.europa.eu/digital-agenda/events/cf/ios12/home.cfm	16 Oct 2012		Scientific Community		
89	Blog Post		Benjamin Mandler "IoTaaS 2015: 'The IoT realm is about to explode'", EAI blog (http://blog.eai.eu/mr-mandler-iotaaS-2015-the-iot-realm-is-about-to-explode/),	22 Apr 2015		Scientific Community, Industry		
90	Blog Post		Benjamin Mandler "Driving innovation in IoT(Internet of things)" Cordis blog (https://ec.europa.eu/digital-agenda/en/blog/driving-innovation-iotinternet-things)	20 mar 2015		Scientific Community, Industry		
91	Blog Post		"COMPOSE: The marketplace of the Internet of Things", Cordis news (http://ec.europa.eu/digital-agenda/en/news/compose-market-place-internet-things)	20 Mar 2015		Scientific Community, Industry		
92	Video		COMPOSE video Playlist on YouTube https://www.youtube.com/playlist?list=PLSwDTVw_H8PxQtj_w738stbx_Cyd48GzS					
93	Video		"Introduction to the Internet of Things in IBM": http://www-05.ibm.com/il/businessconnect/index.html					
94	Virtual MeetUp		"Create Your IoT Services & Applications with COMPOSE Platform" (http://www.meetup.com/Virtual-IoT/events/226102468/?a=co2_grp&gj=co2&rv=co2).	29 th Oct 2015	Virtual meetup	Scientific Community, Industry		

Section B (Confidential or public: confidential information to be marked clearly)

Part B2

Type of Exploitable Foreground	Description of exploitable foreground	Confidential Click on YES/NO	Foreseen embargo date dd/mm/yyyy	Exploitable product(s) or measure(s)	Sector(s) of application	Timetable, commercial or any other use	Patents or other IPR exploitation (licences)	Owner & Other Beneficiary(s) involved
General advancement of knowledge	SERVICITY AND ISERVE COMPONENTS	NO		IOT LAB	ACADEMIC	2015	OPEN SOURCE	OU
General advancement of knowledge	SERVICITY COMPONENT	NO		IOT MONITORING OF BSC SUPERCOMPUTING	ACADEMIC	2015	OPEN SOURCE	BSC
General advancement of knowledge	COMPOSE SECURITY FRAMEWORK	NO		TRAINING COURSES AND RESEARCH OPPORTUNITIES	ACADEMIC	2015		PASSAU
Exploitation of R&D results via standards	DATA MODEL OF COMPOSE WEB OBJECTS	NO		WEB THING MODEL	IOT	2016		EVT

COMPOSE is a framework of multiple open source, cutting edge technologies enabling a **truly open Platform as a Service (PAAS) model for the IoT.**

On the basis of the achieved project results, it is possible to identify two main kinds of exploitable results: tangible and intangible results.

Specifically, COMPOSE **tangible results** may be exploited in two ways:

- As a complete platform made of the integration of all technologies developed under the project, whose combination can be provided as a service to multiple third parties within a reference (open source or commercial) release.
- Individually, in which each independent technology developed under the COMPOSE project can be considered as a single product to be exploited.

COMPOSE **intangible results** represent the know-how that partners have gained throughout the project and can be exploited to e.g. create training and consultancy services.

In the following, we describe the main assets of COMPOSE (i.e. the exploitable tangible foreground) that are at the basis of the exploitation measures started by the partners (and reported in the table above). More details are reported in D9.2.3 (final exploitation plan).

- **ServioTicy** mainly deals with data streams coming from the IoT; offering the necessary means for properly managing and accessing objects as services. The component is accessible via specific APIs and the GUI of glue.things. Key features of the component are: *the openness (i.e. any type of object can be connected), the high scalability in managing IoT data streams and the simplicity of managing IoT objects and their data.* Main contributor of this component has been BSC (with the support of all technical partners), but, given the centrality of this component in the COMPOSE solution, all partners will adopt it in the respective exploitation paths. For example, BSC is using it for an internal project (monitoring the BSC supercomputing) in cooperation with other university departments, while CN is adopting it for a cooperation with an external third party (IoT platform provider) and develop a rapid prototyping IoT platform.
- **iServe** enables the discovery, recommendation and composition of services, may they be COMPOSE Service Objects, applications, or even external services (i.e. services available on the Web). The component is accessible via specific APIs and the GUI of glue.things. Key features of the component are: *the openness (i.e. the possibility to deal with any type of service accessible via APIs); the fast service discovery and composition; and the simplicity of managing service discovery and composition.* Main contributor of this component has been OU (with the support of INN for the trust filtering part), which is adopting it (together with the ServIoTicy component) for implementing its IoT Lab for academic collaborations. However, given its semantic reasoning capabilities, this component represent an important element for the exploitation of the whole COMPOSE platform. Therefore, partners aiming to follow this exploitation path (e.g. INN and FOKUS) will adopt it.
- **glue.things** offers an integrated environment for software developers to create, manage, deploy and run service objects and applications in the COMPOSE platform. The component offers the necessary GUI for a *quick and easy access to all main platform capabilities, as well as a unified access to all platform APIs.* Main contributor of this component has been FOKUS (with the support of all technical partners), which is collaborating with external third parties (in the electronic manufacturing and energy sectors) to further develop and adapt it to specific vertical applications. It is worth to highlight, that this component is the main front-end of all COMPOSE functionalities, therefore it can be fully exploited with the full COMPOSE back-end in place (i.e. the whole platform). Moreover, the COMPOSE SDK (including mobile and M2M interfaces) is a part of this component that will be exploited by all partners that aim to exploit COMPOSE functionalities (to both inbound and outbound).
- **COMPOSE Security Framework** introduces a comprehensive set of security and privacy features to *guarantee access control, policy control at multiple levels (data, applications, workflows), reputation and provenance management, flow analysis to prevent malicious and dangerous use of applications available in COMPOSE.* Main contributor of this component has been PASSAU (with support from INN for the trust part), which is exploiting it for academic purposes. In addition, RETE is investigating the possibility

to exploit it internally (i.e. in their existing SmartBrain product) and the other partners aiming to exploit the whole COMPOSE platform (e.g. INN and FOKUS) will adopt it.

- **COMPOSE Cloud platform** is the underlying *efficient and open cloud environment* (based on Cloud Foundry), offering hosting and managing capabilities to enable from a small to a large scale development. This component actually implements the *PaaS approach of COMPOSE*, i.e. delivering an integrated yet comprehensive set of components that can readily be used and customized, based on a specific scenario being targeted by a platform customer. Main contributors of this component have been IBM and RETE (with the support of all technical partners) that will exploit it internally. Specifically, IBM is patenting some of the elements (communication of the cloud platform) and will push these technologies in IBM products; while RETE will introduce some of its elements (e.g. Service Life Cycle Management) in its SmartBrain product. Moreover, partners that aim to exploit the PaaS approach of COMPOSE (i.e. INN) will adopt it.

It is worth to highlight that all COMPOSE components have been released under open source licenses (Apache 2.0), with the exception of the internal communication technology of the COMPOSE cloud platform, developed by IBM, that will be covered by patents (currently 4 patent applications have submitted), and the scalable consistency services.

Moreover, as additional to the concrete tangible results, we also consider *the outcomes of the pilot and demonstration activities*:

- The smart territory, smart space and smart city pilots developed innovative IoT applications (namely: **Go2ski**, **RetailerIN** and **SmartVAO**) and associated enabling technologies (e.g. **SDK**) that, with the opportune further developments, have a commercial value and the respective partners (namely: CN, UH, EVT, RETE) are including them in their exploitation paths. For Go2ski a second experimentation is planned by CN in the next winter season and concrete cooperation agreements have been set with some potential customers (Trentino Ski Areas). For RetailerIN, UH gained a lot of experience in the sector and has been able to spinoff a new company that is dealing with such a market (with some customers already), while EVT will push some of the COMPOSE capabilities (experimented in the pilot) in some of the company products (some of the existing customers demonstrated interests). For SmartVAO, RETE has already an agreement with the city of Tarragona and will extend the experimentation in other parts of the city.

In parallel to the “official” pilot activities, partners developed other IoT applications for specific demonstration purposes. Similarly to the pilot outcomes, the **Smart Library** use case will be exploited by PASSAU, as well as the **final COMPOSE demonstrator** could be part of the smart space solutions of UH/EVT or exploited as a distinct innovative application. In particular, PASSAU aims to continue the experimentation of the smart library, with also the possibility to engage with other libraries in the Bavarian region

